**AWS**

**Why Cloud :**

1. Earlier we had to buy the server, which comes very costly.
2. Troubleshooting problems can be tedious and may conflict with business goals. We should focus on our business not infrastructure related problems.
3. Since traffic is varying, so servers will be idle most of the time.

**Cloud Computing :**

1. Storing data/applications on remote servers
2. Processing data/applications from servers
3. Accessing data/applications via Internet

**Service Models :**

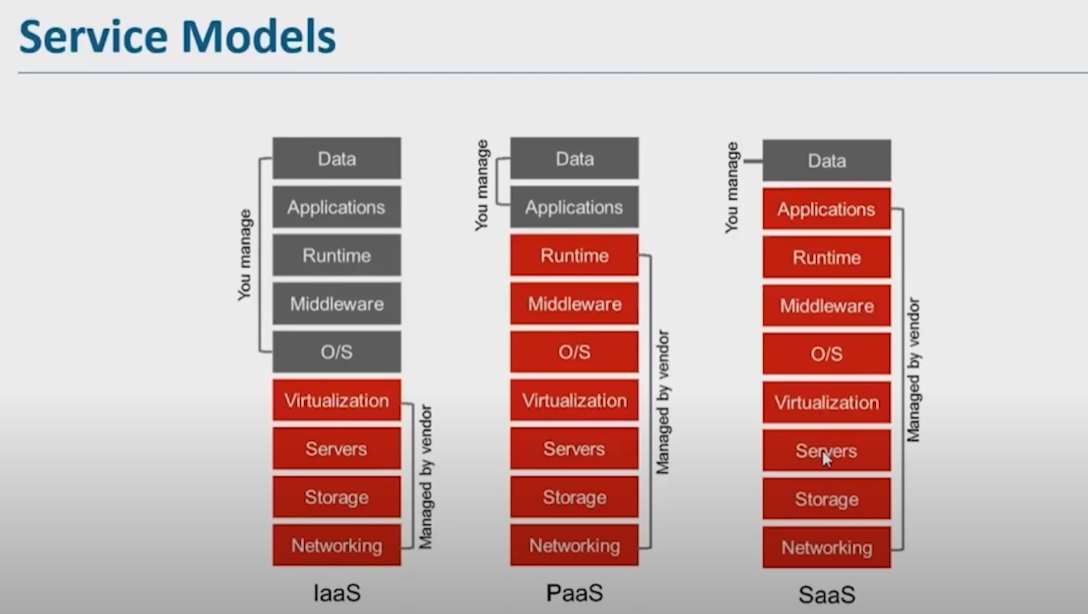
* **Saas :** Software as a Service
  + We are just using applications maintained by cloud to its clients. We don’t need to think about upscaling or downscaling, or any secutiy issues. We just need to consume it.

Ex – Gmail.

* **Paas :** Platform as a Service
  + It provides the platform where we can build our application and others can access it.It doesn’t have any control over underlying architecture,i.e OS,servers etc

Ex – Google App Engine

* **Iaas :** Infrastructure as a Service
  + The entire Infrastructure is provided to you as per your requirements to build your applications. It is something like new physical system one click away.



**Deployment Models :**

* **Public Cloud :** A service provider makes resources available to general public over internet. Easy and inexpensive set-up because hardware, application and bandwidth costs are covered by provider.
* **Private Cloud :** Offers hosted service to a limited no of people only behind firewalls, so it minimizes the security concerns.
* **Hybrid Cloud :** It is basically the combination of public and private cloud to leverage the best advantage out of it.

**Advantages of AWS :**

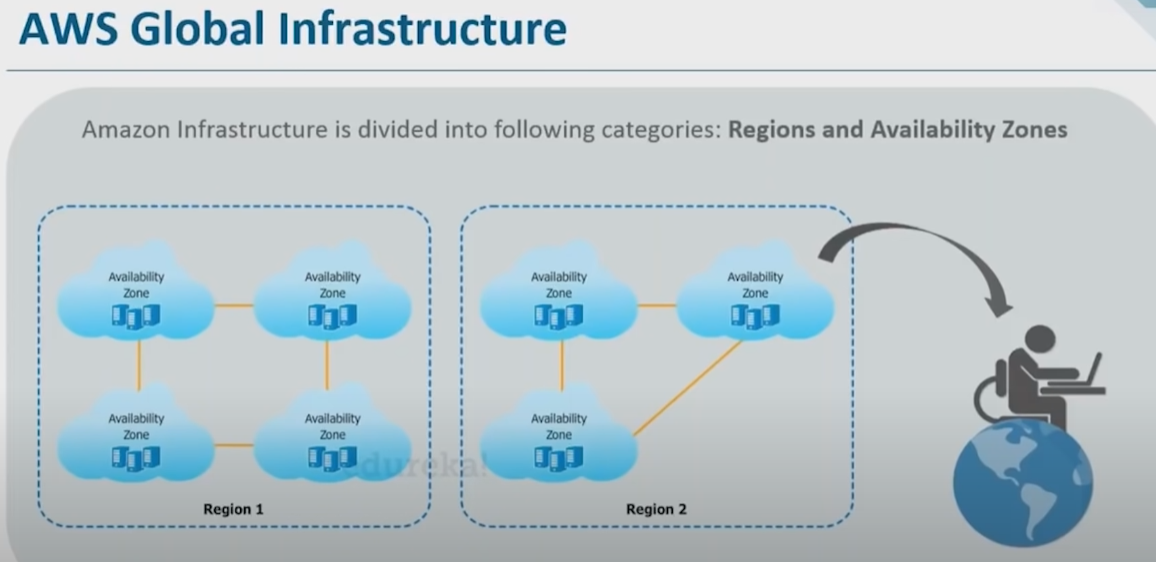
* Flexibility
* Cost Effective
* Scalability
* Security

**AWS Global Infrastructure :**

It is divided into Regions and availability zones.

Region : Region is where amazon is having its datacentres.

Availability zones : In one region, there can be more availability zones.

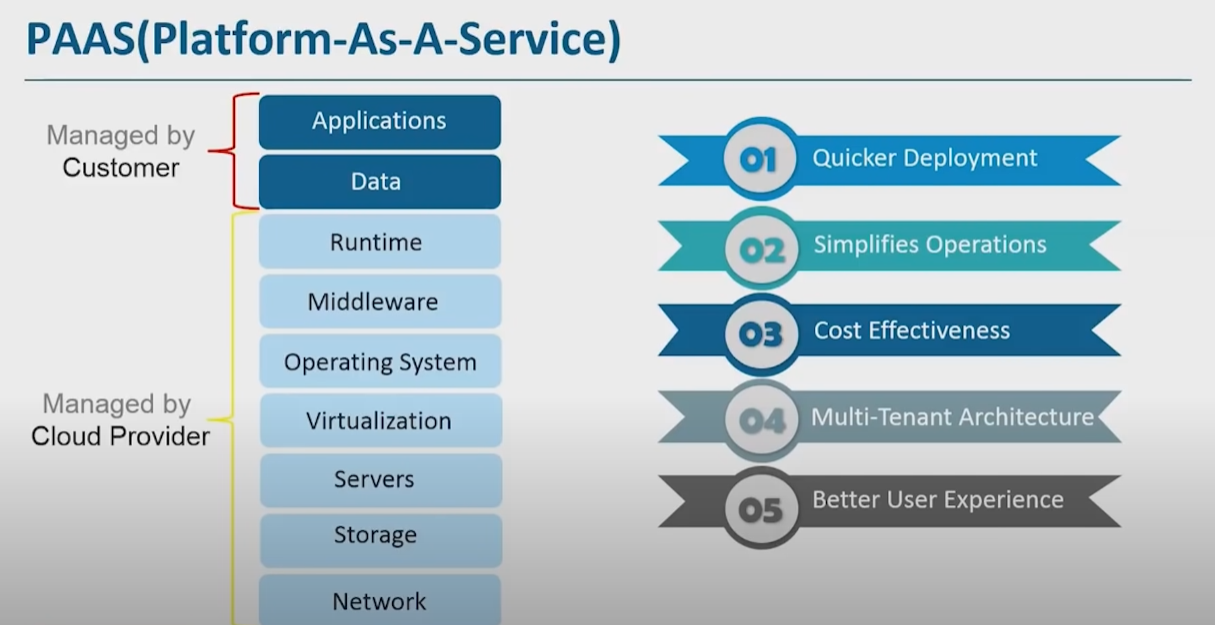


**Domains :**

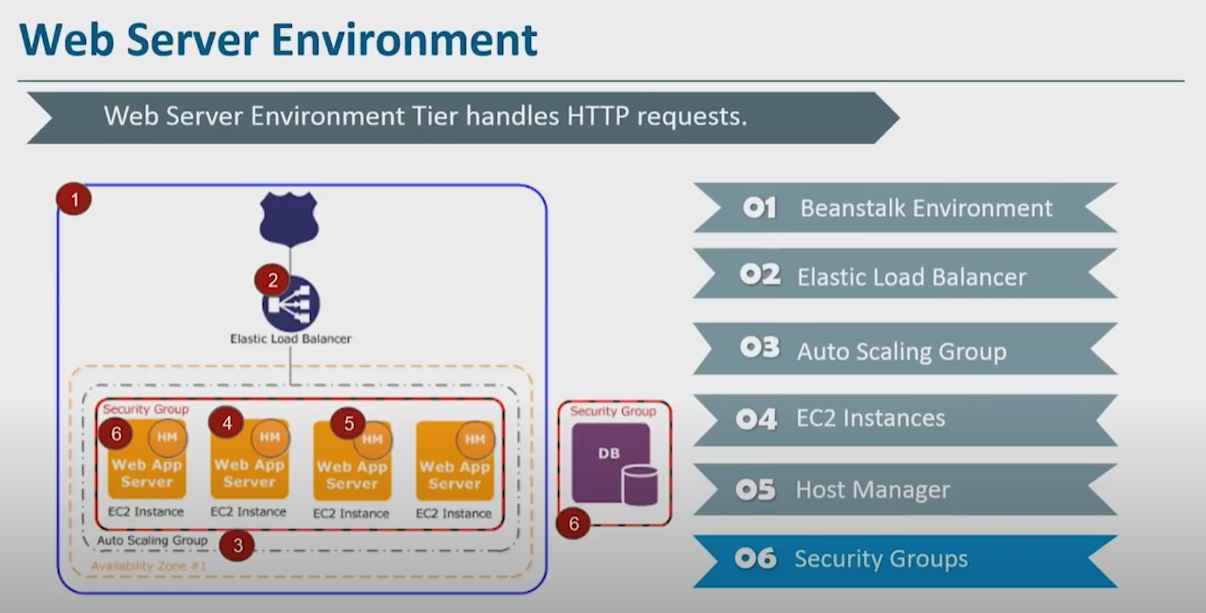
* **Compute**
* **Migration**
* **Security and compliances**
* **Storage**
* **Networking**
* **Messaging**
* **Database**
* **Management Tools**

**Compute:**

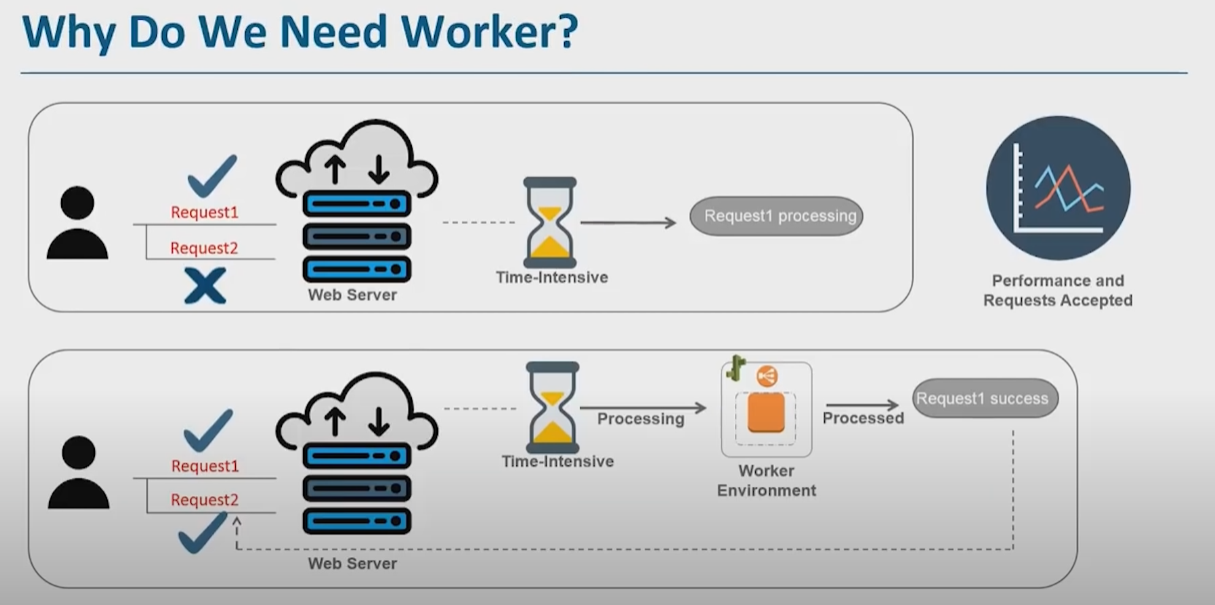
* **EC2 :** Elastic Compute Cloud
  + Instance **:** An instance is a virtual server for running applications on Amazon’s EC2.It can also be understood like a tiny part of large computer, a tiny part having its own hard drive and network connections. But actually it is all virtual.
  + EC2 is a web service from Amazon that provides resizable compute capacity in the cloud.
* **AWS Lambda :** 
  + Automated version of EC2.
  + No worries about underlying architecture
  + Not used for any deployment, it is used for executing background tasks.
  + It is a serverless compute service, meaning the developers, they don’t have to worry about which AWS resources to launch or how will they manage them, they just put the code on lambda and it runs
* **AWS Elastic BeanStalk :**
  + It is a PAAS service used for deploying and scaling web applications and services developed with JAVA, .NET,PHP etc on familiar services such as Apache, Nginx, Tomcat etc.
    - Two options if we buy a new computer :
      * You can go to a computer warehouse and buy different components according to your requirements and assemble them.
        + Deploying an application without using Elastic Beanstalk
      * You can visit a computer retail store and buy a new computer
        + Deploying an application with Elastic Beanstalk

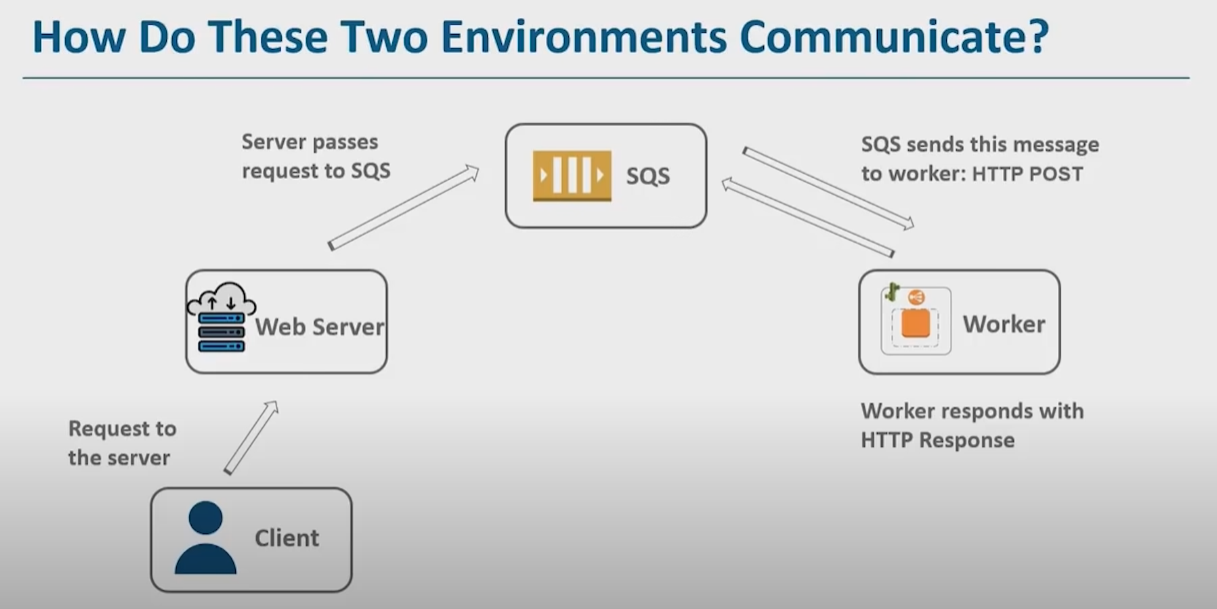


* **Fetaures of Elastic Beanstalk**
  + It is the fastest and simplest way to deploy your application on Cloud.
  + It enables you to focus on writing code rather than spending time managing and configuring servers.
  + It automatically scales application up and down based on application specific’s needs.
  + It gives freedom to select AWS resources like EC2 instance type that are optimal for your application.



* **Worker Environment :** 
  + It is a process that handles background tasks during resource intensive or time intensive operations.
    - Email Notifications
    - Generate Reports
    - Clean Up Databases

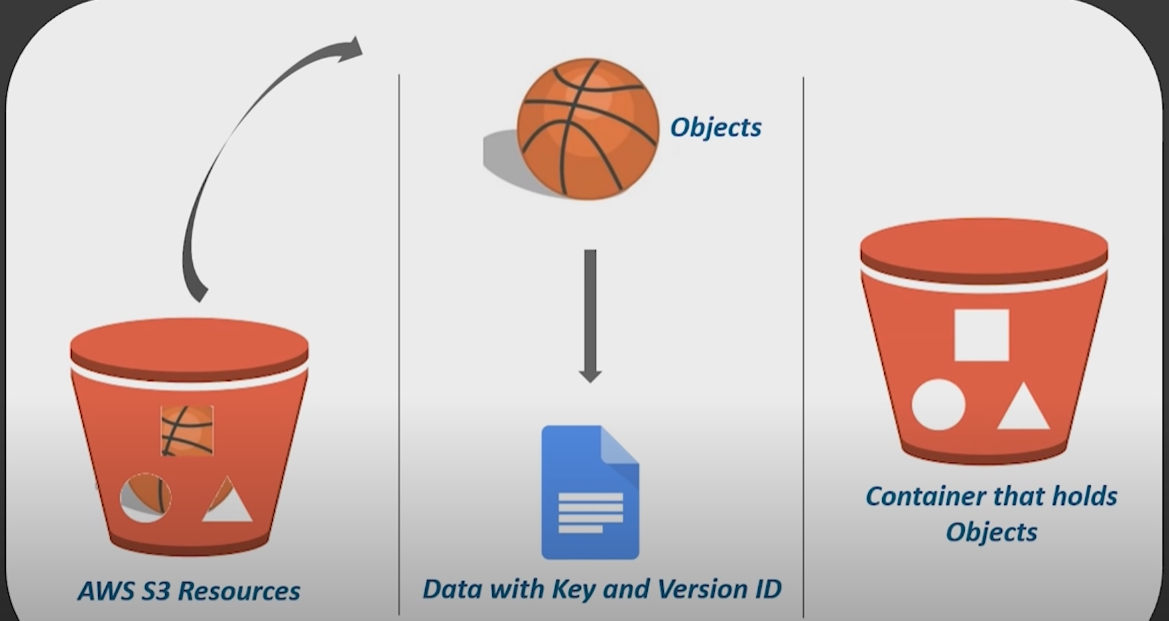




**Storage :**

Benefits :

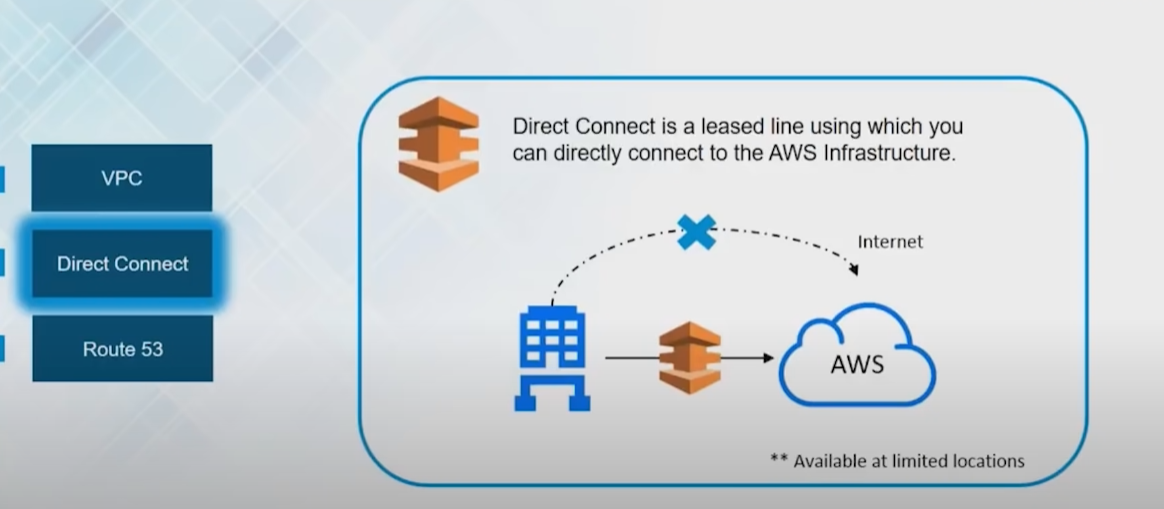
* Customer Friendly
* Pocket Friendly
* Secure
* **S3 :**
  + Amazon S3 is a simple web service interface that you can use to store and retrieve any amount of data, at any time, from anywhere on the web.
    - Durable
    - Flexible
    - Available
    - Cost Efficient
    - Scalable
    - Secure



* + S3 storage class has three types of storage :
    - Standard
    - Infrequent Access
    - Glacier
  + Versioning : It can be used to keep multiple version of an object in one bucket.
    - By default, versioning is disabled
    - It prevents overwriting or accidental deletion
  + Cross Region Replication : It enables automatic, asynchronous copying of objects across buckets in different AWS Regions.
  + S3 Data Transfer Acceleration : It enables fast, easy and secure transfer of files over long distances between your client and an S3 bucket.

**Networking :**

* **VPC :**
  + It enables to launch AWS resources into a virtual network that you have defined. It closely resembles a traditional network that you would operate in your own data centre, with benefits of using scalable infrastructure of AWS.
    - E.g if an EC2 is launched, it’ll be launched in one particular VPC.
* **Direct Connect :**



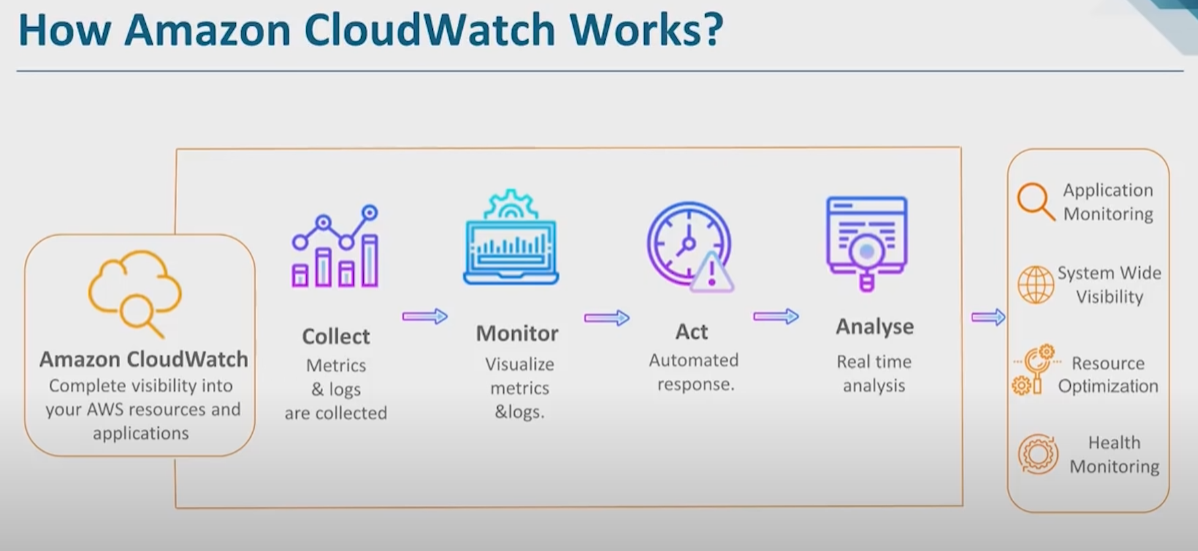
* **Route 53 :**
  + Amazon Route 53 is a highly available and scalable cloud [Domain Name System (DNS)](https://aws.amazon.com/route53/what-is-dns/) web service. It is designed to give developers and businesses an extremely reliable and cost effective way to route end users to Internet applications by translating names like www.example.com into the numeric IP addresses like 192.0.2.1 that computers use to connect to each other. Amazon Route 53 is fully compliant with IPv6 as well.
* **CloudFront** : (Content Delivery Network)
  + It has helped users to request data resulting in low latency, low network traffic, and quick data access with minimal cost. It securely transfers content such as software, SDKs etc to the clients with high speed.
  + **Benefits** :
    - It will cache your content in edge location and decrease the workload, thus, resulting in high availability of applications.
    - It is simple to use and ensures productivity enhancements.
    - It provides high security with the ‘Content Privacy’ Feature.
    - It facilitates GEO targeting service for content delivery to specific end users.
    - It uses HTTP or HTTPS for quick delivery of contents.
    - It is less expensive, as it only charges for the data transfer.
  + **Working :**
    - The client accesses a website and request to download a file.
    - The DNS routes the client request to nearest edge location through CloudFront to serve the user request.
    - At edge location, cloudfront looks for its requested cache file. Once the file is found, it responds back to server.
    - But if the file is not found, it compares the requirements with the specifications and shares it with respective server.
    - The web server responds back with the result at cloudFront edge location.
    - As soon as cloudFront receives the file, it shares it with the client and adds the file to the edge location.

**Monitoring and Management**

**Need of monitoring the cloud**



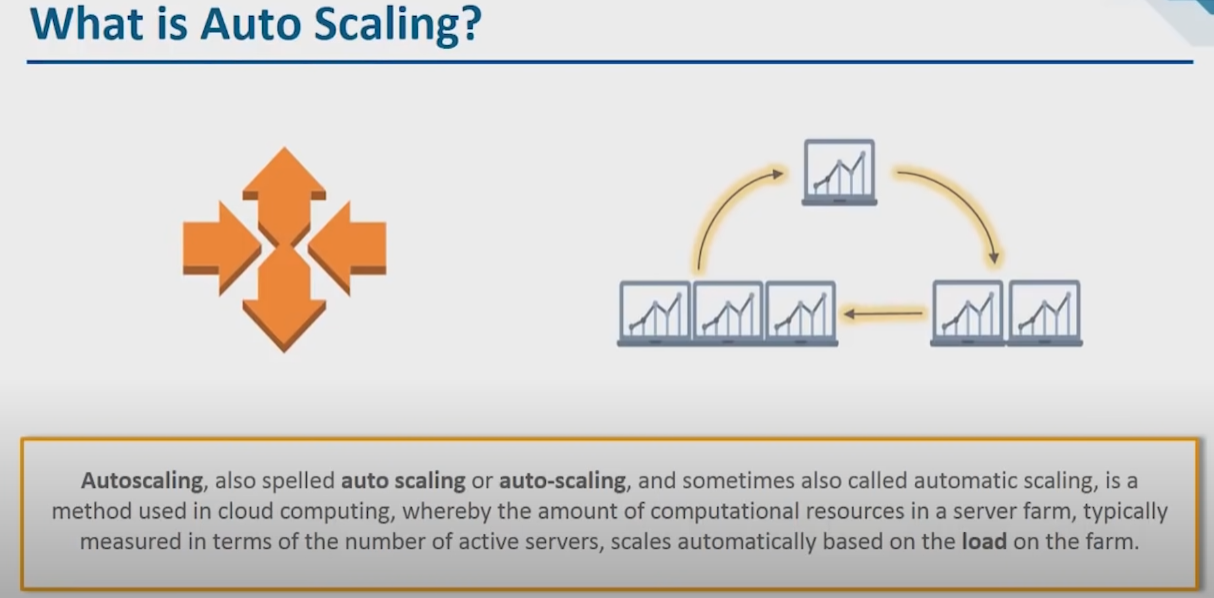


* Provides a catalogue of standard reports that you could use to analyse trends and monitor system performance
* To reduce unnecessary system costs by sending notification in case of unreasonable pricing issues.
* Detect problem early to prevent disasters. Also prevents and reduces downtime and business losses.
* Monitor the end user experience and create reports which help to make improvements.
* **AWS Cloudwatch :**
  + It is the component of AWS which provides real time monitoring of AWS resources and computer applications running on AWS infrastructure.
  + Basic monitoring is free but detailed monitoring is chargeable.
  + 
  + A *metric* represents a time ordered set of data points that are published to cloudwatch.
  + A *dimension* is a name/value pair that uniquely identifies a metric. They can be considered as categories of characteristics that describe a metric.
  + *Statistics* are metric data aggregation over specified period of time. Aggregations are made using the namespaces ,metric names, dimensions with the time period you specify.
  + An *alarm* can be used to automatically initiate actions on your behalf. It watches a single metric over a specified period of time and performs one or more specified actions.
  + 
* **AWS CloudFormation :**
  + AWS CloudFormation is a service that helps you model and set up your AWS resources so that you can spend less time managing those resources and more time focusing on your applications that run in AWS. You create a template that describes all the AWS resources that you want (like Amazon EC2 instances or Amazon RDS DB instances), and CloudFormation takes care of provisioning and configuring those resources for you. You don't need to individually create and configure AWS resources and figure out what's dependent on what; CloudFormation handles that.
  + We can create VPC with CloudFormation with the following benefits :
    - Infrastructure as a Code
    - Updateable
    - Reusable
    - Disposable
    - Drift Detection
  + There is some difference between Elastic beanstalk and Cloudformation.
    - Both CloudFormation and Elastic Beanstalk are a way of deploying your application on AWS. But they’re fundamentally different approaches. Beanstalk is PaaS (platform as a service) while CloudFormation is IaC (infrastructure as code). Beanstalk is similar to tools like Heroku and Engine Yard – a way of making provisioning easy. CloudFormation is the opposite! But once you manage the complexity, it has a lot to offer.
    - IaC refers specifically to the capability to specify infrastructure as code, the IaC service then provisioning the infrastructure described in that file.
    - The user writes a YAML or Json template specifying the AWS services used. Once uploaded, CloudFormation can then create and deploy the specified resources. AWS resources collected together to run one application are known as a stack, and it’s possible to reuse your template to deploy that stack in multiple environments.

**Snapshots and AMIs**

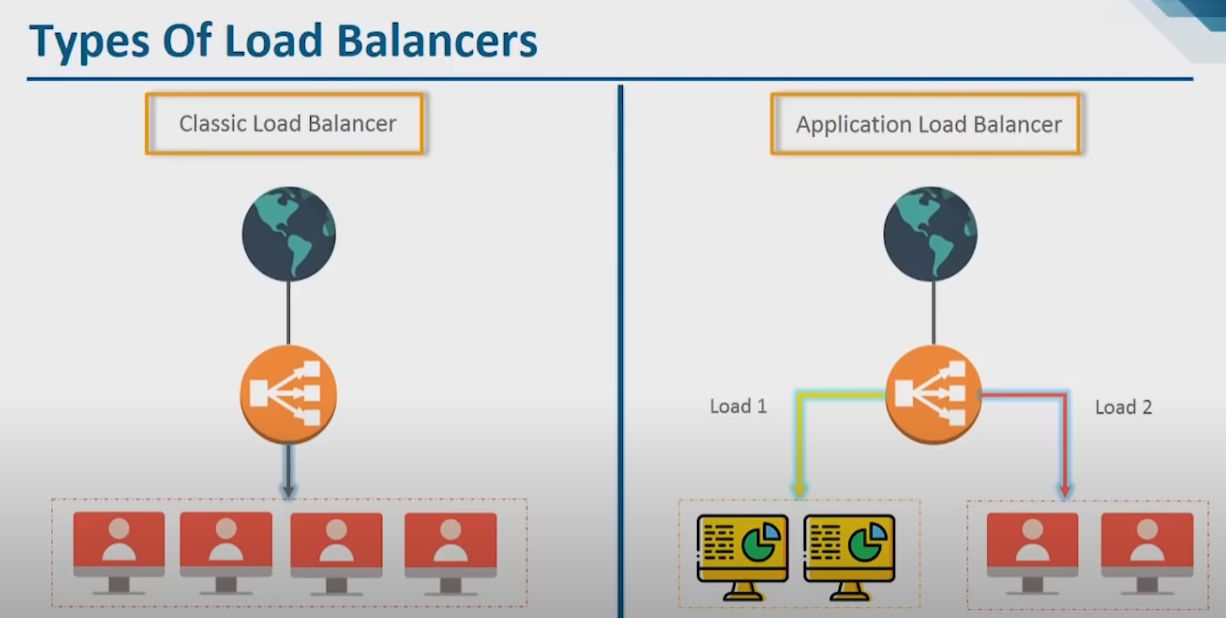
* Snapshots only capture data that has been written to your EBS volume at the time the snapshot command is issued.
* An amazon machine image (AMI) is a special type of virtual appliance that is used to instantiate or create a virtual machine within EC2. It serves as the basic unit of deployment for services delivered using EC2.

**Autoscaling**



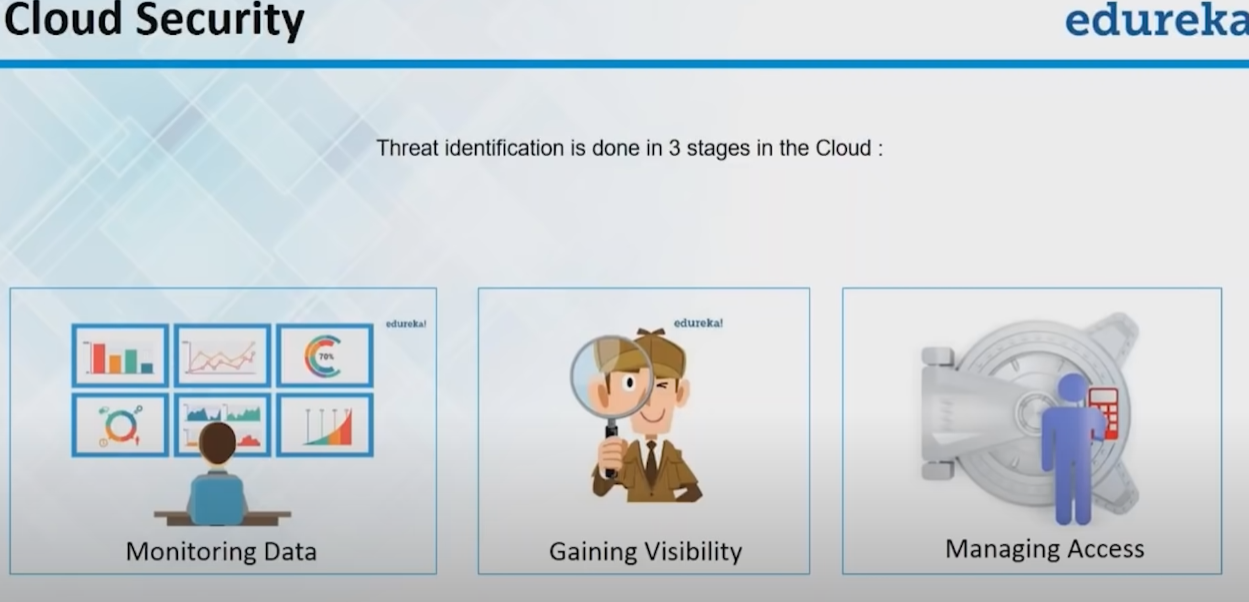
**Load Balancers :**

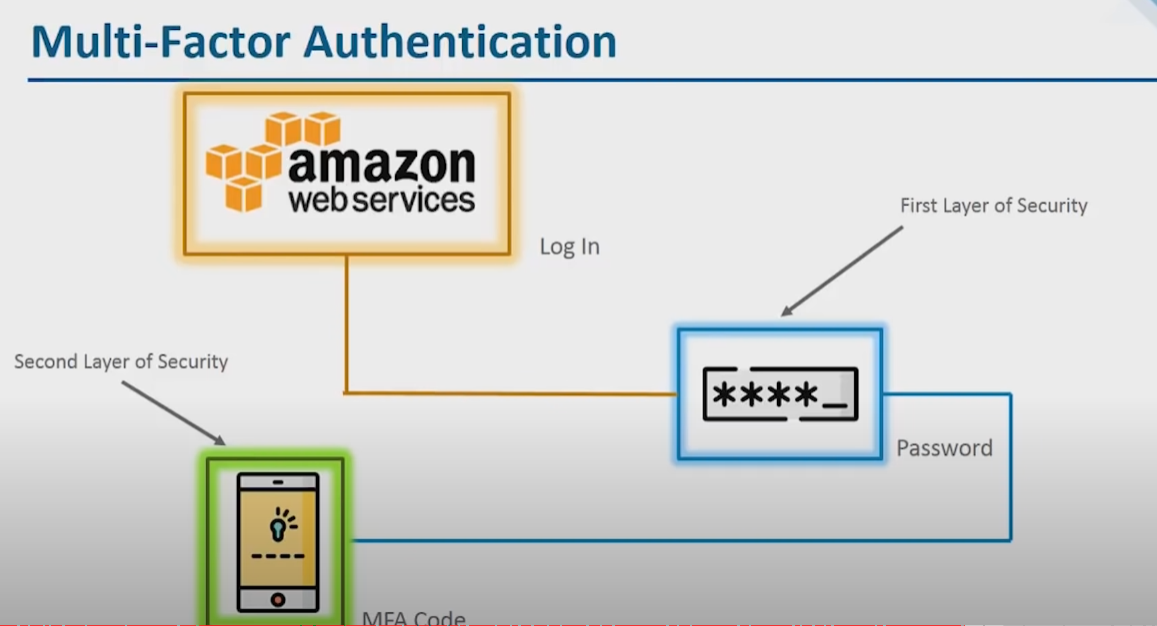
A load balancer is a device that acts as a reverse proxy and distributes network or application traffic across a number of servers. They are used to increase capacity (concurrent users) and reliability of applications



**Cloud Security**

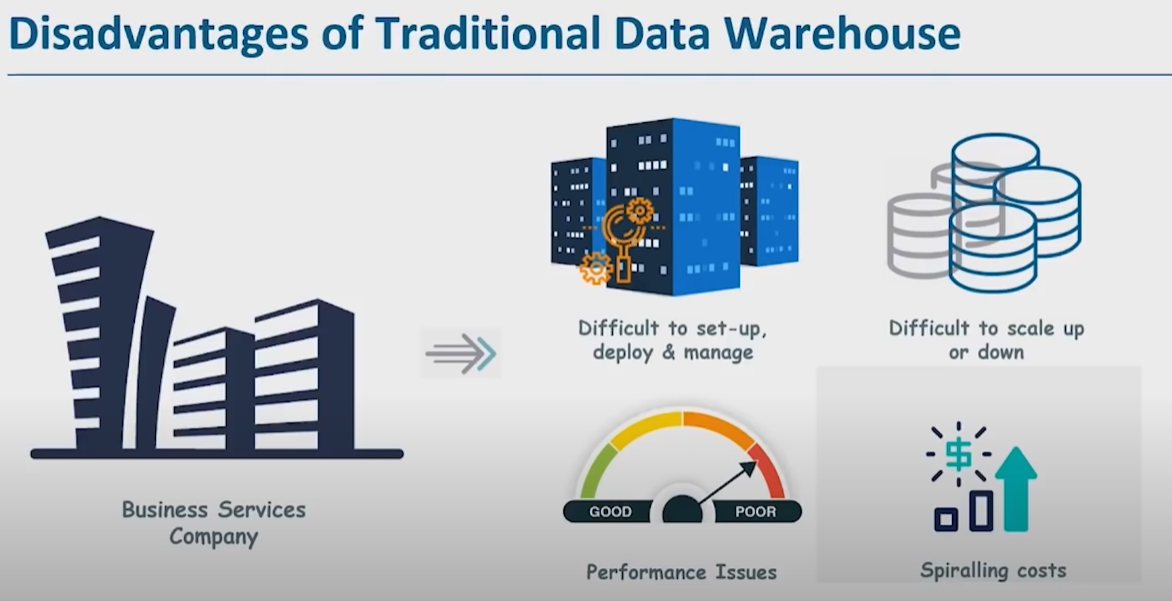
It is the use of latest technologies and security techniques to protect your data, applications and infrastructure associated with cloud computing.



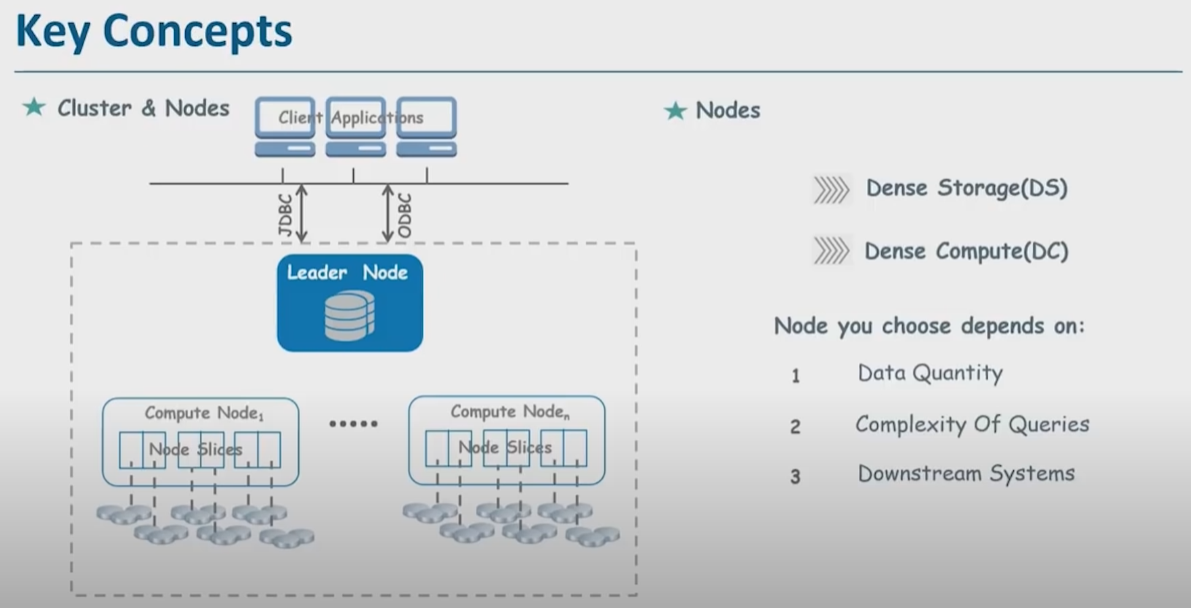
* **Monitoring Data**
  + **AWS CloudWatch:**
    - Monitor EC2 and other AWS resources
    - The ability to monitor custom metrices
    - Monitor and store logs
    - Set alarms
    - Set graphs and statistics
    - Monitor and react to resource changes
* **Gaining Visibility**
  + **AWS CloudTrail :**
    - CloudTrail is a logging service which can be used to log the history of API calls
    - It can also be used to identify which user from AWS Management console requested the particular service.
    - It can be used to identify notorious hackers.
* **Managing Access**
  + **AWS IAM :**
    - Granular Access
    - Secure access to applications running on EC2 instance
    - AWS Identity and Access Management is a web service that helps you securely control access to AWS resources for your users. You can use IAM to control who can use your AWS resources (authentication) and what resources they can use and in what ways (authorization).
    - Basic Components :
      * Users : Using IAM , you can create and manage AWS users and use permissions to allow and deny their access to AWS resources.(basically individual)
      * Groups : The users created, can also be divided among groups and then the rules and policies that apply on the group, apply on the user level as well.
      * Roles : An IAM Role is an IAM entity that defines a set of permissions for making AWS service requests. IAM roles are not associated with specific users or groups. Instead, trusted entities assume roles, such as IAM users, applications or AWS services such as EC2.( basically applications )
      * Policies : To assign permissions to a user, group, role or resources, you create a policy which is a document that explicitly lists permissions.
    - 

**Database Access**

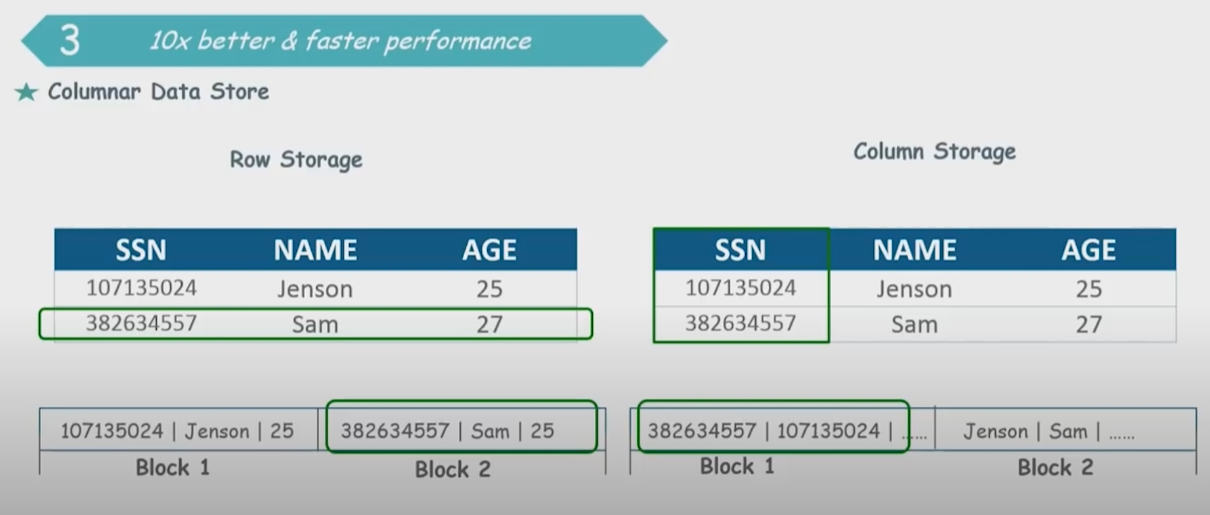
* **Amazon Redshift :**
  + A data warehouse is a subject oriented , integrated, time-variant and non-volatile collection of data in support of organisations decision making process.



* + Redshift is a massively parallel, column-oriented database deployed on the AWS platform that makes it simple and cost effective to analyse all your data across your data warehouse and data lake.

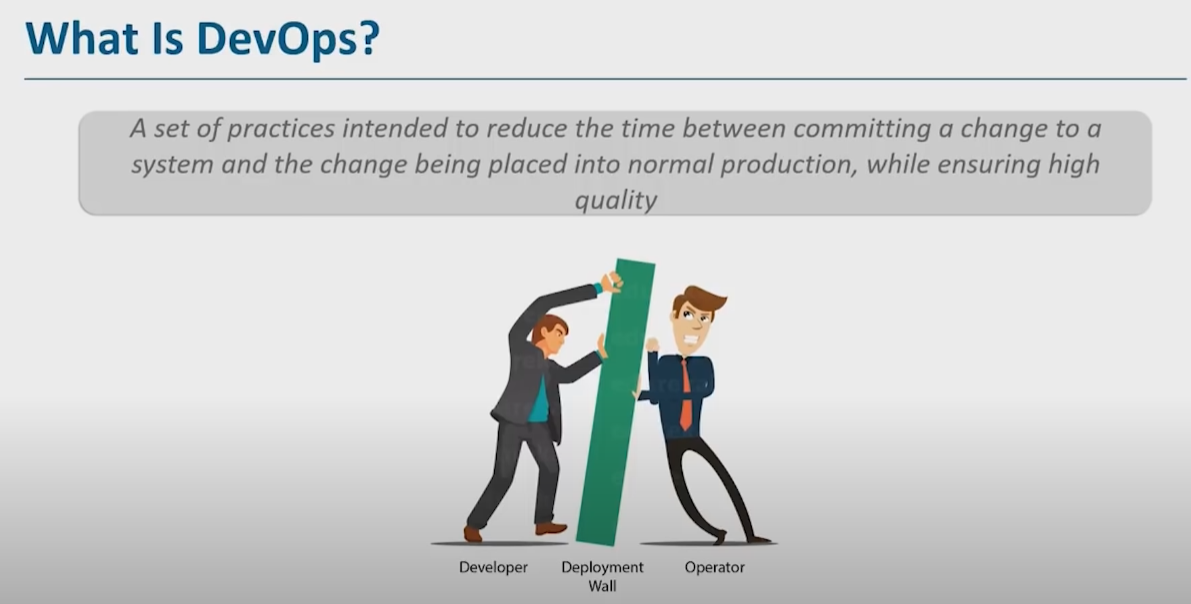


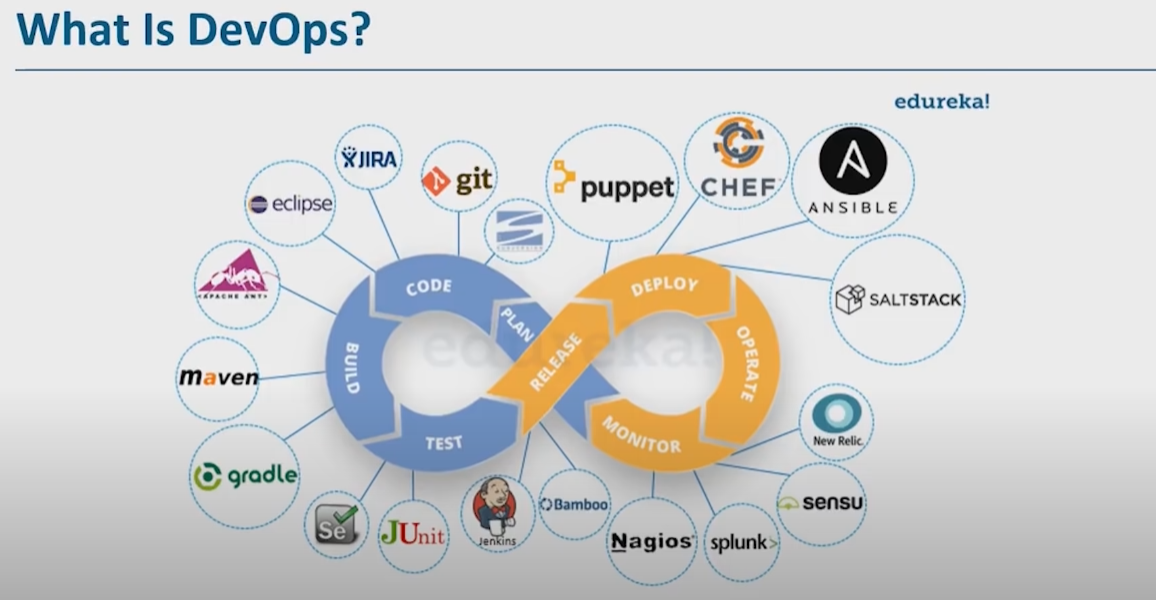
* + Benefits :
    - Easy to use, deploy and set-up
    - Scales quickly to meet your needs.
    - 10 times better and faster performance

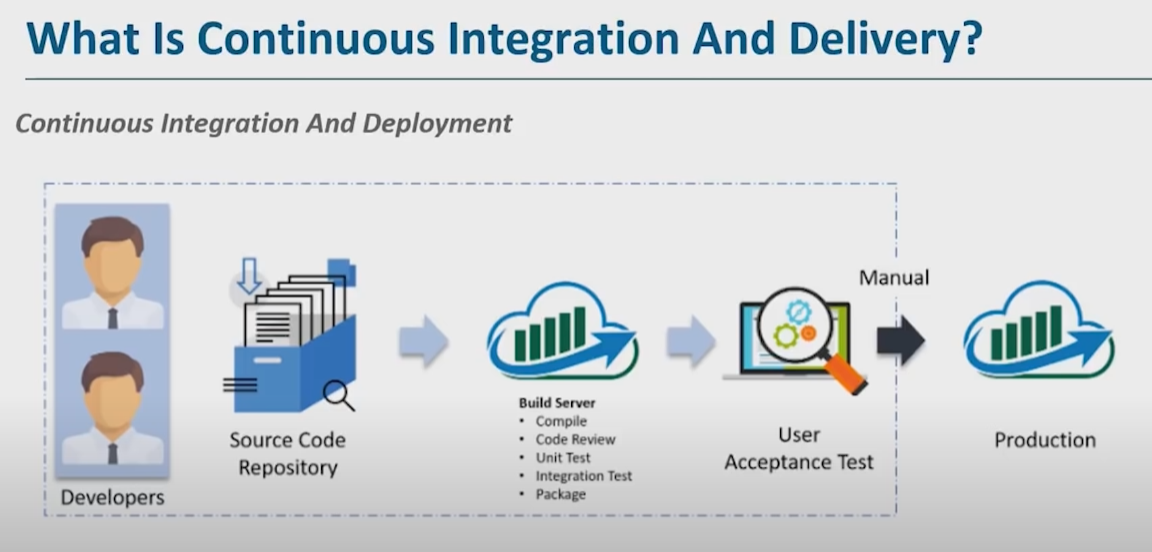


* + - Cost effective
    - Allows to query from data lake
    - Data is secure in Redshift
      * Backup and recovery
      * Encryption

**Devops on AWS**







* **AWS CodePipeling :**
  + It is a continuous delivery service you can use to model , visualise and automate the steps required to release the software.
    - Monitor your processes in real time
    - Ensure consistent release processes
    - Speed up delivery while improving quality
    - View pipeline history details
  + 