Week 2: Modern Cloud Tools

Sources:

1. “Cloud Computing Tutorial for Beginners | Cloud Computing Explained | Cloud Computing | Simplilearn.” YouTube, 18 May 2018, www.youtube.com/watch?v=RWgW-CgdIk0.
2. “Cloud Computing in the Year 2020.” YouTube, 9 Dec. 2019, www.youtube.com/watch?v=1pBuwKwaHp0. Accessed 23 Apr. 2020.
3. Simplilearn. “AWS In 10 Minutes | AWS Tutorial For Beginners | AWS Training Video | AWS Tutorial | Simplilearn.” YouTube, 2020, www.youtube.com/watch?v=r4YIdn2eTm4.
4. Varnum, David, et al. “Fundamentals of Google Compute Engine (GCE) & Platform (GCP)!” Network Management Software - Reviews & Network Monitoring Tools, 28 Mar. 2019, www.networkmanagementsoftware.com/fundamentals-google-compute-engine-gce/. Accessed 17 Oct. 2020
5. “Compute Engine: Virtual Machines (VMs).” Google Cloud, cloud.google.com/compute/. Accessed 17 Oct. 2020.
6. “App Engine | Google Cloud.” Google Cloud, 2019, cloud.google.com/appengine/.

Lecture 1: Introduction to Clouds

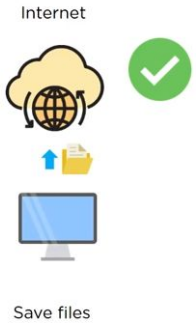
1. What is cloud computing? Why do we need cloud computing?
2. Types of clouds: deployment and service models
3. Cloud Providers
4. Demo

Source: [1]

What is Cloud Computing?**

Cloud computing is the delivery of on-demand computing services over the Internet on a pay-as-you-go-basis.

Rather than managing files on a local storage device, cloud computing makes it possible to save them over internet



Source: [1]

Why is it the better option?*

On-premise vs Cloud Computing

- Higher pay, less scalability
- Allot huge space for servers
- Appoint a team for hardware and software maintenance
- Poor data security
- Less chance of data recovery

- Pay for what you use
  - Scale up= pay more
  - Scale down= pay less
- No server space required
- No experts required for hardware and software maintenance
- Better data security
- Disaster recovery

Source: [1]

Why is it the better option?

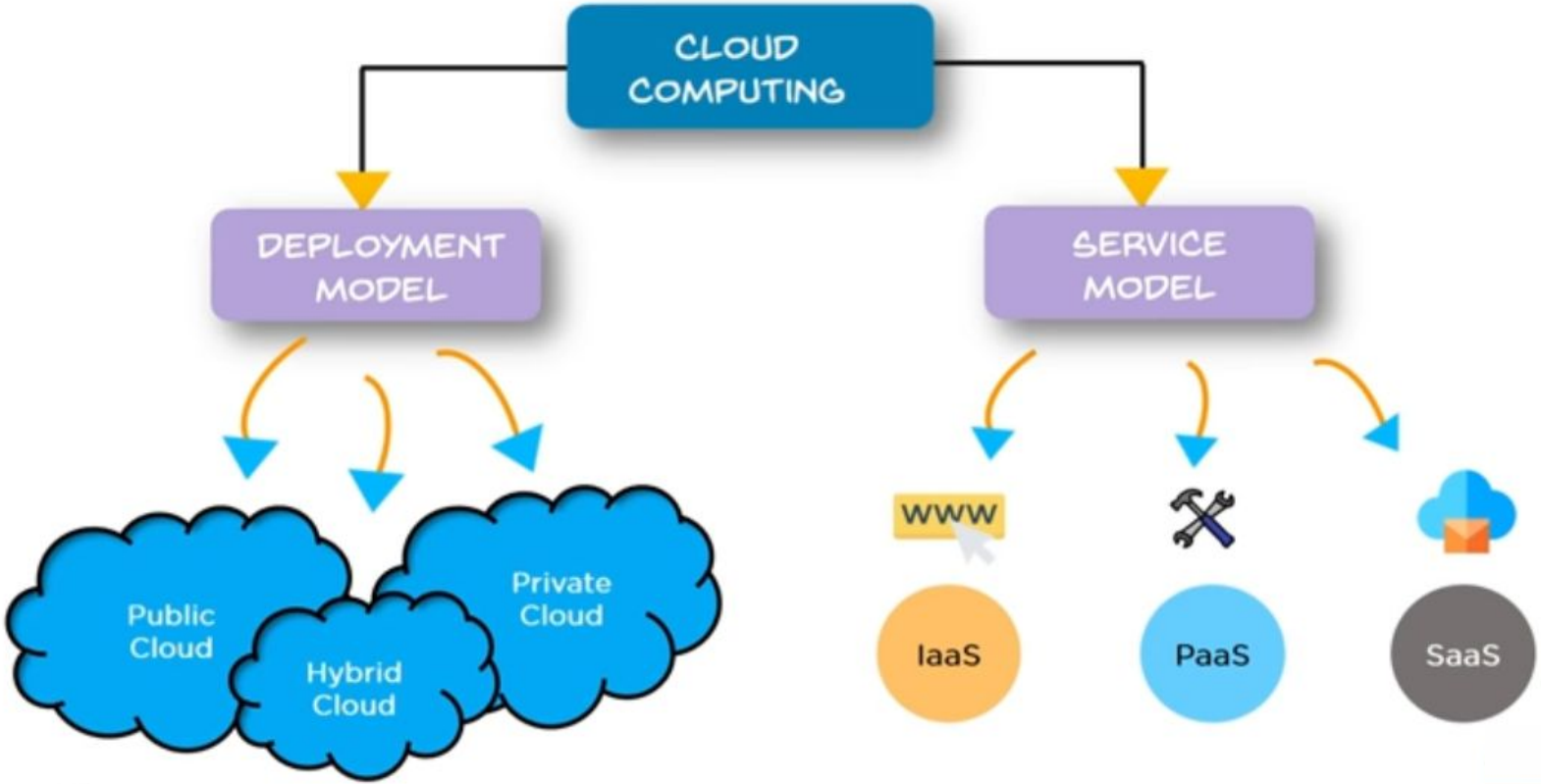
On-premise vs Cloud Computing

- Lack of flexibility
- No automatic updates
- Less collaboration
- Data cannot be accessed remotely
- Takes longer implementation time

- High Flexibility
- Automatic software updates
- Teams can collaborate from widespread locations
- Data can be accessed and shared anywhere over the internet
- Rapid implementation

Source: [1]

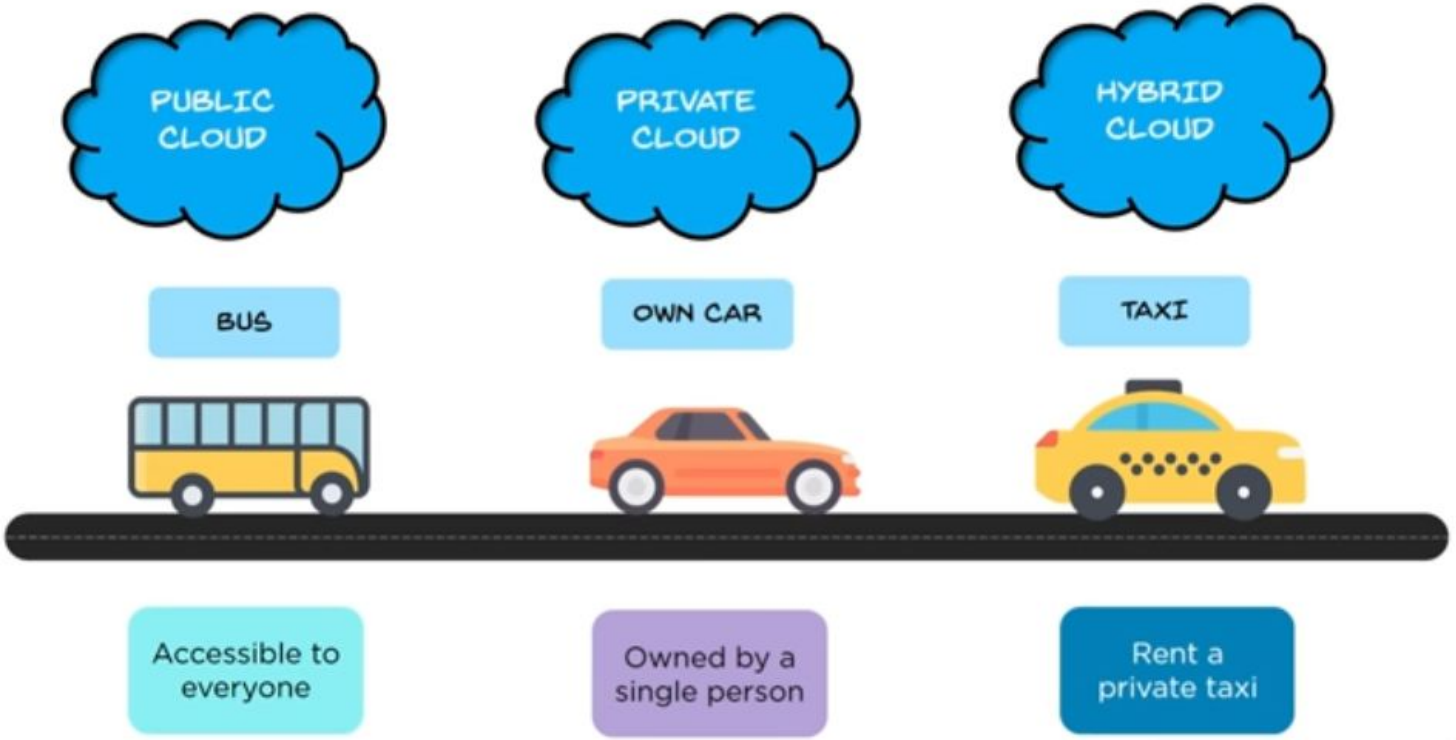
Types of Cloud Computing**



Source: [1]

Deployment Models

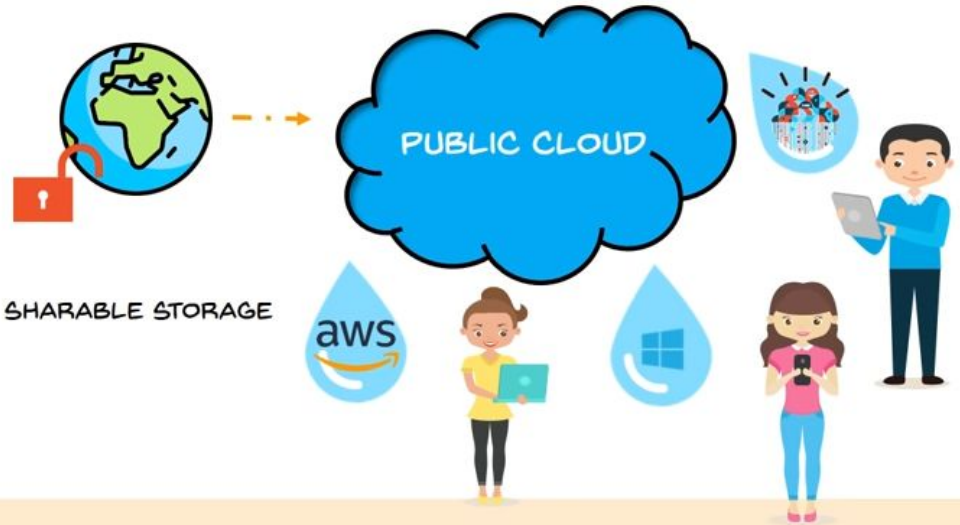
Example of vehicles used for commuting:



Source: [1]

1. Public Cloud:*

The cloud infrastructure is made available to the general public over the Internet and is owned by the cloud provider.



Top three examples in 2020:

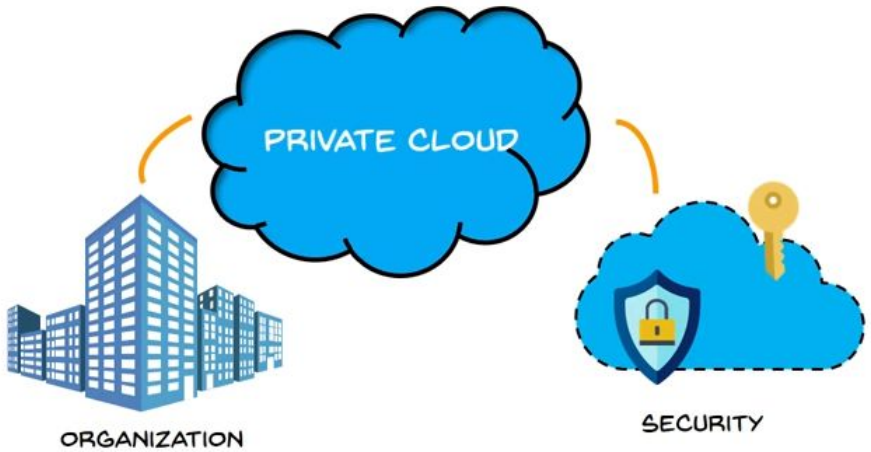
1. AWS (33%),
2. Azure (16%)
3. GCP (8%)

Source: [1]

2. Private Cloud:*

The cloud infrastructure is exclusively operated by a single organisation.

- It can be managed by the organisation or a third-party and may exist on premises or off-premises.



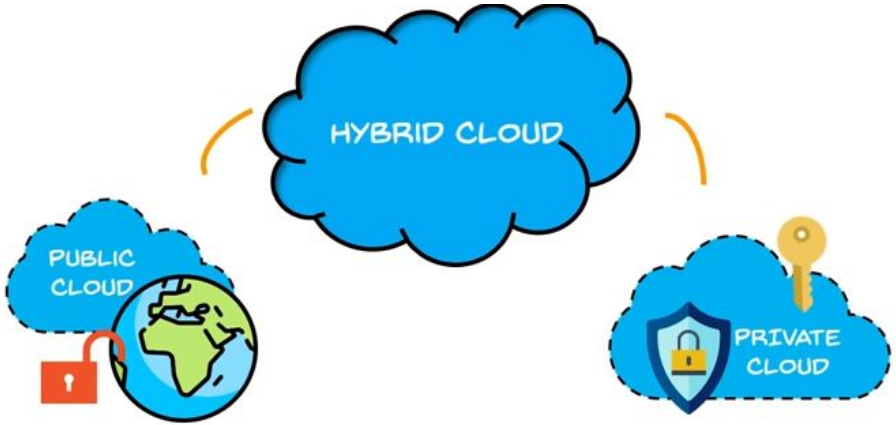
Examples:

1. AWS
2. VMware

Source: [1]

3. Hybrid Cloud:*

It consists of the functionalities of both public and private cloud.



Examples:

- Dropbox
- National / Federal agencies use private cloud when sensitive information is involved. Also, they use the public cloud to share datasets with general public or other government departments.

Source: [1]

Service Models

Which cloud service is suitable for you?

I just want a VM!

IAAS

If your business needs a virtual machine, opt for Infrastructure as a Service

I want a platform/interface!

PAAS

If your company requires a platform for building software products, pick Platform as a Service

I want a finished product!

SAAS

If your business doesn't want to maintain any IT equipment, then choose Software as a Service

Source: [1]

1. Infrastructure-as-a-Service (IaaS):*

- IaaS is a cloud service that provides basic computing infrastructure
- Services are available on PAY-FOR-WHAT-YOU-USE model
- IaaS providers include Amazon Web Services, Microsoft Azure and Google Compute Engine
- Users: IT Administrators

Source: [1]

2. Platform as a Service (PaaS):*

- PaaS provides cloud platforms and runtime environments for developing, testing, and managing applications
- It allows software developers to deploy applications without requiring all the related infrastructure
- Users: Software Developers

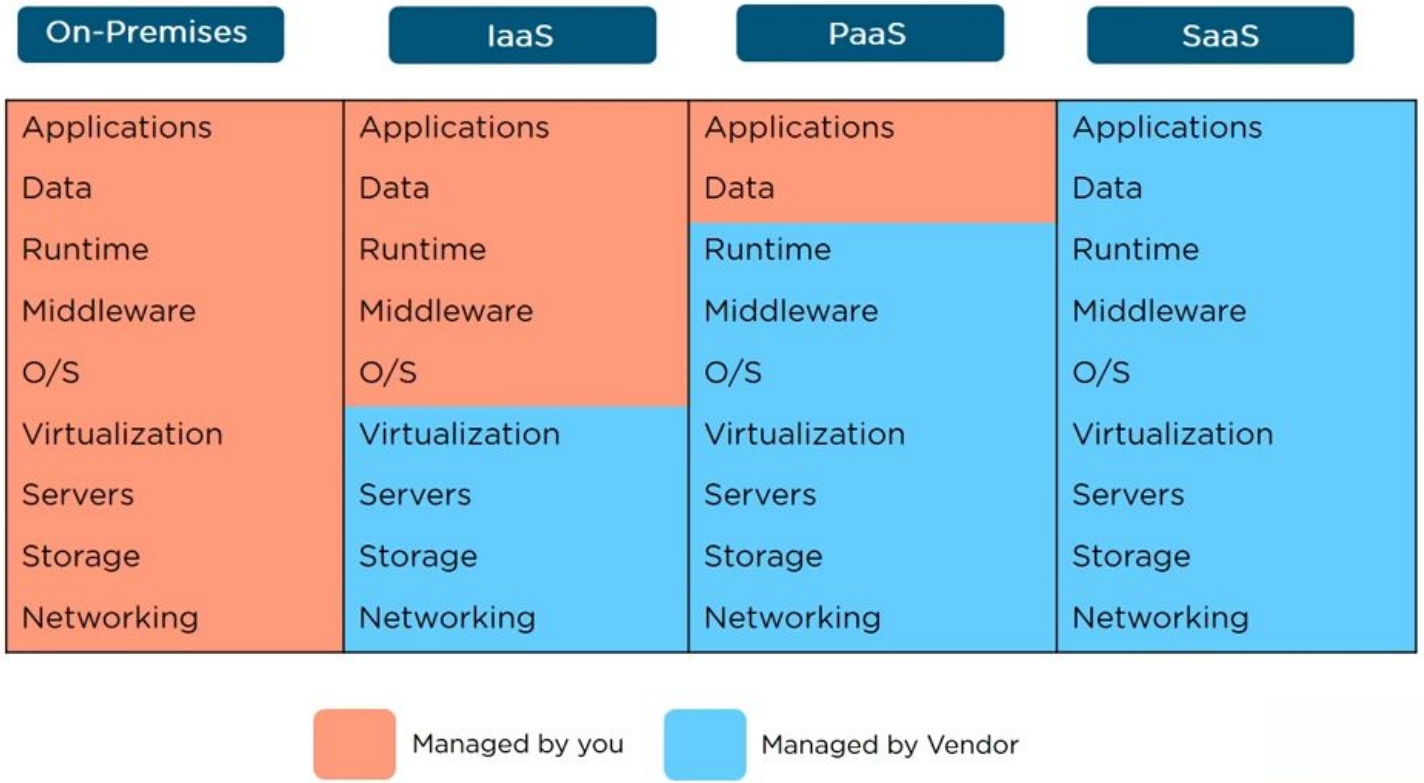
Source: [1]

3. Software As a Service (SaaS):*

- In SaaS, cloud providers host and manage the software application on a pay-as-you-go pricing model
- All software and hardware are provided and managed by a vendor so you don't have to maintain anything
- Users: End Customers

Summary and comparison

Source: [1]



Source: [1]

Example:

Task: you want to bake a cake

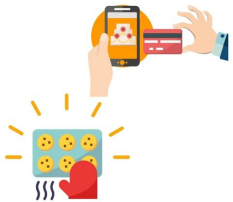
On-Premises

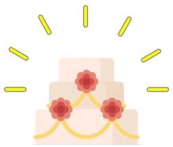
Made at Home

- Dinning table
- Water
- Electricity
- Oven
- Cake Pan
- Flour
- Sugar
- Butter
- Eggs

Managed by you Managed by Vendor

IaaS	PaaS	SaaS
Buy & bake	Cake delivery	Dine out
Dinning table	Dinning table	Dinning table
Water	Water	Water
Electricity	Electricity	Electricity
Oven	Oven	Oven
Cake Pan	Cake Pan	Cake Pan
Flour	Flour	Flour
Sugar	Sugar	Sugar
Butter	Butter	Butter
Eggs	Eggs	Eggs





Managed by you

Managed by Vendor

Source: [1]

Cloud Providers*

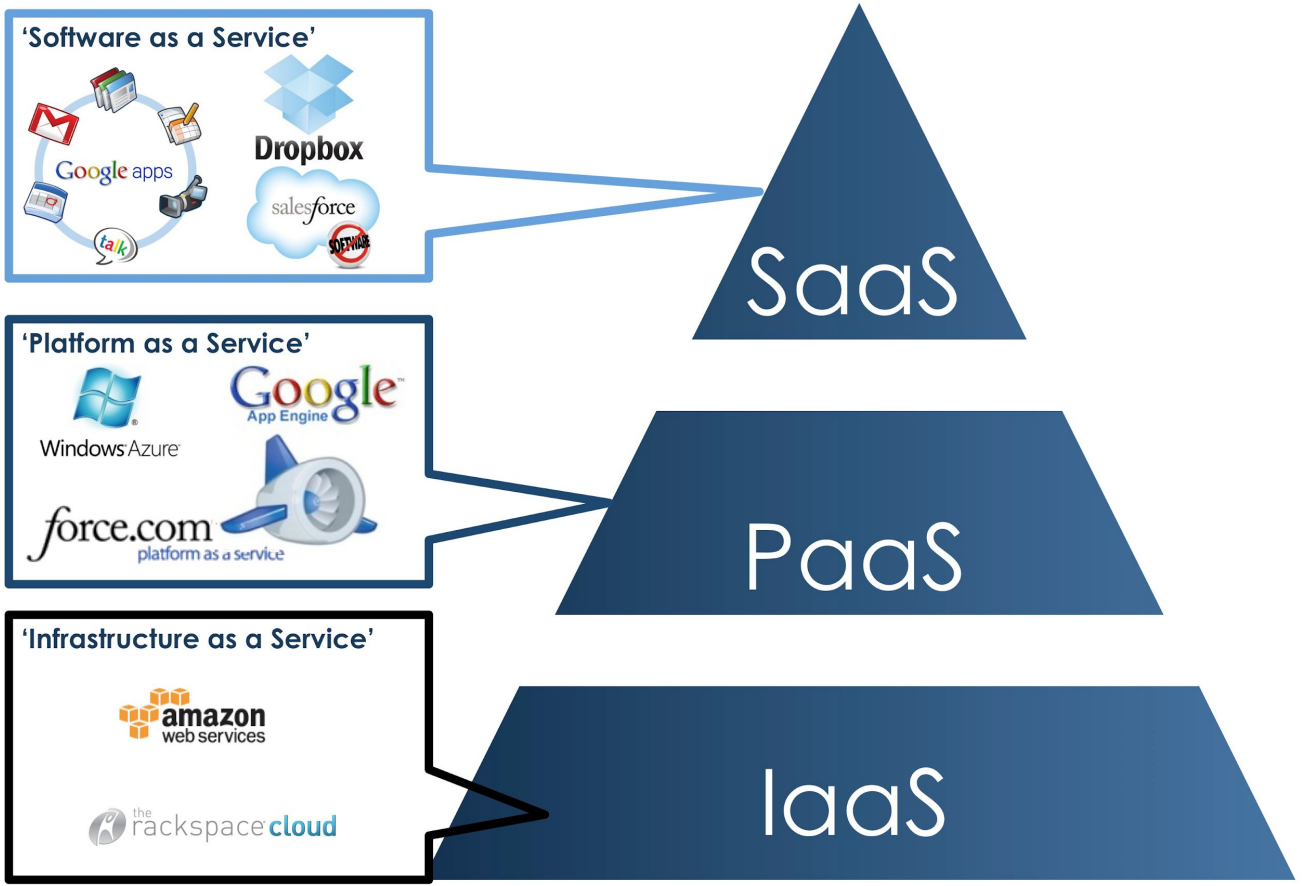
Source: [1]

Source: [1]

Classification according to service model:

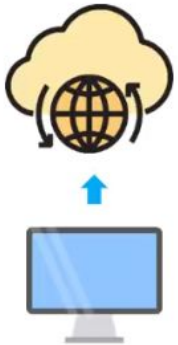
Cloud providers like AWS, Azure, and GCP have multiple deployment and service offerings.

AWS' flagship is IaaS, Azure is mostly PaaS, and Google mostly provides SaaS.



Source: [1] 19:12 min-23:37 min

Demo: AWS EC2 and AWS S3 *



AWS EC2

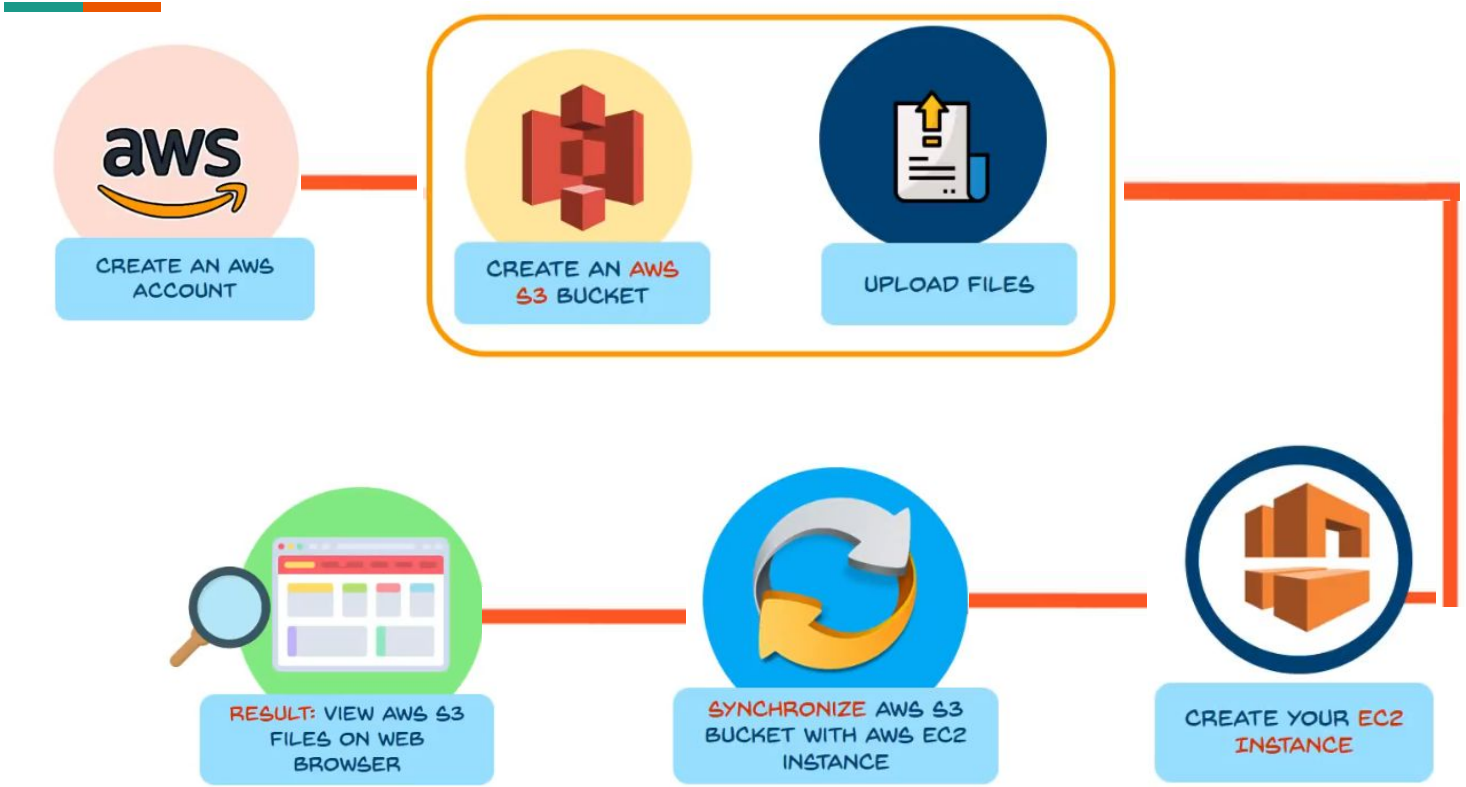
- AWS EC2 is a web service that provides a secure and resizable compute capacity in the cloud
- It can be used to launch as many virtual servers you need

AWS S3

- AWS S3 is a Simple Storage Service provided by AWS
- Using Amazon S3, you can store and retrieve any amount of data at any time on the web

Source: [1]

*


CREATE AN AWS ACCOUNT
CREATE AN AWS S3 BUCKET
UPLOAD FILES
CREATE YOUR EC2 INSTANCE
SYNCHRONIZE AWS S3 BUCKET WITH AWS EC2 INSTANCE
RESULT: VIEW AWS S3 FILES ON WEB BROWSER

Source: [2]

Service Level Agreement (SLA)

A contract between cloud provider and user.

- Provider guarantees a certain uptime and error rate for the service and will generally provide a financial credit / refund back to the client if it fails to meet those SLA requirements.
- Client usually has a quota that they must stay within when using the service.

Monthly Uptime Percentage	Service Credit Percentage
Less than 99.99% but equal to or greater than 99.0%	10%
Less than 99.0% but equal to or greater than 95.0%	30%
Less than 95.0%	100%

Source: [2]

Advantages*

1. High Availability & Redundancy

- **Region**: Cloud Campus Physical Location
  - Each Region is actually a collection of multiple zones
- **Zone**: Isolated data centres
  - E.g. if a very important data, you can store it in multiple zones in a region. So if one zone gets damaged you have the other zone in that region to fallback to.

2. Pay-as-you-go

- Billed by the second

3. Vertical Scale vs Horizontal Scale - all cloud providers have load balancing services.

- Vertical Scaling - Make things bigger (e.g. add more CPU cores, main memory, etc. to make a VM more powerful) - limit is easily reached.
- Horizontal Scaling - make more things (e.g. have more VMs and balance the load between them).

Source: [2]

Disadvantages*

1. Vendor lock-in: you get hooked

Solution?

- Multi-cloud (use more than one cloud provider)
  - Having multiple providers in a market prevents vendor lock-in and better price discrimination.

2. Scaling

- Solution 1 : Containerisation - allow you to take the application and deploy it on any cloud you want.
  - Also helps prevent lock-in
  - Use container registries / repositories - a place to store containers.
  - Kubernetes - organises containers into groups of pods. It can automatically scale those pods up or down based on the amount of traffic or utilisation.
- Solution 2: Serverless (functions-as-a-service) launched in 2014...

Lecture 2: Cloud Computing Solutions

1. Advantages & Disadvantages
2. Evolution
3. Lifecycle of Cloud Computing Solution & Tools

Source: [2]

Evolution and Major Players*

	Major Feature(s)/Points:	Vendors:
IaaS	• Provides a virtual compute environment made up of low level building blocks (e.g. VMs, Storage Buckets) • Upto the developer to manage and scale h/w	VM: Google's Compute Engine VM: Amazon's Elastic Compute Cloud (EC2) Storage Buckets: Amazon's Simple Storage Service (S3)
SaaS	• Provides s/w environment • Upto the developer to manage and scale s/w	Dropbox
PaaS	• Provides h/w and s/w abstraction with configuration • Also takes care of security, scale and management	AWS's Elastic Beanstalk, Google's App Engine, Heroku

Evolution and Major Players

	Major Feature(s)/Points:	**Vendors:**
BaaS (Backend-as-a-Service)	Provides SDKs that brings cloud directly into front-end applications (i.e. abstract backend - so developer does not need to write detailed backend code).	Google's Firebase, AWS' Amplify
FaaS (Function-as-a-Service)	**2014: Serverless** → servers (that you don't think about) Solving **scaling issue** (no need to deal with scaling of backend e.g. use of containers, VMs, etc.). Allows you to run code in response to events. The code will scale automatically. Pay-by-go → pay by each function invocation. **2019: Anything stateless can be serverless** Cost effective	AWS' Lambda AWS' Runtime API, Google's Cloud Run Service

Source: [2]

Source: [1]

Lifecycle of a Cloud Computing Solution

Step 1: Define the Purpose

Understand the requirements of the business and determine what type of applications to run on the cloud

Get a proper/sound understanding of the requirements. Only then will you be able to properly pick the right service offered by the provider.

Source: [1]

Step 2: Define the Hardware

Choose a compute service that will provide the right support where you resize the compute capacity in the cloud to run application programs

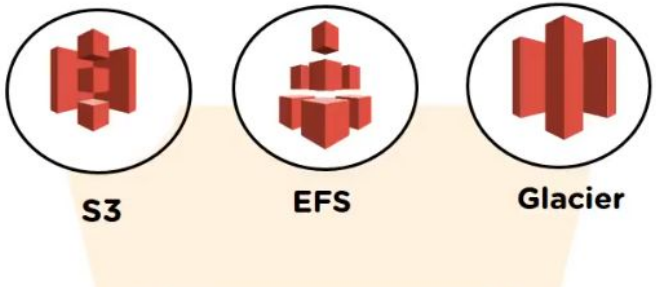
There are different hardware services for different needs. Pick the right one for what is needed:

- Amazon's Elastic Compute Cloud (EC2) for IaaS
- AWS' Lamda for serverless computing
- Amazon's Elastic Container Service (ECS) for containerised service

Source: [1]

Step 3: Define the Storage

Choose a storage service where you can backup and archive your data over the internet



Choose the appropriate storage service. Pick one for:

- Backup e.g. Amazon's Simple Storage Service (S3)
- Archive locally within the cloud or over the Internet e.g. Glacier

Source: [1]

Step 4: Define the Network

Define a network that securely delivers data, videos, applications etc. with low latency and high transfer speed

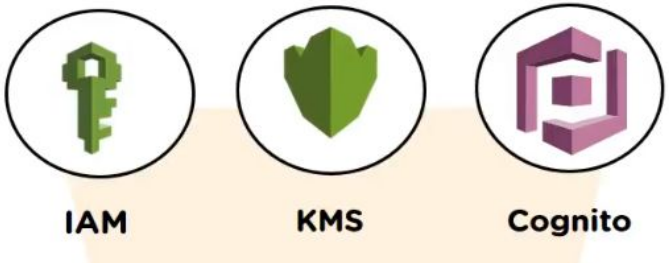
Define and identify the network services properly:

- e.g. VPC for network
- E.g. Route 53 for DNS
- E.g. Direct Connect for Private P2P line (e.g. from office to data centre).

Source: [1]

Step 5: Define Security

Set up your security service which enable services for user authentication or limiting access to a certain set of users on your AWS resources



Clouds also provide a variety of security features/options:

- IAM (Identity and Access Management)
- Policy (Security Rules for Resources)
- Roles (Assignable Access Levels)
- Service Account (Machine to Machine Permission)

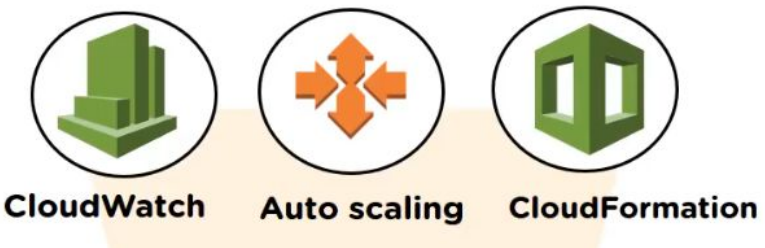
Pick the right security product for your needs:

- e.g. IAM for Authentication and Authorisation
- e.g. KMS for data encryption at rest

Source: [1]

Step 6: Define Management Processes and Tools

You can have complete control on your cloud environment by defining management tools which monitor AWS resources and the customer applications running on AWS platform



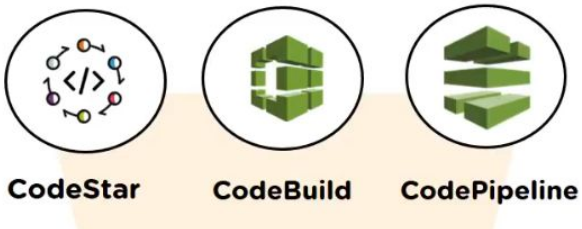
There are a variety of deployment, automation, and monitoring tools that you can pick from:

- e.g. CloudWatch is monitoring
- e.g. Auto scaling is for automation
- e.g. Cloud Formation is for deployment

Source: [1]

Step 7: Testing the Process

Verify the process using AWS developer tools where you can build, test and deploy your code quickly

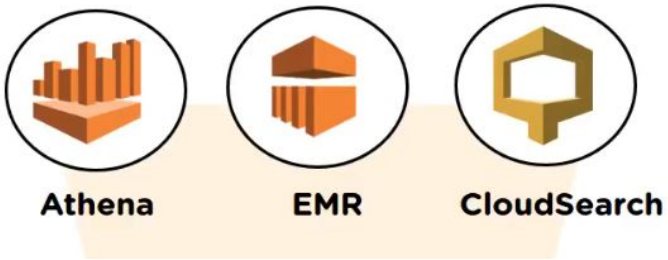


e.g. CodeStar, CodeBuild, Code Pipeline allow a developer to build, test, deploy your code quickly.

Source: [1]

Step 8: Analytics

Finally, analyze and visualize data by using analytics services where you can start querying data instantly and get results



- e.g. Amazon's Athena - to visually view what is happening in your environment; an interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL.
- e.g. Amazon's Elastic MapReduce (EMR) - to analyze vast data sets.
- E.g. Amazon's CloudSearch - to set up, manage, and scale a search solution for your website or application.

Lecture 3: GCP & AWS

1. GCP: Introduction
2. GCP: Compute Services (GCE & GAE)
3. AWS: Introduction
4. Why has AWS been so successful?
5. Commonly Used AWS Services

Google Cloud Platform (GCP)*

Source: [1]

- GCP is a set of services that enables developers to build, test, and deploy applications on Google's reliable infrastructure.
- Explored in Lab 2

- Free trial is available:
  - How To Create Trial Google Cloud Platform Account with Free $300 Credit." Storage Tutorials, www.storagetutorials.com/create-free-trial-google-cloud-platform/.
  - Credit Card is must to create Google Cloud Platform (GCP) account
  - Console: https://console.cloud.google.com/home/dashboard?project=integrated-will-292812
- Can be explored at https://cloud.google.com/

GCP Compute Service: GCE*

Source: [4. 5]

Google Compute Engine

- Google's Compute Engine (GCE) is part of Google's Infrastructure-as-a-Service (IaaS) offering, where you can build high-performance, fault-tolerant, massively scalable compute nodes to handle your application's needs.
- Standard or Custom VM images
- Variety of predefined hardware selection (CPU, RAM)
- Three interfaces: CLI, Web-based, via app (using REST API)

More information at "The Fundamentals of Google Compute Engine (GCE)", www.networkmanagementsoftware.com/fundamentals-google-compute-engine-gce/

GCP Compute Service: GAE*

Source: [6, 7]

Google App Engine

- Google App Engine (GAE) is a fully managed, serverless platform for developing and hosting web applications at scale.
- Started as Platform-as-a-service - instead of getting servers, you get an application framework.
- You can choose from several popular languages, libraries, and frameworks to develop your apps, then let App Engine take care of provisioning servers and scaling your app instances based on demand.
- Demo: http://compute-demo.appspot.com/

Amazon Web Services (AWS)

Source: [1]

- Amazon Web Services (AWS) is a cloud service from Amazon
- It provides services over the internet
- AWS services can be used to create and deploy any type of application in the cloud
- AWS uses the subscription pricing model (pay for what you use)

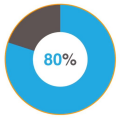
Source: [3]

- Global cloud platform

Allows you to host and manage your services on the Internet.

Used to host their infrastructure and loads of services provided to customers.

- Used by around 80% of Fortune 500 Companies



- Infrastructure as a service

- Platform as a service

Java, Ruby, PHP as a service so you don't have to manage the runtime environment.

- Software as a service

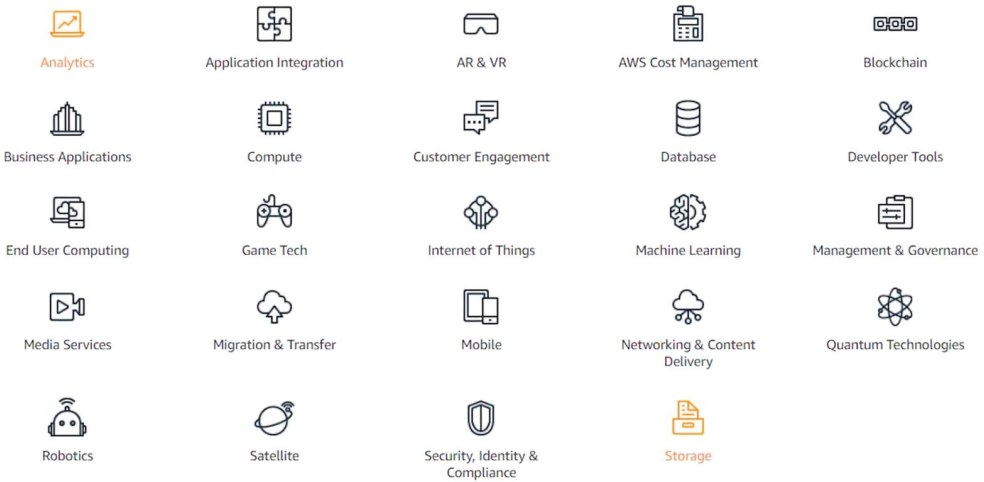
Email sending capabilities e.g. Simple Email Service (SES), Message queueing services e.g. Simple Queuing Service (SQS)

- Cloud storage platform

Storage options like Simple Storage Service (S3), Elastic Block Store (EBS), and Elastic File System (EFS)

Source: [2]

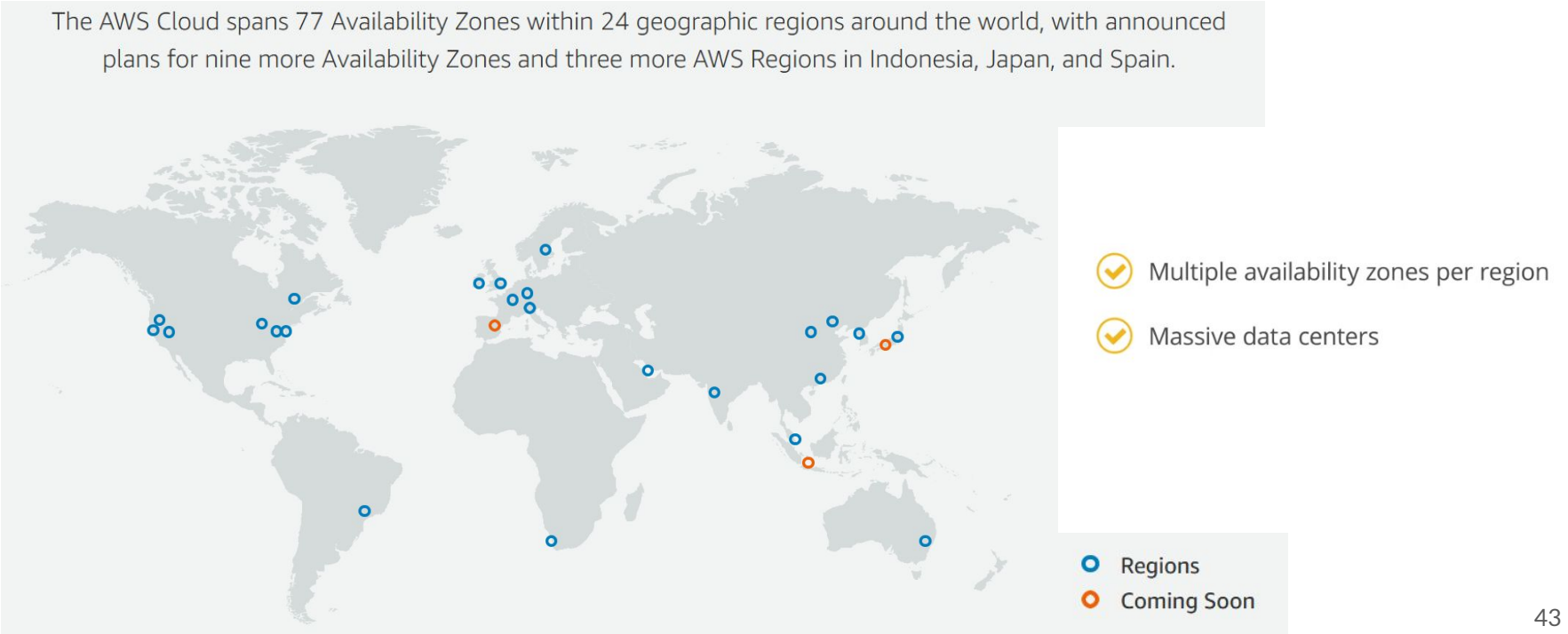
- AWS offers more than 150 different services and APIs



- 64 Services currently
- Launching new services in all domains
- Focus on machine learning
- Focus on SAAS products
- Reduction in costs

Source: [2]

- According to https://aws.amazon.com/ ⇒ 24 regions (global footprint)



Why has it been so successful?

Source: [3]

- Per hour billing

Very clear and transparent billing structure. Also some regions are cheaper than others (e.g. Oregon and Virginia). Long-term sign up is cheaper than on-demand basis..

- Easy sign up process

Very fast!

- Simple billing

Integrated billing dashboard with reports that can be customised.

- Stability

Stable service: very few, region-specific, and short outages.

- Trusted vendor

Used by almost everyone in the industry and seen by almost everyone as a trusted provider.

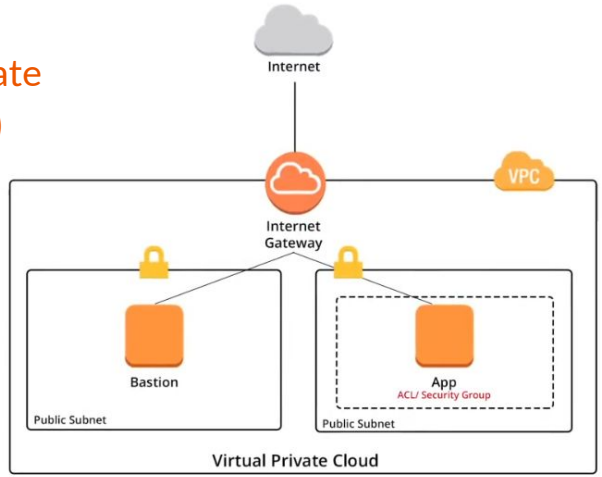
Commonly Used Services*

Source: [3]

1. Elastic Compute Cloud (EC2)

- Give bare server. A machine that can be launched and you can run your software on it.
- You can get small or big machine based on your requirements.

2. Virtual Private Cloud (VPC)



- They do not allow you full control of their cloud. Instead, they give you chunks of their cloud i.e. VPC.
- VPC allows you to create networks in the cloud and then run your servers in those networks.

Source: [3]

3. Simple Storage Service (S3)

- Allows you to upload and share files. Mostly a file storage and sharing service.

4. Relational Database Service (RBS)

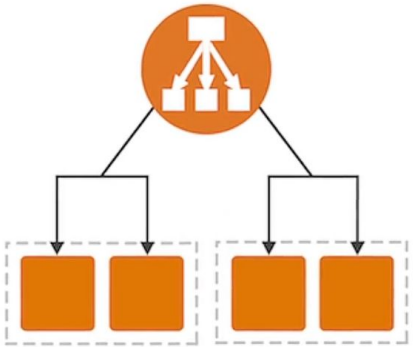
- Allows you to run and manage databases on the cloud.
- It has almost all the major flavours of the databases e.g. SQL, Oracle, and MySQL, etc.

Source: [3]

5. Route 53

- For DNS - a managed DNS service where you can point your DNS to Amazon and they take care of the stuff.
- A global and scalable DNS service.

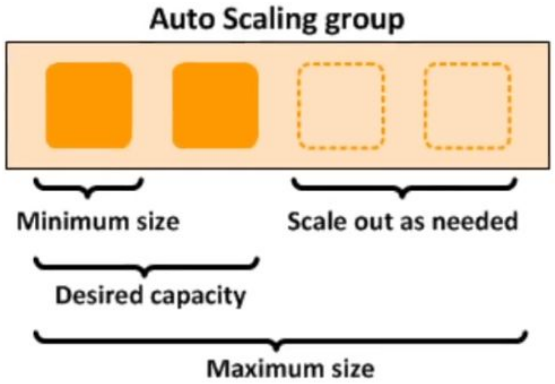
6. Elastic Load Balancing (ELB)



- A service that give you the opportunity to load balance incoming traffic onto multiple machines. This way you can scale up your Web applications to any number of users.

Source: [3]

7. Autoscaling



- Adds capacity on the fly to elastic load balancers so that your Web site or your application is never down due to load.