

# Table of contents

What are Traditional IT Services Like? .....	2
Amazon Web Services Versus Traditional IT .....	7
Amazon Web Services .....	10
Cloud Computing Economics.....	19
Principles of Cloud Native for AWS .....	24
AWS Security .....	27
AWS Security Compliance .....	30
Identity and Access Management (IAM) .....	32
AWS Security Resources.....	39
Global Infrastructure .....	47
AWS Networking .....	52
Firewalls .....	59
AWS Storage.....	63
Compute Services .....	72
AWS Lambda .....	78
Compute Container .....	83
AWS Databases .....	87
SQL Data stores (Structured).....	91
AWS NOSQL Data Stores .....	97
AWS Database SUMMARY .....	105
Application Integration .....	107
AWS Migration Services .....	113
AWS Management Services.....	119
AI/ ML .....	122
Business Application Services in AWS .....	134
Customer Engagement Services .....	137
AWS Developer Services.....	140
AWS End User Computing Services .....	149
AWS frontend & Mobile .....	151
AWS IOT .....	154
AWS Billing .....	158
AWS EC2 Billing .....	162

# What are Traditional IT Services Like?

Imagine you are producing an Idea with new technology, and you need three servers. This sparts you to order for:

- Hardware, Power, Cooling , Space
- Operational and Security installation
- Access granted to the requester
- Software installation by the requester.

This can take day or weeks or months . Any you will have to take care of:

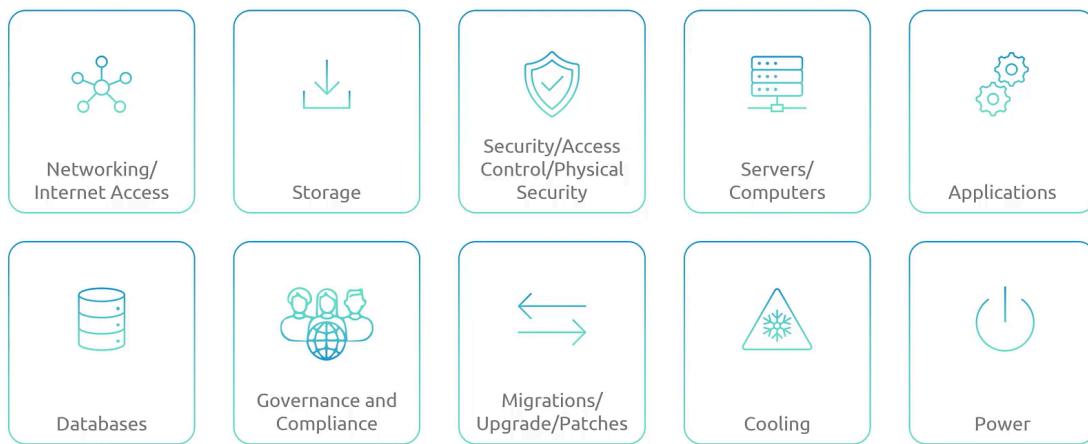


image.png

## Pros to Traditional IT Services

- Improved Security
- Greater Customization
- More Control

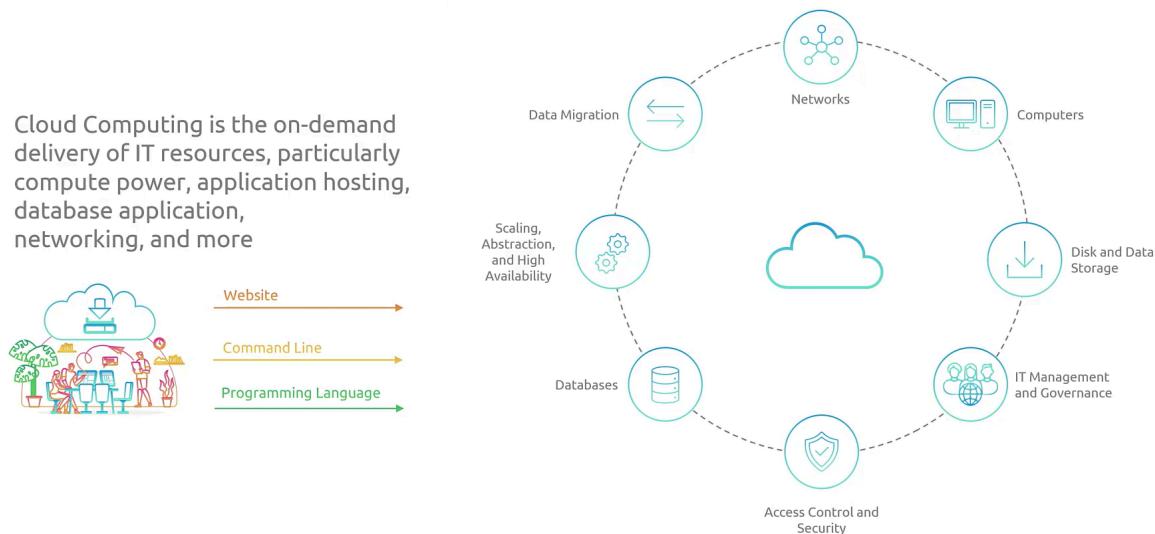
# Cons Of Traditional IT Services

- Increased cost / Poor ROI
- Limited scalability
- Responsible for everything
- Limited location
- More personnel
- Long Provisioning times

Clear the Cons outweigh the pros. What is the Solution?

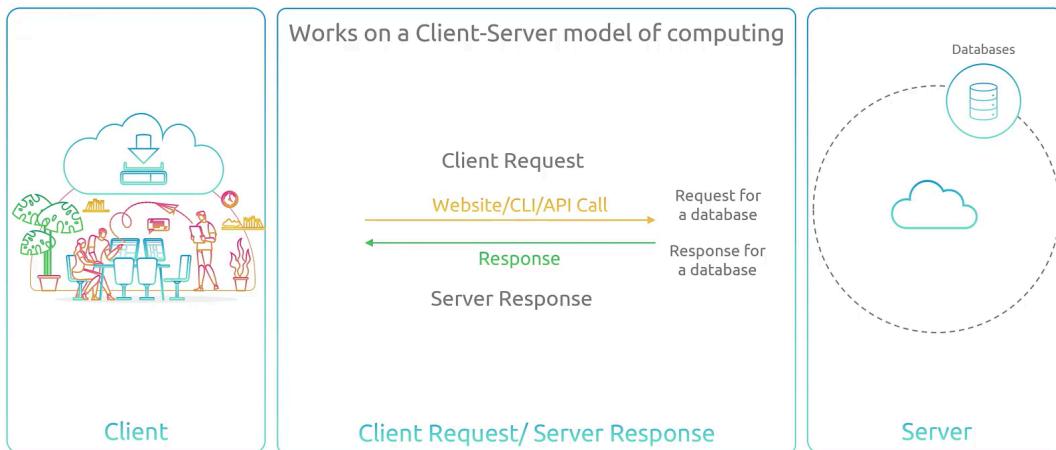
# Cloud Computing

## What Is Cloud Computing?



image\_1.png

For example if you want a Server to run a Database or Application , all you have to do is make a request using the AWS console or AWS CLI or API calls and a response will be given back.

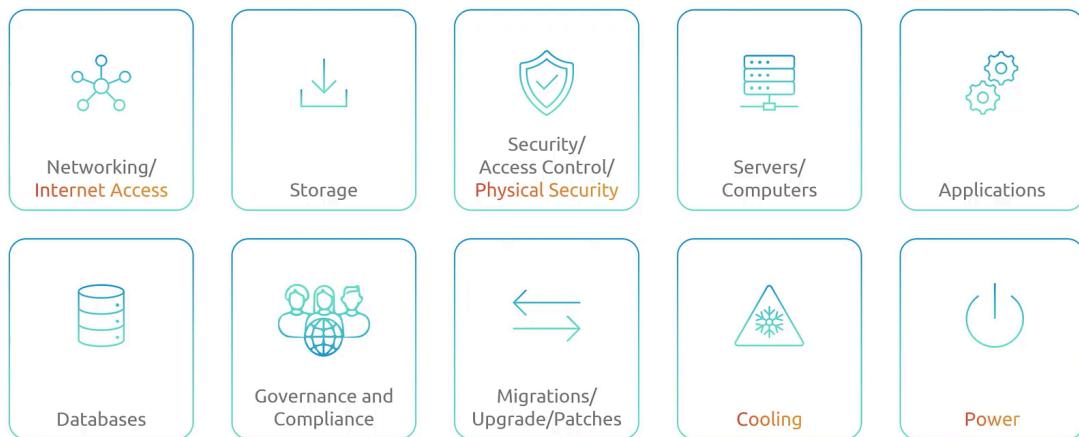


image\_2.png

Here again you will pay for what you request or use.

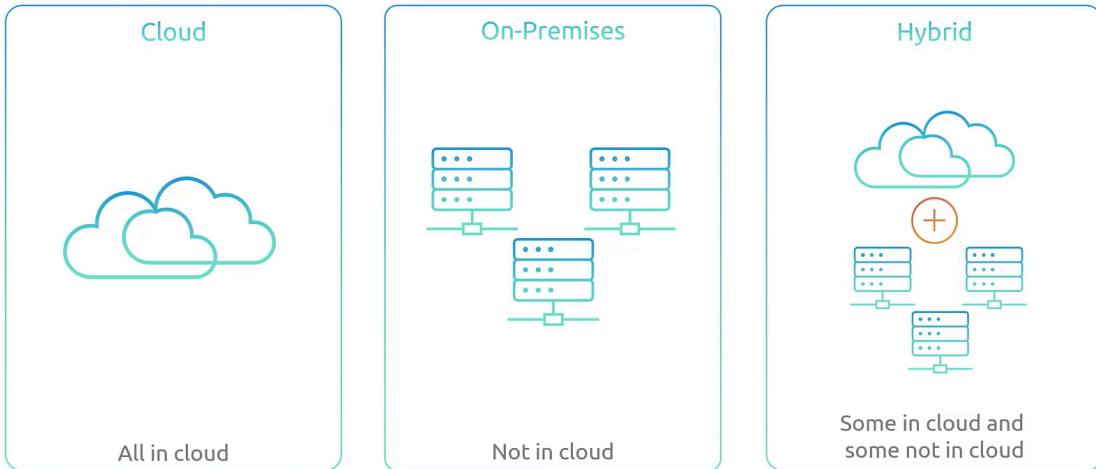
You will access your requested resources in seconds or minutes or even instantly.

How does Cloud Computing Help? All the items in orange are taken care of by the cloud



image\_3.png

## Cloud Computing Models



image\_4.png

## Cloud Model

- Usually includes startup of all sizes.
- Runs everything "in the cloud"
- Shared Responsibility for security and operations

## On Premises Model

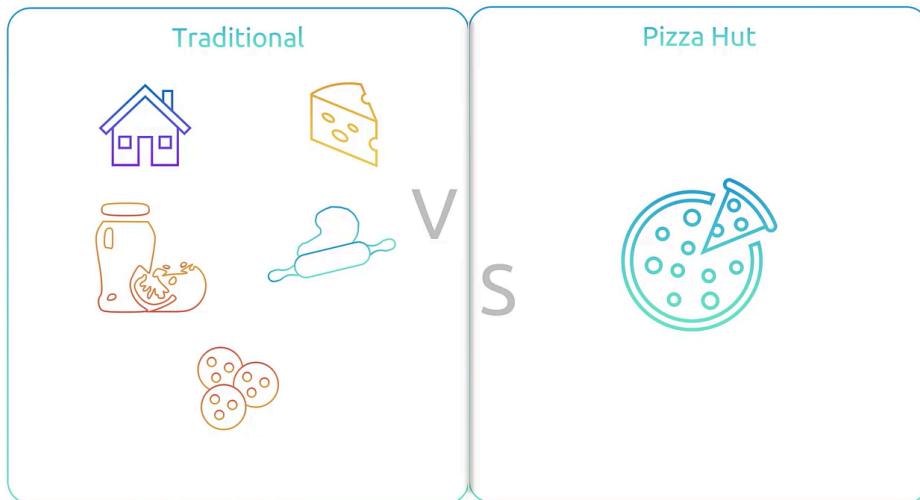
- Minimal to no cloud computing usage
- All projects run in a owned data center or a rented data center
- Responsible for all security and operations
- Can be legacy companies that haven't had a reason to move to the cloud
- Can be companies that need strict control and security over the entire infrastructure e.g. Companies dealing with Government Data .

## Hybrid Model

- Runs some parts of the application in cloud; other parts are on-premise

- Migrates existing applications to the cloud; may leave legacy applications on-premise
- Most new applications are designed and built for the cloud
- Usually fast connection between on-premise and cloud resources

# Amazon Web Services Versus Traditional IT



image\_5.png

Its more of making Pizza from Scratch vs Ordering one.

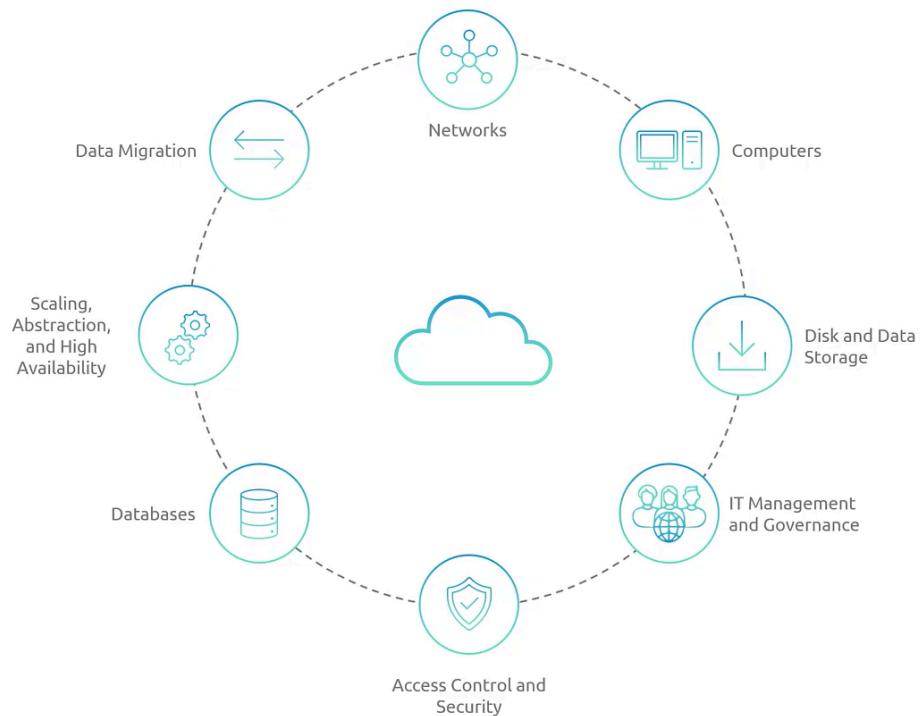
In Traditional Services if you have an idea with new technologies and you are using the traditional way, you have to request for:

- Hardware
- Network Cables
- Power Cable
- Server placement with cooling
- Operational and Security Installation
- Access granted to requestor in 3-6 weeks
- Software installations by requestor

**⚠** But imagine instead of provisioning hardware, you provisioned a hardware service that included networking and power

**⚠** Instead of managing physical objects, you get access to services that you can configure with ease.

Now imagine all this:



image\_6.png

Was Replaced with:



image\_7.png

Now the above is Amazon Web Services.

# Amazon Web Services

AWS has some Code service categories .



Compute



Networking and Content Delivery



Storage



Database



Security, Identity, and Compliance



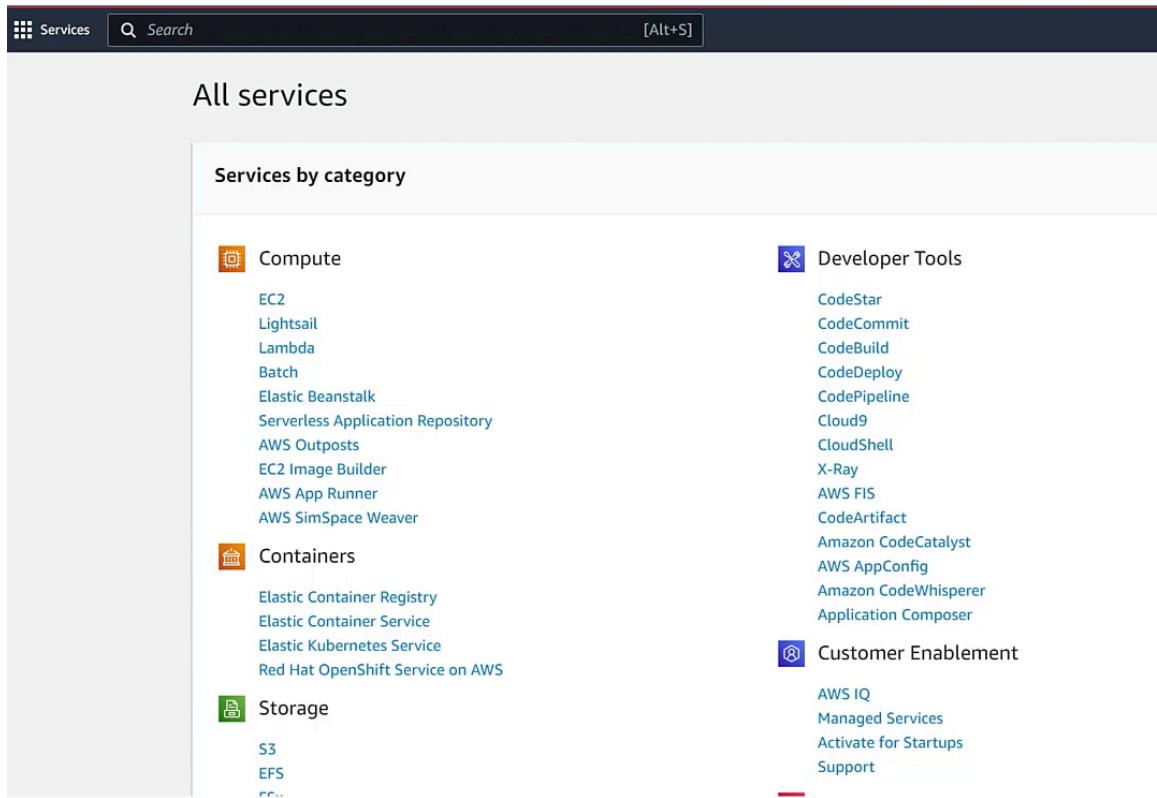
Management and Governance

image\_8.png

## Three Ways to Interact with AWS

### Console

It's Great to Learn and confirm.It more of accessing AWS through a website.



image\_9.png

## AWS command Line Interface (CLI)

It's Great for engineers . Allows you to type command that get and manipulate information in AWS



```
$ aws ec2 describe-instances
$ aws ec2 start-instance --instance-ids i-1348636c
$ aws sns publish --topic-arn arn:aws:sns:us-east-1:546419318123:OperationsError --message "Script Failure"
$ aws sqs receive-message --queue-url https://queue.amazonaws.com/546419318123/Test
```

image\_10.png

## AWS SDK

It's great for developers and different types of coders. SDKs take the complexity out of coding by providing language specific APIs for AWS services. Think of these as Libraries just like any other library e.g Nodemailer in JavaScript



image\_11.png

## AWS Summary

- ✓ Amazon Web Services or AWS was the first large-scale cloud provider
- ✓ AWS was launched in 2006, with S3 as the first service
- ✓ Since then, AWS has grown to 300+ services
- ✓ Signing up is free; all services usually are pay-to-use
- ✓ AWS has one of the largest communities, market positioning, and growth in the industry

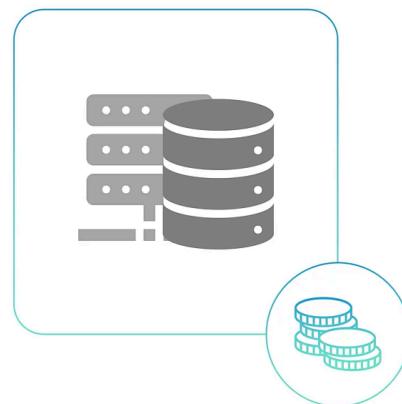
image\_12.png

## Benefits of Clouds

### Trade UpFront Expense for Variable Expense

#### Upfront Expenses (CAPEX)

Large upfront investment in hardware before use

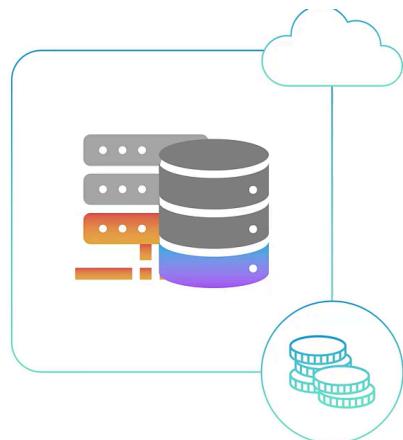


image\_13.png

## Variable Expenses (OPEX)

Pay for usage and requests

Give back what you don't need

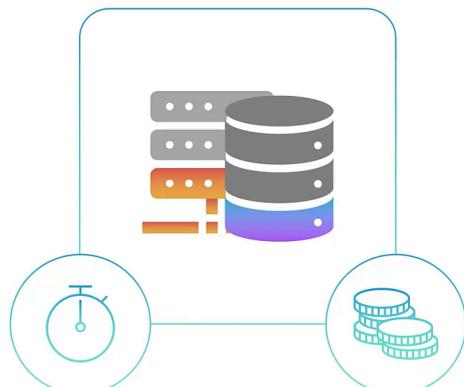


image\_14.png

## Stop focusing on Data Centers

### Focus on Data Centers

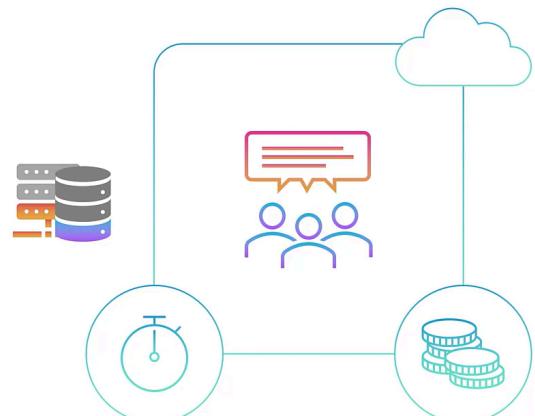
Invest time and money; focus on data centers



image\_15.png

### Focus on Customers

Invest time and money; focus on customers and applications



image\_16.png

## Stop Guessing Capacity ( or easily fix bad guesses)

### Limited Scalability

Bound by hardware limitations or vendor supply limits



image\_17.png

### Extreme Scalability

Scales in and out as needed; experiments with load and performance as needed

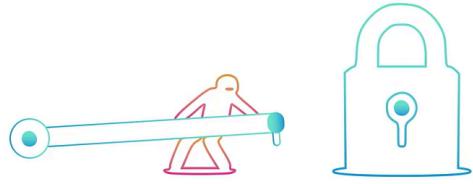


image\_18.png

## Benefit from massive Economies of Scale

### Smaller Scale

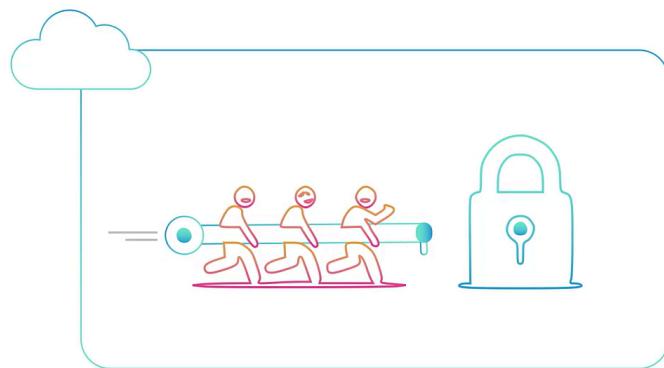
Pay higher prices based on only your own usage, with no price drops per unit



image\_19.png

### Economies of Scale

Benefit from customers' aggregated usage, with price drops per unit used

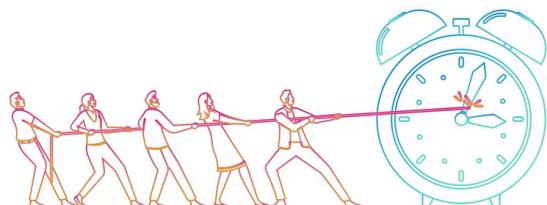


image\_20.png

### Increase provision speed and Business agility

### Data Center Requests

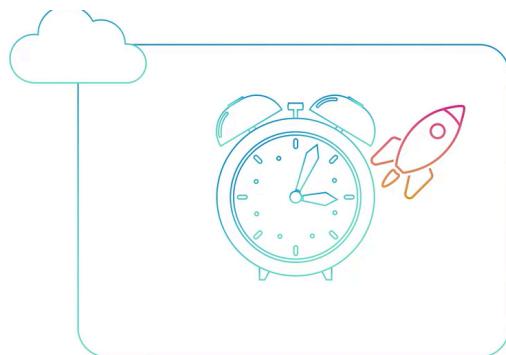
Often provision in days or weeks or months



image\_21.png

### Cloud Computing Requests

Always provision in seconds or minutes



image\_22.png

### Go Global( with your app) in minutes

#### Limited Deployment Options

Limited to places where data centers are available



image\_23.png

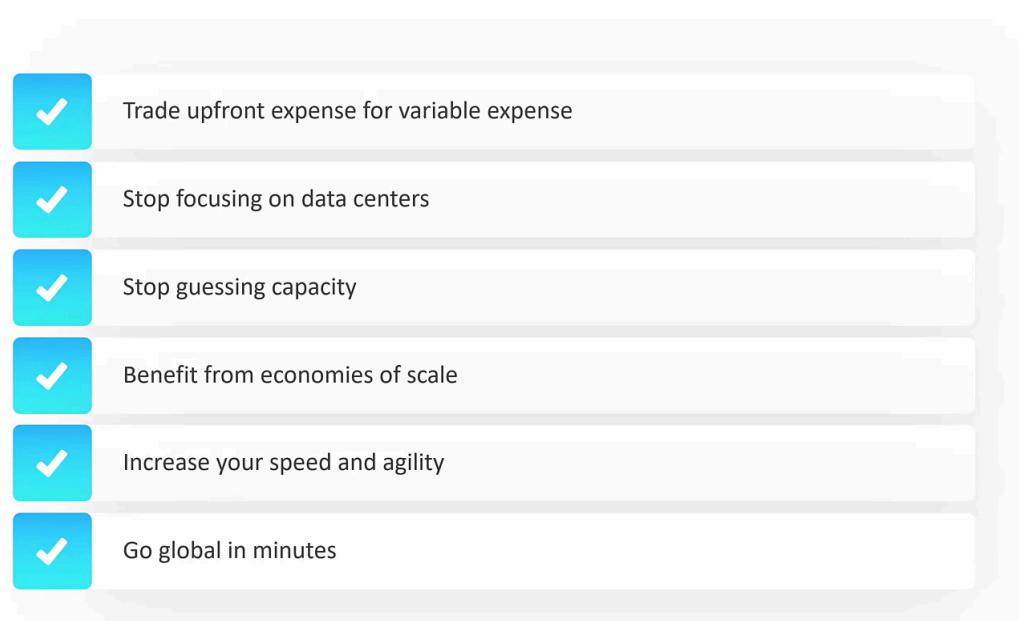
#### Global Deployment Options

AWS global infrastructure available at 31 locations globally



image\_24.png

## Summary



image\_25.png

# Cloud Computing Economics

Free Tier

Volume Discounts

On-demand

Price Drops

Reservations

image\_26.png

## Free Tier

Some services are free forever , some for 12 months after new account creation.



<b>COMPUTE</b>		<b>STORAGE</b>		<b>DATABASE</b>	
Free Tier	12 MONTHS FREE	Free Tier	12 MONTHS FREE	Free Tier	12 MONTHS FREE
<b>Amazon EC2</b>		<b>Amazon S3</b>		<b>Amazon RDS</b>	
<b>750 Hours</b>		<b>5 GB</b>		<b>750 Hours</b>	
per month		of standard storage		per month of database usage (applicable DB engines)	
Resizable compute capacity in the Cloud.		Secure, durable, and scalable object storage infrastructure.		Managed Relational Database Service for MySQL, PostgreSQL, MariaDB, or SQL Server.	
750 hours per month of Linux, RHEL, or SLES	▼	5 GB of Standard Storage	▼	▼	
<b>DATABASE</b>		<b>MACHINE LEARNING</b>	<small>NEW</small>	<b>COMPUTE</b>	
Free Tier	ALWAYS FREE	Free Tier	FREE TRIAL	Free Tier	ALWAYS FREE
<b>Amazon DynamoDB</b>		<b>Amazon SageMaker</b>		<b>AWS Lambda</b>	
<b>25 GB</b>		<b>2 Months</b>		<b>1 Million</b>	
of storage		free trial		free requests per month	
Fast and flexible NoSQL database with seamless scalability.	▼	Machine learning for every data scientist and developer.	▼	Compute service that runs your code in response to events and automatically manages the compute resources.	
25 GB of Storage	▼	250 hours per month of ml.t3.medium on	▼	▼	

image\_27.png

## On-Demand

You pay for the Size of the machine you use. For Example here you will pay for using 2 processors , 16Gb of storage and it ran for 10 hours.

2 Processor  
16 Gigabyte  
10 Hours



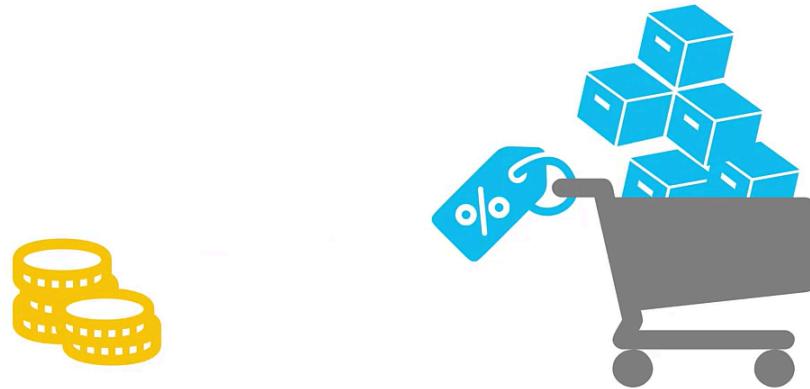
image\_28.png

## Reservation

You make a one year or 3 year commitment to rent a virtual machine.

## Volume Discounts

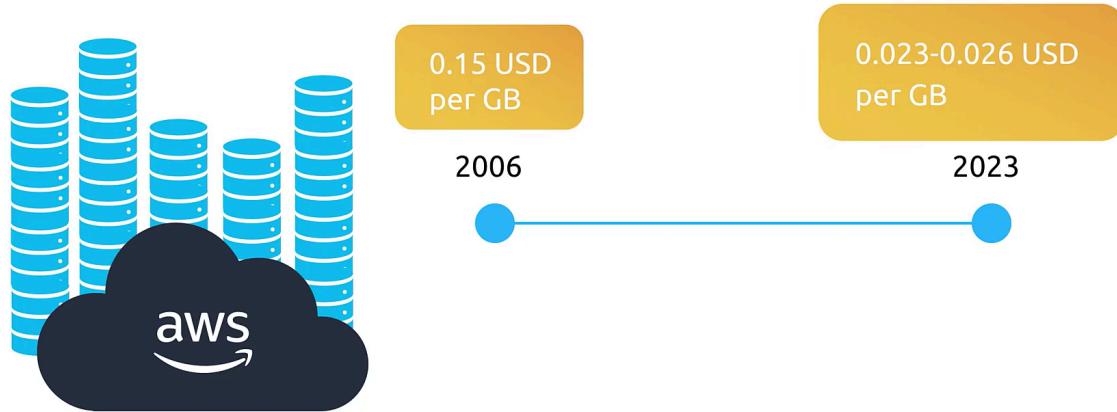
As your per unit consumption goes up amazon will give you discount for spending more. The more you use the less it costs.



image\_29.png

## Price Drops

AWS will drop price as they keep upgrading their resources.



© Copyright KodeKloud

image\_30.png

## Summary

- ✓ Free Tier means that certain services are always free, and some are free for 12 months after new account creation
- ✓ On-Demand is full pricing and pay-as-you-go, but no contract and very flexible
- ✓ Reservations are contracts that you enter with Amazon for 1-3 years
- ✓ Volume Discounts are pay less per unit as you use more
- ✓ Price Drops are random price cuts that AWS does every few years on its services

image\_31.png

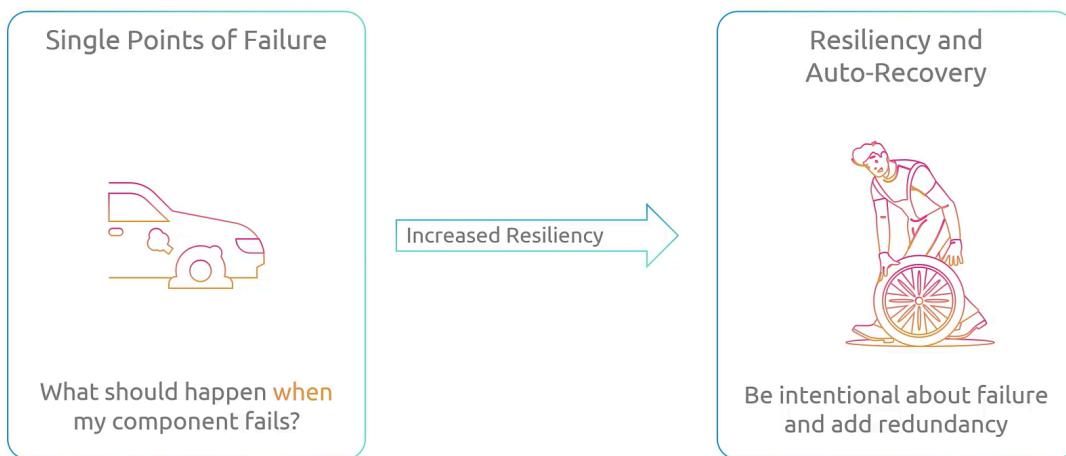
# Principles of Cloud Native for AWS

There are four design principles in AWS :

- Design for failure
- Decouple components
- Implement elasticity
- Think parallel

image\_32.png

## Design for Failure



image\_33.png

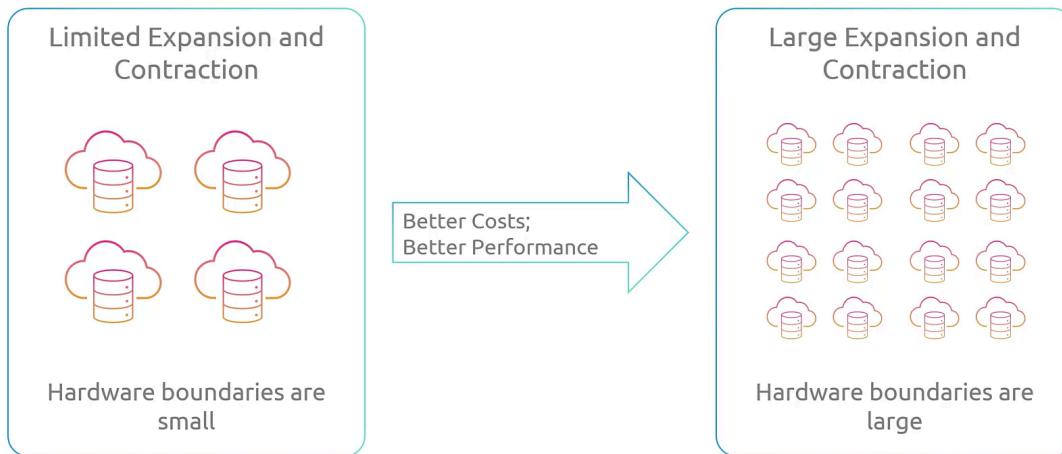
We want to increase resiliency of our applications.

## Tight Coupling



image\_34.png

## Implement Elasticity



image\_35.png

Acquire resource when you need them and release them when you don't.

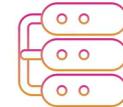
## Think Parallel

Do Things in Series



Do you want 1 server to do it  
in 36 hours?

Do Things in Parallel



Do you want 3 servers to do  
it in 12 hours?

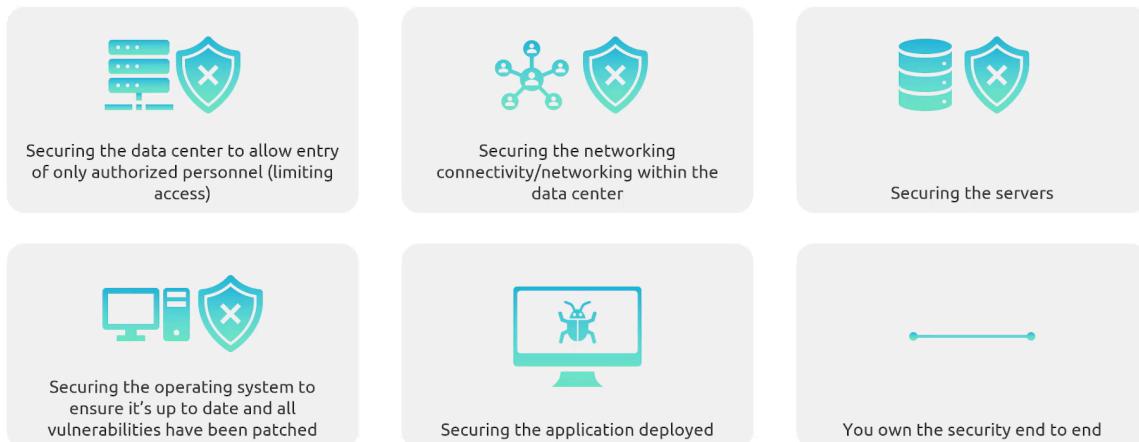
Increased Concurrency →

image\_36.png

# AWS Security

## Security The Traditional Way

The traditional security way you are responsible for:



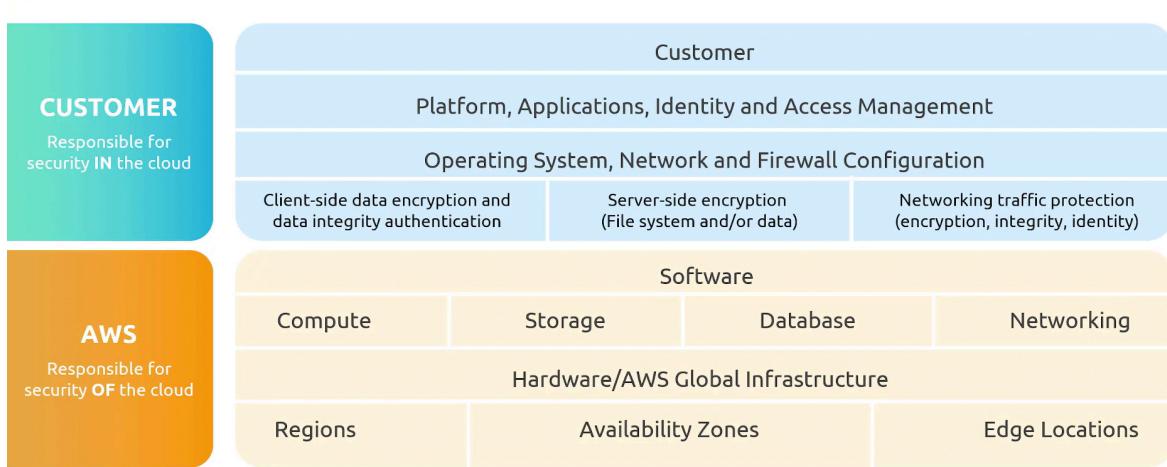
image\_37.png

But Security in the Cloud is a shared responsibility model between:



image\_38.png

## Shared Responsibility Model



image\_39.png

## Infrastructure as a Service(IaaS)

AWS will give you a virtual machine that you can use to run application. An example is Amazon Elastic Compute (Amazon EC2)



image\_40.png

## Platform as a Service (PaaS)

This is where AWS will provide you a platform that you can use to run your code for example. They will set up the operating system and the networking for you and all you need is to just run your code.



image\_41.png

## Software as a Service (SaaS)

Here AWS provides you a software and they own everything except the application data.



image\_42.png

# AWS Security Compliance

## What is Compliance

Organizations in specific industries must adhere to certain rules and guidelines specific to that industry (Finance, Health, Federal Government).

Compliance and regulatory frameworks are sets of guidelines and best practices.

Organizations follow these guidelines to meet regulatory requirements, improve processes, strengthen security, and achieve other business goals

## AWS Compliance

AWS undergoes certification reviews and audits to meet regulatory requirements too.

Audit reports are available on **AWS Artifacts**. Customers can review and accept agreements in the Artifact.

## Customer Compliance Center



image\_43.png

## AWS Audit Manager

AWS Audit Manager continuously collects data to prepare for audits and ensures that you are achieving compliance with regulatory standards. It helps build audit-ready reports.

## AWS Config

It tracks how the resource is configured and records the previous configuration states, so you can see how the configs for it have changed over time

With AWS config it keeps track of all the changes made to a resource . Example if I launch an EC2 instance , then add a security group, then add an EBS volume . AWS config will keep track of all these configurations that im making to the ECS. This will be a great tool for auditing and compliance.

## Short Summary



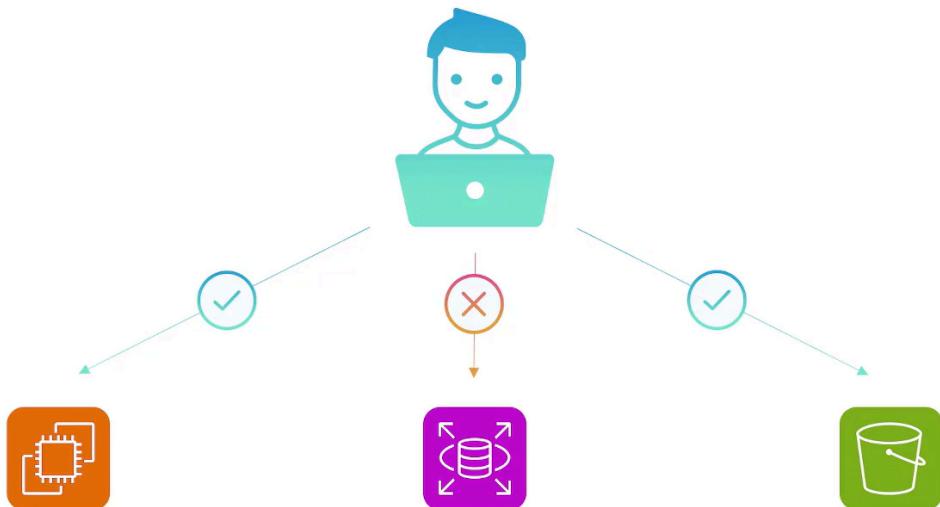
image\_44.png

# Identity and Access Management (IAM)

To create an AWS account you must provide a unique email, account name and a credit card. You can use same credit card for different accounts.

When you create an account you get a Root user, The Root user has access to everything in the account and can perform any action on any resource within the account.

**A** But assuming You have a new employee in your organization and you want to provide access and also maybe restrict access to certain resources.



image\_46.png

This is where IAM comes in.

## Identity and Access Management (IAM)

IAM manages access to AWS resources. IAM handles who is authenticated and what they are authorized to do.

There are three types of Identities:

- Users
- Groups
- Roles

**IAM policies** define what resources a user/group/role has access to and what actions they can perform on them.



image\_47.png

## User

IAM user is just an organization or user who wants to access a set of services within AWS

But what do users have access to?

**A** Users have to be granted explicit permissions to access specific services/resources.

Users are implicitly denied all permissions by default

## Policies

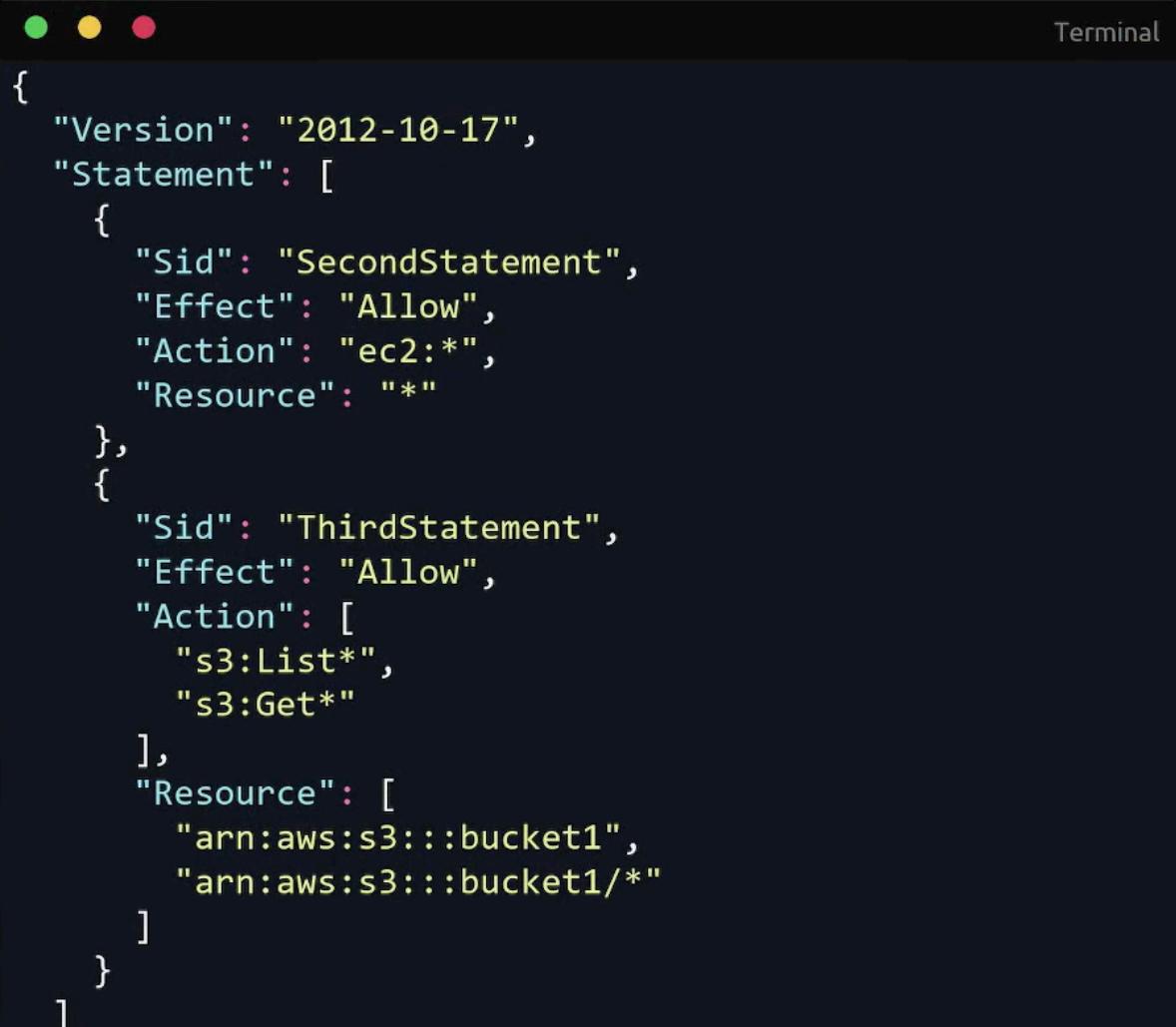
To grant users access to resources, IAM policies need to be applied to the user, giving them permissions.

Policies are documents that either grant or deny access to specific AWS services/resources.



image\_48.png

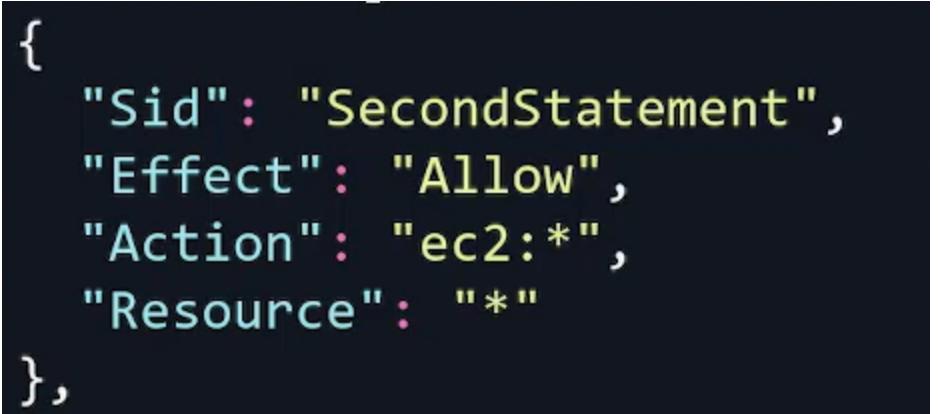
## Policy Example



```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "SecondStatement",  
      "Effect": "Allow",  
      "Action": "ec2:*",  
      "Resource": "*"  
    },  
    {  
      "Sid": "ThirdStatement",  
      "Effect": "Allow",  
      "Action": [  
        "s3>List*",  
        "s3:Get*"  
      ],  
      "Resource": [  
        "arn:aws:s3:::bucket1",  
        "arn:aws:s3:::bucket1/*"  
      ]  
    }  
  ]  
}
```

image\_49.png

This policy has a version which can change based on the recent AWS policy version.  
Statement is what permissions you want to grant access to.



```
{  
  "Sid": "SecondStatement",  
  "Effect": "Allow",  
  "Action": "ec2:*",  
  "Resource": "*"  
},
```

image\_50.png

The above policy has a unique id and allows a user to access to all EC2 resources.

```
{  
  "Sid": "ThirdStatement",  
  "Effect": "Allow",  
  "Action": [  
    "s3>List*",  
    "s3:Get*"  
,  
  "Resource": [  
    "arn:aws:s3:::bucket1",  
    "arn:aws:s3:::bucket1/*"  
]
```

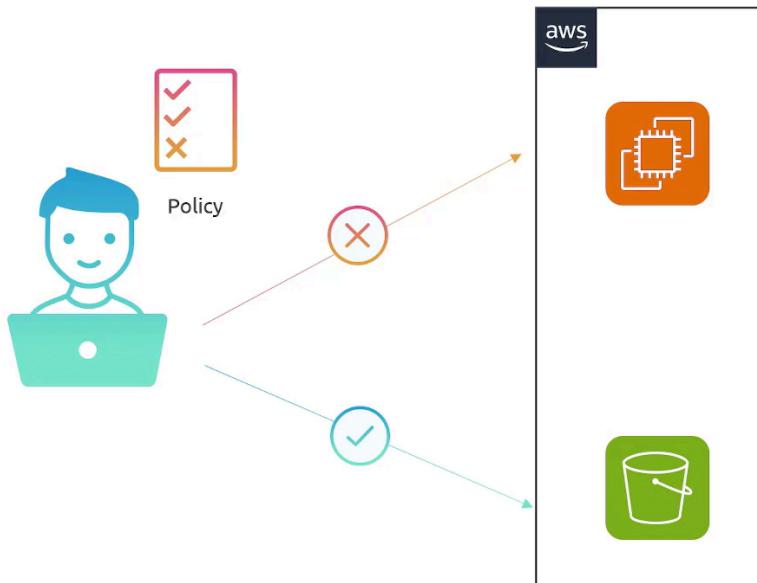
image\_51.png

The second permission allows a user to list and get all objects in a S3 bucket

**⚠** Now we have a user and policies

## Users and Policies

We now attach a policy to a user since by default they don't have access to anything



image\_52.png

## Least Privilege Permission

Identities/ users should only be granted the minimum permissions necessary for the to perform their job.

**⚠** For every new user created you have to give them certain permissions and you have to remember to always give them permission.

If I have a bunch of users that require same permission we can create groups.

## Groups

Groups are a collection of IAM users.

- Policies can be added to groups
- IAM users within a group automatically inherit all the policies from the group
- Example – having a separate group for each department

## Roles

Roles allow users, applications, or services to assume temporary permissions.

- Roles are assigned permissions/policies just like users and groups
- When someone assumes an IAM role, they inherit the permission of the role temporarily and return to their original permission when done

## Multi-Factor Authentication (MFA)

To login to an AWS root account you must provide an Email and a Password. If they leak or hacked then other people will have access to your account and might do malicious acts.

To prevent this we can create an Extra security , a code that will be generated on an MFA device/app which changes periodically. This code will then be sent to AWS to verify the user. So they can have the email and password but if they dont have a code then they cant access your account.

# Organizations

Organizations help manage multiple AWS accounts Organizational units (OUs) allow you to group accounts with similar business or security requirements

Service Control Policies (SCPs) restrict what an account can do

- SCPs can be applied to individual accounts or OUs
- When applied to OUs, all AWS accounts within the OU inherit the policies

# AWS Security Resources

## Prevention Security Resources

### Web Application Firewall (WAF)

Monitors HTTP requests and prevents a variety of attacks like SQL injection and XSS attacks.

### AWS Shield

AWS Shield detects and mitigates sophisticated distributed denial of service (DDoS) attacks. A DDoS attack is when a perpetrator seeks to make a machine or network resource unavailable to its intended users by temporarily or indefinitely disrupting the services of a host connected to a network.

- Denial of service is typically accomplished by flooding the targeted machine or resource with fake requests to overload systems and prevent some or all legitimate requests from being fulfilled.

### AWS Network Firewall

It is a stateful managed firewall and intrusion detection and prevention service for VPCs. It monitors traffic going into and out of a VPC.

## Detection Security Resources

AWS Inspector AWS inspector scans workloads running on AWS for software vulnerabilities and undesired network exposure.

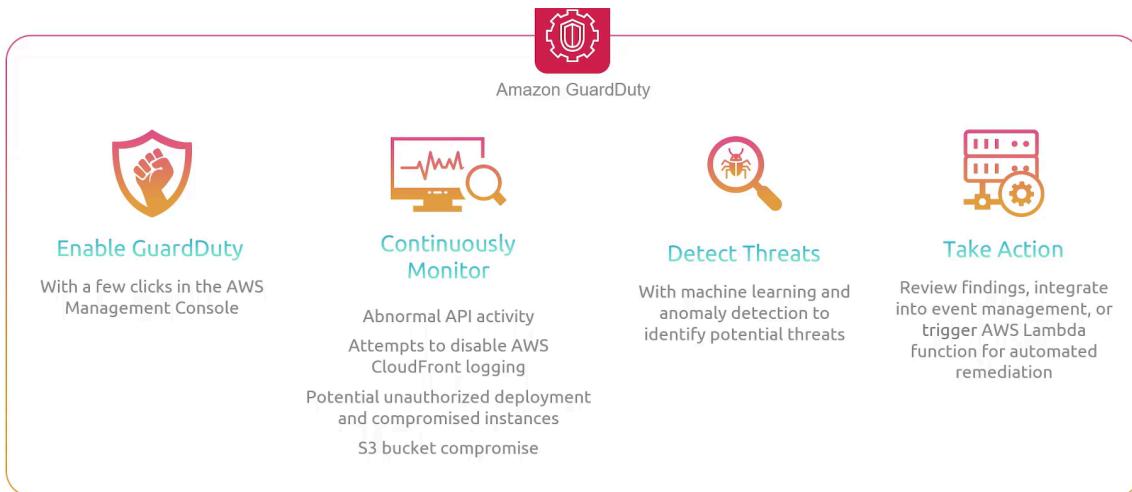
- It automatically discovers and scans EC2 instances, container images in ECR, and Lambda function.
- It continues to assess your environment throughout the lifecycle of your resources by automatically rescanning resources in response to changes that could introduce new vulnerabilities.

### 1. Installing new package

2. Installing a patch

3. New common vulnerabilities and Exposures (CVE) are released • If vulnerabilities are identified, AWS Inspector produces a finding that you can investigate.

## Amazon GuardDuty



image\_53.png

## Amazon Detective

Amazon Detective helps you quickly analyze and investigate security events across one or more AWS accounts.

1. It ingests data from VPC flow logs, CloudTrail logs, and GuardDuty findings.

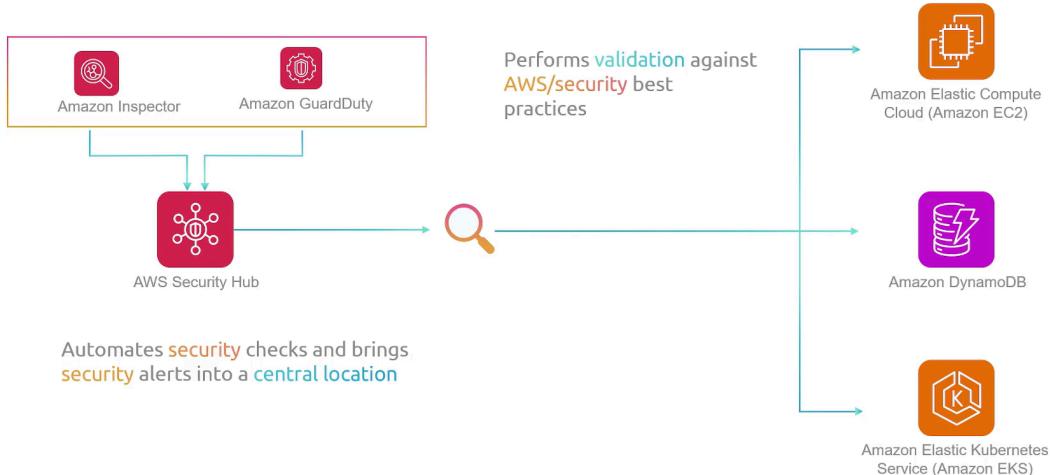
2. It uses machine learning and statistical analysis to create advanced visualizations that show resource behaviors and interactions over time. Amazon Detective simplifies the process of investigating security incidents by providing interactive dashboards and visualizations.

## AWS Config

It tracks how the resource is configured and records the previous configuration states, so you can see how the configs for it have changed over time

AWS Security Hub It automates security checks and brings security alerts into a central location:

- Ingests data from other services like AWS Config, GuardDuty, Firewall Manager, Inspector, etc.
- Performs validation against AWS/Security best practices



- image\_54.png

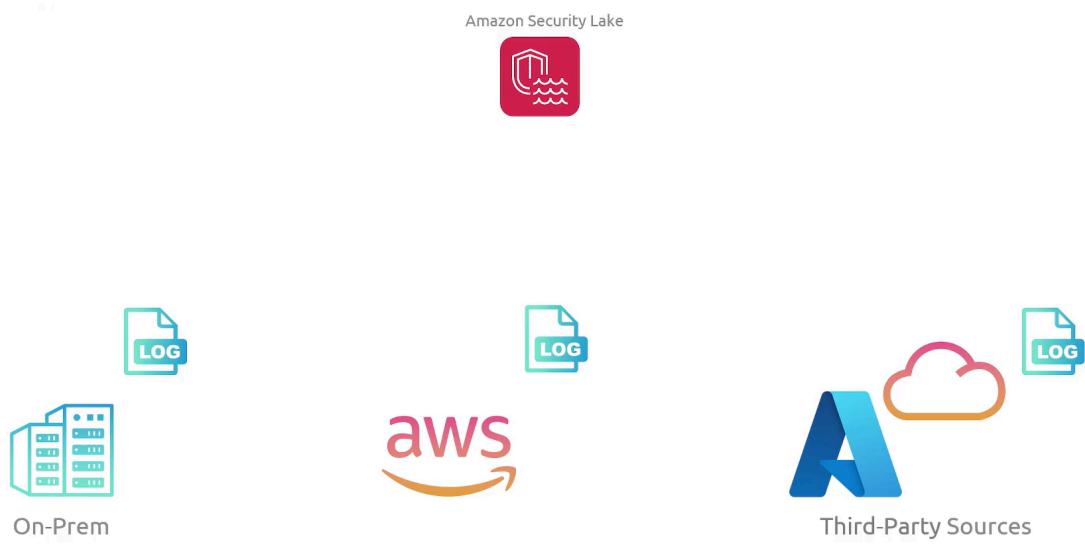
## CloudTrail

CloudTrail tracks user activity with an AWS account.

- It tells us who is doing what.
- Anytime a user performs an action within AWS, an event is logged in CloudTrail, such as:
  1. Logging in
  2. Modifying security policies
  3. Accessing/creating/deleting a service
- It logs actions taken in the console, command line, and AWS SDK/APIs.

## Security Lake

It collects security logs and events from multiple sources including on-premises, AWS services, and third-party services. It transforms the data into storage and query-efficient Parquet format



image\_55.png

## AWS Macie

AWS Macie uses pattern matching and machine learning to automatically discover sensitive data. Macie will generate an inventory report of S3 buckets and scan objects for sensitive data and notify you.

## Detection Summary

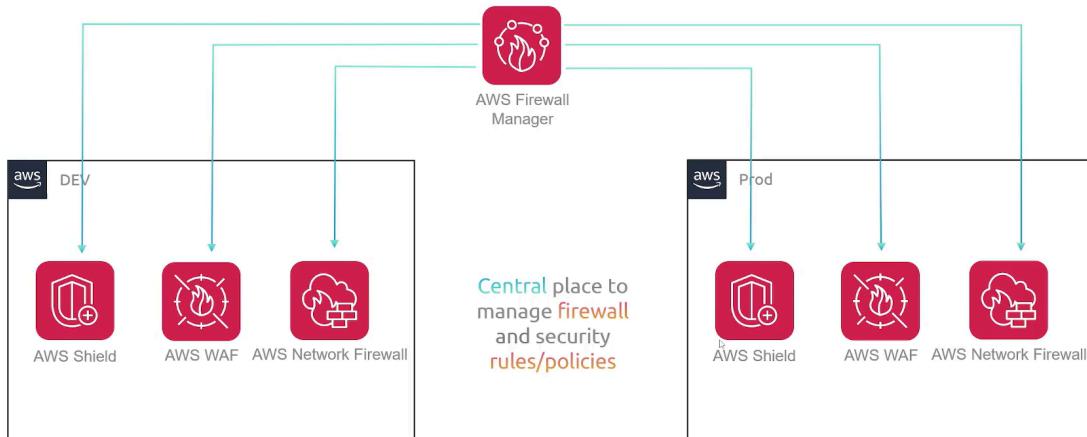
-  GuardDuty monitors and detects suspicious activity and potential threats in your AWS environment
-  Detective helps analyze and investigate security-related events by collecting and visualizing data
-  CloudTrail logs and monitors all user and API activity within an AWS account
-  AWS Config tracks and audits the configuration of AWS resources over time
-  Security Hub automates security checks and brings alerts to a central location. Also performs validation on AWS best practices
-  Security Lake collects logs from a variety of locations and transforms them into a query-efficient format
-  AWS Macie scans S3 buckets for sensitive data and notifies users of findings

image\_56.png

## Management Security Resources

### Firewall Manager

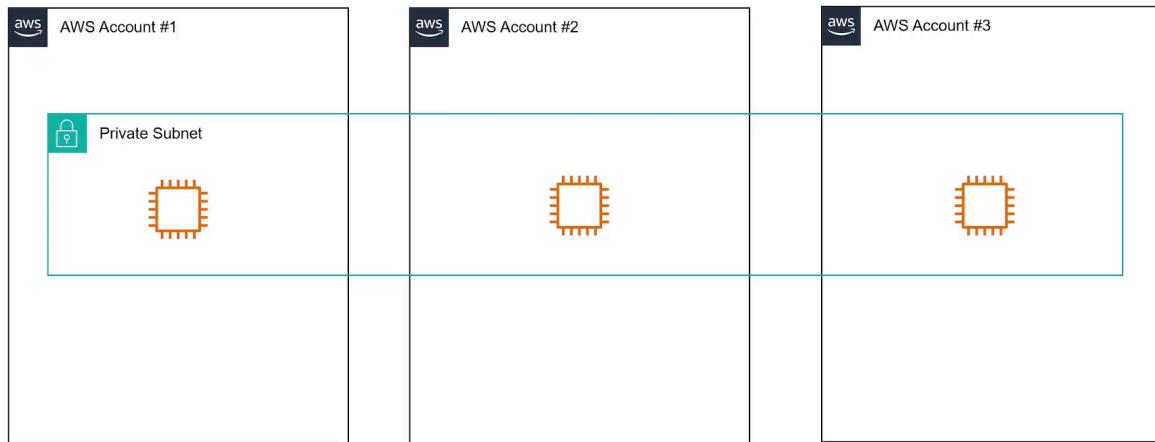
It helps manage security services (WAF, security groups, network firewall) across multiple AWS accounts. It configures rules just once and has them available across all accounts.



image\_57.png

## Resource Access Manager

It helps you securely share resources across accounts, organizations, and OUs. So, you can create a resource once and have the Resource Access Manager make that resource usable by other accounts.



image\_58.png

## AWS Cognito

AWS Cognito helps implement customer identity and access management for mobile and web applications. • It manages all user credentials. • It makes adding signup/login functionality easy without social authentication on any app.

## IAM Identity Center

It simplifies managing users across multiple AWS accounts. It also allows you to manage sign-in security for your users centrally and grant them access across all AWS accounts and resources.

## Secrets Manager

The Secrets Manager helps retrieve and rotate secrets and other sensitive data like credentials and Auth tokens.

- No need to hard code credentials in application source code
- Makes it more difficult to leak or compromise sensitive credentials
- Can configure automatic rotation schedule for your secrets

- Having short-term secrets decreases the risk of compromise

## AWS Certificate Manager (ACM)

It handles the complexity of creating, storing, and renewing public and private SSL/TLS certificates and keys that protect AWS websites and applications.

- If users need to communicate with an ELB or API gateway, you now no longer need to purchase a paid certificate from a third party. -ACM can generate them for you, and end-user traffic will be encrypted.
- You can issue certificates directly from ACM or import third-party certificates.

## AWS Private Certificate Authority

It provides users with a private CA that is managed by AWS.

- Issues certificates for authenticating internal users, computers, and applications
- Certificates issued by a private CA are trusted only within your organization and not on the internet
- Saves the hassle of having to set up, configure, and maintain your own internal certificate authority

## Key Management Service (KMS)

KMS helps create and manage cryptographic keys used for encrypting/decrypting data.

- It allows you to provide granular control over who has access to the keys
- Key rotation can be configured

## CloudHSM

CloudHSM provides customers with a hardware security module in the cloud.

- All cryptographic keys are securely stored on the HSM, and they never exit the device
- Data is sent from the clients to the HSM for encryption/decryption

- Because all keys are securely stored on the HSM in a central location, it helps minimize the risk of keys getting leaked or stolen

# Global Infrastructure

## region

An AWS region is a geographic location where resources can be deployed. Each region is designed to be isolated from each other for:

- Fault tolerance
- Stability



But all the regions are not the same

## Selecting a Region

- Not all services are available in all regions.
- GovCloud regions designed for U.S. Government and customers who need to meet regulatory and compliance requirements.
- You should select a region as close to your customers as possible to minimise latency.
- Cost of services varies from region to region
- Due to compliance and legal requirement , you may need to run your infrastructure in certain regions.

## Availability Zones (AZ)

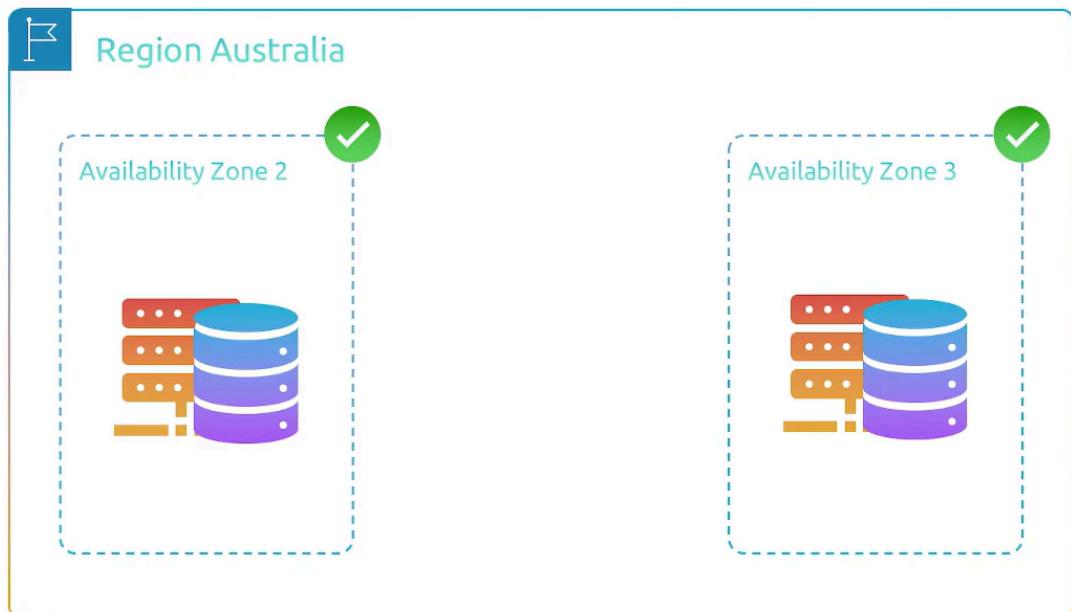
Within a region there are multiple Availability zones. An Availability zone is one or more discrete data centers with redundant power, networking, and connectivity in an AWS region. Availability zones provide redundancy within a region.

If you have an **application** deployed in only one **AZ**, then the entire **region** is affected in case of a failure in that **AZ**



image\_59.png

But instead we can have the application in multiple availability zones so that if one is down , another can cover it up.



image\_60.png

## Global Content Delivery and Edge Locations



image\_61.png

If we have a web server in N.America and a customer from the same country is accessing a resource then this will be faster but if the customer is accessing from India then that's a longer distance and we might experience latency.

AWS noticed this and Solved it using Edge Locations:



image\_62.png

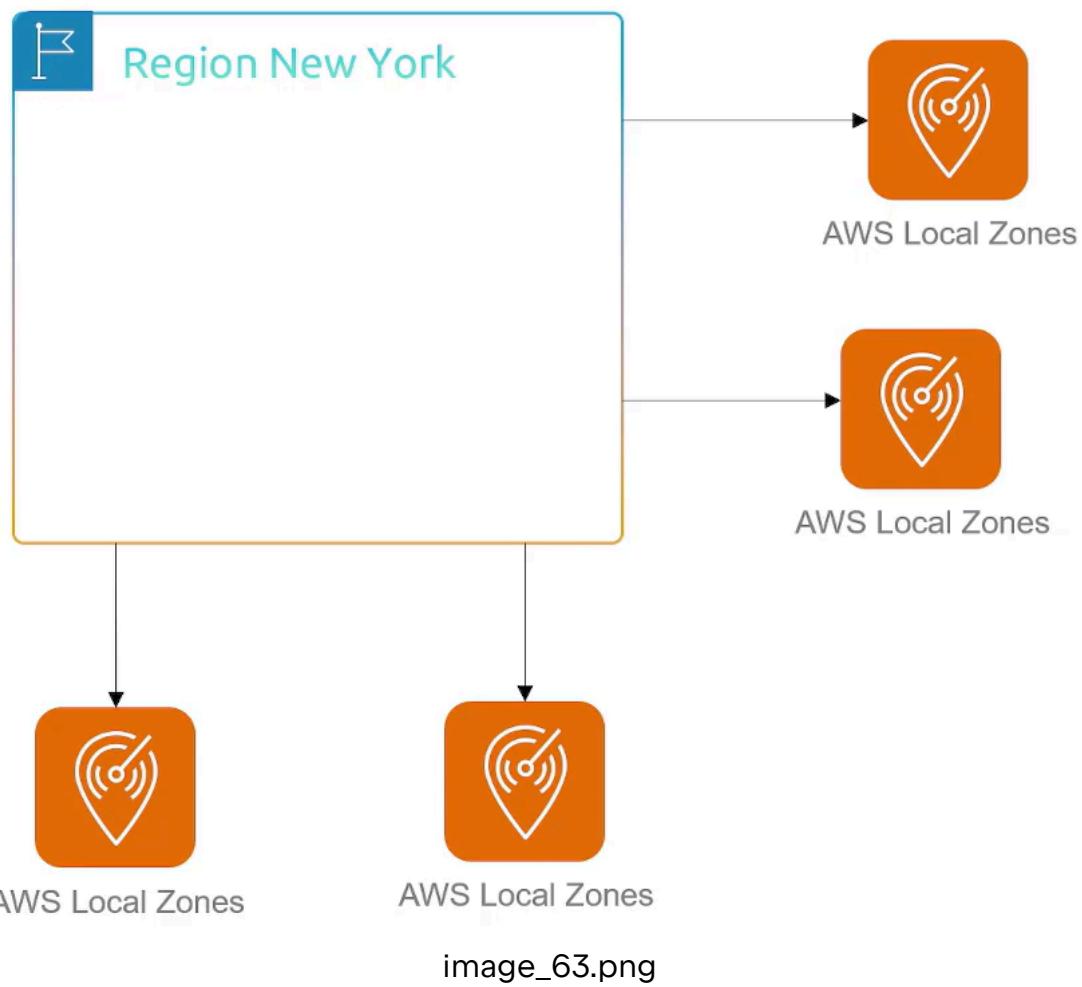
## Edge Locations

AWS Edge Locations are strategically positioned points in the AWS network optimized for low-latency content delivery, ensuring that data reaches users swiftly. In contrast, AWS regions are larger geographic areas with multiple Availability Zones.

Now instead of the request being sent to N.America, Its going to be sent to an edge location near India hence low-latency.

## Local Zones

Local zones are extension of an AWS Region. local zones allow you to use select AWS services like compute, and storage closer to end-user. Local Zones provide high-bandwidth , secure connection to parent AWS region. Great for low latency applications like real-time gaming, live-streaming and virtual workstations for the local city.



## Edge Locations vs Local Zones



Edge location

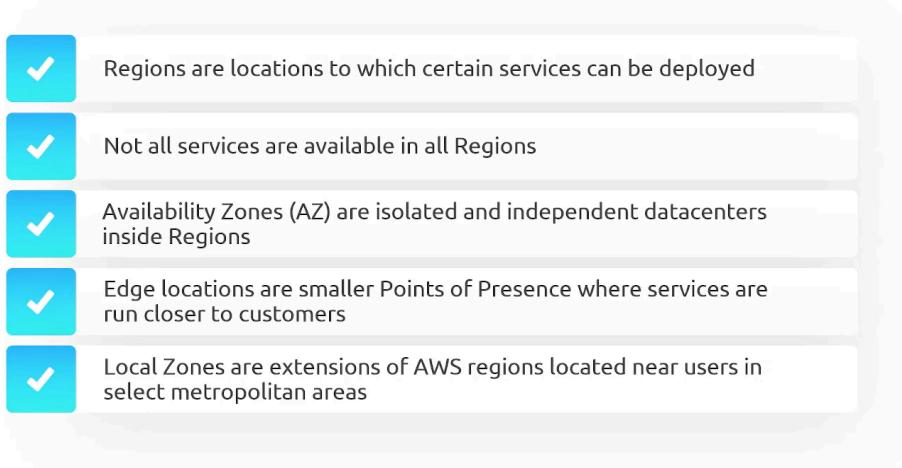


AWS Local Zones

- Both help get **services** closer to **end-users** to improve user experience and minimize **latency**
- **Local Zones** are extensions of a **regions** physically located in **major cities** across the world
- **Local Zones** have their own isolated infrastructure but are connected to the parent AWS region through a high bandwidth network link
- **Local Zones** Provides access to subset of **services** like **EC2 & EBS**
- **Edge Locations** are small geographically dispersed compute sites that primarily support services like **CloudFront**(content delivery network), **Route 53**, **AWS WAF**
- **Edge Locations** are limited to mostly **CDN** services and do not provide full **compute** services
- There are hundreds of **Edge Locations** as they are not full sized datacenters; there are over 20+ **local zones**

image\_64.png

## Short Summary



image\_65.png

# AWS Networking

In AWS you rent a slice of a machine because renting the whole machine. That means that one machine can have several customer applications. But does that mean that they will be able to communicate with one another?

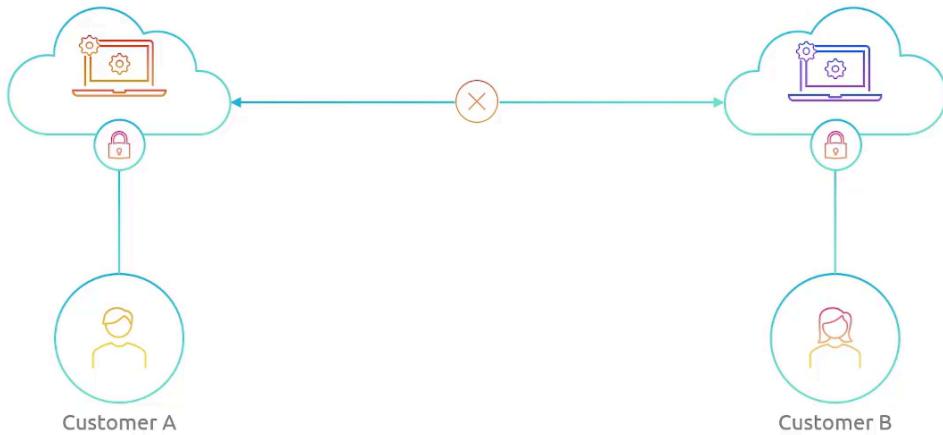


image\_66.png

Amazon thought about this and came up with VPC

## VPC

Every customer will have their application running on a private cloud which can't communicate with another private cloud.



image\_67.png

Virtual private cloud(VPC) is a secure, isolated network segment hosted within AWS. VPC isolates computing resources from other computing resources It gives the customer full control of the networking in the cloud and provides:

- Sub-netting
- Routing( Routes Tables)
- Firewalls (NACLs and Security Groups)
- Gateways

But does it mean every customer gets one VPC?

No you can create multiple VPC and you can create a VPC for different environments or applications.

## VPC and Regions

A VPC is specific to a single region

## Subnets

Subnets are a group of IP addresses in your VPC. For example if you deploy a website on an EC2 it means that the application will be running on one of the IP address of a subnet.

A subnet also resides within a single availability zone



image\_68.png

A Subnet can either be public or private to allow external access to resources within them.

A private subnet means that the subnet can't talk to the internet and also the internet can't talk to the subnet. It's good for something like a database

A Public subnet on the other side means that the subnet can talk to the internet and you can also access it from the internet.

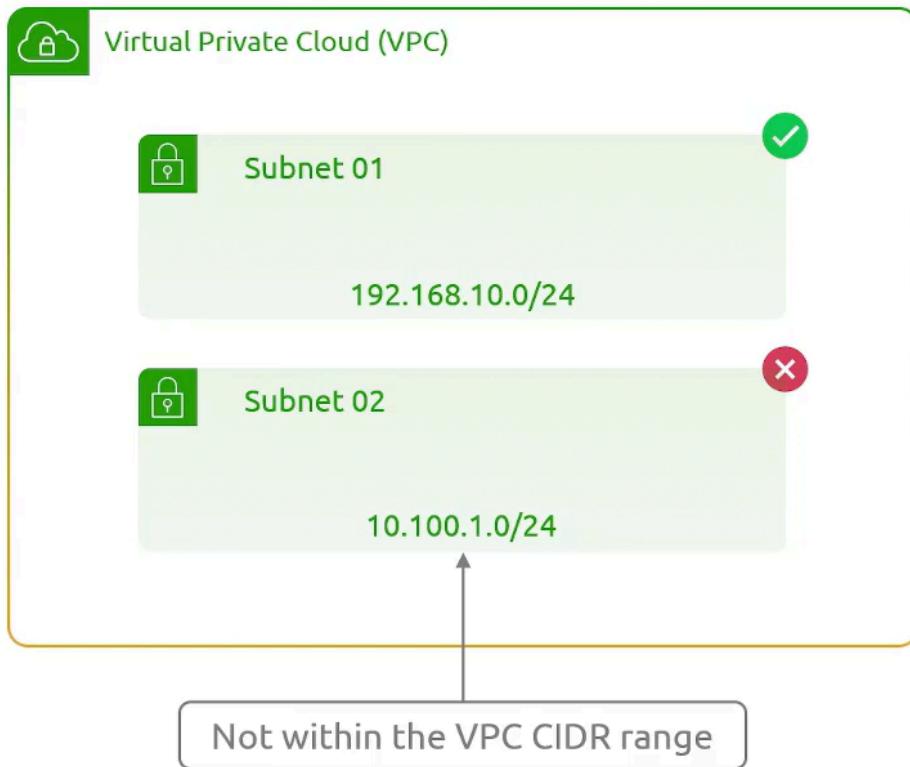
Every VPC has a range of IP addresses assigned to it called the CIDR block.

A CIDR block defines the IP addresses that resources in the VPC can use

A CIDR block size can be anywhere from /16 to a /28

e.g. !92.168.0.0/16 : Will give you a range from 192.168.0.0 - 192.168.255.255 Subnet within a VPC must be within the CIDR range:

CIDR = 192.168.0.0/16



image\_70.png

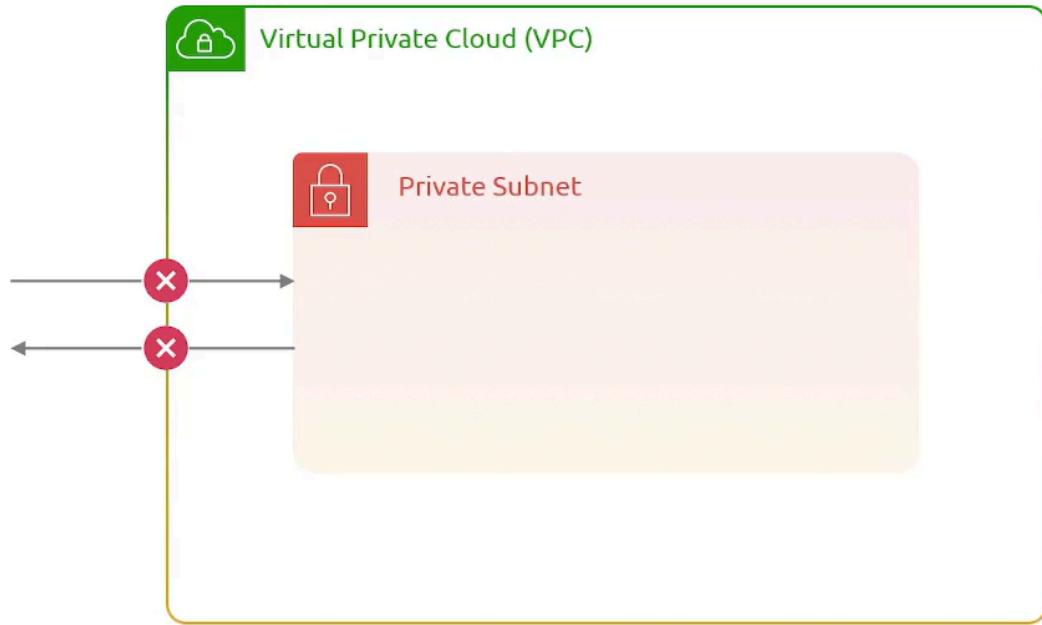
The first 4 IP addresses of a subnet are reserved and cannot be used.

- 192.168.10.0 : Network address
- 192.168.10.1 - 192.168.10.3 : Reserved for AWS
- The Last IP address of a subnet is reserved as the broadcast address : 192.168.10.255
- The first address ready for use should be 192.168.10.4

## VPC External Connectivity

### Internet Gateway

Subnet when first created are private and nor exposed to the internet.



image\_71.png

Then if private it means that the resources running inside it cannot be accessed from the internet.

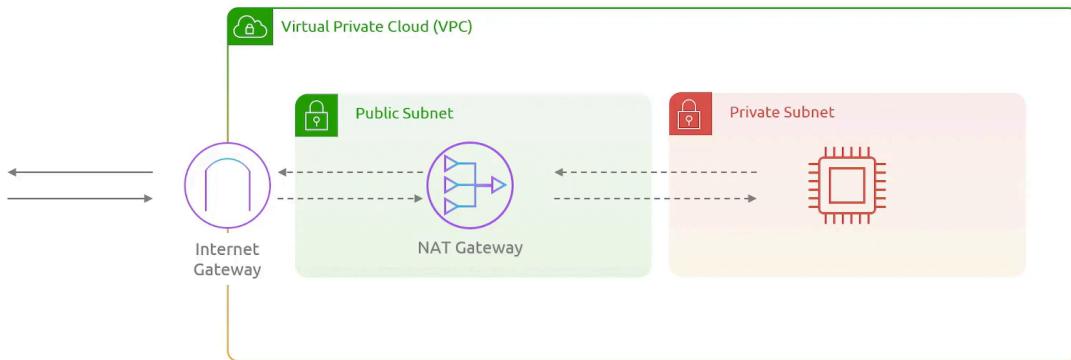
To make a private subnet public an Internet gateway is attached to a VPC to allow subnet in a VPC to communicate with the internet and vice versa.

The ability to access an internet gateway is what determines whether a subnet is public or private.

## NAT Gateway

With NAT Gateway a connection must be initiated from within the VPC, if the connection is from the internet its going to be blocked unlike internet gateway.

This means that a private subnet can use the NAT Gateway for Communication with the internet.



image\_72.png

To connect to a private subnet you use a Virtual Network Gateway or Direct connect.

## Default VPCs

There are two types of VPC

- Default
- Custom

### Default VPC

Every account has a default VPC in each region.

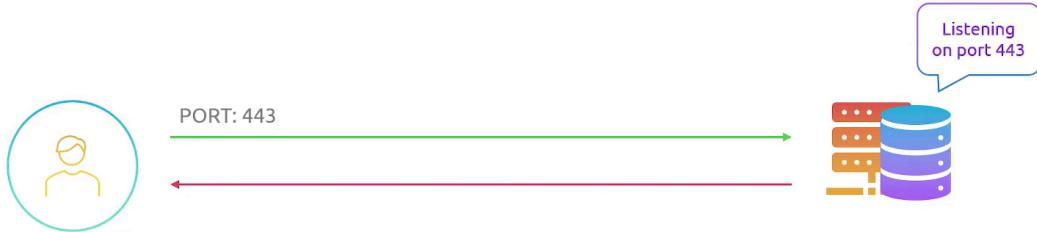
- Default VPCs have configurations defined by AWS.
- Default VPC uses CIDR block 172.31.0.0/16 (65,536 addresses)
- In each availability zone we will have a subnet with a /20 ( 4096 addresses)
- An Internet gateway is attached to the VPC
- A route that points all traffic (0.0.0.0/0) to the **internet gateway**
- All the subnets are going to be public
- A Default security group is also created

- A Default network access control list too.

## **Custom VPC**

Custom VPC allow you to create.modify all configurations associated with the VPC.

# Firewalls



image\_73.png

If your application is running on port 443 then we need to make sure that we only allow traffic from that port and not from other ports. That's where firewalls come in.

Firewalls monitor traffic and only allow traffic permitted by a set of predefined rules.

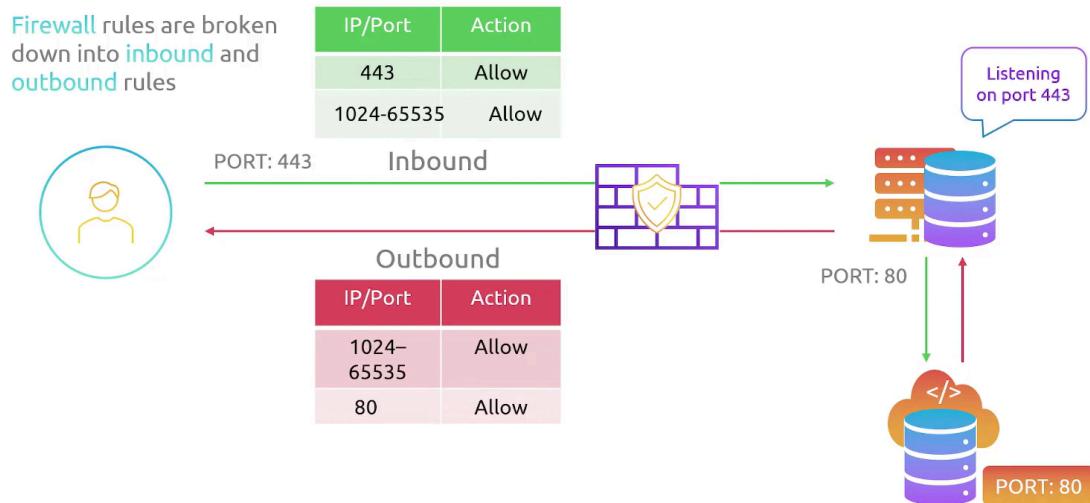
Firewalls rules are broken down into inbound and outbound rules.

**Inbound rules** - are a set of rules that defines what traffic is allowed from the internet into the resources protected by the firewall

**Outbound rules** - are a set of rules that define the traffic allowed to the internet

## Stateless Firewalls

A Stateless firewall is configured to allow both inbound & outbound traffic.

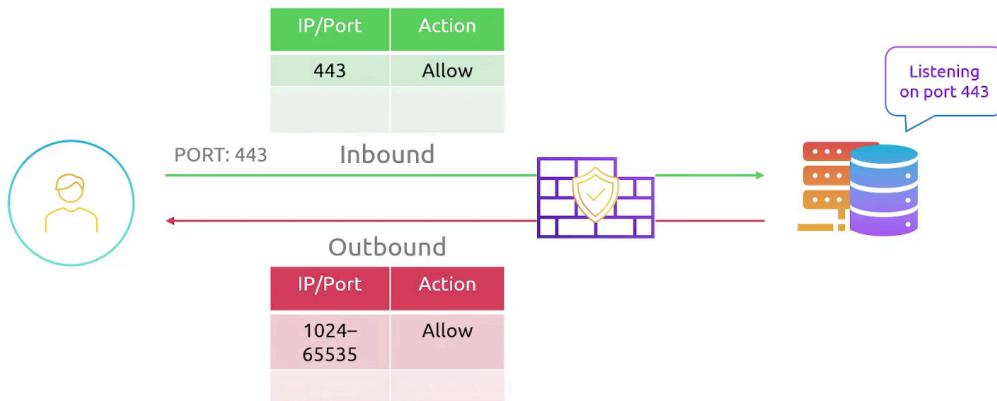


image\_74.png

## Stateful Firewalls

Stateful firewalls are intelligent enough to understand which request and response are part of the same direction.

If a request is permitted the response is automatically permitted as well in a stateful firewall.



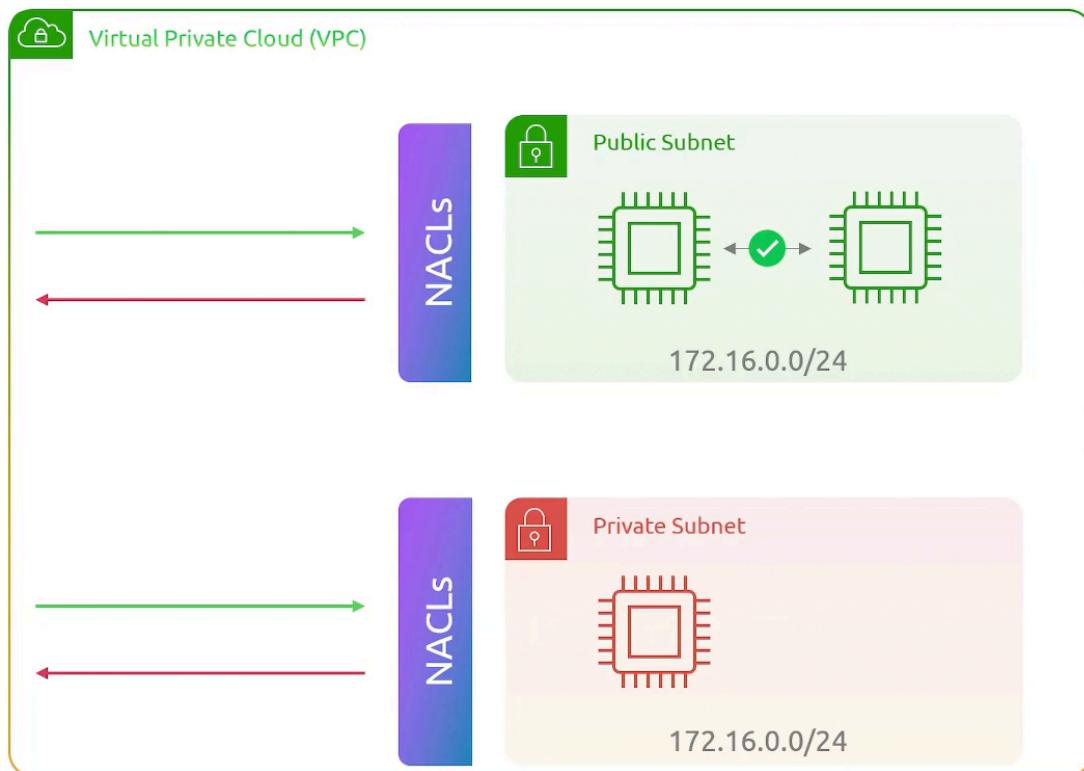
image\_75.png

## Network Access Control List (NACL)

NACLs filter traffic entering and leaving a subnet.

NACLs do not filter traffic within a subnet ( If you have two server on one subnet they can talk to each other)

NACs are stateless firewalls so rules must be set for both inbound and outbound traffic.



image\_76.png

## Security Groups

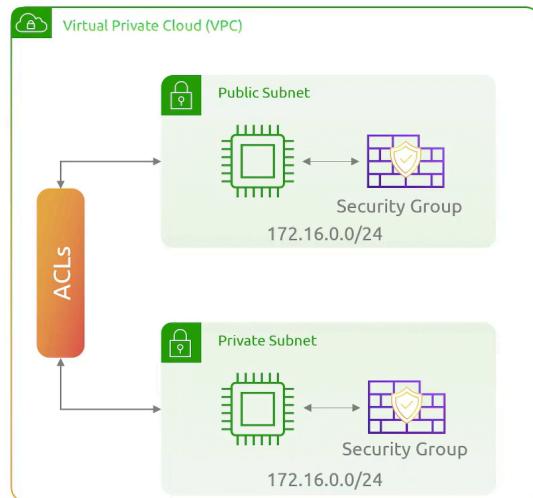
Security groups acts as a firewall for individual resources (EC2, LB, RDS)

Security groups are stateful so only the request needs to be allowed.

## NACLS VS Security Groups

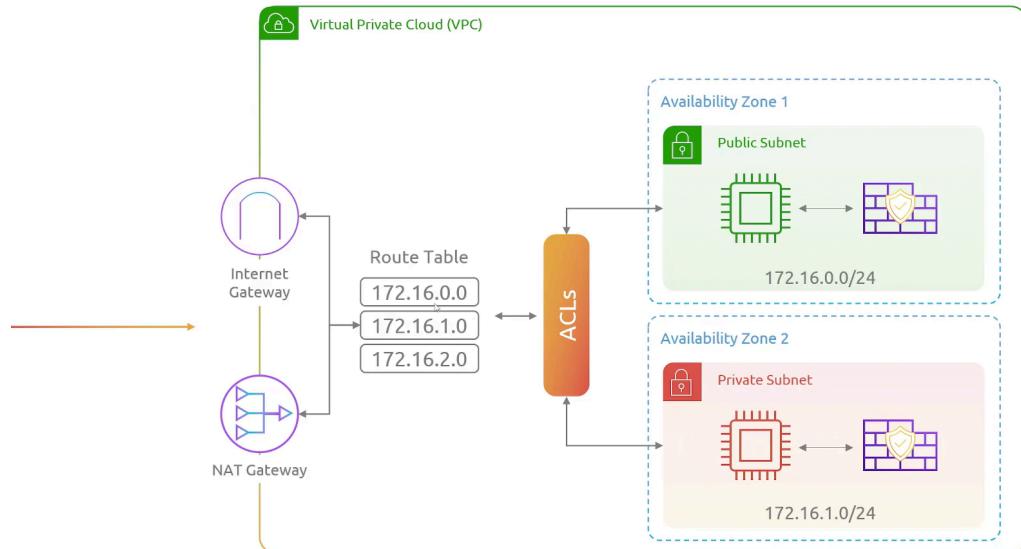
## NACLs vs Security Groups

- **NACLs** monitor traffic entering and leaving a **subnet**
- **NACLs** are stateless **firewalls**
  - Traffic must be permitted in both **ingress** and **egress** directions
- **Security groups** act as personal **firewalls** for individual **resources**
- **Security groups** are stateful
  - Only the direction of the **request** needs to be permitted
  - The **response** will automatically be permitted as well



image\_77.png

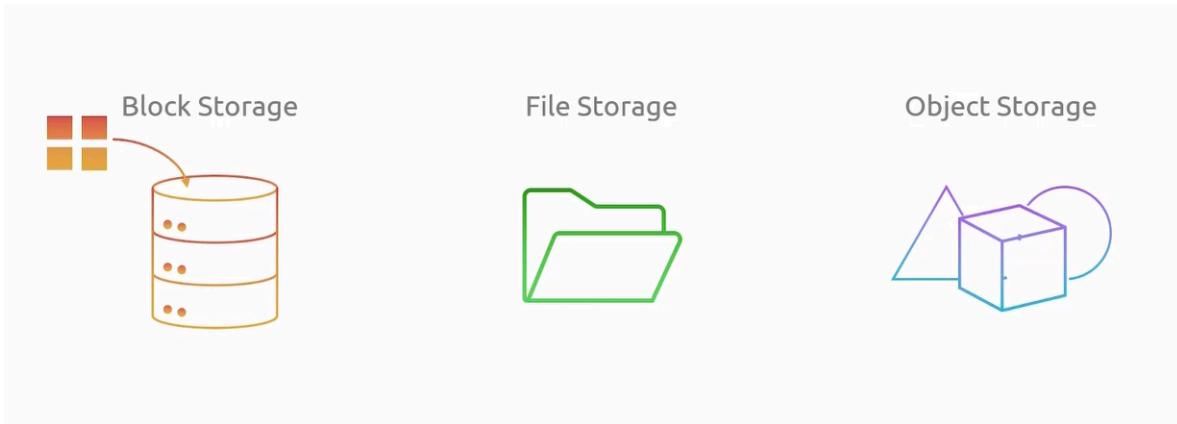
Her is like the whole picture of Networking



image\_78.png

# AWS Storage

## Types of Storage

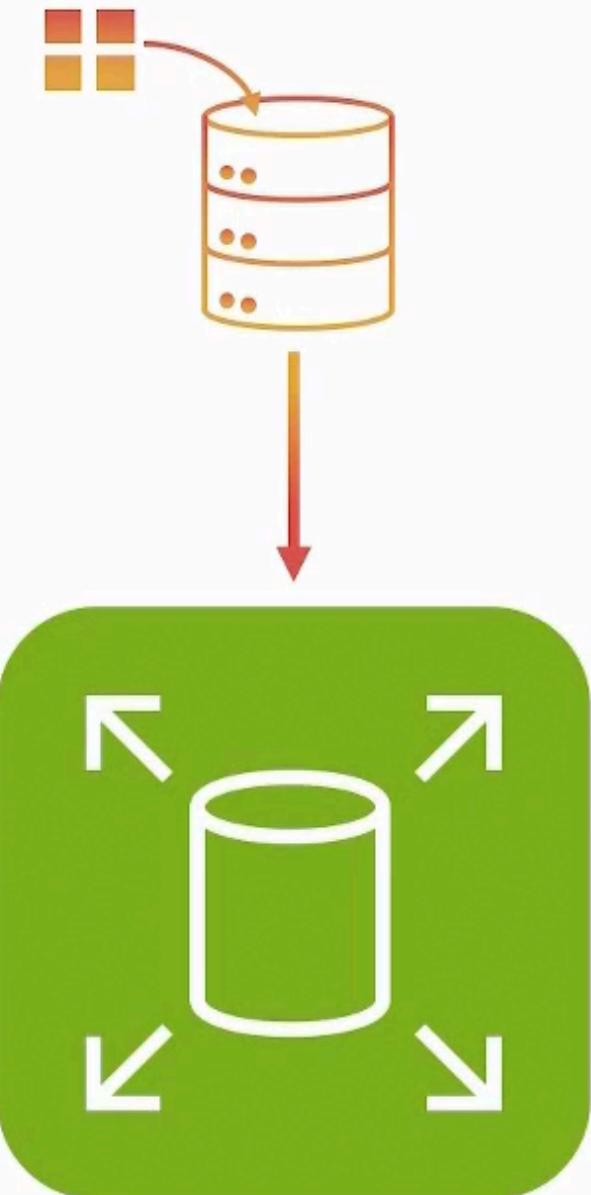


image\_79.png

### Block Storage(Elastic Block Store)

Amazon has a service for block storage called **Elastic Block Store**

## Block Storage



Amazon Elastic Block Store  
(Amazon EBS)

image\_81.png

Block storage breaks up data into blocks and then stores those blocks as separate pieces, each with a unique identifier.

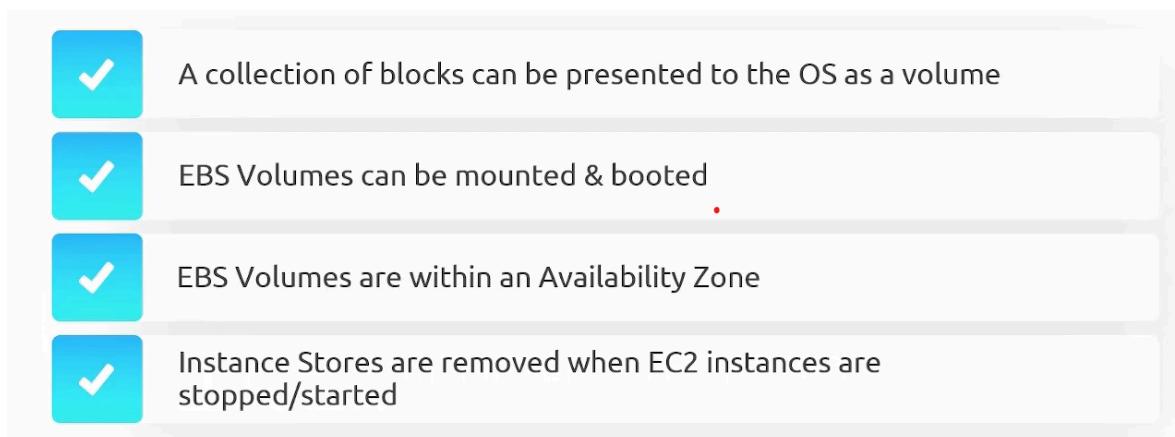
A collection of blocks can be presented to the os as a volume

- The operating system the creates a filesystem on top of it.

A collection of blocks can be presented as a hard drive

- Block storage is bootable( Only storage that is bootable), which means **operating systems** can be installed on it.

EBS volume and EC2 Must be in one availability zone to connect.



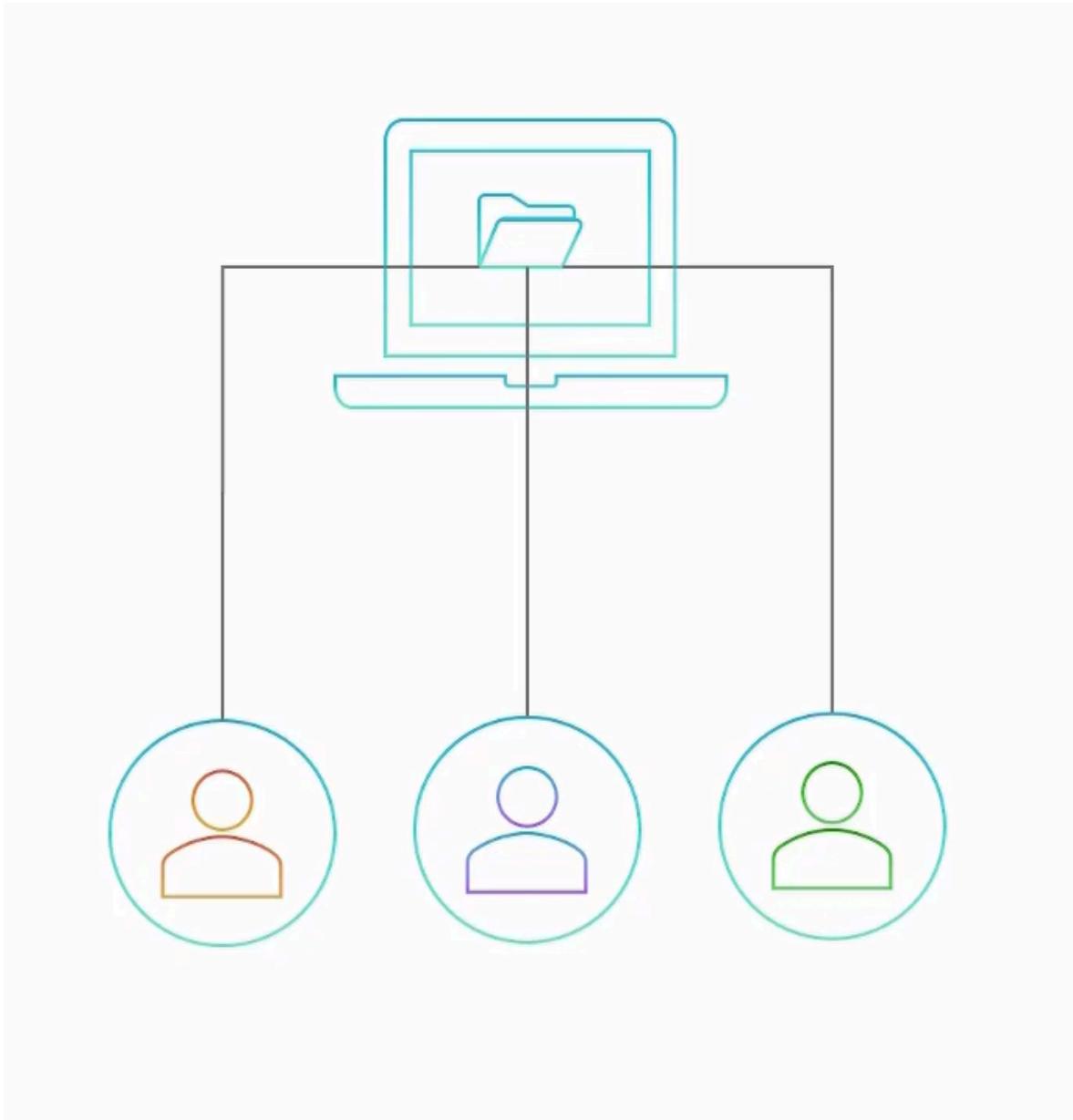
image\_80.png

## File Storage(Elastic File Storage)

File Storage - Stores data in a hierarchical structure of files and folders

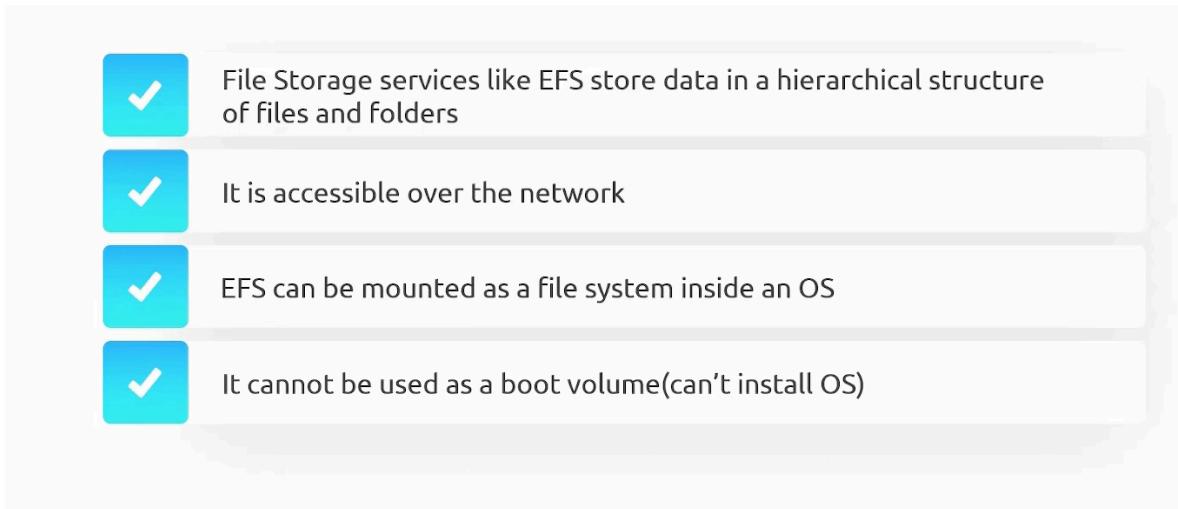
- similar to how your computer stores files/folders.

Filesystem is accessible remotely, multiple clients can access the same data.



image\_82.png

Files storage can be mounted but its not bootable , so you cannot install an operating system



image\_83.png

## S3- Object storage

Object storage stores objects

- Objects are nothing more than files
- Can store any type of file Does not have a folder structure ( Think of it as Google drive or DropBox)
- Flat file system - where everything is in the same folder Since there is no folder structure, object storage cannot be mounted or booted Great for storing logs and media files

## Types of S3 Storage Classes

S3 has different classes because the following factors make S3 Classes different:

- Data Access
- Resiliency
- Cost

### S3 Standard(Default)

- This can handle two simultaneous AZ failures because data is stored across multiple AZs
- This is the most expensive S3 storage class
- Data will be accessed instantly
- You will be charged based on the number of gigabytes you use in a month.

if you don't need to access data frequently you can use:

### **S3 Standard-IA**

- This can handle two simultaneous AZ failures because data is stored across multiple AZs
- You will be charged based on the number of gigabytes you use in a month.
- Cheaper than standard
- You have a retrieval fee
- Has a minimum duration charge of 90 days -If you store an object in these classes and delete it before the minimum duration ,you will still be charged as if the object was stored for the full 90 days.
- Minimum size charge of 128 KB per object- If you store an object in these storage classes that is smaller than 128 KB, Amazon S3 will still charge you as if the object were 128 KB in size.
- If data is accessed frequently or files are too small then you will pay more

### **S3 One Zone-IA**

- Similar to S3 Standard IA but the difference is data is stored in one Availability Zone but replicated in multiple data centers which makes it cheaper
- You have a retrieval fee
- Has a minimum duration charge of 90 days

- Minimum size charge of 128 KB per object

### **S3 Glacier-Instant**

- This can handle two simultaneous AZ failures because data is stored across multiple AZs
- Low-cost option for rarely accessed data
- Performance same as that of S3 standard
- You have a retrieval fee but higher than the previous ones.
- Has a minimum duration charge of 90 days
- Minimum size charge of 128 KB per object
- Cheaper than the previous
- 

### **S3 Glacier-Flexible**

- This can handle two simultaneous AZ failures because data is stored across multiple AZs
- Data is not immediately available , it takes sometime to retrieve the data
- You have a retrieval fee
- How long to wait for the data to be retrieved depends on the option selected:
  - Bulk : 5-12 Hours
  - Expedited: 1-5 Minutes
  - Standard: 3-5 Hours ( The less the time the higher the cost)
  - During retrieval data is stored in S3 standard-IA class temporarily.
- Has a minimum duration charge of 90 days
- Minimum size charge of 40 KB per object

## **S3 Glacier Deep Archive**

Cheapest storage class in S#

- You have a retrieval fee
- How long to wait for the data to be retrieved depends on the option selected:
  - Bulk : 12Hours
  - Standard: 48Hours
- Has a minimum duration charge of 90 days
- Minimum size charge of 40 KB per object

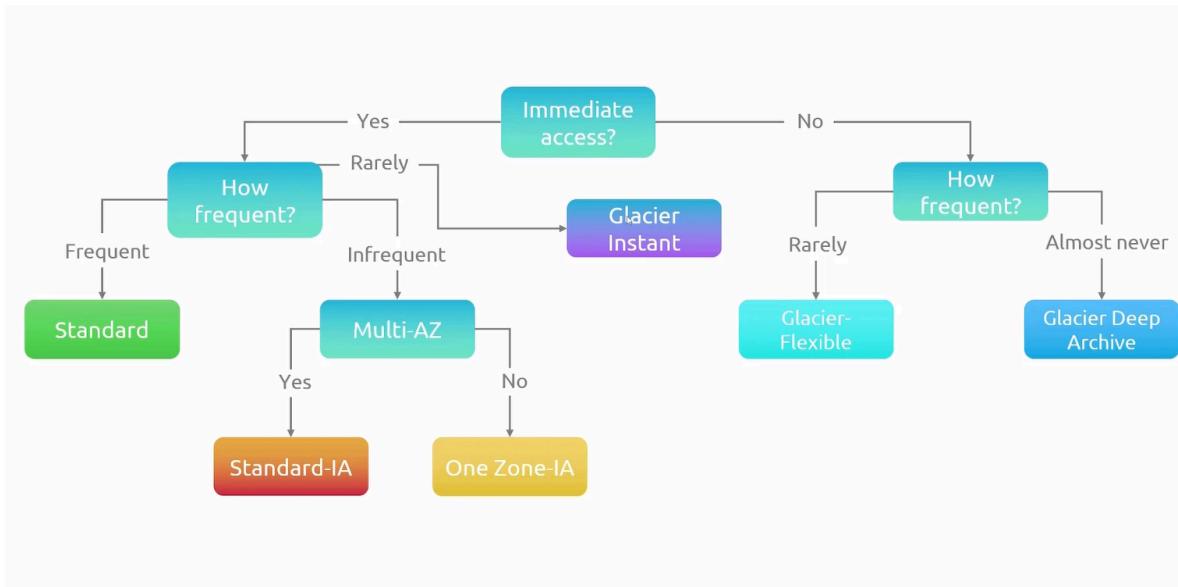
What happens when you don't know the storage class to use?

## **S3 Intelligent-Tiering**

- Here AWS will analyse the access pattern on a file and move it to a fitting category.
- Automatically reduces storage costs by intelligently moving data to the most cost-effective access tier
- Apart from the cost of the storage class and all objects also incur a monitoring/automation cost per 1,000 objects.

## **How to Select the Best Storage Class**

How do you select the best storage class , use the diagram below to guide you.



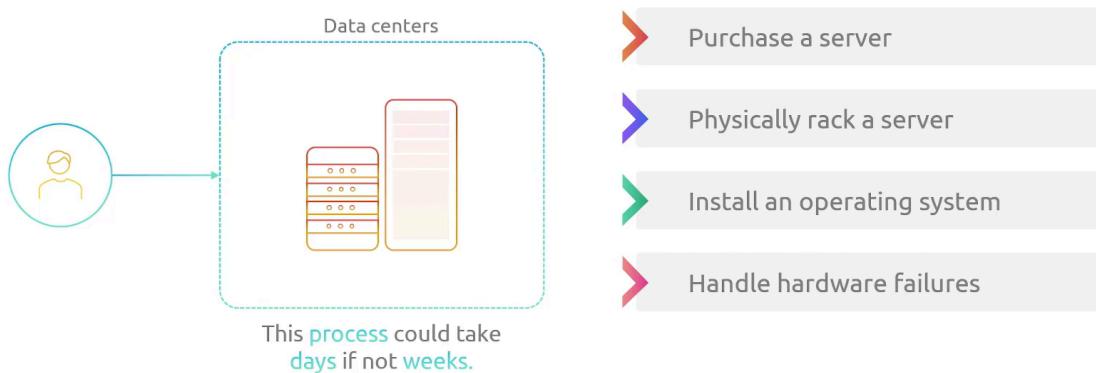
image\_84.png

# Compute Services

In order to run an application we need servers in order to deploy our code.

Before AWS :

You needed to find a data center or build a data center then:



image\_85.png

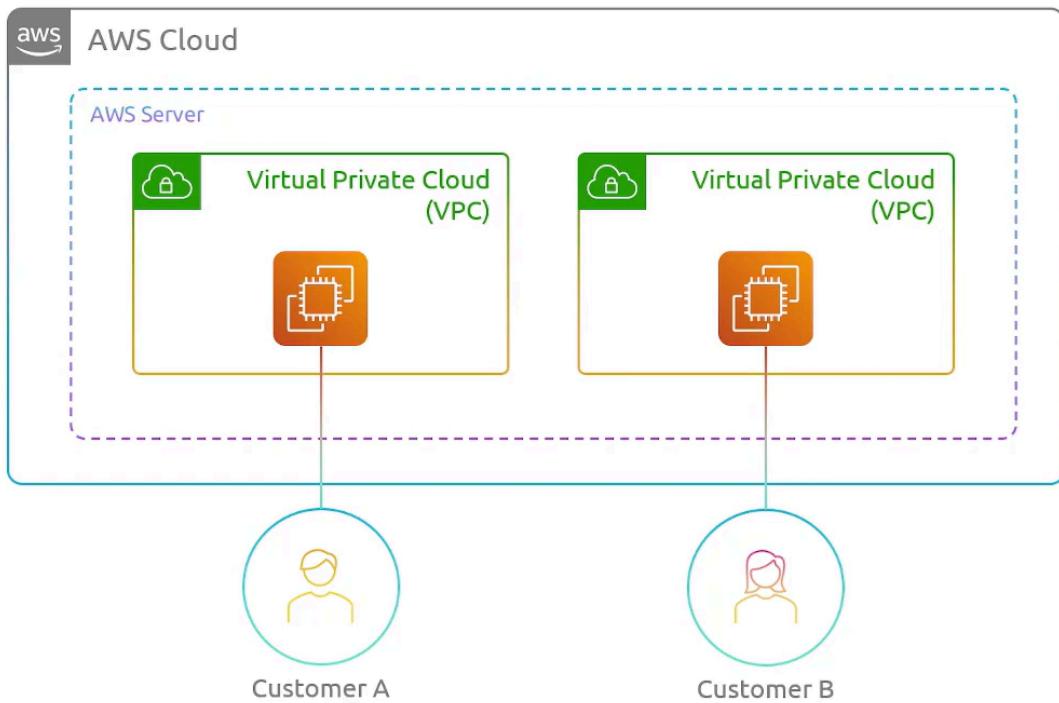
This is time-consuming , costly and involved multiple people.

With AWS a Server is Just a Click-away. The service responsible for providing servers in the cloud is an EC2.

## Amazon EC2

EC2 allows you to provision a server on AWS within minutes, it provides secure, resizable compute capacity in the cloud

There is a potential that An AWS server has several EC2 instances deployed there. VPCs are used to logically isolate each customers' infrastructure.



image\_86.png

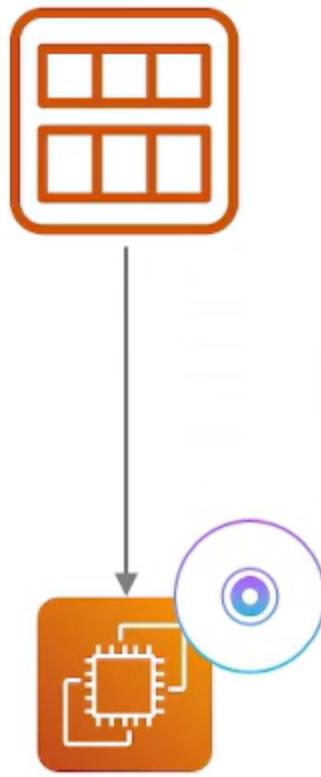
If you are using a physical server you will have to install an operating system but what about EC2 instances?

### Amazon Machine Images(AMI)

AMI is an image that provides the information required to launch an EC2 instance.

AMI is a blueprint that contains information such as what operating system and what software/packages should be installed on an instance.

We can use one AMI for multiple EC2 instances. Think of AMI as an installer disk used for installing a new Server



image\_87.png

AMI Can be fully customized to:

- Add application source code
- Add dependencies
- Customize OS firewall

If you are shopping for a physical server there will be specifications that you will consider e.g. Memory, CPUs even storage Now how does that happen in the cloud.

## Instance Types

EC2 provides a wide selection of instance types optimized to fit different use cases.

Instance types have varying combination of CPU, Memory, storage, and Networking capacity. They are just different server models to select from.

There are different types:

### **General Purpose**

- Provide a good balance of compute, memory and networking resources
- Can be used for diverse workloads
- Ideal for applications that use resources in equal proportions

### **Compute Optimized**

- Optimized for compute-heavy applications
- Contain high-performance CPUs
- Ideal for batch processing workloads, media transcoding, machine learning and gaming servers.

### **Memory Optimized**

- Deliver fast performance for memory-intensive workloads
- Suited for database

### **Storage Optimized**

- Optimized for workloads that require high, sequential read and write access to large data sets on local storage.
- Can deliver tens of thousands of low-latency, random I/O operations per second (IOPS)

### **Accelerated Computing**

- Utilize hardware accelerators to perform expensive calculations
- Great for graphics processing and data pattern matching

## **EC2 Pricing**

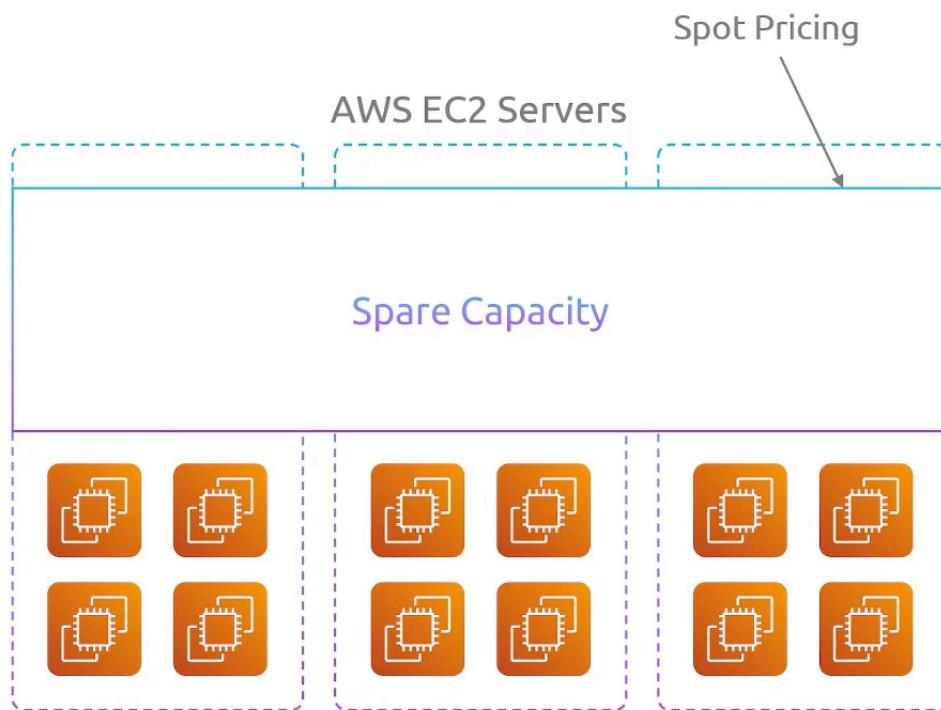
## On-demand Pricing

Here you pay for compute capacity by the hour. Only billed when instance is running:

- Still charged for storage attached to instance NO upfront payment or long-term commitment
- Great for short-term, irregular or unpredictable workloads.

## Spot Pricing

Amazon tries to maximize the usage of their servers. For example if they have a machine with 64GB and the EC2 instances running inside the machine only use 20 GB then 44GB is idle.



image\_88.png

Instead of it being idle, Amazon offers spare compute capacity at discounted rates. Spot instances are recommended for:

- Applications that have flexible start time and end time

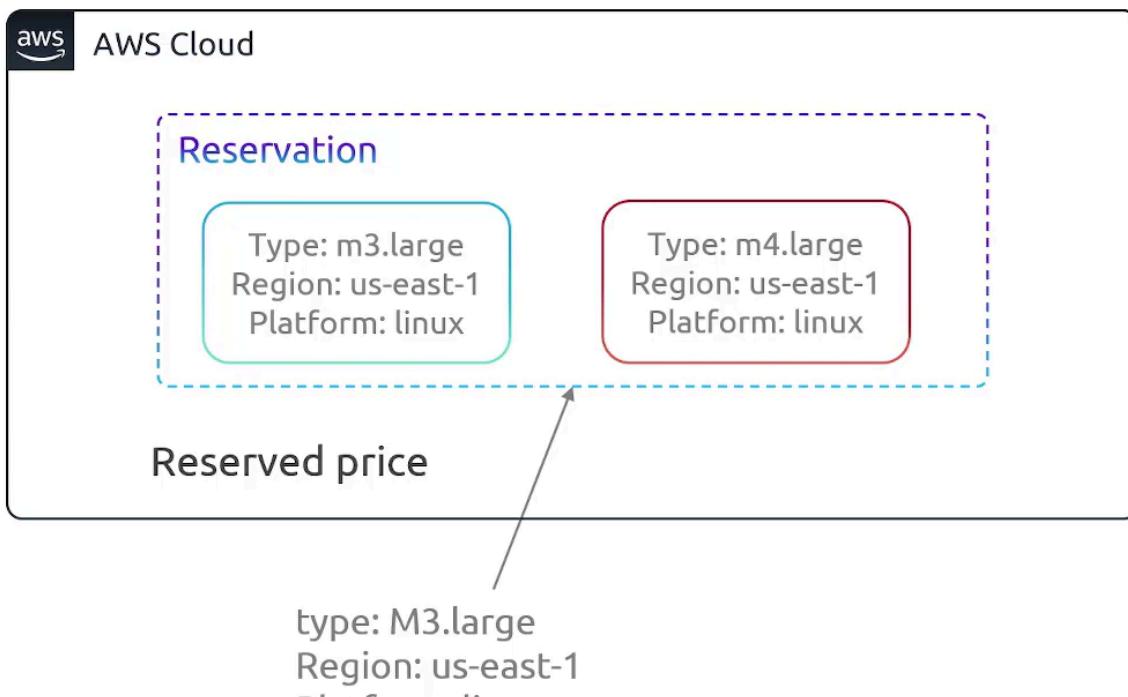
- Applications that need low compute prices Not suitable for application that cannot tolerate interruptions.

## Reserve Pricing

Reserve instance is a billing discount that allows you to save on your EC2 costs. Offers discounted rates when reserved for a certain period (1year or 3 year contract)

When purchasing a reserved instance, you are not actually buying an instance. You are merely committing to using an on-demand instance for a long-term period.

When you deploy an on-demand instance with matching attributes as the reservation(instance type, region, platform), it will be charged at the reserved price and not the default.

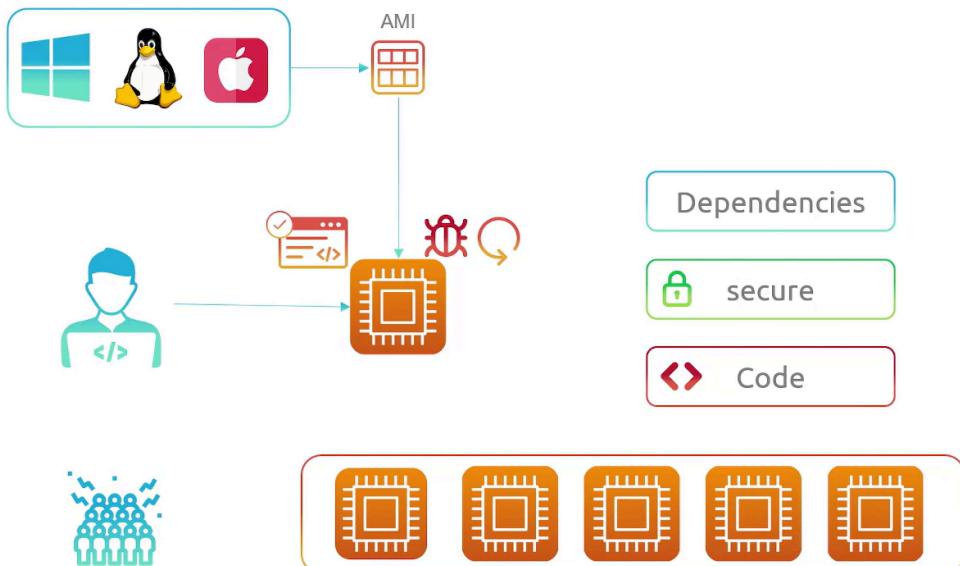


image\_89.png

In the above diagram, what reserved instance means is any instance that is M3.Large you will be charged as a reserved instance but any other type of instance you will be charged as an On-demand instance.

# AWS Lambda

Before we check on Lambda let check at what we have right now:



image\_90.png

When we want a server running, we must choose the right AMI, we must add the right dependencies , we must copy the code and Even after the application is running we must watch out for updates and software patches.Not forgetting that in case of traffic we need to spin up or down more servers. That's a lot of work and a developer just wants to write code. This is where AWS Lambda Comes in

## What is Lambda

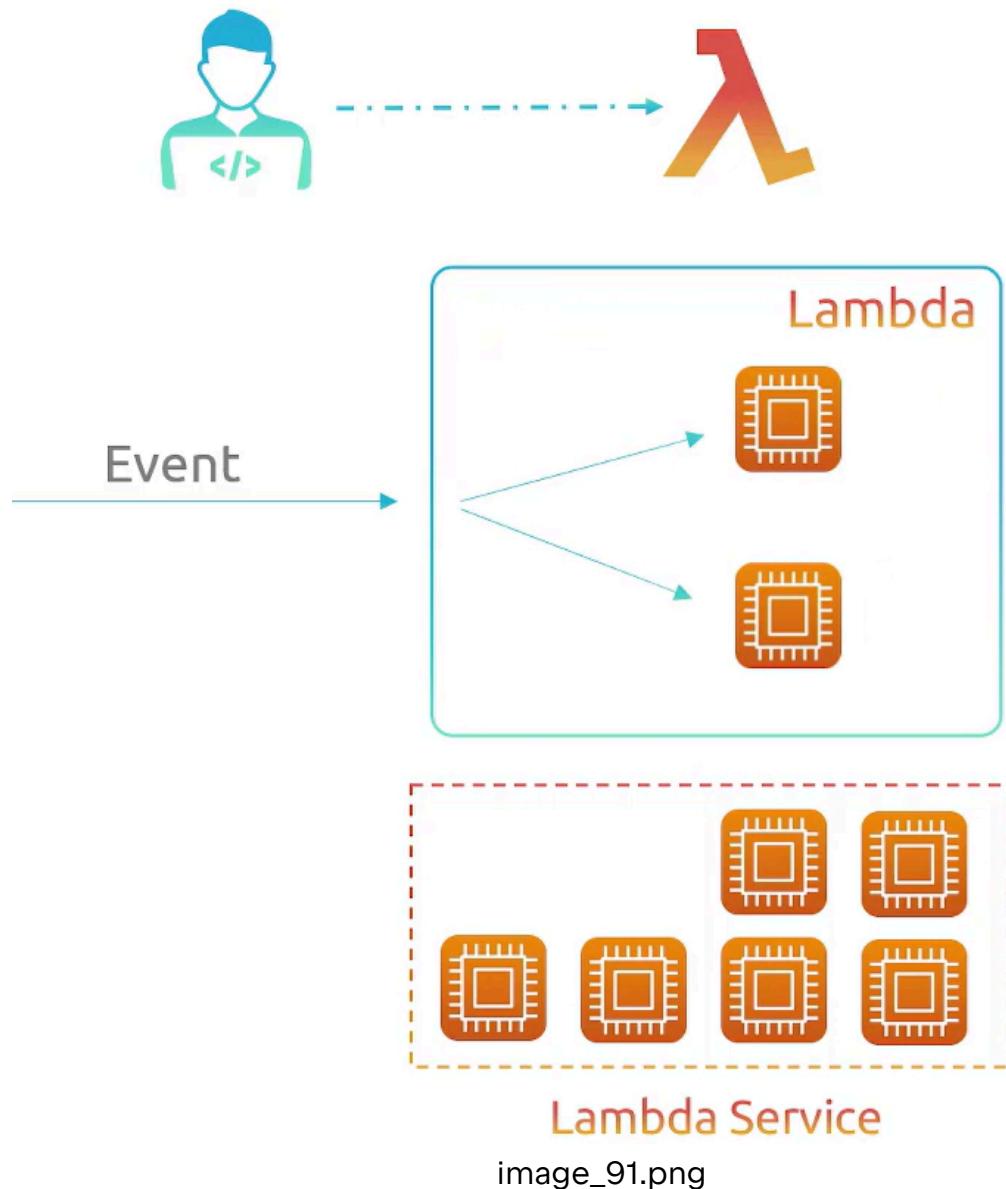
AWS Lambda is a compute service that let's you run your code without having to provision or manage servers.You just copy the code to AWS and AWS will manage the underlying infrastructure like starting a server and running it. Lambda is AWS' serverless offering:

- AWS manages the server maintenance, scaling, capacity provisioning and logging.

But serverless , does that mean there is no servers?

## Serverless

No, to run an application there will always need to be a server but with Lambda serverless means from the user perspective you just upload your code and then AWS will manage the infrastructure.



image\_91.png

What happens is AWS will have a Lambda service reserve pool that contains several servers and there will be an event that will be triggered for your code to run, when the event is triggered then some servers are pulled from the list of servers in the service pool

, they will run the code and when done they will be freed and taken back to the service pool.

## Lambda Use Case

- File processing - Eg resizing an image everytime an image is uploaded.
- Stream processing
- Web application
- Mobile/ Web backend

## Lambda Components

### Lambda Function

This is basically your code

```
exports.handler = async function
(event, context) {
    console.log("Lambda function ran");
    return;
};
```

image\_92.png

### Trigger

Trigger determines when your lambda function should run.

✓ Upload file S3

✓ Request to API Gateway

✓ DynamoDB update

image\_93.png

## Event

Event is the information of what trigger took place, eg of an event is about file upload then the event could include the name of the file .

## Benefits of Lambda

- No Server to manage
  - No need for an infra team
  - No patching/upgrading
  - Faster development
- Auto scale to handle traffic spikes
- Pay for what you use
  - pay per invocation

- No extra cost during low traffic

## Downsides of Lambda

- No Local state
  - Will need a separate database to store data that needs to persist
- Limited execution duration
  - Function can run at most for 15 minutes
  - Not good for long-running tasks
- Cold starts
  - Cold start occur due to the time it takes to initialize and load a function.

## Lambda Pricing

You will be charged for:

- Number of time the function ran
- How long did it run for
- How much memory/ CPU did it require.

# Compute Container

## What is a container first

Container are a tool that allows you to package an application and all the necessary files, libraries and dependencies the applications need to run.

Containers can be deployed on a machine and since they have everything the application needs to run there is nothing else that needs to be done. They are just small lightweight version of virtual machines.

## Container Challenges

Ultimately we want our application to be in multiple hosts, we need to restart faulty container, we need to scale up and down,we also need to replace a host if its not working.. In that case we need something to manage the hosts. This is where container Orchestrators comes in.

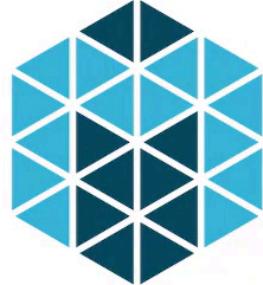
## Container Orchestrators

Container orchestrators are the brains of a containerized environment. Responsibilities include:

- Deploying containers across all available servers
- Load balancing request to containers
- Providing container to container connectivity
- Restarting failed containers
- Moving containers when hosts fail Here is example services for containers orchestrators



Kubernetes



Apache Mesos



ECS

image\_95.png

## Elastic Container Service (ECS)

Fully managed container orchestration service that helps manage, and scale containerized applications.

- ECS is managed by AWS
- Containers run on EC2 instances or Fargate ECS is proprietary software (only available in AWS)
- Migrating to a different cloud provider will be more difficult.

## EKS

Lets first understand Kubernetes

### Kubernetes

Kubernetes is an open-source container orchestrator. Kubernetes cluster has two types of nodes

- Control-plane nodes - managers of the cluster
  - Watch over cluster and make sure cluster is kept in a working state
- Worker nodes- responsible for actually running the containerized workload

Managing Control planes and the worker nodes is difficult:

- Running & scaling control-plane
- Properly securing the control-plane

AWS Elastic Kubernetes Service is a managed Kubernetes service

- EKS will manage the control-plane for you
- Users are still responsible for managing worker nodes
- If you use Fargate AWS will manage worker nodes too

## Benefit of EKS

- Runs and scales control-plane across multiple Availability Zones
- Scales control-plane instances based on load
- Can integrate with other AWS services
  - IAM for authentication
  - Elastic load balancer
  - ECR for container Images

## ECS VS EKS

ECS is proprietary to AWS so migrating to another cloud provider can be difficult EKS is kubernetes which is open-source and can run on any platform.

- Keep in mind the more AWS service you configure your cluster to interact with the harder it will be to move to another provider because of service specific configuration.

ECS has a simple architecture

- Simpler API

- Easier to ramp up new team members Kubernetes is very complex and has a steep learning curve
- With EKS you'll have to learn kubernetes as well as EKS specific features.

Kubernetes has a larger community

- More support online
- More tooling like Helm/ Kustomize/ArgoCD

ECS pricing - No cost for control-plane , only pay for infrastructure running applications (EC2/ Fargate , EBS) EKS pricing - more expensive, you have to pay for control-plane and worker nodes/ Fargate.

# AWS Databases

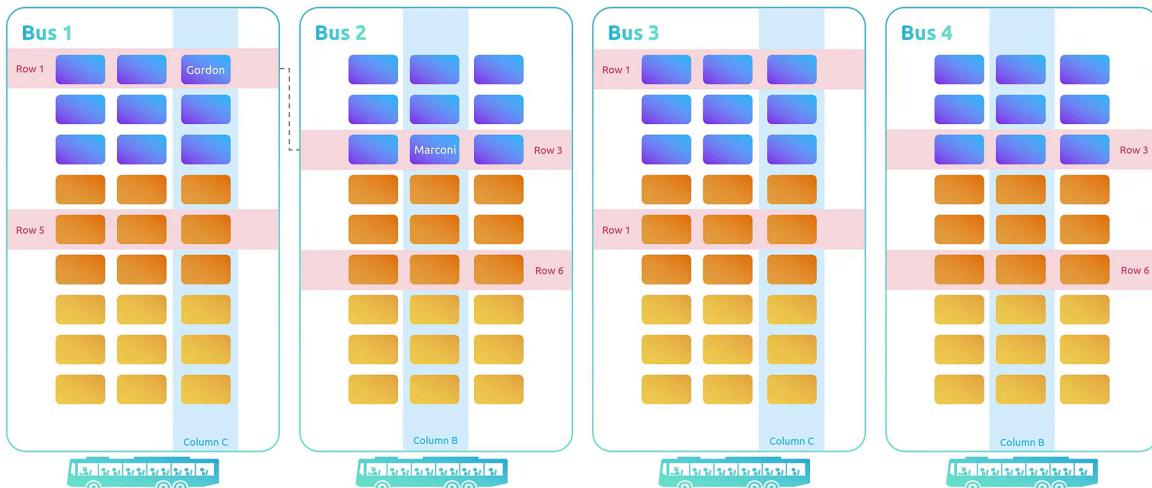
Database is basically an application that stores data.

## Types of Databases (Datastores)

There are three types of databases:

- Self-Managed Data stores
- SQL/Relational Data stores
- NoSQL Data stores

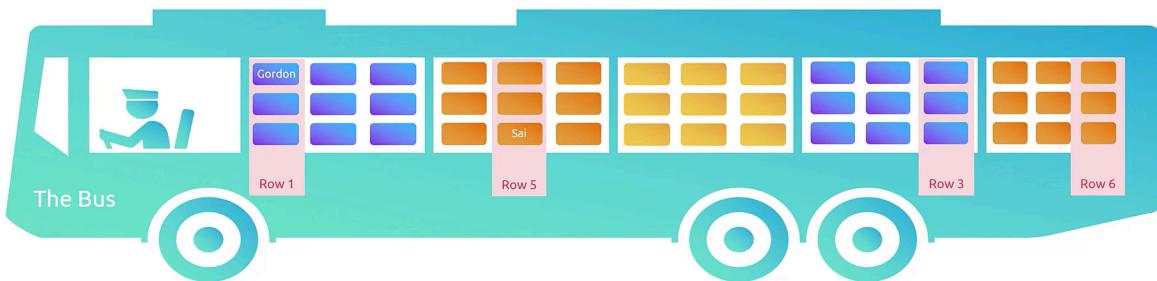
## Structured Data VS Unstructured Data



image\_96.png

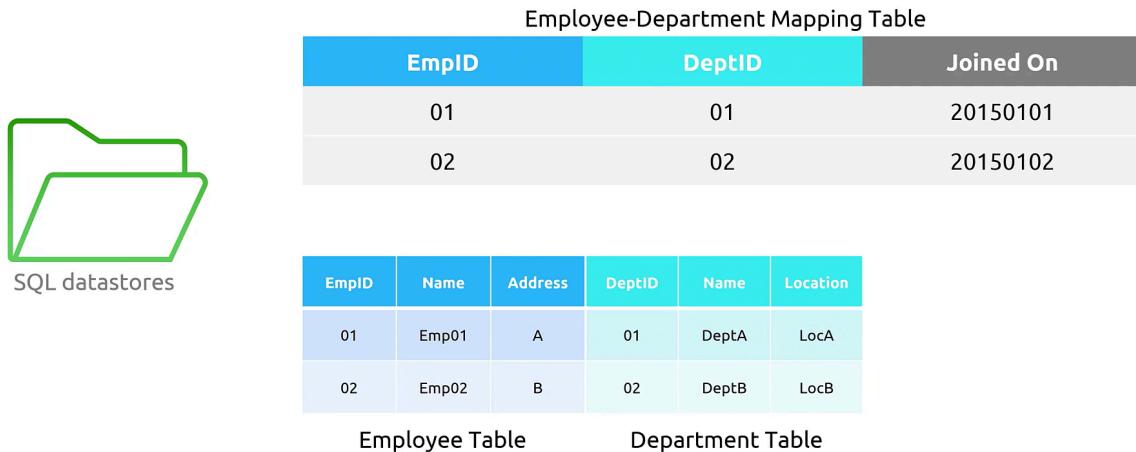
Lets use the above diagram to explain a relational database. Assuming we have buses and in one bus we have Gordon and in the other we have Marconi . If we need to find Gordon we need to go to Row 1 Column 3 Assuming there is a relationship between Marconi and Gordon( friends ) Then if marconi moves rows we have to update Gordon where marconi is

This is structured data and what SQL datastore are used for.



image\_97.png

Assuming now we just come and ask if there is Gordon in the bus and Gordon raises his hand. Here we just use a key like 'name' to search



image\_98.png

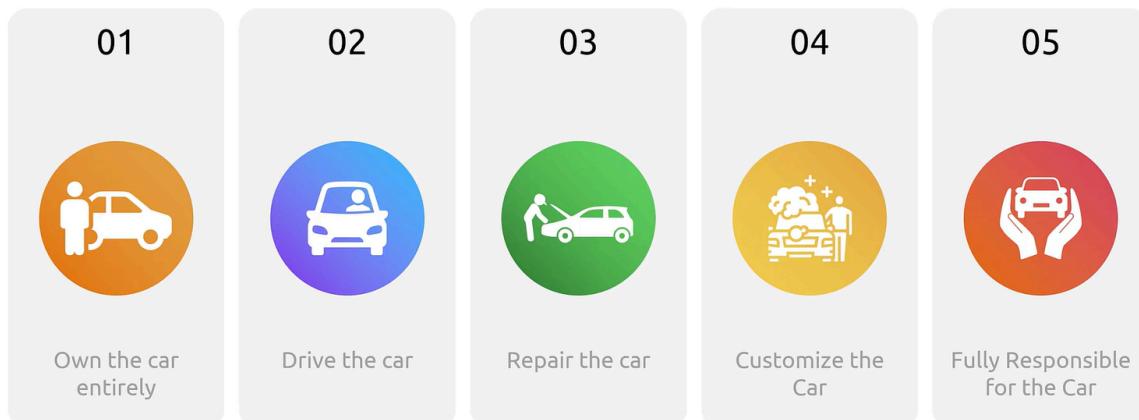
In Structured data stores , the buses represents tables and the tables have relationship with each other. Data here is a bit structured and organized. Used primarily when you have complex relationship with your data Think Transactional (like banking) or reporting use cases for these data stores

Partition Key	Sort Key	Attributes		
Product ID	Type	Odyssey	Homer	1871
1	Book ID	6 Partitas	Bach	
2	Album ID	Partita No. 1		
3	Album ID: Track ID	The Kid	Drama Comedy	Chaplin
4	Movie ID			
Primary Key		Products		

image\_99.png

On the other side we just need to know a key like 'movielid' or 'bookName' and just search it. NoSQL is good for search. Used when you have simple but specific needs for data Think search, High performance, Documents, Relationship use case.

## Self-Managed Data stores



image\_100.png

Assuming a database is a car and you want full control of everything Own the car entirely (Own the database) Drive it (In charge of the database) Repair the car ( Repair database in case of any issue) Customize the Car (Customize it how you want it, make updates too) Fully Responsible for the car ( You in charge of everything about the database)

If you decide to use EC2 Virtual Machines or ECS or EKS that is a Self-Managed Database.

- Can run your own database software on EC2 or any container services (ECS or EKS)
- Increased control and Flexibility

- Increased Operational overhead and responsibility
- Mainly used when you have specific software or security requirements.

 EC2 instance contents	 Amazon Elastic Kubernetes Service (Amazon EKS)
<ul style="list-style-type: none"><li>• Most “unmanaged” option</li><li>• More control</li><li>• More responsibility</li><li>• Cost less in service dollars</li><li>• Cost more in Operational Overhead</li></ul>	<ul style="list-style-type: none"><li>• Most “unmanaged” option</li><li>• More control</li><li>• More Responsibility</li><li>• Cost less in service dollars</li><li>• Cost more in Operational Overhead</li><li>• Can mitigate that with Fargate instead of EC2 cluster</li></ul>

image\_101.png

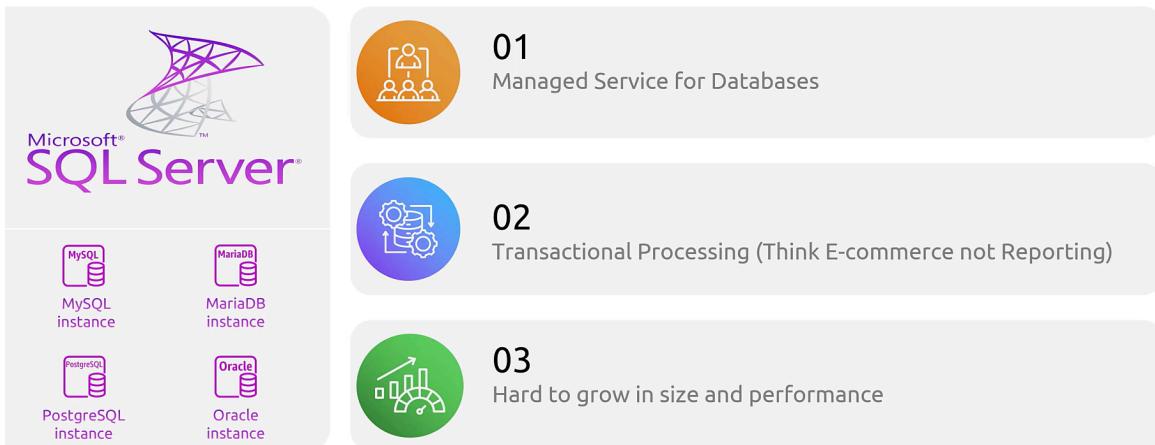
# SQL Data stores (Structured)

Back to the Car database



image\_102.png

We have 5 databases in this case:

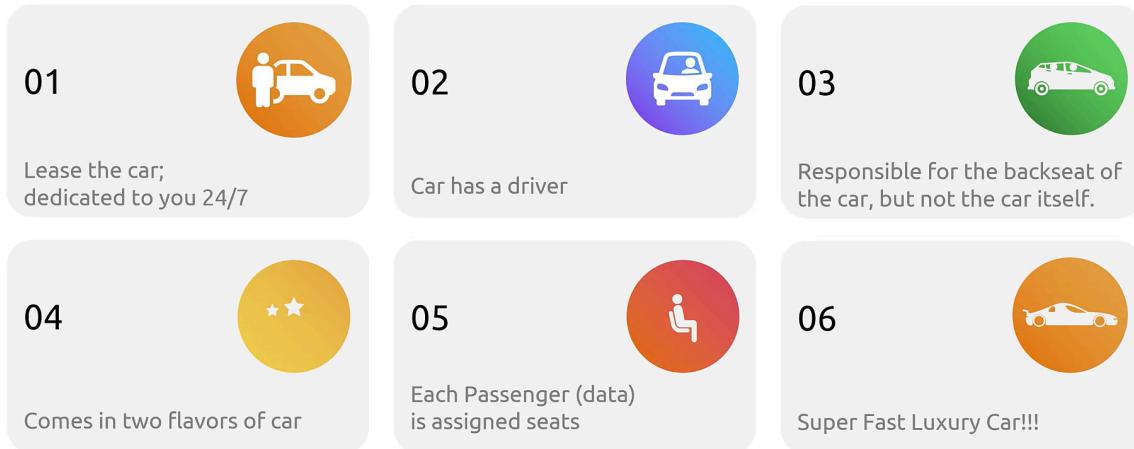


© Copyright KodeKloud

image\_103.png

But it hard to grow in terms if write but easy to grow in terms of read.

But what if you want something that is a bit faster .



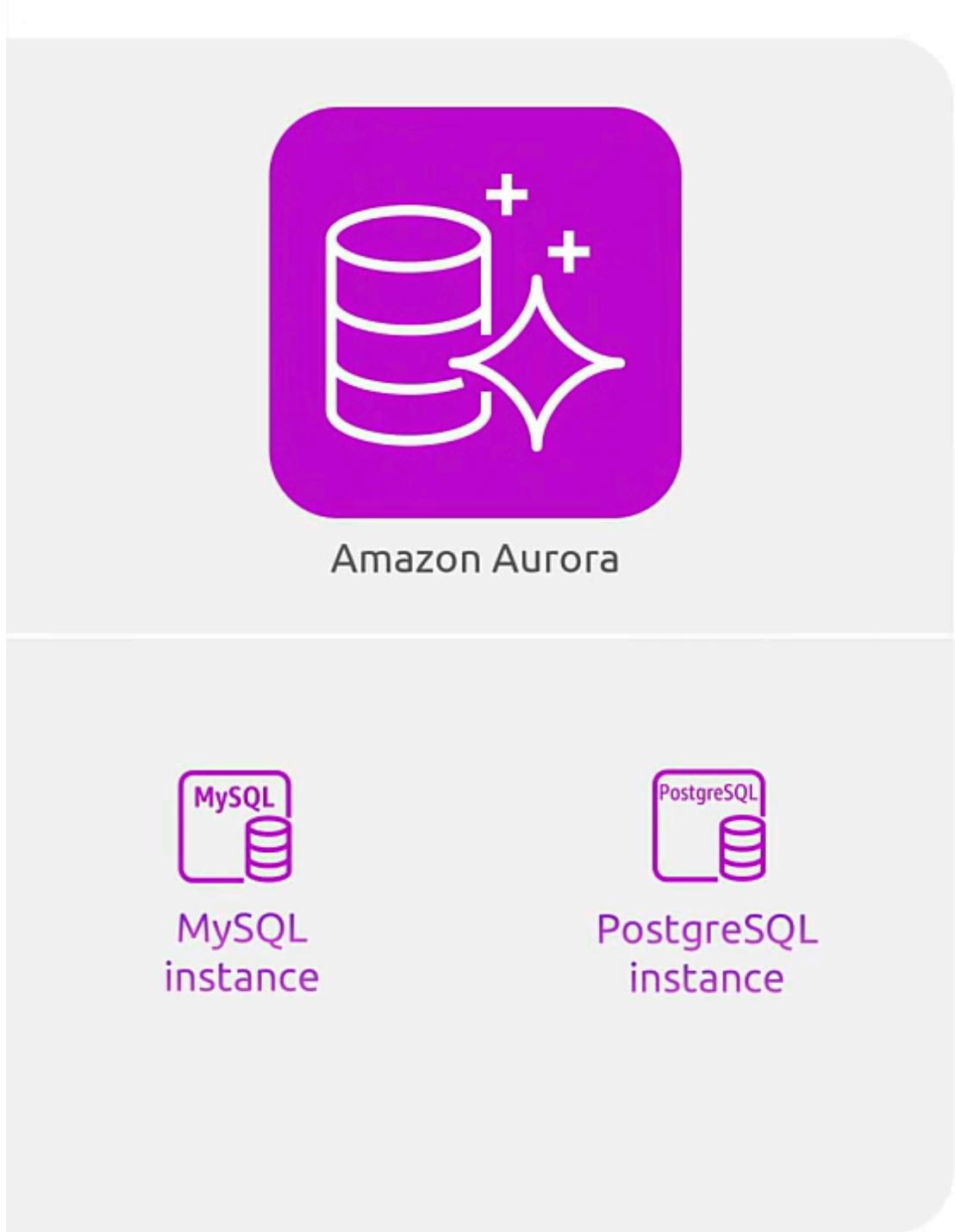
image\_104.png

This ia where Aurora comes in

## SQL Data stores - Aurora ( really RDS Aurora)

Here they took Postgres and MySQL from the traditional RDS service :

- Higher capacity and higher performance
- Grows more easily than the main RDS service
- Managed services for Databases
- Cloud Native



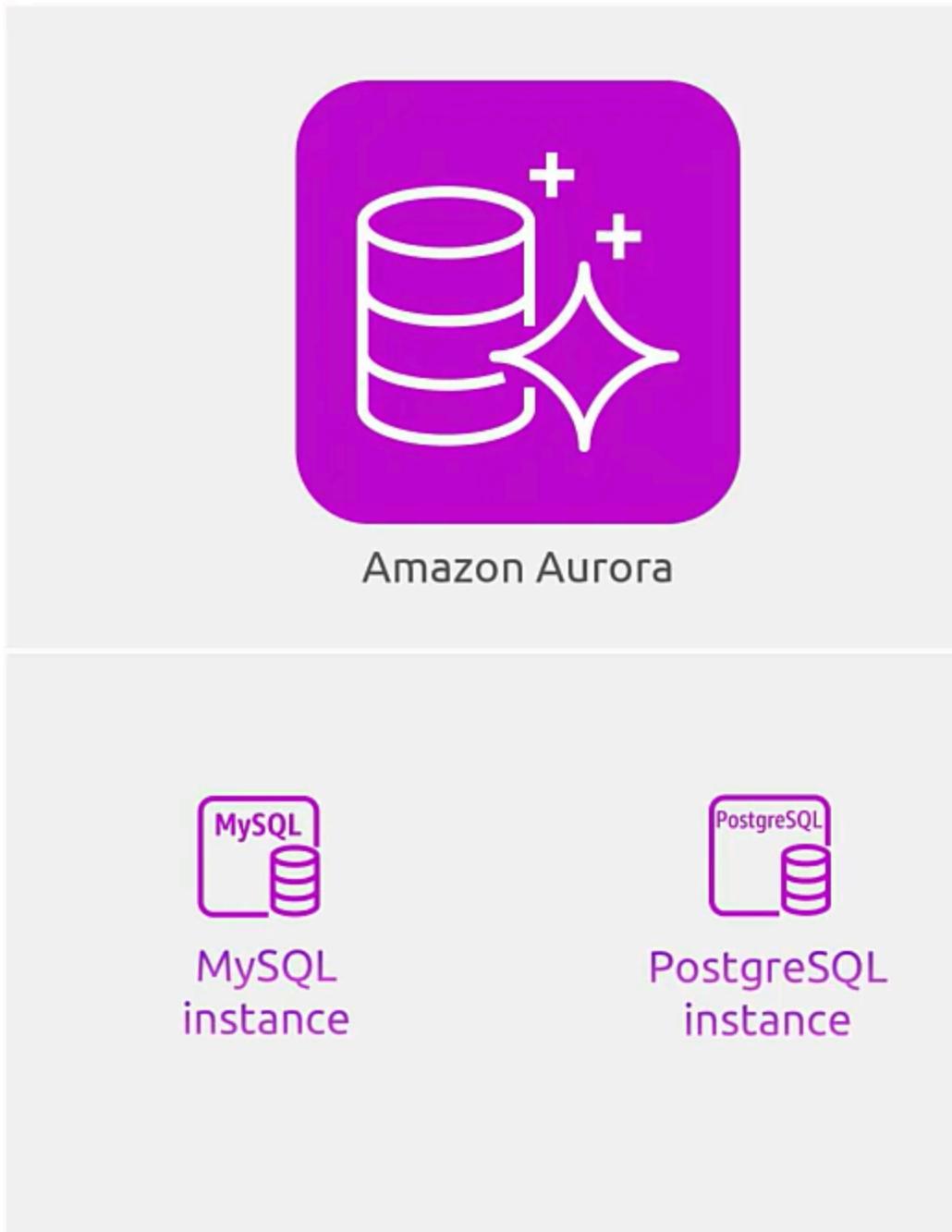
image\_105.png

But assuming again you want something that you only pay for it when you use it ( more of an uber here ) Here we have:

## Aurora Serverless V2

Again here we have a :

- Managed service for databases
- Cloud native
- Higher capacity and higher performance
- Capacity can go up and down easier than other RDs servicePay little storage, but not compute when you are not using it



image\_106.png

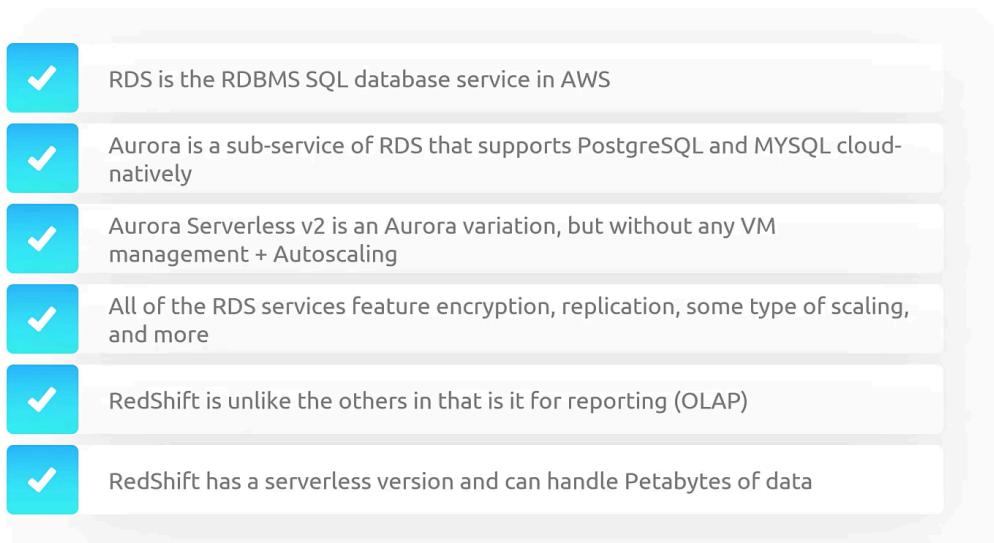
But sometimes you have a lot of data that needs analyzing and processing reports (you need a bus not a car)

## RedShift

This is the SQL data warehouse service is SQL. Sometimes you don't want transactional database but a data warehouse. RedShift is used when:

- When you need a data warehouse not a transactional data store
- it's the SQL data warehouse in AWS
- Petabyte scale
- Serverless and 'Server' ed versions
- Think reporting and analyzing not transactional

## Short Summary



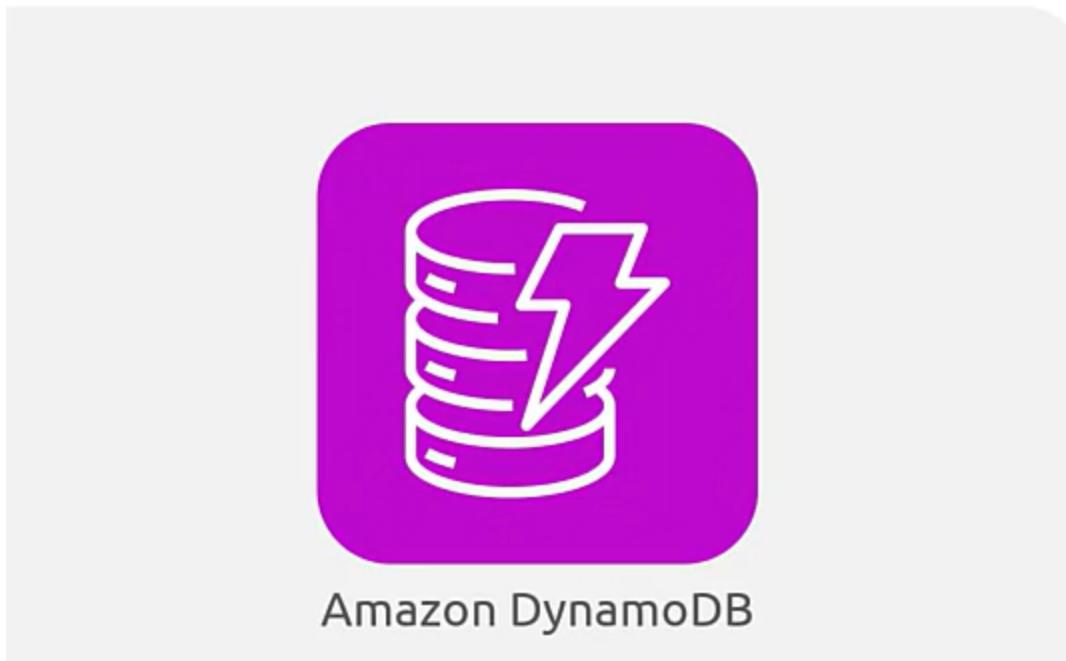
image\_108.png

# AWS NOSQL Data Stores

## Amazon DynamoDB

This is 'The lightning-fast king of key-value at AWS'

If you want to create blobs of data that you can search for with a single keyword or phrase we use **Amazon DynamoDB**



## DocumentDB

We use this Service when we want to retrieve a document like essays, profiles and more like collections of data.



Amazon DocumentDB  
(with MongoDB compatibility)

image\_109.png

## Keyspaces (Cassandra Compatibility)

you use these when you want a database that can run in many different locations across the planet and you need large-scale unstructured data.

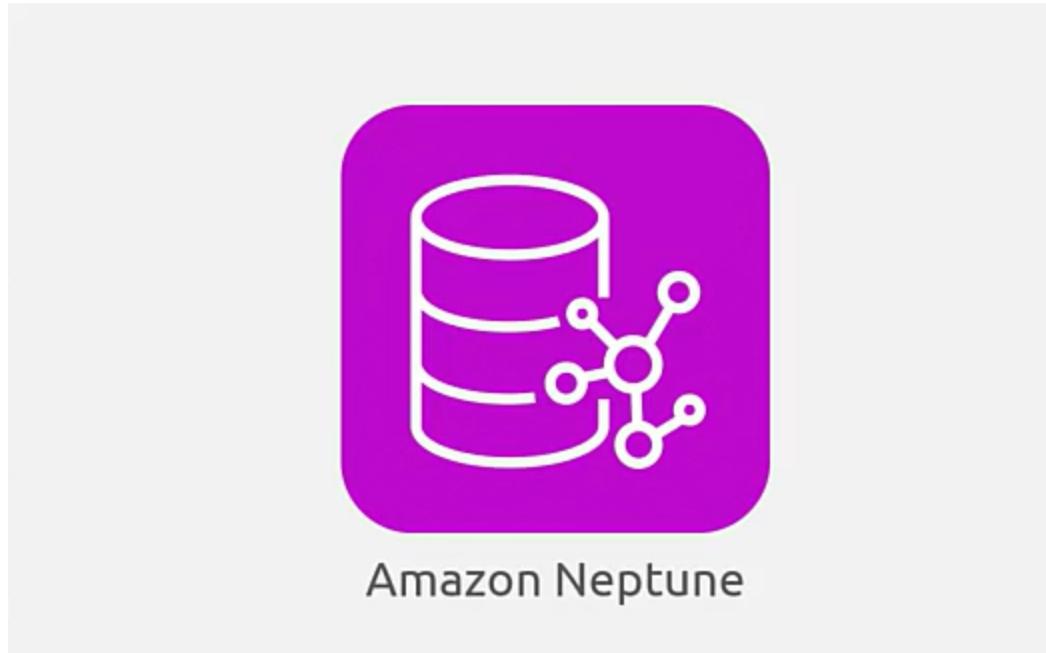


Amazon Keyspaces  
(for Apache Cassandra)

image\_111.png

## Neptune

When you need a database that will detect relationship between data like fraud detector or social network relationship this is when you use **Amazon Neptune**



image\_112.png

## ElasticCache

Sometimes you need a database that can cache expensive database results and be able to get it faster. Here you need to store data in a location that is faster than my regular database. We have two databases:



Amazon ElastiCache



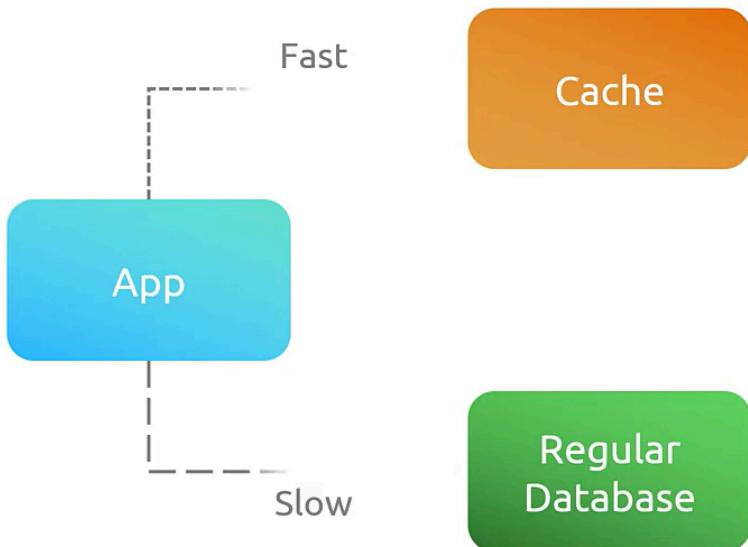
ElastiCache for  
Memcached



ElastiCache  
for Redis

image\_113.png

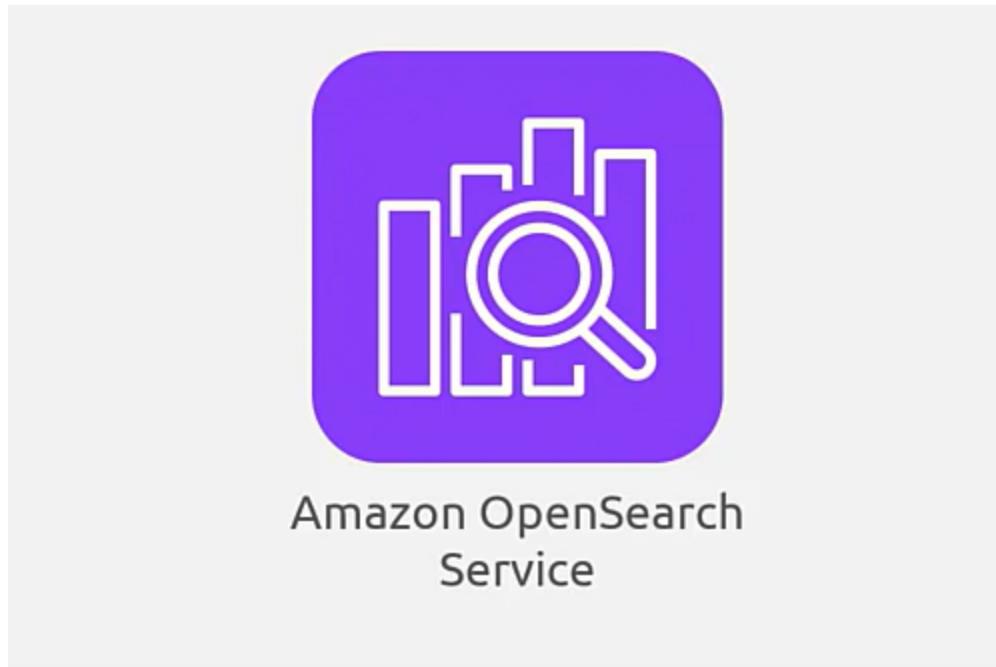
Here you will have your app listening to the cache but if something is not in the cache then we will have to go to the database but it will be a bit slow.



image\_114.png

## OpenSearch Service

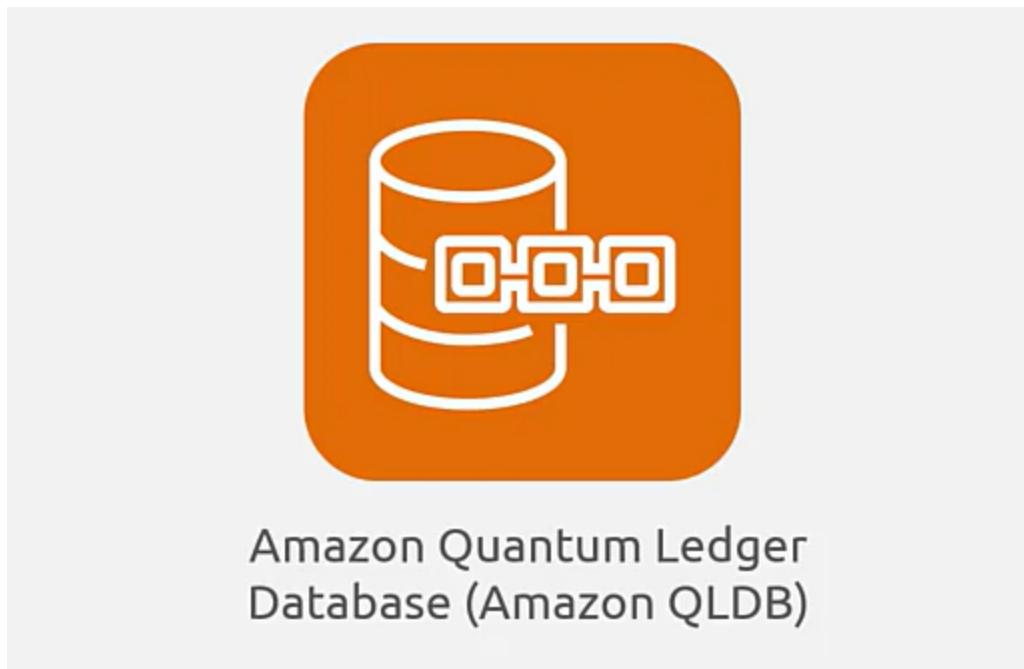
When you want a database that you can use to search through a bunch of information like a google search that give you relevant or related results, here you use **Amazon OpenSearch Service**



image\_115.png

## Amazon Quantum Ledger Database Services

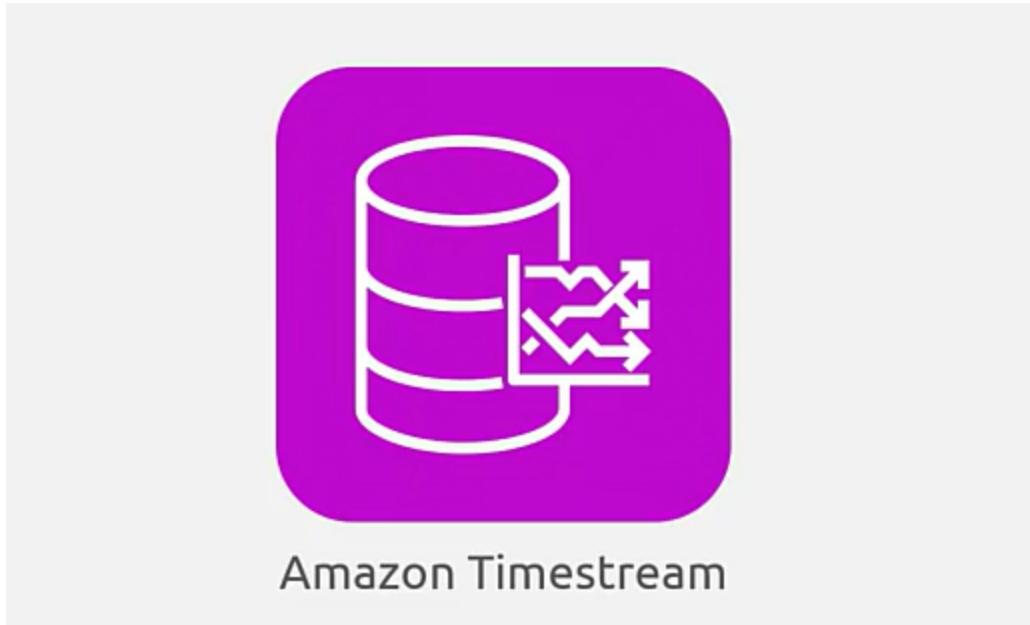
A database where everytime you modify a number or a transaction it keeps a record of it (more like blockchain).It's great for security consistency. Example databases:



image\_116.png

## Timestream

This is the database you use when collecting data from IOT devices and you need a database that captures data from various sources at high scale and maintains the timestamp.



image\_117.png

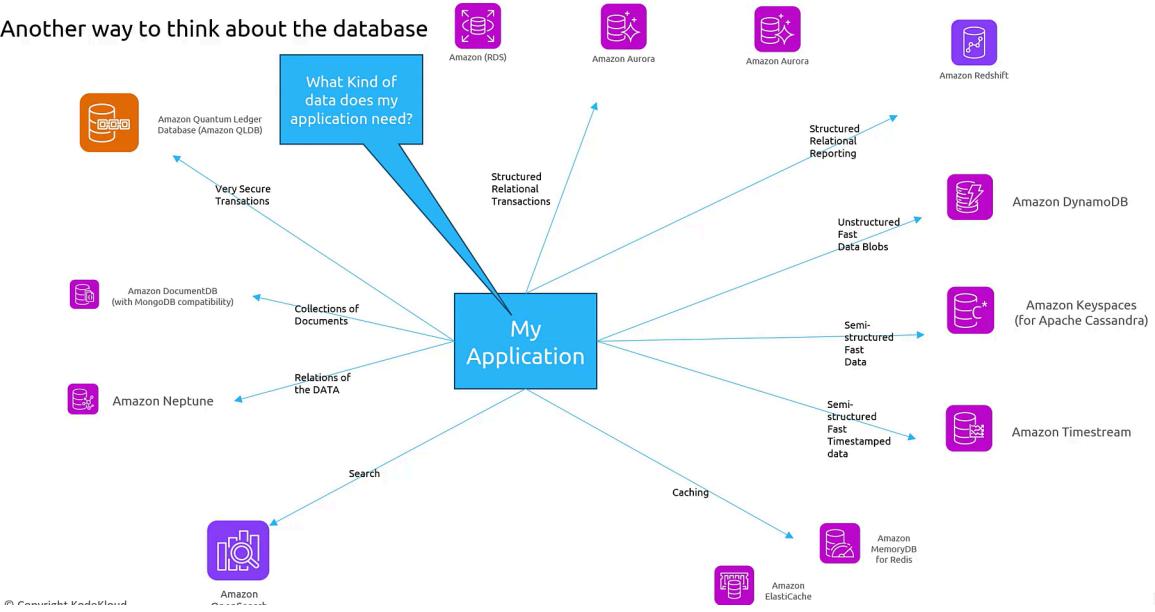
Good for collecting Logs from multiple devices



image\_118.png

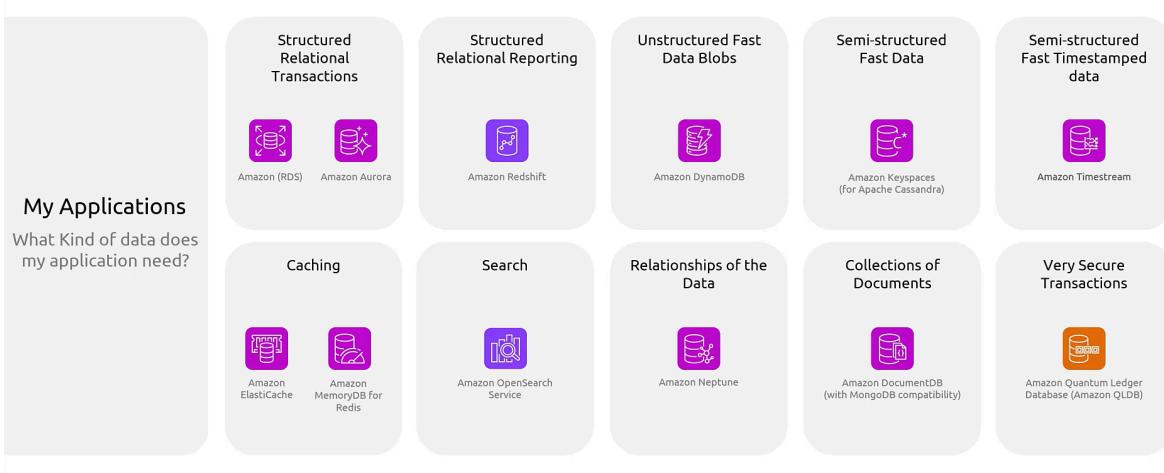
# AWS Database SUMMARY

Another way to think about the database



image\_119.png

Another way Would be:



image\_120.png

Summary



Self-Managed is an option if you need management and control



RDS and RedShift are the primary SQL database systems



RDS has five engines supports: Oracle, MySQL, MariaDB, MS SQL, and PostgreSQL



DynamoDB, DocumentDB, and others are NoSQL services that are fully managed



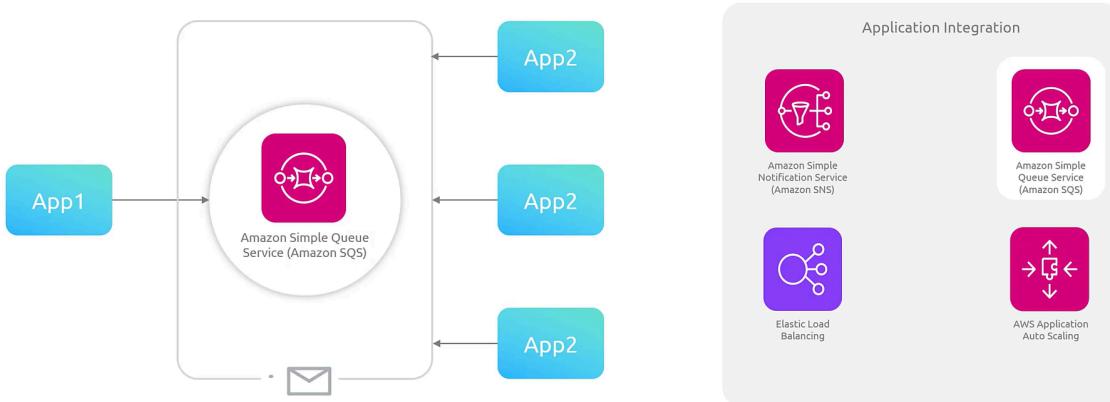
Make sure you look at the previous slide for use cases for each service

image\_121.png

# Application Integration

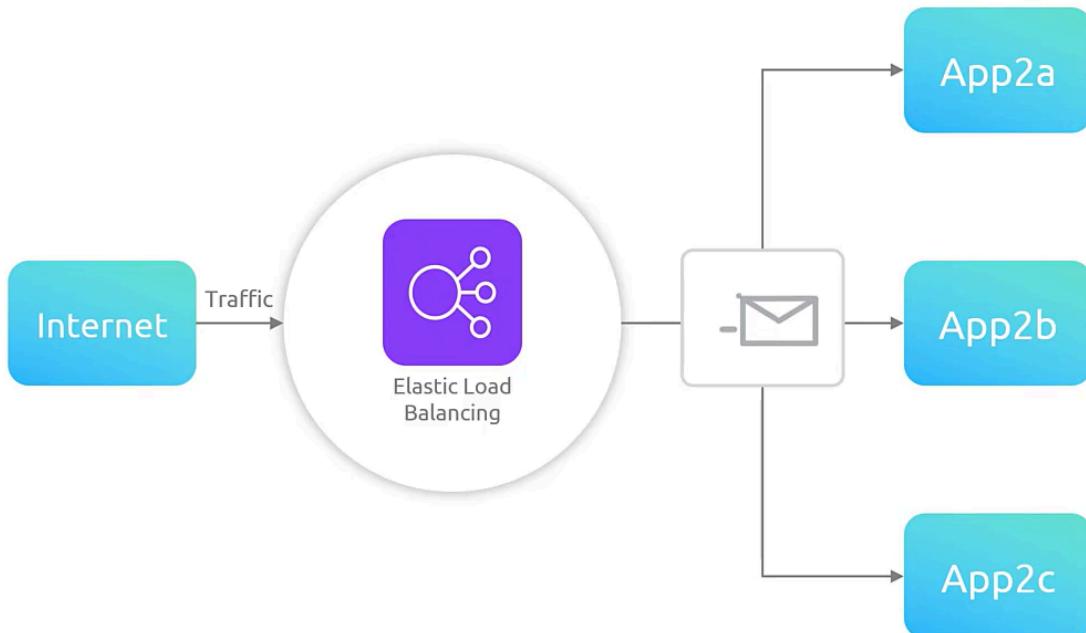
Application integration is something that manages or enhances the communication between application.

## App to App



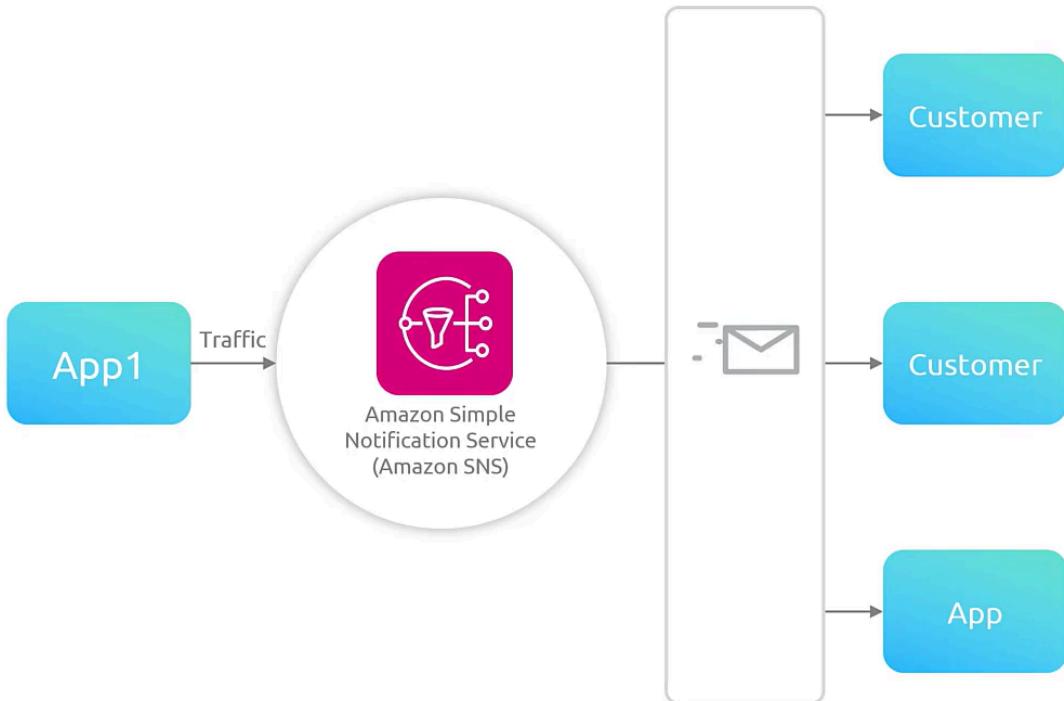
image\_122.png

## Internet to App



image\_123.png

## App to customer



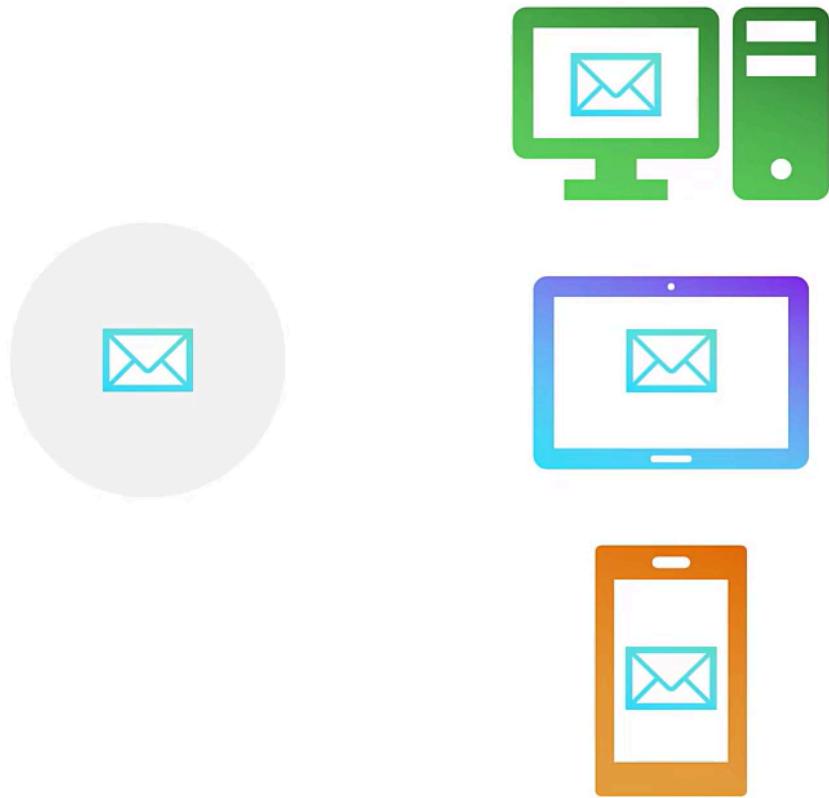
image\_124.png

Application integration is managing the flow, quality, buffering, speed, rate of communication between two application

## Amazon Simple Notification Service

Is a service used between two application in order to duplicate a message and send to multiple customers.

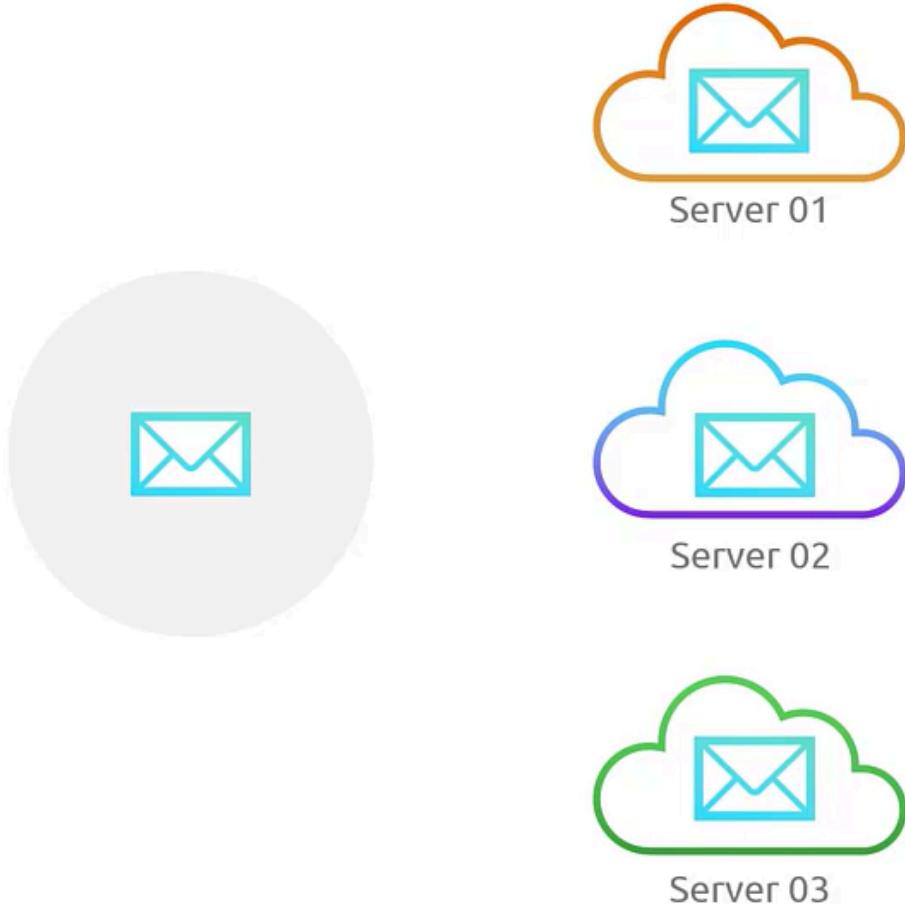
SNS is used when you want an application to send messages to customers via text, email, or mobile push or when you want to copy a single message to multiple applications.



image\_125.png

## Amazon Simple Queue Service

SQS is used when you want to send a message to another application, but there is a chance that a sudden increase in user traffic could generate a large amount of message. The orders will just queue until your backend can process them.



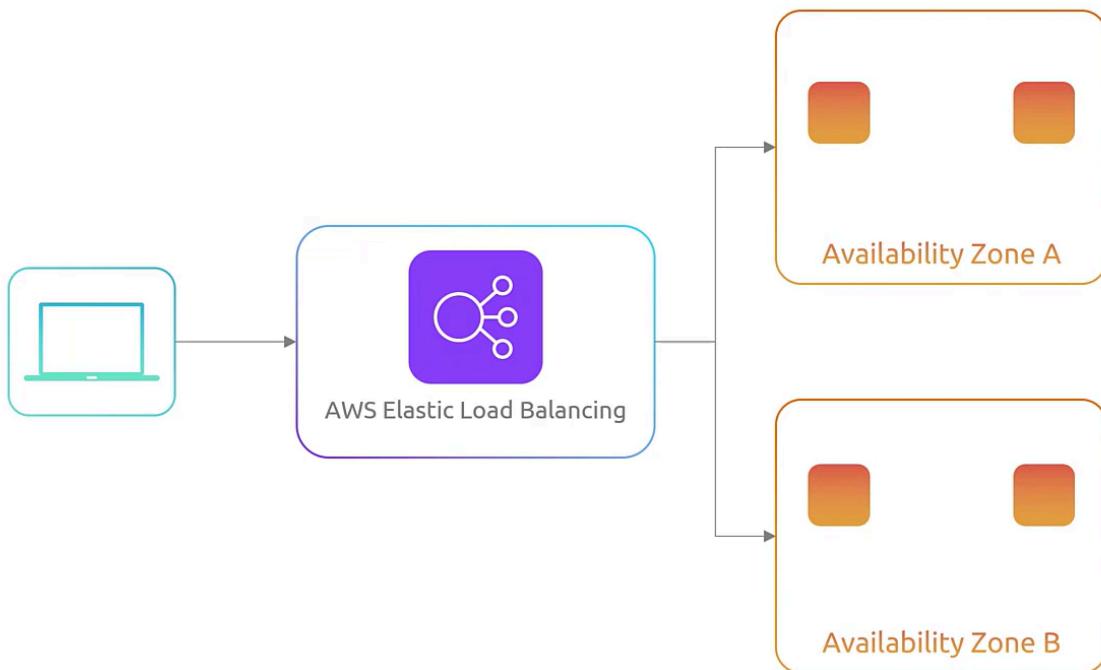
image\_126.png

While SNS duplicate SQS will make sure that you won't overload your backend.

## Amazon Elastic Load Balancing

Used to direct traffic to backend servers and distribute workloads evenly across servers.  
Unhealthy servers are not available if failing.

Can be used with EC2, ECS, EKS and lambda among others.



image\_127.png

## Amazon AutoScaling

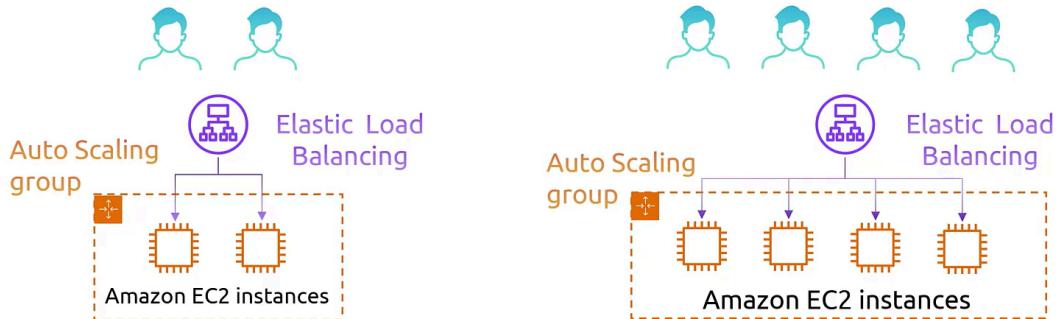
A companion service that is used with elastic Load Balancer (ELB). It monitors traffic and checks the number of people waiting in line and if I get a certain number of people then it's going to add more servers to allow that traffic to be handled.

Many applications have autoscaling like DynamoDB and EC2.

Allows for scale up and down to numbers you specify.

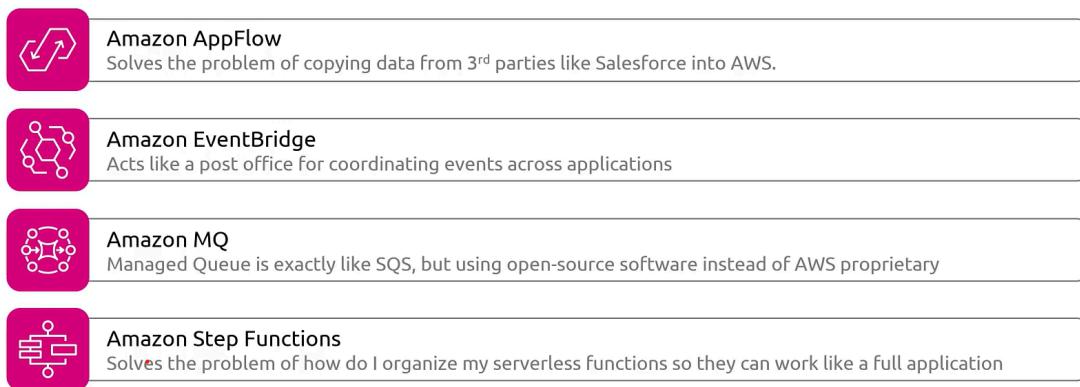
Like 2 instances minimum and 4 instances maximum( shown below)

Scales as you need to within your limits.



image\_128.png

## Other services



image\_129.png

# AWS Migration Services

## Overview



image\_133.png

## Cloud Adoption Framework



image\_132.png

*What if you needed to track all the migration you had*

## Migration Hub

Allows you to centralize and see all migration you have in one place via AWS services.

*What method would you use to migrate*

Aws has six methods or patterns for migration. Which one you use depends on your business, time , money, effort, or other factors important to your business

### **Rehosting (Lift-shift)**

Take the server as it is and copy it in the cloud without changing anything the app or database or even operating system. Has least amount of benefit but its easy and fast.

### **Replatforming ( Lift, Tinker, and Shift)**

Here you change the underlying system eg operating system or managed services

### **Refactoring(or Rearchitecting)**

Take traditional server change it radically and you re-imagine your system architecture.

### **Repurchasing**

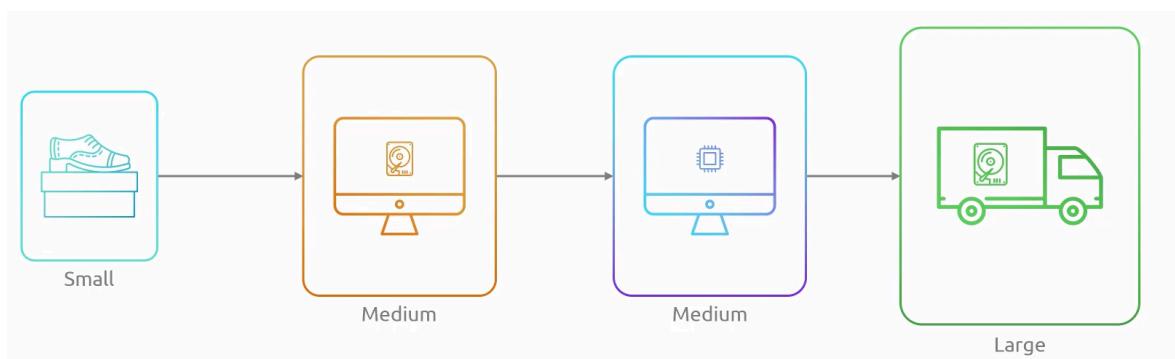
Here you have a licence on traditional side but you Just repurchase on AWS and copy the data maybe.

### **Retaining (Do Nothing)**

Keep traditional data center till it retires

### **Retire**

*What if you need to move data to AWS \_What if you needed a device that could plug into your network , copy data and sent to AWS when full to upload it for you.*



image\_134.png

## **The Snow Family**

### **Snowball Edge Storage Optimized**

Small Compute Medium Data Data Transfer 80TB

## Snowball Edge Compute Optimized

Medium Compute Small Data Portable Compute

## Snowmobile

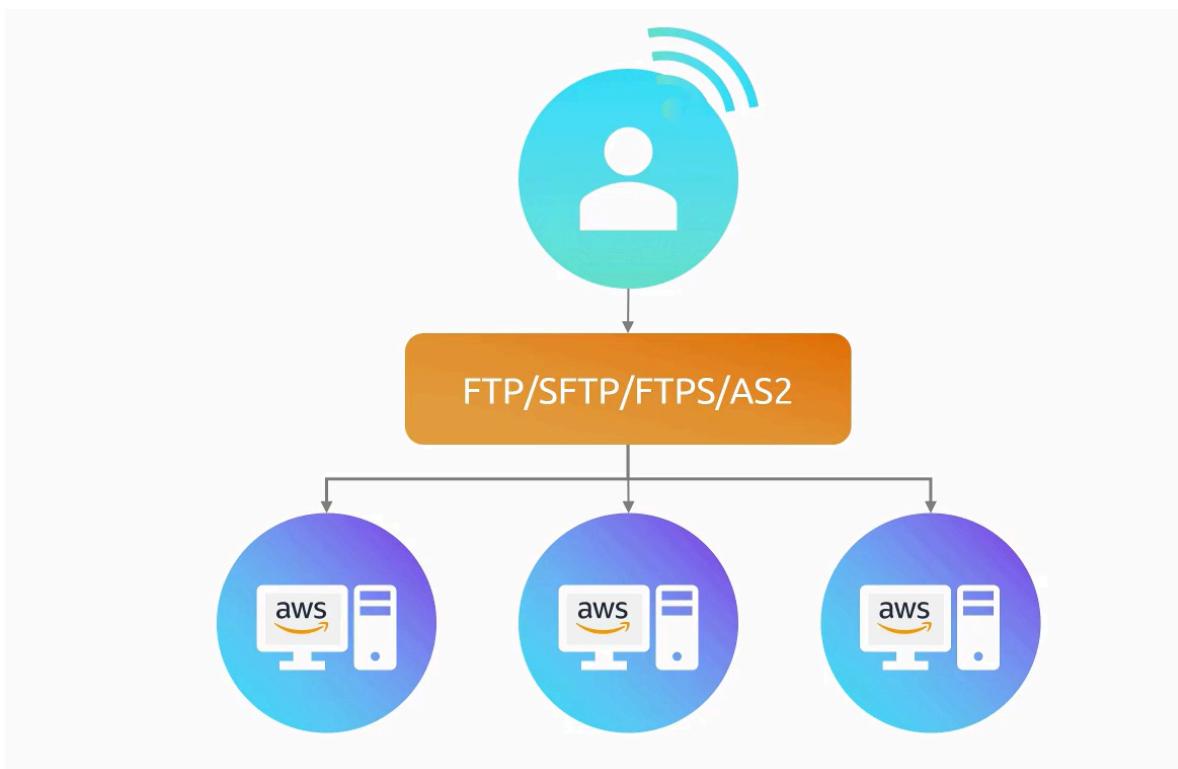
No compute Large Data 100PB of data

## Snowcone

Small compute Small storage Portable compute

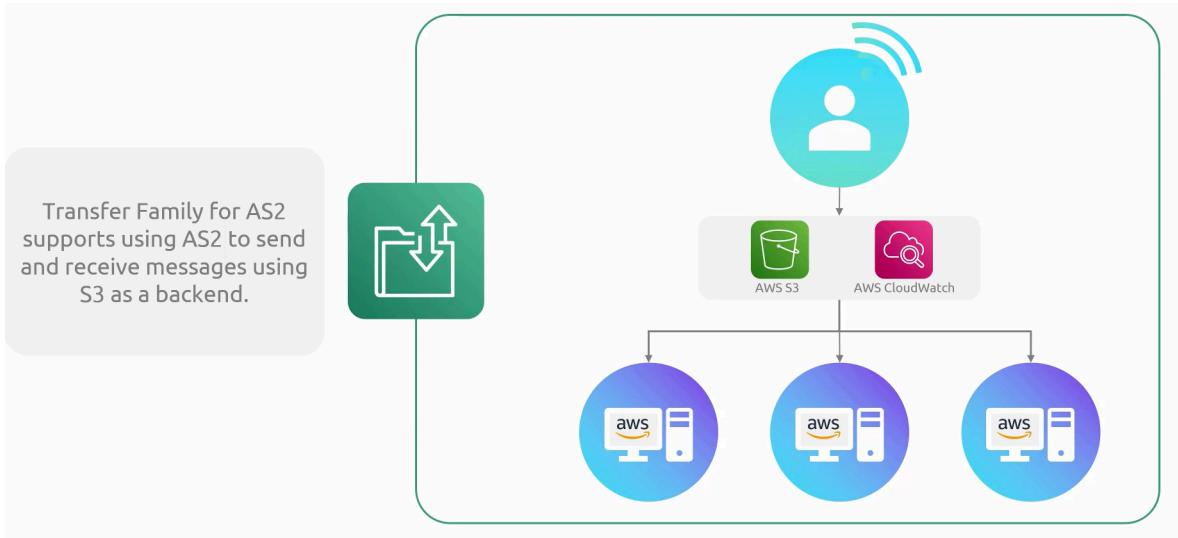
*What if you needed to move data online?*

Transfer Family for FTP supports FTPS,SFTP, and more for secure Transfer to S# or EFS.



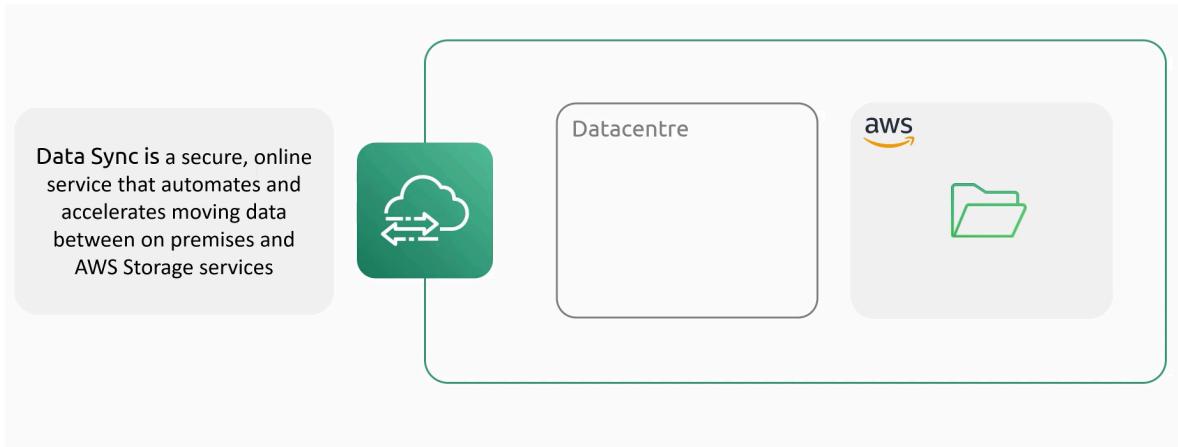
image\_135.png

AS2 version



image\_136.png

## AWS DataSync

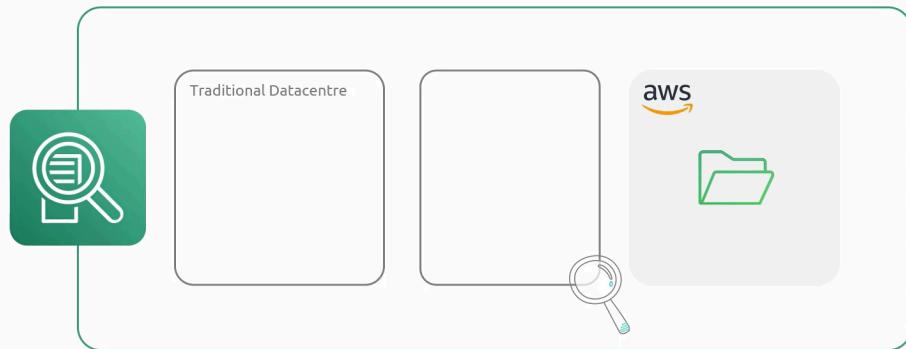


image\_137.png

Can We Transfer a whole datacenter?

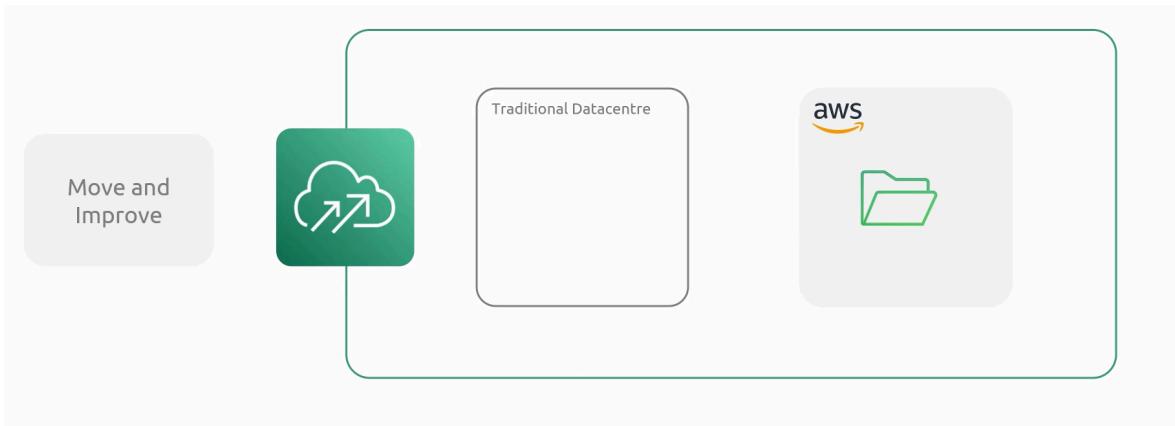
First Discover

What can I use to discover information about apps I am migrating?



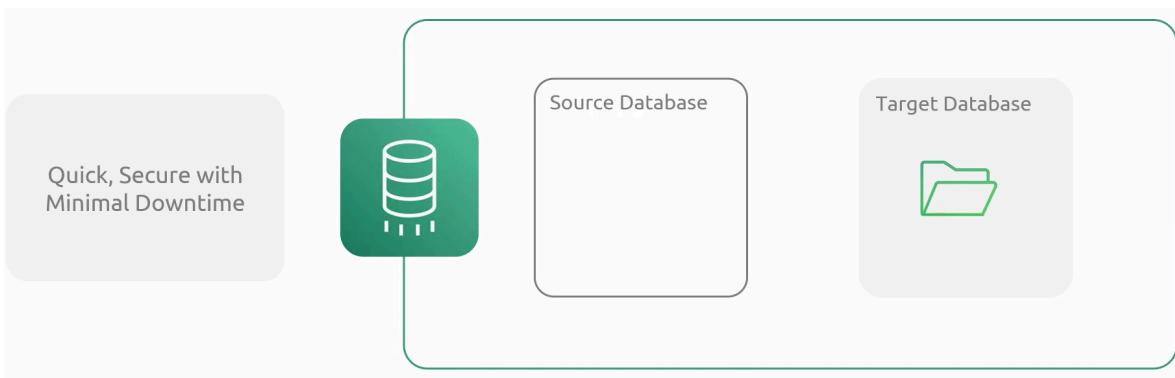
image\_138.png

## Move and Improve



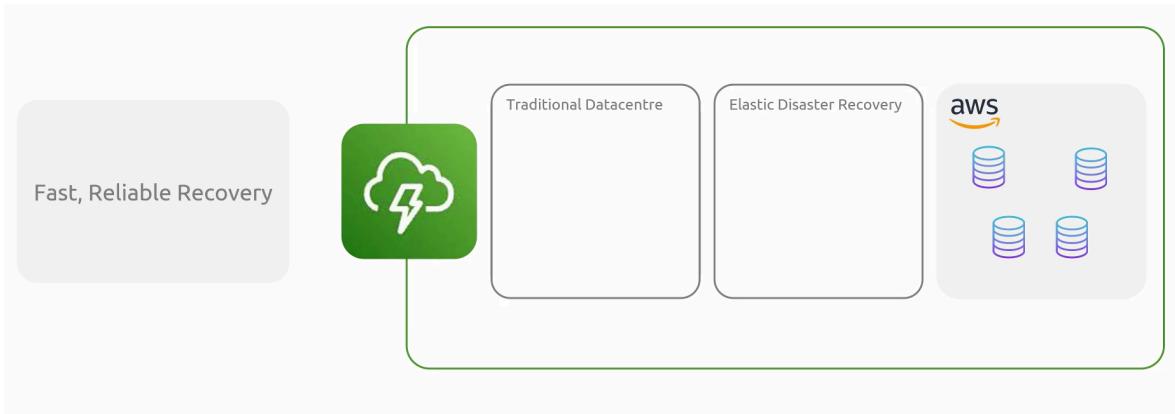
image\_139.png

## For Databases



image\_140.png

## ELastic Disaster Recovery



image\_141.png

## Mainframe modernization



image\_142.png

# AWS Management Services

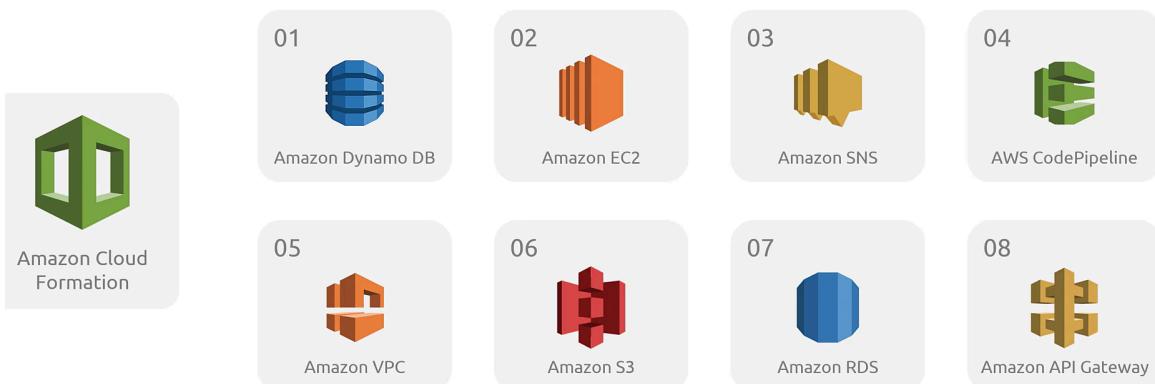
These are services that help manage other services. These services either provision other aws service or optimize other AWS services

How do we create AWS services .. but you need 1000 of them .

*What if we could order services how we order food i.e. give me 10 Hotdogs*

## Cloud Formations

cloud formation creates other services using files.



image\_130.png

You can create all these files by just writing on a file.

\_How now do we create AWS Services and install software on them \_

## AWS OpsWorks

Opswork is a tool that installs specific software into your servers.

*Now you have 100 server how do we manage, patch*

## System Manager

AWS Systems manager is a secure end-to-end management solution for resources(services) on AWS and on-premise environment

*How do you manage all my AWS accounts as if they are one account*

## AWS Organization

AWS Organization is used to centrally govern and manage multiple AWS accounts

*How do I offer my AWS services to my employees like AWS was a vending machine*

## AWS Service Catalog

AWS service catalog allows you to take your cloudFormation and Terraform creation templates and turn them into a vending machine for your users on AWS.

-

## AWS Control Tower

AWS control tower is a service that helps you set up AWS organizations in a secure best practice way, with auditing,logging, and compliance rules in place.

*How do I Manage and audit my accounts for malicious activity and track changes*

## AWS Config & AWS CloudTrail

AWS Config is used to record and evaluate all configuration changes in your AWS service

AWS CloudTrail is a service that tracks and records all user and API activity in your AWS account. Everything is always an API call in AWS whether you are using a console,CLI or SDK.

## Other Services



image\_131.png

# AI/ ML

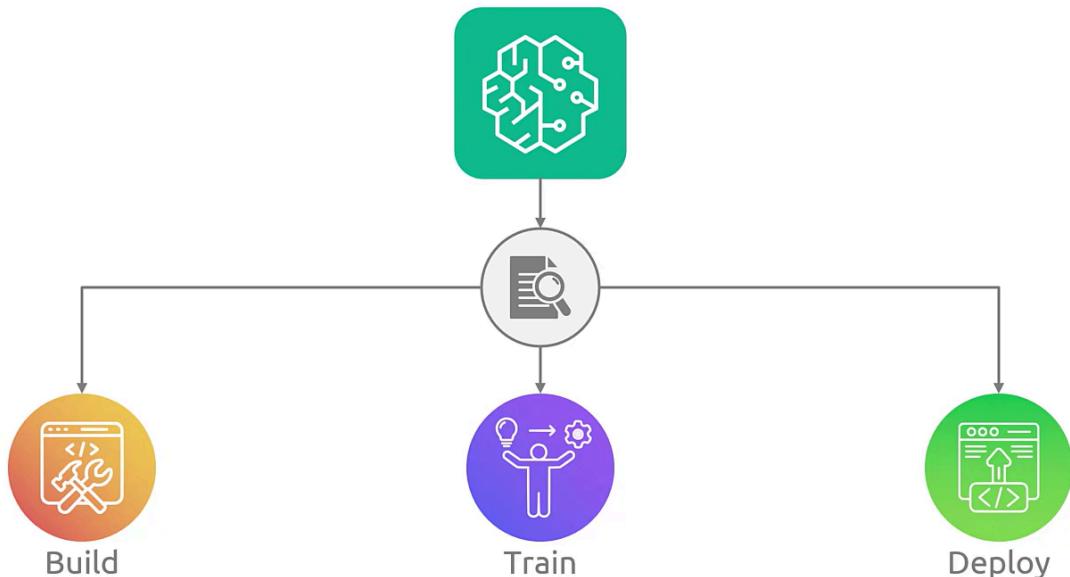
## What is Artificial Intelligence

Anytime we get a machine to do a human task

## What is Machine Learning

Methodology or type of artificial Intelligence.

## Amazon SageMaker



image\_143.png

- Its a fully managed service
- Integrated Jupyter notebook
- Model training and deployments.

## Use Cases

- Predictive analytics and fraud detection

- Recommendation systems
- Facilitating data-driven decisions
- Innovation and automation
- Enhancing customer experiences

## **Relevance to Modern Computing**

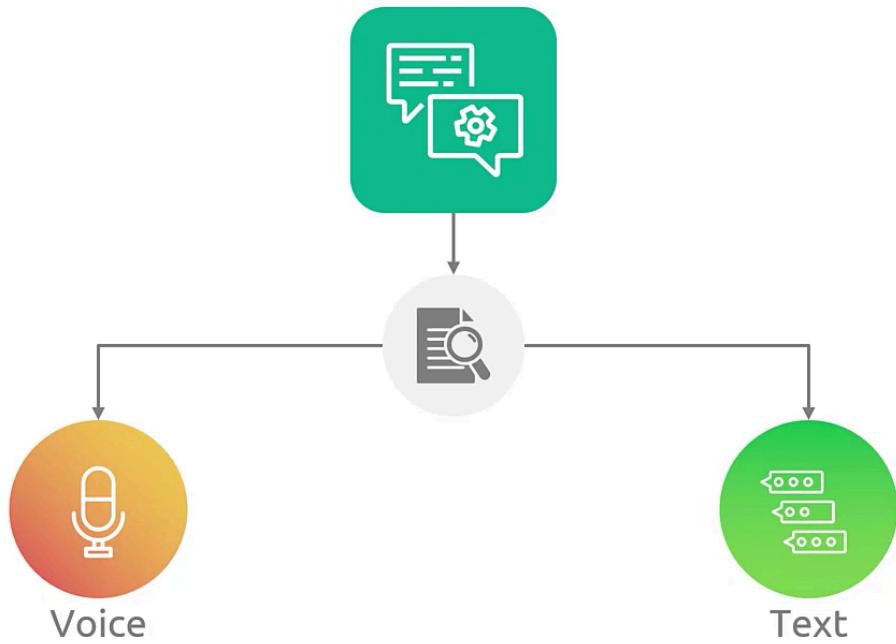
- Enhancing user experiences
- Automation and efficiency
- Accessibility

## **Why choose AWS SageMakers**

- Ease of use Icon
- Scalability
- Comprehensive toolset

## **Amazon Lex**

Think of two people having a conversation except one person is a bot. Lex helps build conversation chatbot using voice and text Natural language Understanding(NLU) and Automatic speech recognition (ASR) are the main technologies used in Lex

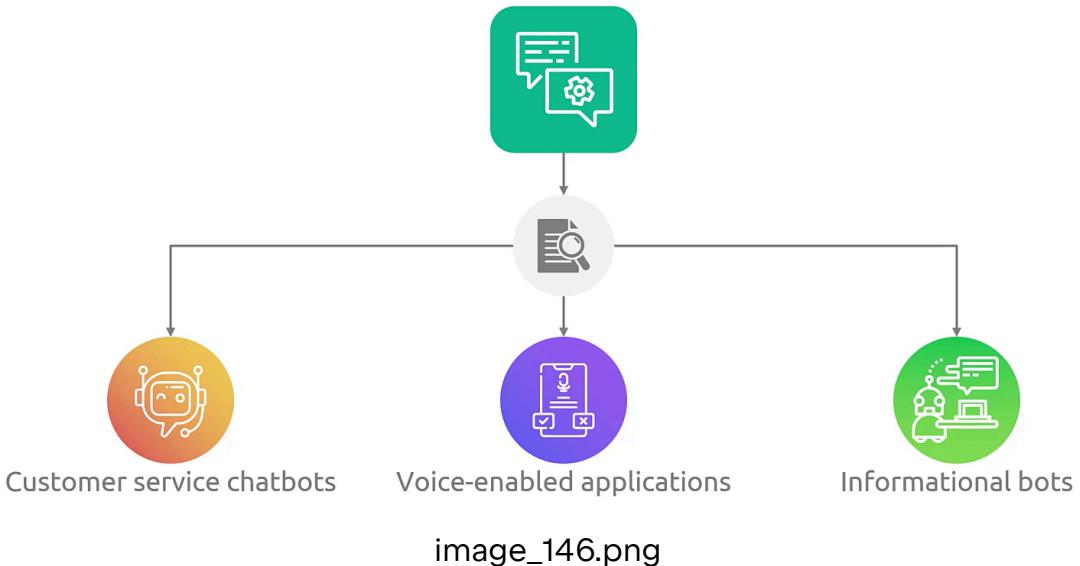


image\_145.png

Used to build chatbots and can easily be integrated with other AWS services

## Use Case

- Customer service chatbots
- Voice-enabled applications
- informational bots



## Relevance to Modern Computing

Enhances user experiences Accessibility Automation and efficiency

## Why choose Lex

- Stands out from other NLP tools
- Building conversational interfaces with lex

## Amazon Kendra

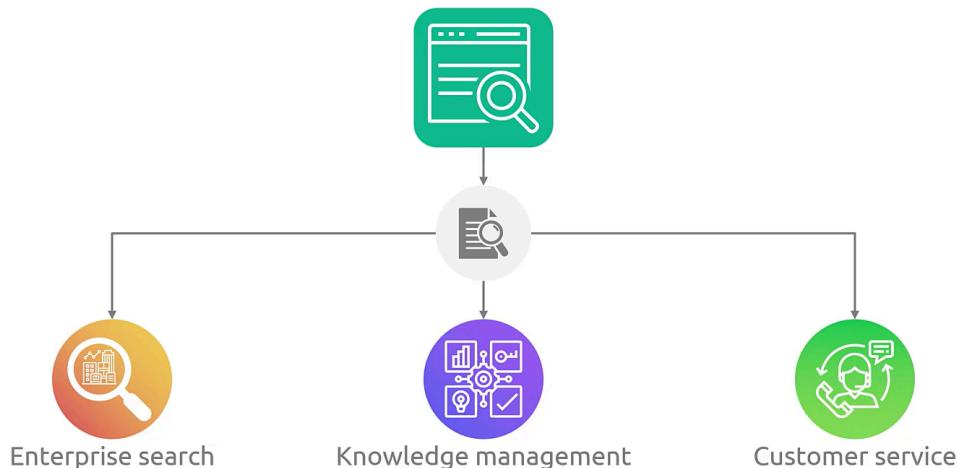
This a highly accurate and easy to use enterprise search service that indexes and manages enterprise documents.



image\_147.png

It's more like a google search engine for your enterprise

## Use Cases



image\_148.png

## Relevance to Modern Computing

- Facilitating informed decision making
- Enhancing productivity
- Improving customer satisfaction

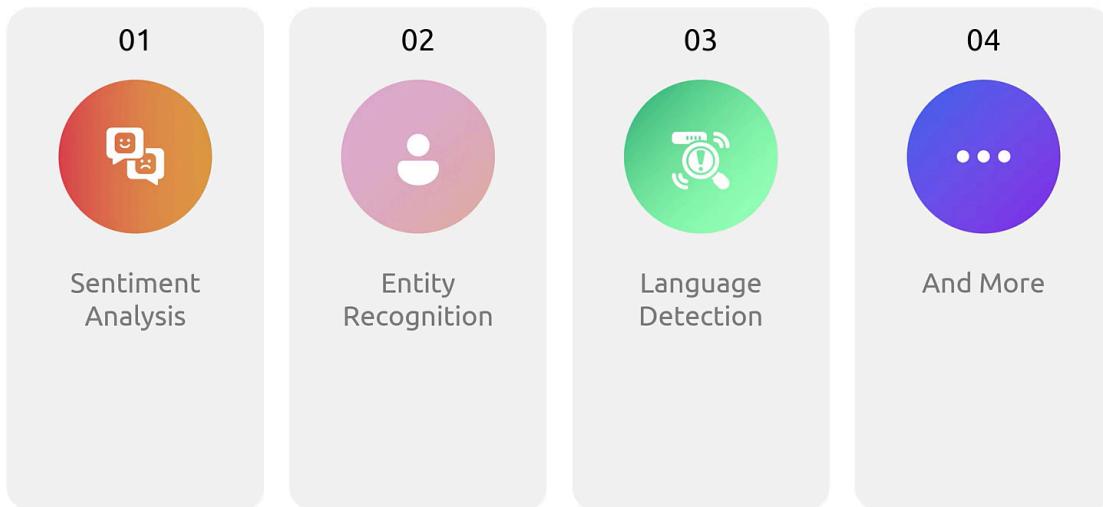
## Why kendra

- Stands out from other Enterprise search solutions
- Accuracy and efficiency

## Amazon Comprehend

This service provided incites and relationship with text

Uses Natural language processing (NLP) and deep learning to extract information such as intend, sentiments and relationships.



image\_149.png

## Use Cases

- Compliance monitoring
- Social media monitoring
- content personalization

## Why Amazon Comprehend (Detective)

- Standouts from other NLP tools

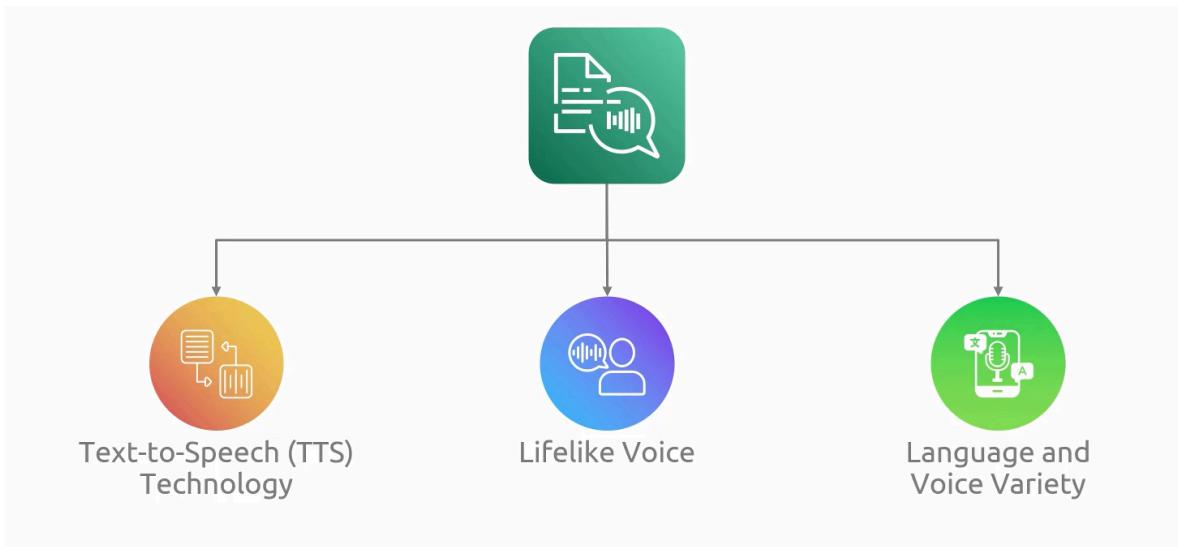
- scalability and security

## Relevance to Modern Computing

- Facilitate data-driven decision
- Enhancing customer experiences
- Enables automation and efficiency
- Aid innovation in product development

## Amazon Polly (Voice studio)

Take text and convert to speech



image\_150.png

## Use Case

- Voice assistant
- Content narration
- Educational tools
- Accessibility

## Relevance to Modern Computing

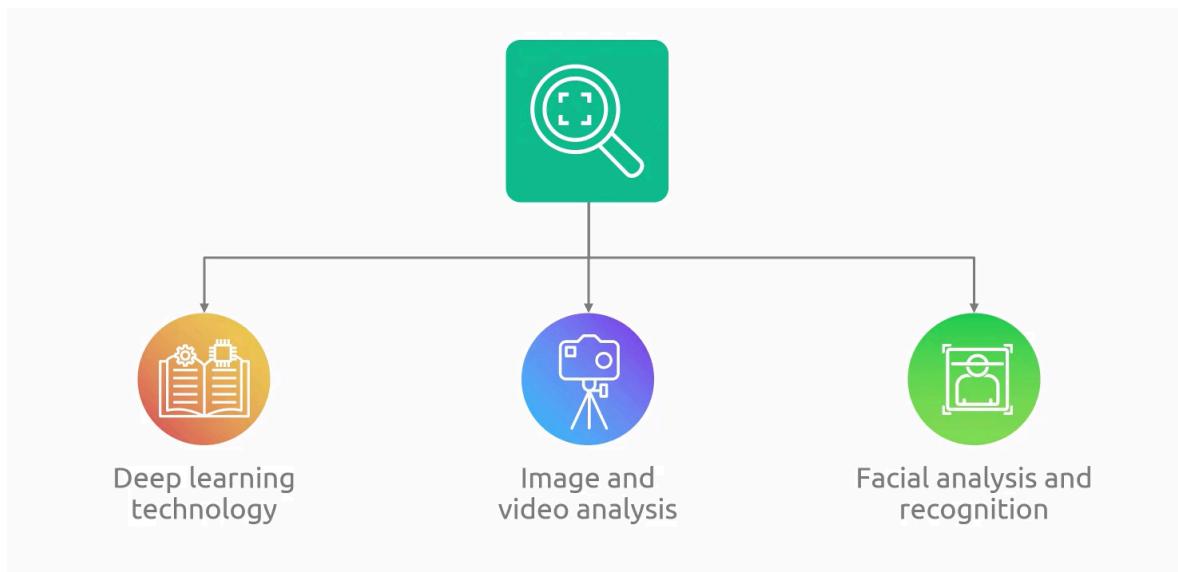
- Enhance user engagement
- Enables accessibility and inclusivity
- Aid innovation in Content Delivery

## Why Polly

- Standouts from other TTS solutions
- High quality voice output and customization options

## Amazon Rekognition

Analyzes Images and Video and tells you what is in the media.



image\_151.png

## Use Case

- Security and Surveillance
- Content moderation
- Customer engagement

- Accessibility

## Relevance to Modern Computing

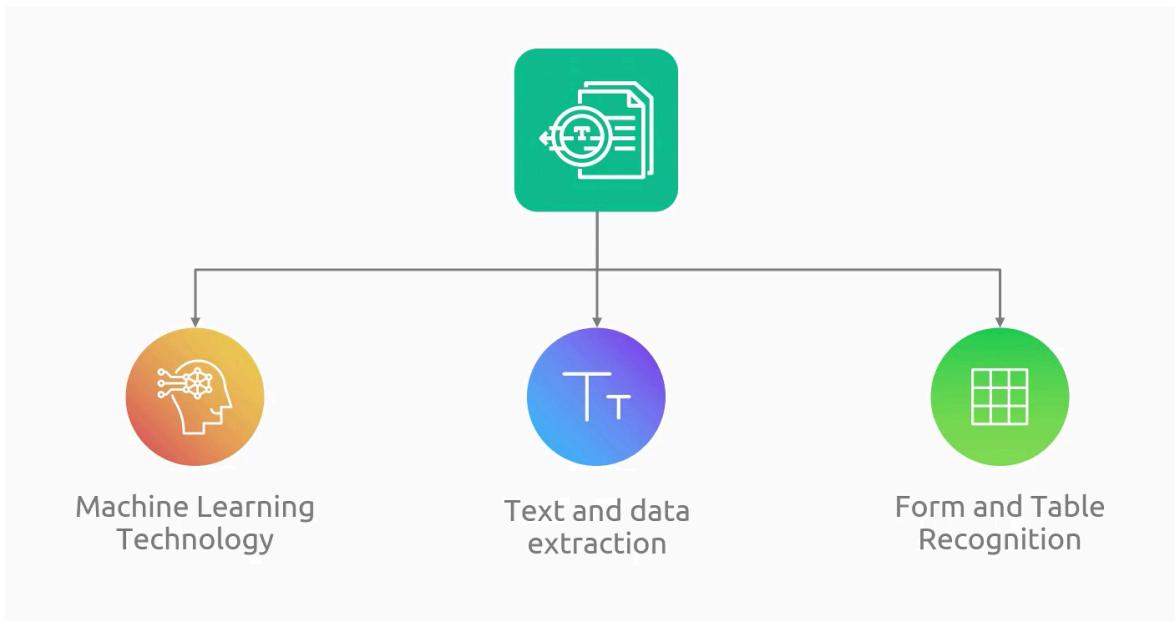
- Enhancing safety and security
- Automation and efficiency
- Aid innovation in customer experiences

## Why Rekognition

- Standouts from other Image and video analysis tools
- Scalability and security

## Amazon Textract

Textract Uses ML technology to extract text and data from documents and its able to recognize form and Tables



image\_152.png

It's an Image to PDF text Extraction

## Use Case

- Automated Data Entry
- Content Migration
- Compliance Monitoring
- Search and Discovery

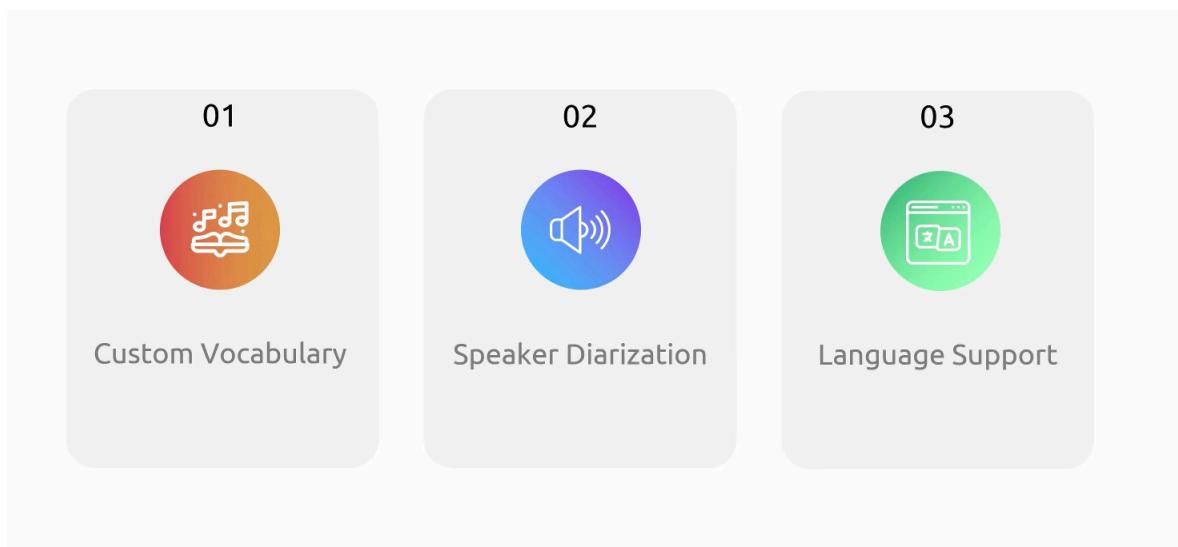
## Why Textract

- Enhance efficiency
- Facilitating digital transformation
- Efficiency

## Amazon Transcribe ( Recording studio)

This is your audio to text recorder

Key Features



image\_153.png

## Use Case

- Content Creation
- Subtitling and captioning

- Voice analytics
- Accessibility

## **Relevance to Modern Computing**

- Enhance accessibility
- Facilitating content creation
- Enabling Innovation in analytics

## **Why Choose Transcribe**

- Stands out from other transcription services
- Offers accuracy and customization

## **Amazon translate**

Translate uses Neural Machine Translation(NMT)technology to bridge language gaps

## **Use Case**

- Content localization
- customer support
- Social media monitoring
- E-learning

## **Relevance to Modern Computing**

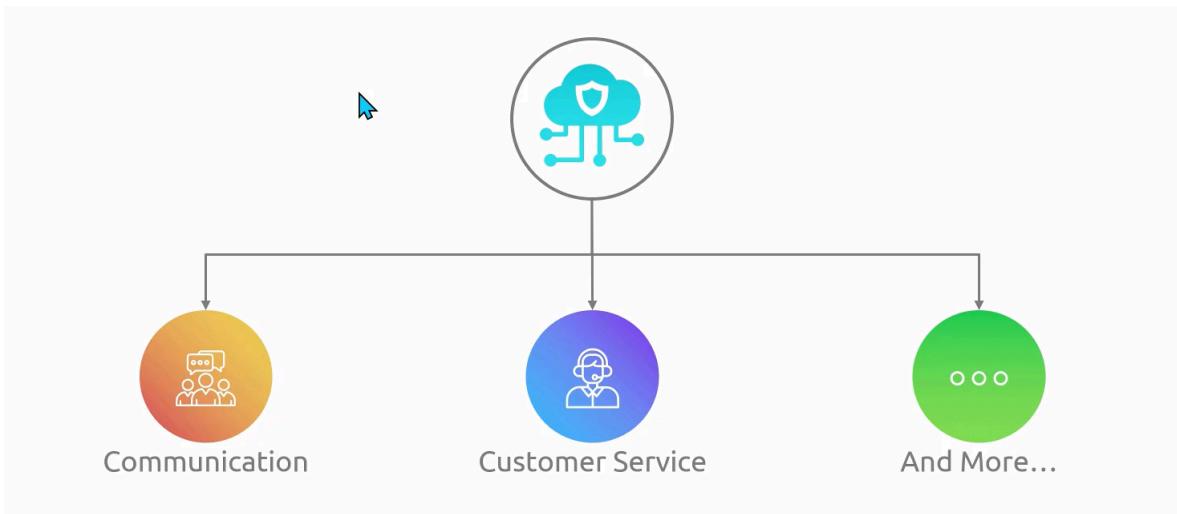
- Global reach
- Facilitating communication
- Innovation in content delivery

## **Why Translate**

- Stands out from other tNLP tools
- Offers accuracy and customization

# Business Application Services in AWS

To solve a business problem we need:



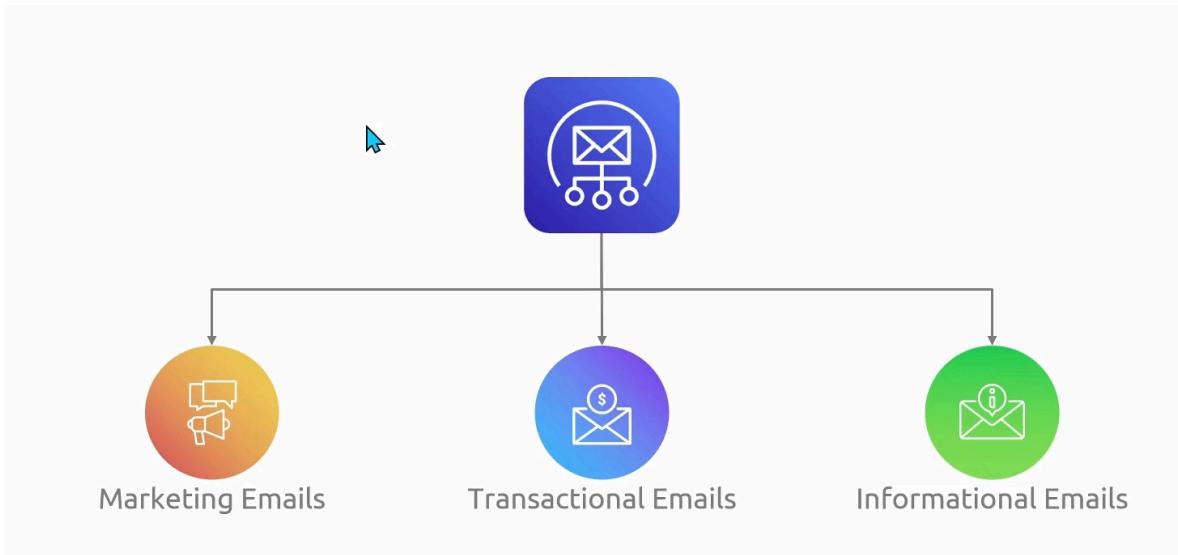
image\_159.png

There are two types of business applications:

- Amazon Simple Email Service - Business Communicator
- Amazon Connect - Customer Service Desk.

## Amazon Simple Email Service

This is a business communicator whose main job is to send mass emails.



image\_160.png

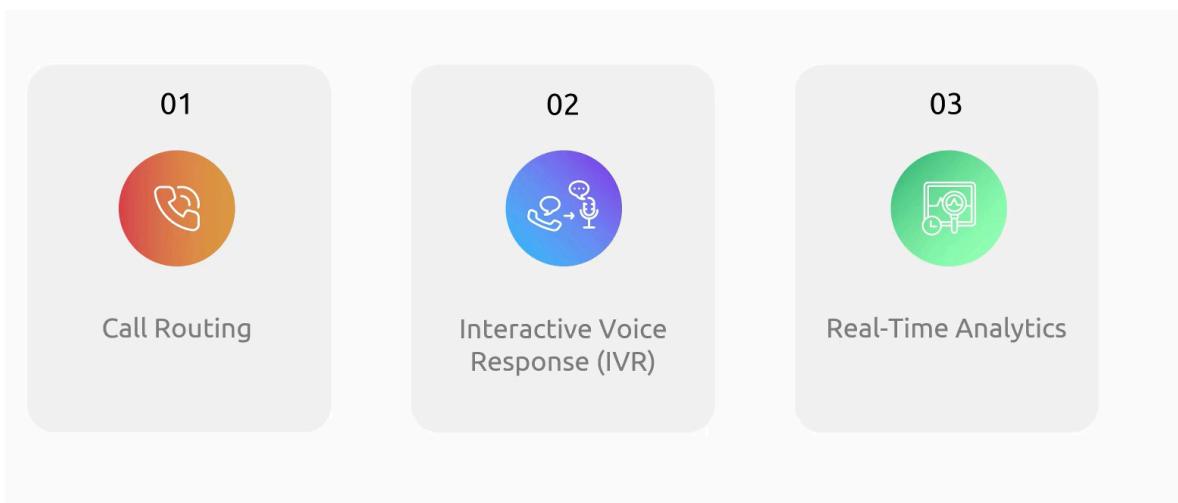
Its main job is just to send emails in a reliable, scalable way possible.

It's main advantage include:

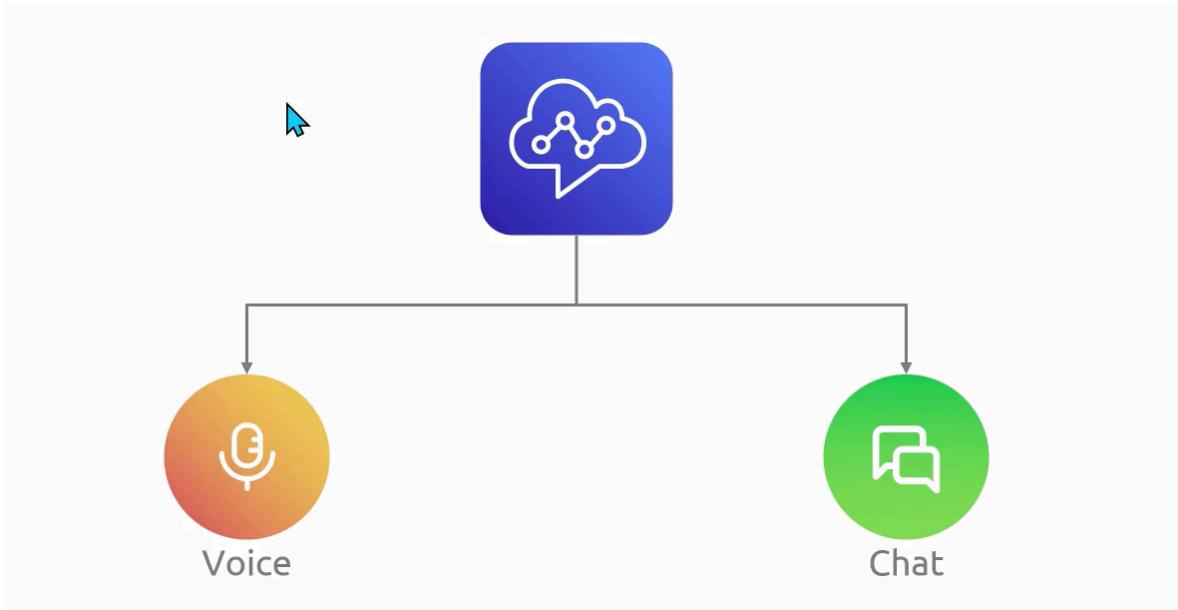
- Scalability
- Integrates seamlessly

## Amazon Connect

It helps in handling customer interactions

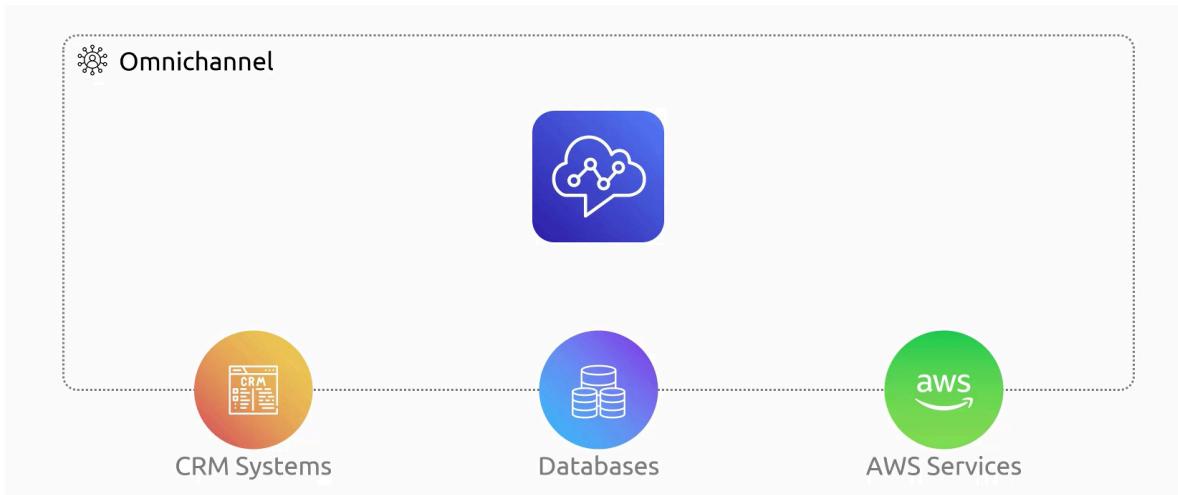


image\_162.png



image\_161.png

it can also connect to other channels to help in giving a better service to customers



image\_163.png

# Customer Engagement Services

These services create engagement between AWS and the customer.

## AWS Activate for startups

Here they give advice and discounts on any project that a start up has and might be too costly for the startup. It offers components and structure plus some guidance for taking your idea from a concept to a piece of software.

It Helps with:

- Providing credits to access AWS services
- Offering technical support and training
- Facilitating access to AWS's startup community

## Use cases

- Early-stage startups leveraging AWS credits to build their infrastructure
- Startups utilizing technical support for scaling their service
- Networking and learning from the AWS startup community
- Utilizing training resources to build a skilled team

## AWS IQ

Connects you to certified experts with a great understanding of AWS.

It Helps with:

- Facilitating connections between AWS experts and clients
- Offering a secure platform for project collaboration.

## Use Cases

- Startups finding their right experts for their AWS projects
- Businesses leveraging AWS IQ for technical guidance and project execution
- AWS experts finding opportunities to work on diverse projects
- Clients receiving personalized assistance and solutions
- 

## AWS Managed Services (AMS)

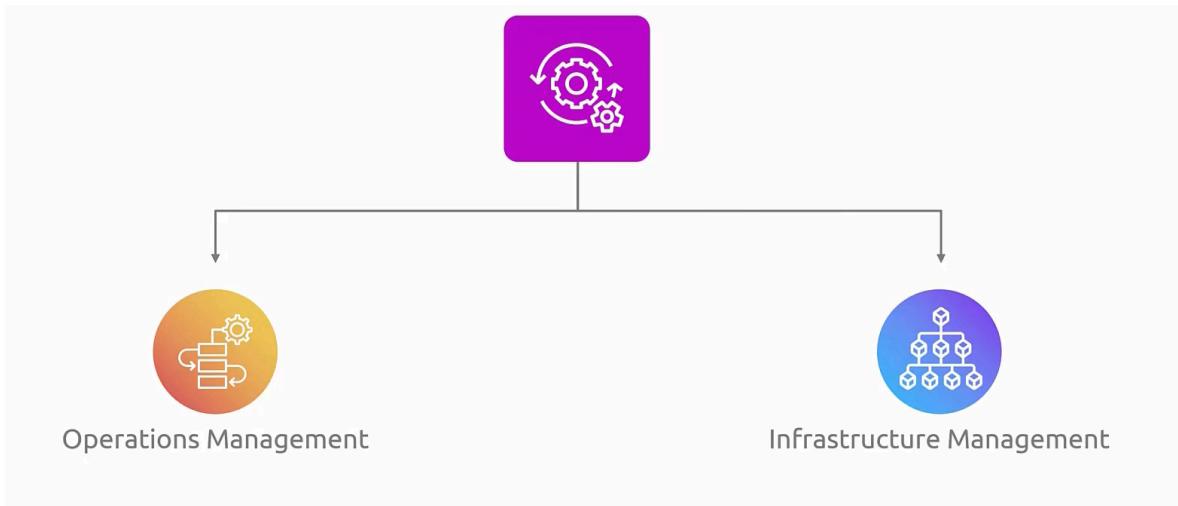
Helps with operations and infrastructure management and provides the best advice about them.

It helps with :

- Infrastructure operations management
- Security assurance and compliance
- Enterprise-level businesses leveraging AMS for infrastructure management.
- Startups utilizing AMS for cost-effective and secure operations

## Use Cases

- AMS adding in migration and cloud adoption processes
- Utilizing AMS for continuous compliance and automated operations



image\_164.png

## AWS Support

Support is a customer touch point which include forms,e-mails supports Has 5 Tiers:  
Basic, developer, business , enterprise

Helps with:

- Offering technical support and guidance
- Providing infrastructure event ,management.

## Use Cases

- Businesses leveraging AWS support for technical guidance
- Startups utilizing AWS support for smooth operations
- AWS Support aiding in crisis management and resolution
- Utilizing AWS support for proactive infrastructure management
-

# AWS Developer Services

## AppConfig

It's going to streamline your configuration :

- Managing configuration
- Deploying Configurations

## Features

- Controlled Deployments
- Validations Checks
- Monitoring

## Use Cases

- Enables applications tuning
- Features flag management for applications
- Ensures safe deployments by gradually rolling out changes

## Relevance to Modern Computing

- Facilitate quick and safe deployments
- Maintains stable application environment
- Enables agile adaptation to changing needs

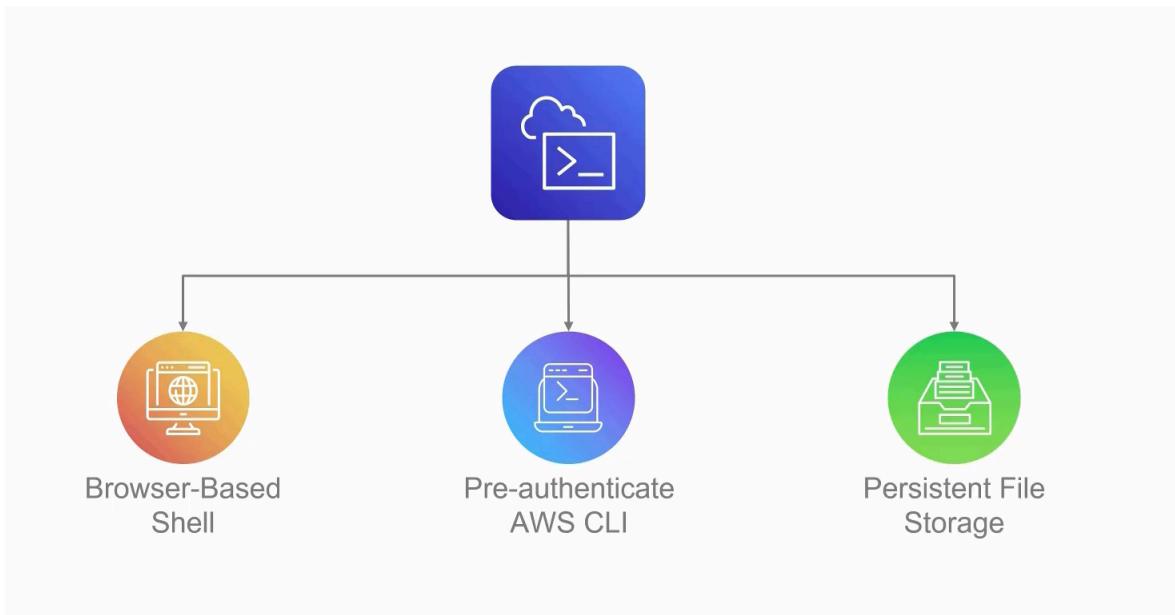
## Why AppConfig

- Integrates seamlessly with development tools

- Helps enhance efficiency and reduce errors

## AWS CloudShell

It's a browser based accessible shell to manage an aws service



image\_165.png

## Use Cases

- Running scripts and commands directly in the browser
- Managing AWS resources more efficiently
- Quick access to AWS services without any setup

## Relevance to Modern Computing

- Facilitating remote work and collaboration
- Enhancing security with pre-authenticated AWS CLI
- Speeding up access to AWS services

## AWS Cloud9 Sky

This basically an integrated developer environment.



image\_166.png

## Use Cases

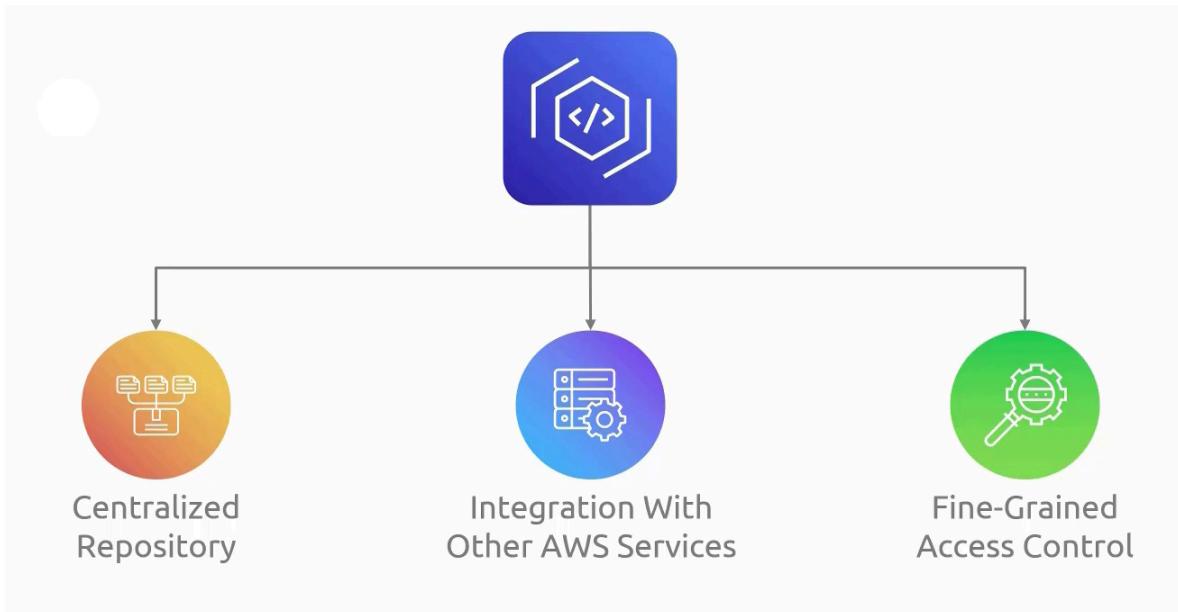
- Collaborative coding
- Serverless application development
- Educational purposes for coding tutorials

## Relevance to Modern Computing

- Facilitates remote work and collaboration
- Offers pre-configured environments
- supports many programming languages

## AWS CloudArtifact

Your managed artifact repository Its main function is to host the build code , not the code( code is in a repository)



image\_167.png

## Use Cases

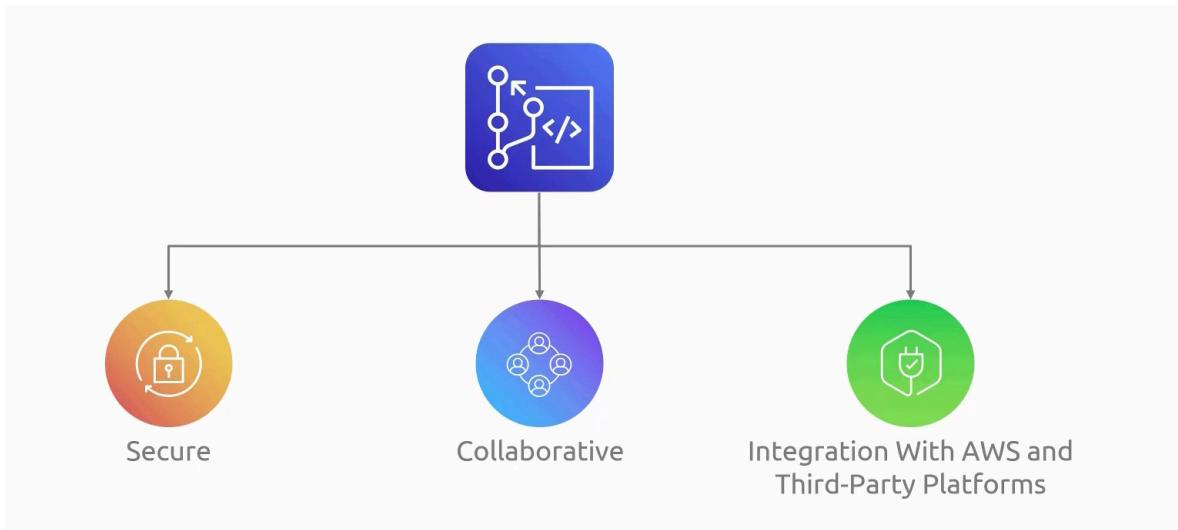
- Ensuring secure and centralized artifact storage
- Streamlining development workflows
- Integration with popular build tools and package managers

## Relevance to Modern Computing

- Facilitating secure and efficient software development
- Enhancing collaboration while maintaining security
- Compliance with industry standards

## AWS CodeCommit

This is where the raw code lives, Acts just like Git repository.



image\_168.png

## Use Cases

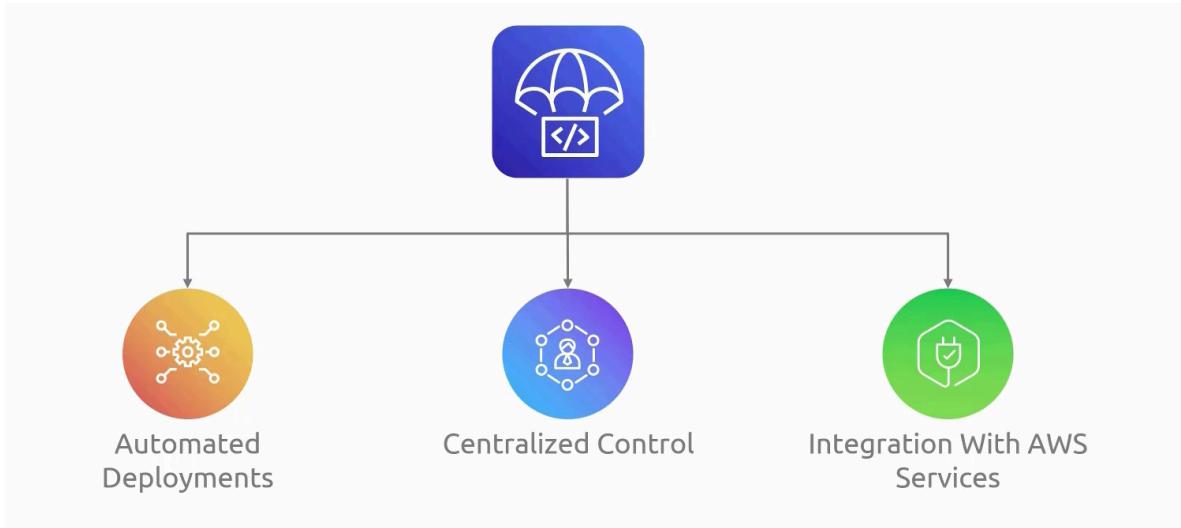
- Source control for enterprise
- Collaborative coding
- Integration in CI/CD pipelines

## Relevance to Modern Computing

- Facilitating secure code storage
- Facilitating DevOps practices
- Compliance with industry standards

## AWS CodeDeploy

Its main function is to deploy the build code.



image\_169.png

## Use Cases

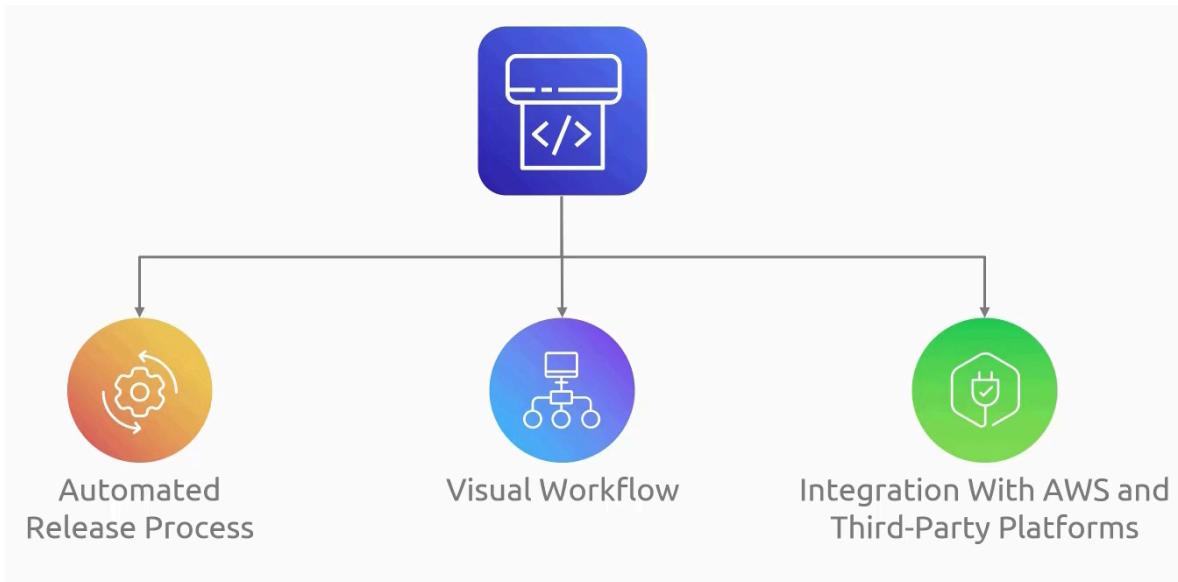
- Rapid release cycles
- Blue/Green deployments
- Automated rollbacks

## Relevance to Modern Computing

- Facilitating DevOps practices
- Ensuring application availability
- Maintaining uptime in deployments

## AWS CodePipeline

It helps with streamlining the release process



image\_170.png

## Use Cases

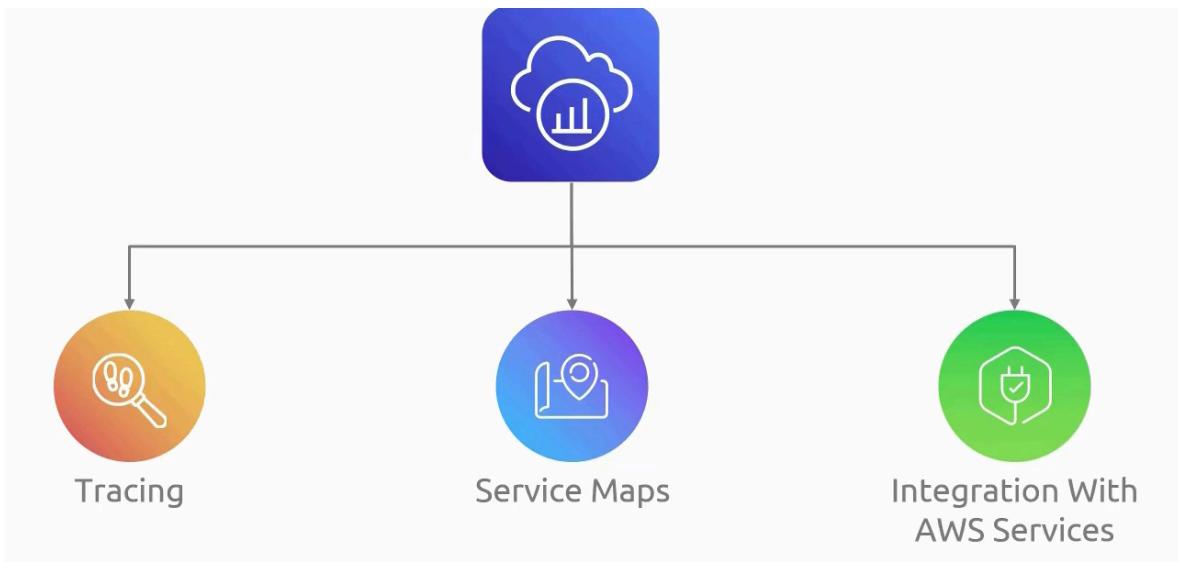
- Continuous delivery and integration
- Rapid iterations
- Collaboration

## Relevance to Modern Computing

- Facilitating DevOps practices
- Speeding up software delivery
- Ensuring code quality

## AWS X-Ray

It runs traces to track what the application is doing



image\_172.png

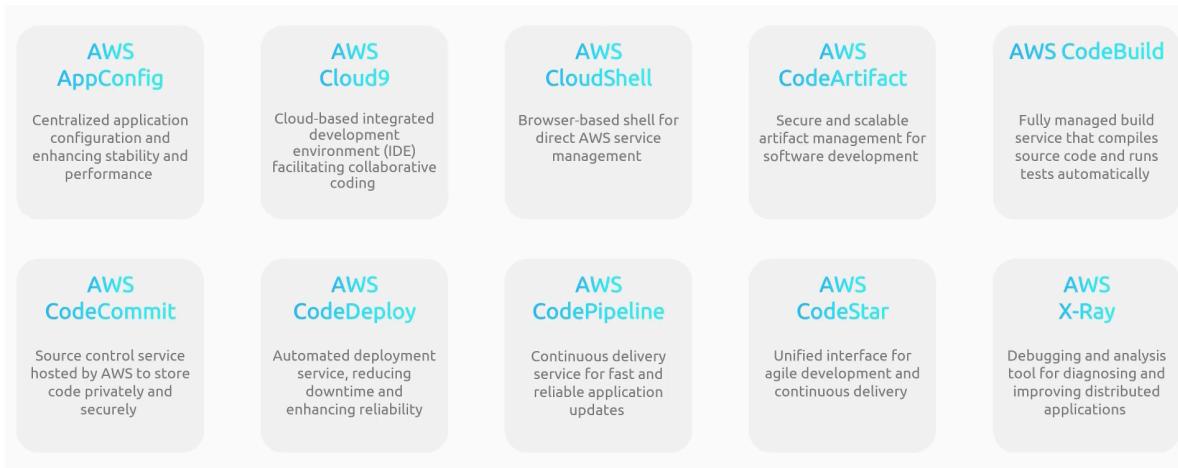
## Use Cases

- Performance optimization
- Debugging
- Monitoring

## Relevance to Modern Computing

- Facilitating DevOps practices
- Ensuring application health
- Compliance with industry standard

## Short Summary



image\_173.png

# AWS End User Computing Services

There are three of them:

- Amazon AppStream 2.0
- Amazon Workspaces
- Amazon WorkSpaces Web

## General Use Case

- Remote Working
- Secure data handling
- Simplified IT management

## Relevance To Modern Computing



image\_174.png

## Amazon Workspaces - Your Virtual Desktop

It's a secure managed desktop service with great customization options This is Desktop as a Service(DaaS)

## Use Cases

- Remote working
- Secure data handling
- Simplified IT management

## Relevance to Modern computing

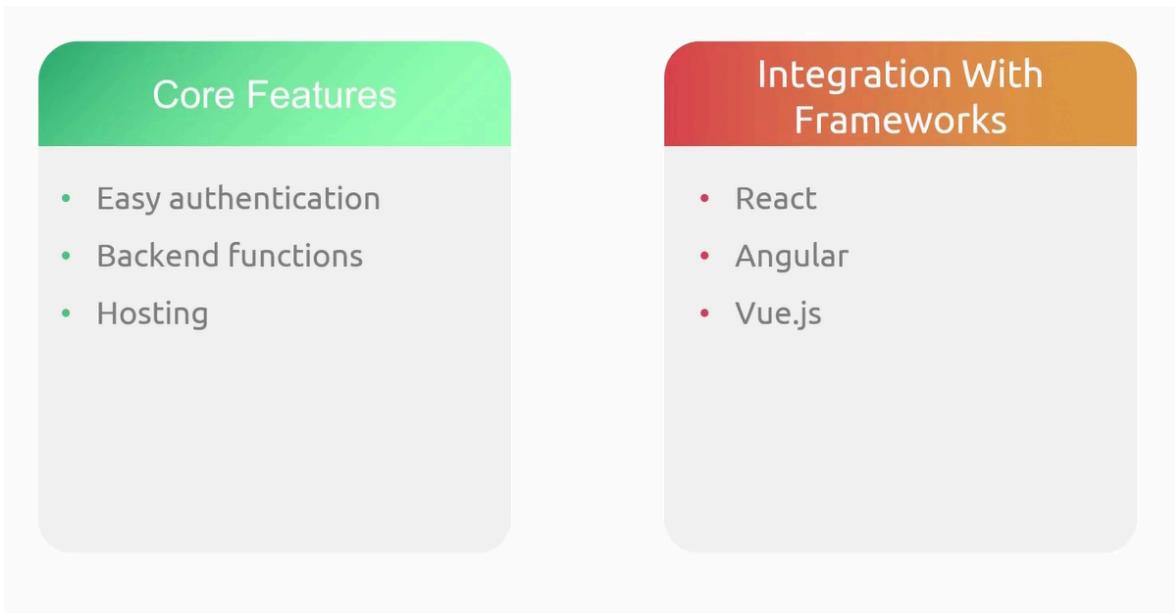
- security
- Scalability

## Amazon Workspaces Web Your Desktop , Anywhere

This is a browser-based access to your workspaces.

# AWS frontend & Mobile

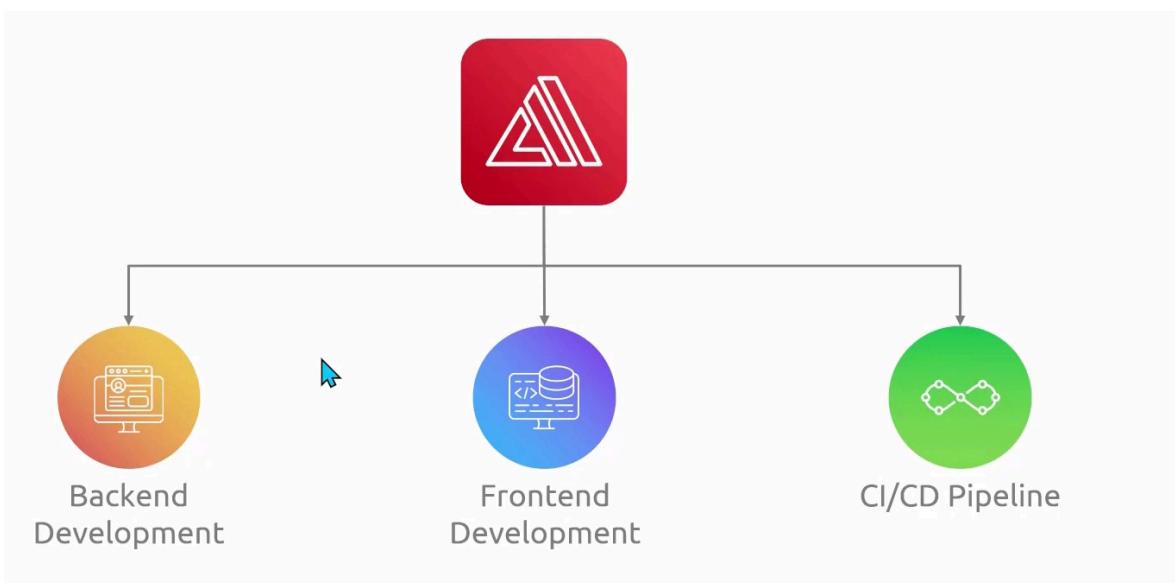
## AWS Amplify



image\_175.png

This is a deployment methodology to build and deploy mobile apps It mainly supports Javascript.

It offers:



image\_177.png

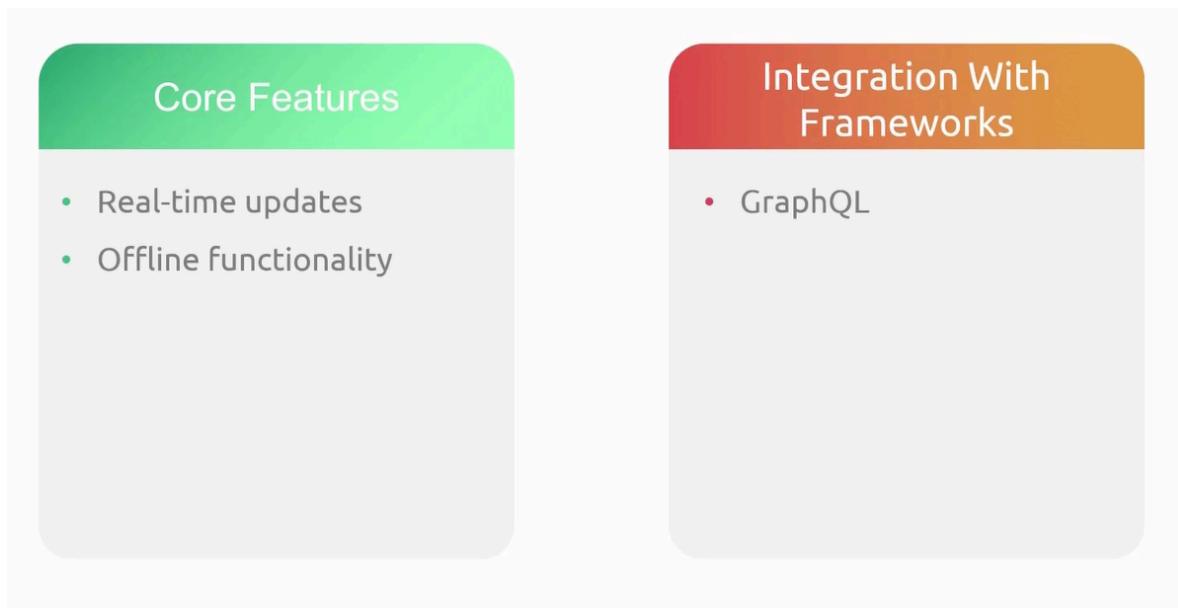
## Use Cases

- Building Dynamic web and mobile apps
- Providing real-time updates

## Relevance to Modern Computing

- Offers speed and efficiency
- Provides scalability
- 

## AWS AppSync



image\_176.png

It's really used for getting API access using GraphQL both online and offline to variety of different API or frontend sources. It's a backend tool that supports GraphQL

## Use Cases

- Building Dynamic web and mobile apps
- Providing real-time updates

## **Relevance to Modern Computing**

- Offers speed and efficiency
- Provides scalability

# AWS IOT

## AWS IOT Core

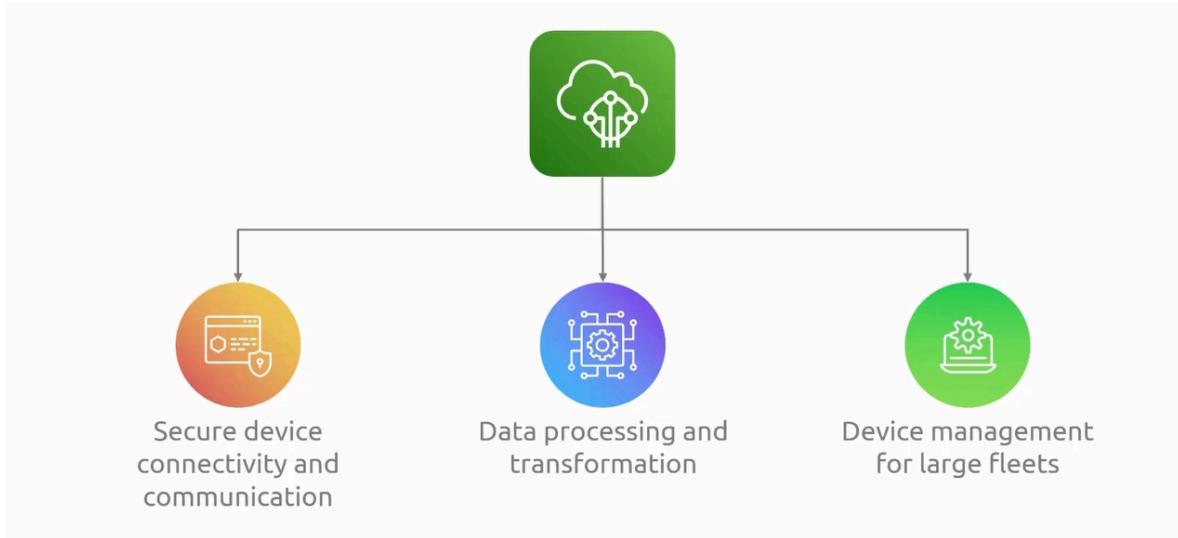
This is the backbone of IOT devices. It is an IOT device management system.

### Core Functionalities

- Device communication
- Security
- Data management

image\_178.png

It accomplishes:



image\_179.png

## Use Cases

- Industrial automation and predictive maintenance
- Smart home solutions
- Connected vehicles and telematics

## Relevance to Modern Computing

- Enabling intelligent IOT solutions
- Facilitating seamless device to cloud integration

## AWS IOT GreenGrass- Your Local IoT Orchestrator

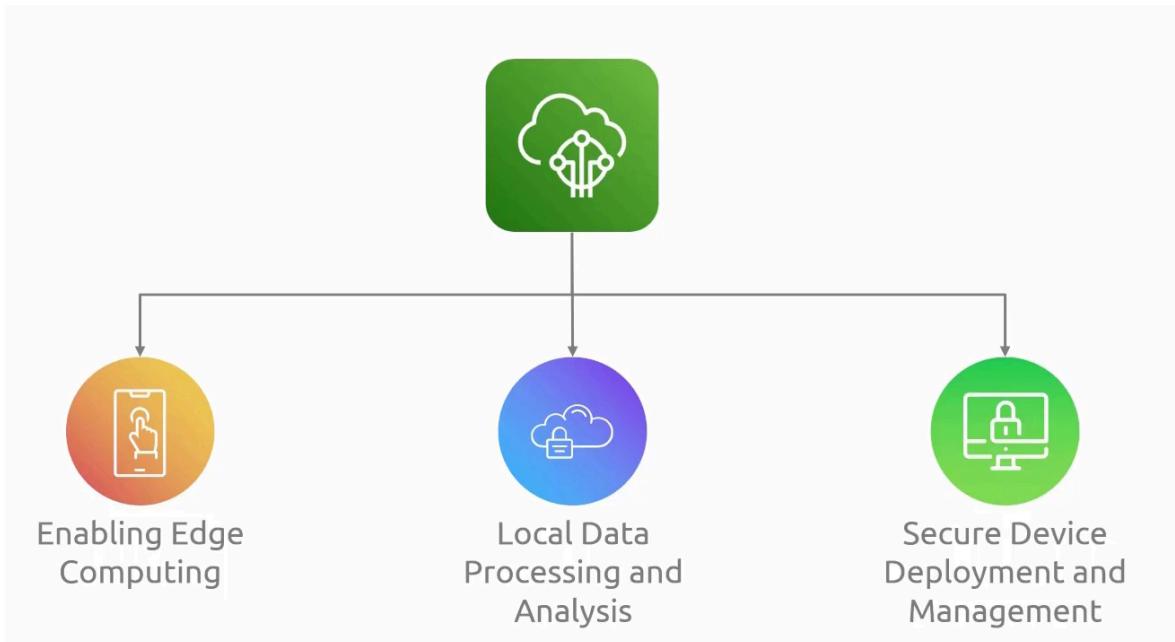
IOT Greengrass is your Local IoT Orchestrator. Its a runtime and also a service Offers:

## Core Functionalities

- Local compute
- Messaging
- Data caching

image\_180.png

Accomplishes:



image\_181.png

## Use Cases

- Industrial automation
- Agriculture
- Healthcare

## **Relevance to Modern Computing**

- Facilitating IOT solutions at the edge
- Enabling offline operations and reducing latency

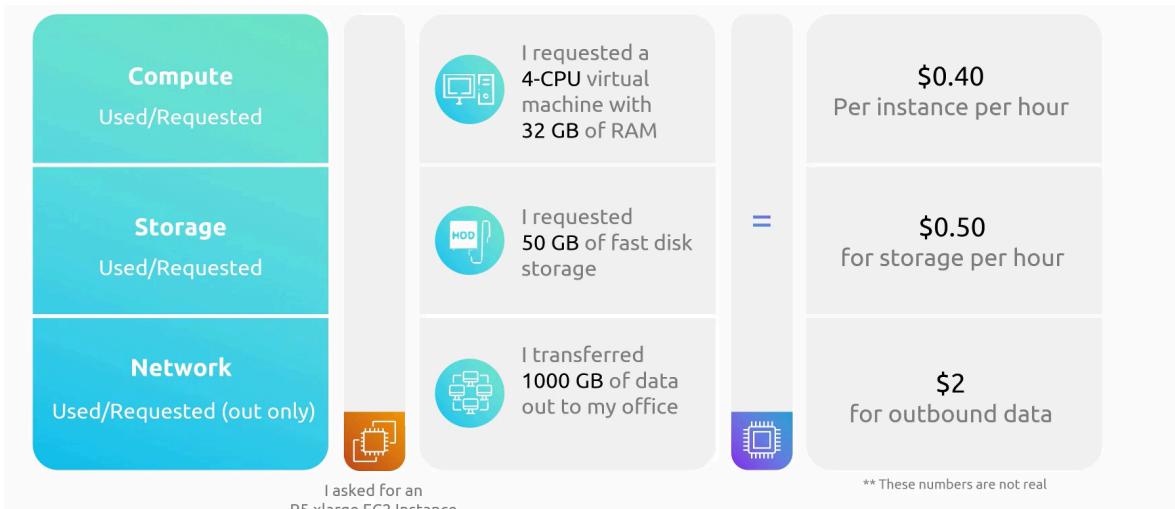
# AWS Billing

Billing is like renting a Car



image\_182.png

## Fundamentals of Pricing



image\_184.png

## Duration Pricing

Just like a car lease, you pay as long as you have the car. An example will be EC2 you will pay as long as the EC2 is running. This gives more consistency

## **Request Pricing**

Like a rideshare, you pay each time you 'ride' in the vehicle. This is more like using serverless you will pay only when it runs . This gives less cost.

## **Cost Optimization**

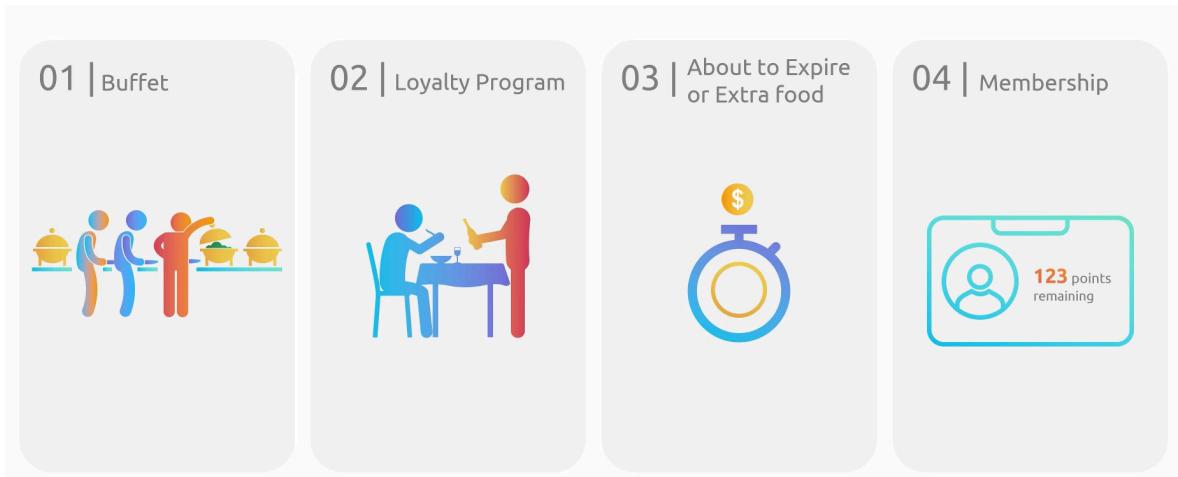
- Read the dimension of pricing before you implement.
- Choose managed services over unmanaged when possible (More request services than duration services)
- Correct expensive sizing
- Use AWS's optimize and save tools when possible

## **Maximize the Power of Elasticity**

Use auto-scaling and managed services more.

- Scale up when needed
- Don't forget to scale down
- Leverage repeatable scripts or programs as much as possible so you can delete services and recreate them.
- Use automatic scaling if available and suitable for your use case.

## **Pricing Models**



image\_187.png



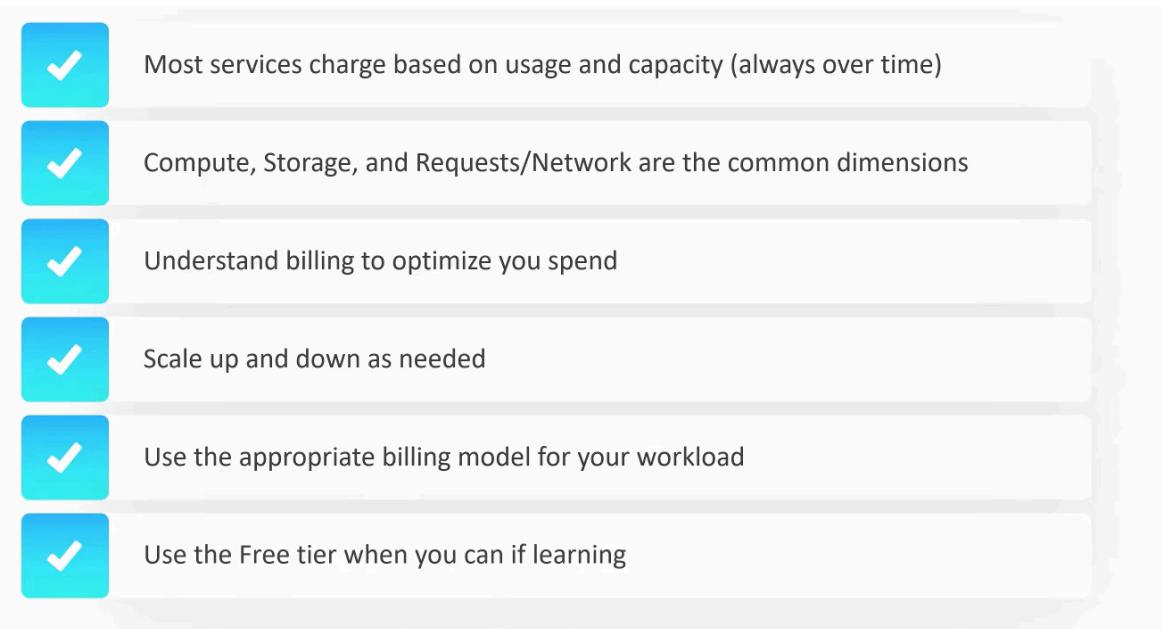
image\_186.png

## The Free Tier

This is for mainly learning and testing

- Free trials - short-term, starting from activation
- Free for 12 Months - For some services, starting from when the account was created.
- Always Free - A certain level of usage is always free for certain services.

## Short Summary



image\_188.png

# AWS EC2 Billing

## Size

The size of an EC2 affects the pricing for example a NANO is going to cost more than a Micro simple because you will have more compute power with a Nano than a micro. To check on the cost of EC2 instances Checkout EC2 (<https://instances.vantage.sh/>)

## Pricing Models

### On-demand

You request a certain size and if you use it for some second you pay for that. Payment can be based on per hour or seconds. The OS determine if you will be charged per minute or second

### Savings Plan

Here you pay for 10,000 into a saving plan for EC2 and AWS will give you 12,000. They you can pay from the \$ 12,000 for 1 or 3 years. It expires if not fully spend within the 1 or 3 years

### Spot

Place a bid on a spot machine; win the bid; get the machine for minutes to days; get up to 90% discount. Not great for application that don't need interruption.

### Reservation

Commit to 1 or 3 years and get up to 72% off on demand cost with reserved instances.

### Dedicated

Your private dedicated EC2 Host. Here they can't run their virtual machine on shared models and best for security but its the most expensive.

## Other features (What did you turn on for Ec2)

- Storage? >>> another service
- Elastic IP addresses? >>> cost only if unused
- Load balancers? >> >another service
- Hibernation? >>> slight cost

More\_features