

## AWS

### \* WHAT IS CLOUD COMPUTING ??



→ The practice of using remote servers hosted on Internet to store, manage, and process data, rather than a local server or a personal computer.

### ON PREMISE

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

### CLOUD PROVIDERS

- Someone else does all these jobs.

### \* WHAT IS AWS ?

- Leading Cloud Services Provider.
- Secure Cloud Platform
  - Provides lots of services.
  - Offers a broad set of global cloud based products
- Provides on-demand access to compute, storage, network, database & other IT resources.
- Offers flexibility.
- Pay-as-you-go
- AWS works together as building blocks.

## → LIST OF SERVICES,

- Amazon EC2
- AWS Lambda
- AWS Elastic Beanstalk
- Amazon EC2 AutoScaling
- Amazon ECS
- Amazon EKS
- Amazon ECR
- AWS Fargate

Compute Services.

- AWS IAM
- Amazon Cognito
- AWS Shield
- AWS Artifact
- AWS KMS

Security, Identity and Compliance

- Amazon S3
- Amazon S3 Glacier
- Amazon EFS
- Amazon EBS

Storage Services

- Amazon RDS
- Amazon DynamoDB
- Amazon RedShift
- Amazon Aurora

Database Services

- Amazon VPC
- Amazon Route 53
- Amazon CloudFront
- Elastic Load Balancing

Networking & Content Delivery.

- AWS trusted advisor
- AWS Cloudwatch
- AWS Cloudtrail
- AWS Well-Architected tool
- AWS Auto Scaling
- AWS CLI
- AWS Config
- AWS Management Console
- AWS Organizations.

Management and  
Governance Services.

- AWS cost & usage reports
- AWS Budgets
- AWS cost Explorer.

AWS Cost-Management Services.

## \* AWS Management Console,

↳ Web-based unified console used to build, manage, and monitor everything from simple web apps to complex cloud deployments.

## \* Service Console,

↳ AWS services each have their own console.

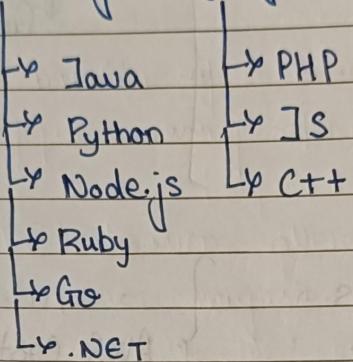
## \* Every AWS account has a unique Account Id.

## \* Amazon Resource Names (ARNs)

↳ Uniquely identify AWS resources.

## \* AWS SDK,

- ↳ Software development kit.
- ↳ Create, modify & interact with AWS resources.
- ↳ Programming languages supported,



## \* AWS Cloudshell,

- ↳ Browser based shell
- ↳ Preinstalled tools
  - ↳ AWS CLI, Python, Node.js, Git, make, pip, sudo, etc.
- ↳ Storage included
  - ↳ 1GB / AWS region

## \* Infrastructure as Code (IaC)

- ↳ You write a configuration script to automate creating, updating or destroying cloud infrastructure.
- ↳ IaC is blueprint of your infrastructure.
- ↳ Allows you to share, version or inventory your cloud infrastructure.

→ AWS has two offerings,

# Devops engineer.

## Cloud Formation

- Declarative IaC Tool.
  - Explicit (what you see is what you get).
  - More verbose, zero chance of mis-configuration.
  - Uses scripting languages JSON, XML, etc.

Bwneed By.

→ Cloud Development kit (CDK),

- You say what you want, the rest is filled in (implicit).
  - Imperative IaC tool.
  - Less verbose, chance of mis-configuration.
  - Does more than declarative.
  - Uses Programming languages eg. Python, ~~Java~~, Java.

## → Shared Responsibility Model,

↳ cloud security framework that defines the security obligations of the customer versus AWS.

## Configuration of Managed Services or 3rd Party software

- ## • Platforms • Applications • IAM

## Configuration of virtual Infrastructure & systems

- Operating System
  - Network
  - firewall

## Security Configuration of data.

- Client side Data Encryption • Server side encrypt. • Customer data  
• Net traff. prot.

# Customer

ANS

## Software

- Compute
  - Storage
  - Database
  - Networking

## Hardware

## • Regions

- Regions
- AZ

- Storage
  - Database
  - Networking
  - AZ
  - Edge location
  - Physical security

→ Computing Services,

↳ Resizable compute capacity.

### \* EC2 (Elastic Cloud Compute)

↳ Allows you to launch Virtual Machines (VM)

A VM is an emulation of a

physical computer using software. ↳ Multiple VMs can run on same physical servers.

↳ Server Virtualization allows you to easily create, copy, resize or migrate your server.

↳ When we launch a VM, we call it an **instance**.

- EC2 is highly configurable server where you choose AMI that affects options such as,

↳ The amount of CPU's.

Pre-defined ↴

↳ The amount of Memory (RAM).

Configuration ↴

↳ The amount of Network Bandwidth.

↳ OS (e.g. Windows, Linux, etc.)

↳ Multiple virtual Harddrives e.g. Elastic block store.

Virtual Machines → An emulation of a physical computer using software.

→ Amazon Lightsail,

- It is a managed virtual server service.
- friendly version of EC2 VM.

Containers → virtualizing an OS to run multiple workloads on a single OS instance. Containers are generally used in microservice architecture.

→ When you divide your application into smaller apps that talk to each other.

→ Elastic Container Service (ECS)

- It is a container orchestration service that supports Docker containers.
- Launches a cluster of servers(s) on EC2 instances with Docker installed.

→ Elastic Container Registry (ECR)

- Storage that has version control.
- It is a repository for container images.
- In order to launch a container you need an image. An image means a saved copy.

→ AWS Fargate, → More Robust than Lambda.

- Serverless orchestration container service.
- It is same as ~~EC2~~ ECR except you pay on demand.

per running container. With EC2S you have to keep an EC2 server running even if you have no containers running.

→ AWS manages the underlying server, so you don't have to scale or upgrade the EC2 server.

### → Elastic Kubernetes Service (EKS),

→ It is a fully managed Kubernetes Service.

→ K8 is an open-source orchestration software that was created by Google and is generally the standards for managing microservices.

**Serverless** → when the underlying servers are managed by AWS. You can ~~only~~ don't worry or configure servers.

→ AWS Lambda, → Short running tasks.

→ It is a serverless functions service.

→ You can run code without managing or provisioning servers.

→ You upload small pieces of code, choose much memory and how long function is allowed to run code without before timing out.

→ You are charged on the basis of runtime of the serverless functions rounded to the nearest 100 ms.

## → Edge and Hybrid Computing services,

### ↳ What is edge computing?

→ When you push your computing ~~out~~ workloads outside of your networks to run close to the destination location. Eg: IoT devices, external servers, mobile phones.

### ↳ What is hybrid computing?

→ When you're able to run workloads on both your on-premise datacenter and AWS Virtual Private Cloud.

→ **AWS Outposts** is a physical stack of servers that you can put on your datacenter. It allows you to use AWS API and services such as EC2 right in your datacenter.

→ **AWS Wavelength** allows you to build and launch your applications in a telecom datacenter. By doing this your applications will have ultra-low latency since they need will be pushed over a 5G network and be closest as possible to the end user.

→ **VMware Cloud on AWS** allows you to manage on-premise virtual machines using VMware as EC2 instances.

→ **AWS Local Zones** are edge datacenters located outside of an AWS ~~zone~~ region so you can use AWS closer to end destination.

## → Cost and Capacity Management Computing Services,

- EC2 spot Instances, Reserved and Savings Plan,

↳ ways to save on computing.

- AWS Batch,

↳ plans, schedules, and executes your batch computing workloads across the full range of AWS compute services, can utilize spot instance to save money.

- AWS Compute Optimizer,

↳ suggests how to reduce costs and improve performance by using ML to analyze your previous usage history.

- EC2 AutoScaling Groups,

↳ automatically adds or removes EC2 servers to meet the current demand of traffic. Will save you money and meet capacity since you only run the amount of servers you need.

- Elastic Load Balancers,

↳ distributes traffic to multiple instances, can route traffic from unhealthy instances to healthy instances. It can route traffic to instances in different availability zones.

- AWS Elastic Beanstalk,

↳ it is easy for developers deploying web applications without developers having to worry about setting up and understanding underlying AWS services.

## → TYPES OF STORAGE SERVICES,

↳ Virtual Harddrive in the cloud.

### • Elastic Block Storage (EBS) - Block

- ↳ Data is split into evenly split blocks.
- ↳ Directly accessed by the operational os.
- ↳ Supports only a single write volume.

When you need a virtual drive attached to a VM.

### • AWS Elastic File Storage (EFS) - File

- ↳ file is stored with data and metadata.
- ↳ Multiple connections via a network share.
- ↳ Support multiple reads, writing locks the file.

When you need a file share where multiple users or VMs need to access.

### • Amazon Simple Storage Service (S3) - Object

- ↳ Object is stored with data, metadata and unique id.
- ↳ Scales with limited no file limit.
- ↳ Supports multiple reads and writes.
- ↳ Not intended for high I/O ps.

When you want to upload files, and not have to worry about underlying infrastructure.

#### What is S3 bucket?

- ↳ Buckets hold objects. It can also have folders which in turn hold objects.
- ↳ Bucket names must be unique.

### • Storage Gateway

- ↳ extends your on-premise storage to cloud.

#### • File Gateway

- ↳ local storage to AWS S3.

#### • Volume Gateway

- ↳ caches your local drives to S3.

- ↳ continuous backup.

#### • Tape Gateway

- ↳ store files to virtual tapes. (cost effective)

→ AWS Snowfamily, (Data is delivered to S3 bucket)

↳ AWS Snowfamily are stored and compute devices used to physically move data in or out the cloud when moving data over the Internet or private connection ~~to it~~ it is slow, difficult or costly.

• Snowcone.



Comes in two sizes:

- 8TB (HDD)
- 14TB (SSD)

• Snowball Edge.



Gen comes in two sizes

- Storage optimized
- 80TB

• Snowmobile.

• 100PB of

storage

• Compute optimized

39.5TB

→ AWS Backup,

↳ managed backup service that makes it easier to centralise & automate the backup of data across multiple AWS services. eg: EC2, EBS, RDS, DynamoDB, EFS storage gateway. You create backup plans.

## → WHAT IS A DATABASE?

- ↳ Data-stores that stores semi-structured and structured data.
- ↳ Requires using formal designs and modeling techniques.

## → WHAT IS A DATAWAREHOUSE?

- ↳ A relational data-store designed for analytical workloads.
- ↳ Generally perform aggregation.
  - ↳ Grouping data. Store.
- ↳ HOT → return queries very fast.
- ↳ Infrequently accessed.
- ↳ Need to consume data from a relational database on a regular basis.

## → What is a key-value store?

- ↳ keyvalue database is a type of Non-relational database (NoSQL) that uses a simple key value method to store data.
- ↳ Dumb and fast.
  - ↳ They lack
    - ↳ Relationships
    - ↳ Indexes
    - ↳ Aggregation.
- ↳ Schemaless.

→ NoSQL key/value store  
→ NoSQL Database service,

- **DynamoDB**,

↳ AWS's flagship db service

↳ Designed to scale to billions of records with guaranteed consistent data return in at least a second.

- **Document DB**,

↳ NoSQL document database, that is MongoDB compatible

- **Amazon Keyspace**,

↳ fully managed Apache Cassandra database.

## AWS Relational Database Services

### \* Relational Database Services,

→ Relational Database Service (RDS) is a relational database service that supports multiple SQL engines. Relational is synonymous with SQL and Online Transactional Processing (OLTP). Most commonly used type of database.

- RDS supports the following SQL Engines,
- MySQL → most popular SQL database (Oracle).
  - MariaDB → Open source copy of MySQL.
  - PostgreSQL (PSQL) → More Complex, rich features.
  - Oracle → Used by Enterprises.
  - Microsoft SQL Server → MS prop. SQL DB.
  - Aurora → fully managed DB service.

### AURORA

### AURORA

#### SERVERLESS

- Server-less on-demand version of Aurora.
- less traffic demand.

- It is a fully managed or either MySQL (5x faster) and PostgreSQL (3x faster) database.
- Highly available.
- Scalable.
- Secure.
- durable.

### \* RDS on VMware

- It allows you to deploy RDS supported engines to on-premise data-center. The datacenter must use VMware for server virtualization.

## → Other database Services,

### → RedShift,

- It is a petabyte-size data-warehouse.
- Data warehouses are for Online Analytical Processing (OLAP).
- They can be expensive because they are keeping data "hot".
  - ↳ Can fetch large amount of data in less time.

### → ElastiCache,

- It is a managed database of the in-memory and caching open-source databases Redis or Memcached.
- used when we need to improve the performance of application by adding a caching layer-in-front of webserver or database.

### → Neptune,

- Managed graph database
- Data is represented as interconnected nodes.
- To understand connections b/w data.

### → Amazon Timestreams,

- fully managed time series database.
- Devices that send a lots of data that are time-sensitive such as IoT devices.
- Measure things that change over time.

## → Amazon Quantum ledger database,

- ↳ It is a fully managed ledger database that provides transparent, immutable and cryptographically verifiable transactions & logs.
- ↳ When we need to record history of financial activities that can be trusted.

## → Database Migration Service (DMS),

- ↳ One can migrate from,
- ↳ On premise database to AWS
  - ↳ from two database in different or same AWS accounts using different SQL engines.
  - ↳ from an SQL to NoSQL database.

## \* Networking,

### → Virtual Private Cloud,

- ↳ It is a logically isolated section of the AWS network where you launch your AWS resources. You can choose a range of IPs using CIDR Range.
- ↳ 65,536.

- ↳ Subnets → a logical partition of an IP network into multiple smaller network segments less IPs

Private ↴ Public ↴ 256

- ↳ Can reach the internet

- ↳ Cannot reach the internet.

→ AWS Cloudfront,

↳ Content delivery network.

↳ Designed to deliver data, videos, apps, and APIs to users globally with low latency, high transfer speed and high availability.

### {Compute}

→ EC2 Instance families,

↳ Different combinations of CPU, memory, Storage and Networking capacity.

- General Purpose → T2, Mac
- Compute Optimized → C5, C4
- Memory Optimized → R4, R5
- Accelerated Optimized → P2, P3
- Storage Optimized → D2, D3

→ EC2 Instant Types,

↳ Particular instance size and instance family.

→ EC2 Pricing Models,

↳ Pay as you go

- On Demand,

(Least Commitment)

- ↳ Low cost and flexible
- ↳ Windows, Linux
- ↳ Only pay / hour or second
- ↳ short-term, spiky, unpredictable workloads.
- ↳ Cannot be interrupted.
- ↳ for first time apps.

↳ Maximize the utility of their idle servers.  
↳ Up to 90%.

↳ AWS Batch

↳ Way to use spot instances

## • Spot, (Biggest Savings)

- ↳ Request spare computing capacity.
- ↳ Flexible start and end times.
- ↳ Can handle interruptions (servers randomly stopping & starting)
- ↳ For non-critical background jobs

↳ Up to 75%.

↳ Instance Type

↳ Region

↳ Tenancy

↳ Platform

## • Reserved, (Best long term)

- ↳ Steady state or predictable usage or required reserved cap.
- ↳ Commit to EC2 over a 1 year or 3 year term.
- ↳ Can resell unused reserved instances

## • Dedicated, (Most Expensive)

- ↳ Dedicated servers.
- ↳ Can be on demand, spot or reserved.
- ↳ When you need a guarantee of isolate hardware.

## \* Identity,

### - Y AWS IAM,

↳ One can create and manage AWS users and groups, and use permissions to allow/deny their access to AWS resources.

#### → IAM Policies

↳ JSON Documents which grant permissions for a specific user, group, or role to access services.

↳ Policies are attached to IAM identities.

#### ↳ IAM Permissions

↳ The API actions that can or cannot be performed.

↳ They are represented in the IAM Policy document.

## • IAM Identities,

### ↳ IAM Users,

↳ End users who log into the console or interact with AWS resources programmatically or via clicking UI interfaces.

### ↳ IAM Groups,

↳ Group up your users so they all share permission levels of the group. e.g. Admin, developers, Auditors.

### ↳ IAM Roles,

↳ Roles grant AWS resources permissions to specific AWS API actions.

↳ Associate policies to a role & assign it to an AWS service.

## → Principle of least Privilege (PoLP)

- ↳ Computer security concept of providing a user, role, or application the least amount of ~~privileges~~ permissions to perform an operation or action.
- ↳ Just enough Access → Permitting only the exact actions for the identity to perform a task.
- ↳ Just in time → Permitting the smallest length of duration an identity can use permissions.

## → AWS Account Root User,

- ↳ A special account with full access that cannot be deleted
- ↳ There can only be one Root user per AWS account.
- ↳ Tasks,
  - ↳ Change your account settings.
  - ↳ Restore IAM user permissions
  - ↳ Close your AWS Account.
  - ↳ Change or cancel AWS Support plan.
  - ↳ Signup for GovCloud.

## → AWS Single Sign On (SSO),

- ↳ Create, or connect, your workforce identities in AWS once and manage access centrally across your AWS organisations.

## → Application Integration,

- ↳ Application Integration is the process of letting two independent applications to communicate and work with each other, commonly facilitated by an intermediate system.
- ↳ Cloud workloads encourage systems and services to be loosely coupled and therefore AWS has many services for the specific purpose of Application Integration.

## → Queuing,

- ↳ what is messaging System ?
  - ↳ used to provide asynchronous communication and decouple processes via messages/ events from a sender to a receiver.
- ↳ What is a Queuing System ?
  - ↳ It is a messaging system that generally will delete messages once they are consumed.
  - ↳ Simple Communication
  - ↳ Not Realtime.
  - ↳ Have to pull.
  - ↳ Not reactive.

## \* sqs (Simple Queueing Service),

↳ Fully managed Queueing service that enables you to decouple and scale microservices, distributed systems, and serverless applications.

↳ Use Case

↳ You need to queue up transaction emails to be sent e.g. Signup, Reset Password.

## \* Streaming,

↳ Multiple users can react to events (messages)

↳ Realtime

↳ Amazon Kinesis,

↳ fully managed solution for collecting, processing & analyzing streaming data in the cloud.

## \* Pub/Sub,

↳ Publish - subscribe

↳ implemented in messaging systems.

↳ send messages to an event bus.

↳ Simple Notification Service (SNS),

↳ receives events from source, routes events to the target.

↳ Highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and serverless apps.

## → APPLICATION INTEGRATION SERVICES,

- Simple Notification Service → pub-sub messaging system
- Simple Queue Service → queuing messaging service.
- Step functions → state machine service.
- Event Bridge (CloudWatch Events) → Serverless event bus.
- Kinesis → real-time streaming data service
- Amazon MQ → Managed message broker service
- Managed Kafka Service → fully managed Apache Kafka service.
- API Gateway → to create, publish, maintain, monitor and secure API
- App Sync → fully managed GraphQL service.

## \* Containers,

- Virtualizing an OS to run multiple workloads in a single OS instance.
- Generally used in microservice architecture.

## \* What is a Microservice / Microservices Architecture,

- Multiple apps which are each responsible for one thing.
- Functionality is isolated & stateless.

## vs. Monolithic Architecture,

- One app which is responsible for everything.
- functionality is tightly coupled.

## \* Kubernetes (K8),

- It is an open-source container orchestration ~~service~~ system for automating, deployment, scaling and management of containers.
- The advantage of Kubernetes over Docker is its ability to run containers distributed across multiple VMs.
- A unique component of Kubernetes ~~over dock~~ are pods.
  - group of one or more containers with shared storage, network resources and other shared settings.
- Ideally for microservice architecture where a company has tens to tons of services they need to manage.



### → Docker,

- |→ Set of platform as a service products that use OS level virtualization to deliver software in packages called containers.
- |→ It was earliest popularized open-source container platform.
  - |→ containers → Docker.

### → Podman,

- |→ container engine that is OCI-compliant & is the drop-in replacement for Docker.

## \* Organisations and Accounts,

### → AWS Organizations,

- |→ Allows the creation of new AWS accounts.
- |→ Centrally manage billings, control access, compliance, security and share resources across AWS accounts.

### → Root Account User,

- |→ Single sign-in identity that has complete access to all AWS services and resources in an account.

### → Service Control Policies,

- |→ give central control over the allowed permissions for all accounts in your organization.

## → AWS Control Tower,

- helps enterprises quickly setup a secure, AWS multi-account.
- Provides you with a baseline environment to get started with a multi-account architecture.

## → Landing Zone,

- It is a baseline environment following well-architected and best practices to start launching production ready workloads.

- AWS SSO enabled, centralized logging for AWS CloudTrail, cross-account security auditing.

## → Account factory,

- Automates provisioning new accounts in your organisations.

- pre-approved account configurations.

- pre-approved network configurations and region selections.

## → Guardrails,

- pre-packaged governance rules for security, operations, and compliance that customers can select and apply enterprise-wide or to specific groups of accounts.

## → AWS Quickstarts,

→ Reduces 100s of manual steps into a few.

- Pre-built templates by AWS and AWS Partners to help deploy wide range of stacks.

## → Composed of,

- Reference Architecture for deployment.

- AWS CloudFormation that automate & configure deployment.

- A deployment guide explaining the architecture & implementation in detail.

## → Tagging,

- ↳ Key/Value pair assigned to AWS resources.
- ↳ Allows you to organize resources in following ways,
  - Resource Management.
    - ↳ Specific workloads, environments.
  - Cost Management and Optimization.
    - ↳ Cost tracking, Budgets, Alerts
  - Operations Management
    - ↳ Business commitments & SLA operations.
  - Security.
    - ↳ Classification of data and security impact
    - ↳ Governance & regulatory compliance
    - Automation, Workload Optimization.

## → Resource Groups,

- ↳ It is a collection of resources that share one or more tags.

## → Serverless Services,

- The underlying servers, Infrastructure and OS is taken care by CSPs.
- Highly available
- Scalable
- Cost-effective
- Pay for what you use.

- \* **Dynamo DB** → serverless NoSQL keyvalue & doc database.
- \* **Simple Storage Service (S3)** → serverless object storage service
- \* **ECS Fargate** → serverless orchestration container service
- \* **AWS Lambda** → serverless functions service.
- \* **Step Functions** → statemachine service
- \* **Aurora Serverless** → serverless on-demand version of Aurora
  - Logging Services,
    - \* log streams represent a sequence of events from the monitored instance.
  - \* Cloud trail → logs all API calls (SDK, CLI) between AWS services
    - Where
    - When
    - Who
    - What.
    - detect developer misconfiguration
    - Detect malicious actors.
    - Automate responses.
  - \* Cloud Watch → collection of multiple services.
    - Cloudwatch Logs → log data storage
    - Cloudwatch Metrics → time ordered set of data points
    - Cloudwatch Events → trigger an event based on conditions defined threshold.
    - Cloudwatch Alarms → triggers notification based on metric
    - Cloudwatch Dashboard → create visualizations based on metrics.
  - \* AWS xray → distributed tracing systems.
    - ↳ used to pin point issues with microservices

## \* Machine learning and AI Services,

### → Amazon SageMaker,

↳ It is a fully managed service to build, train and deploy machine learning models at scale.

- Apache MXNet on AWS → open-source DL framework
- Tensorflow on AWS → open-source ML library
- Pytorch on AWS → open-source ML framework

↳ Amazon SageMaker Ground Truth is data-labelling service. Humans label a dataset that will be used to train machine learning models.

→ Amazon Augmented AI human intervention review service. When SageMaker's user ML to make a prediction is not confident it has the right answer queue up the prediction for human review.

→ Amazon Code Guru → ML code analysis service.

↳ suggest changes to improve the quality of code.

→ Amazon Lex → Conversation interface service.

↳ voice & text chatbots.

→ Amazon Personalize → real-time recommendations service

↳ used by amazon to make product recomm.

→ Amazon Polly → text to speech services.

↳ image and video recog.

→ Amazon Rekognition → Extract text from scanned docs.

→ Amazon Translate → neural ML translation services.

↳ do deliver more accurate & natural sounding translations.

- Amazon Lookout → for equipment / Metrics / Vision, ML for quality control.
- AWS Newton → SDK used to run DL workloads on AWS Inferentia and AWS Trainium based instances
- Amazon Textract → Extract text from scanned documents.
- Amazon Comprehend → NLP service. find relationships between text to produce insights.
- Amazon Transcribe → speech to text service.
- Amazon Forecast → time-series forecasting service
- AWS Deep Learning AMI → EC2 instances pre-installed with popular deep learning frameworks.
- AWS Deep Learning Containers → Docker images instances pre-installed with popular DL frameworks.
- AWS DeepComposer → ML enabled musical keyboard.
- AWS DeepLens → Webcam that uses DL to do things.
- AWS DeepRacer → toy racecar powered with ML to perform autonomous driving.
- Amazon Elastic Inference → allows you to attach low cost GPU powered acceleration to EC2 instances to reduce cost.
- Amazon Fraud Detector → fully managed fraud detection service.
- Amazon Kendra → Enterprise ML search engine service. Uses NLP.
- Amazon Bedrock → Large language Model {like chatGPT}
- Amazon CodeWhisper → predict code {copilot}
- Amazon DevOps Guru → analyze operational Database to detect abnormalities.
- Amazon Monitor → use IoT sensors to monitor vibrations.

## \* CUDA → Compute Unified Device Architecture.

Page No.:

Date:

## → Big Data and Analytics Service,

- \* Amazon Athena → serverless interactive query service.  
↳ It can take JSON / CSV file in an S3 bucket & load them to SQL.
- \* Amazon CloudSearch → full text search service.
- \* Amazon Elasticsearch Service → Elasticsearch is an open-source full text search engine.
- \* Amazon Elastic MapReduce → for Data processing and analysis.  
↳ Uses Hadoop. ↳ Structured to Unstructured.
- \* Kinesis Data Streams → real-time streaming data service.
- \* Kinesis Firehose → serverless and simpler version of Data Streams.
- \* Amazon Kinesis Data Analytics → allows you to run queries against data that is flowing through real-time stream.
- \* Amazon Kinesis Video Streams → to analyze and apply processing on real-time streaming video.
- \* Managed Kafka Service (MKS) → similar to Kinesis but more robust.
- \* Amazon Quicksight → business intelligence dashboard.  
↳ Similar to Tableau
- \* AWS Data Pipeline → automates the movement of data.
- \* AWS Glue → Extract, transform, load service.
- \* AWS Lakeformation → centralised, curated, and secured repos that stored all your data.
- \* AWS Data Exchange → catalogue of 3rd party datasets.

### \* AWS Pricing Calculator,

↳ free cost estimate tool.

### \* Migration evaluator,

↳ estimate tool used to determine an org. existing on-premise cost so it can compare it against AWS costs for planned ~~cost~~ cloud migration.

### \* VMware Export/Import,

↳ Allows user to import VM images into EC2.

### → AWS Free Services,

- IAM
  - Amazon VPC
  - AutoScaling
  - CloudFormation
  - Elastic Beanstalk
  - OpsWork
  - Amplify
  - AppSync
- CodeStar
  - Organizations and Consolidated Billing
  - AWS Cost Explorer.

### → AWS Support Plans,

0\$/month	29\$/month	100\$/month	15000\$/month
Basic	Developer	Business	Enterprise.

- Email support for Billing & Account

Tech Support via email

- No 3rd Party Support. chat/phone 24/7

Gen' Guidance < 24 hrs

System Impaired < 12 hrs.

Prod. System Impaired < 4 hrs

Prod. System Down < 1 hr.

TAM  
Personal Concierge

## → Consolidated Billing,

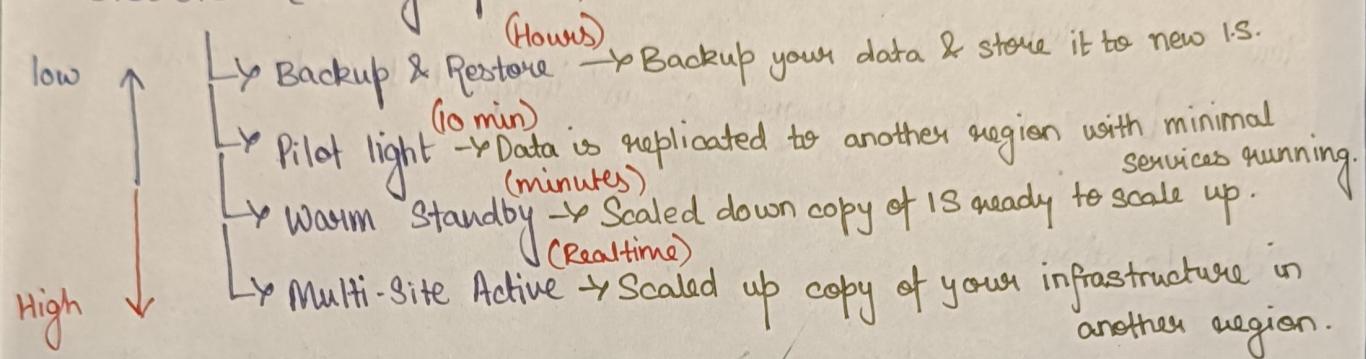
lets you  
take  
advantage

- One bill for multiple AWS accounts in an organization.
- Master/root account that pays the charge.
- AWS has volume discounts.

## \* CloudEndure,

↳ Disaster Recovery Service that continuously replicates your machines into a low cost staging area in your target AWS account and preferred Region enabling fast and reliable recovery in case of IT data center failure.

## \* Disaster Recovery Options,



## \* RTO (Recovery Time Objective)

↳ It is the maximum acceptable delay between the interruption of service & restoration of service.

↳ Defined by the organizations.

## \* RPO (Recovery Point Objective)

↳ It is the max acceptable time since the last data recovery point.

↳ Defined by organisations

## \* AWS API

↳ Two apps or services communicate  
↳ https requests.

↳ Postman

## \* AWS Management Console

↓  
Web Interface

★ AWS SDK  
↓  
interact with API  
using your fav pr.  
language

## \* AWS CLI.

↓  
interact with the API  
via a terminal/shell  
program

## \* Fault Tolerance,

↳ A fault tolerance is a section of a network that is vulnerable to damage if a critical device or system fails.

\* Each AZ is isolated, but the AZ in a region are connected through low latency links.

↳ Each AZ is designed as an independent failure zone.

## \* Edge Locations,

- AWS Global Accelerator

↳ They use edge locations as an ~~on~~ off ramps to quickly reach AWS resources in other regions by traversing the

- AWS S3 Transfer Accelerator

↳ They use edge locations as off ramp to provide fast AWS Global Network. Edge storage and compute near the end user

- Amazon Cloudfront