AWS Cloud Practitioner Certification Bootcamp

Week - 3

Session 3 - Amazon Compute & Application Integration Services

29th January, Saturday
7:00 PM to 8:30 PM IST

WOMEN iN BIG DATA
BANGLADESH

aws USER GROUP
BANGLADESH

AWS
User Groups

Speakers

Sanchit Jain

Lead Architect - AWS at Quantiphi

AWS APN Ambassador

Agenda

Re-cap of
Last session

Amazon Ec2

Amazon ECS

Amazon EKS

Amazon Lambda

Re-cap of Last session

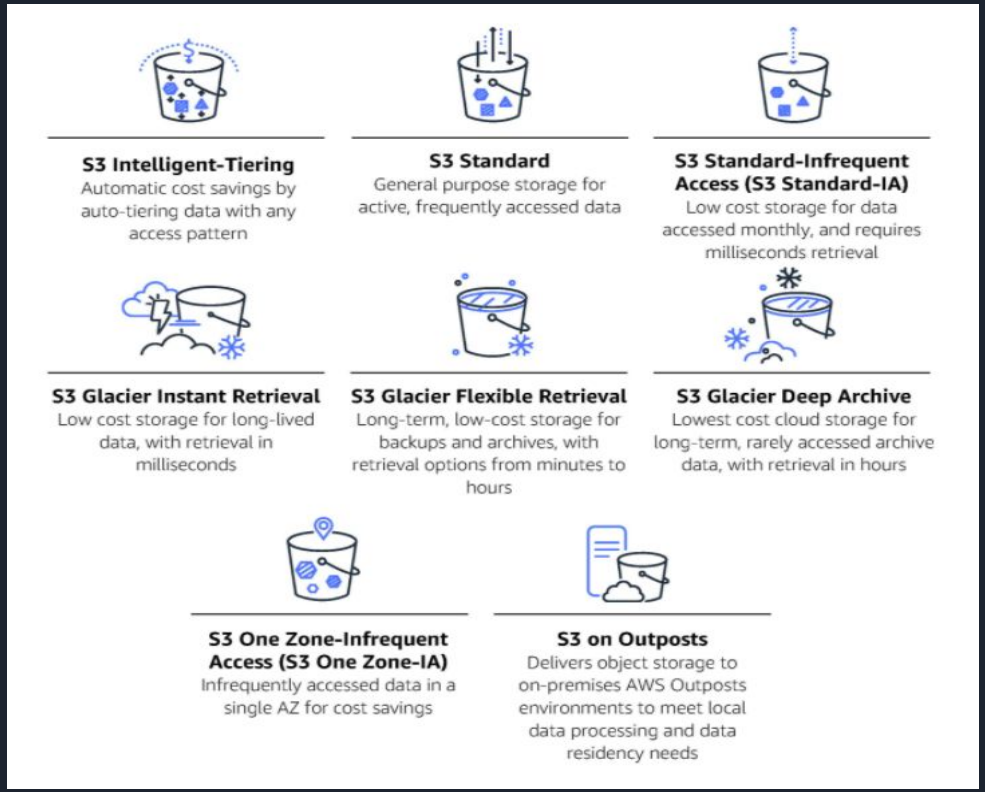
Re-cap

AWS
User Groups

Amazon S3 Overview

- Infinitely scaling storage
- Unlimited storage space & pay-as-you-use model.
- Amazon S3 allows people to store objects (files) in “buckets” (directories). S3 resources for e.g. buckets and objects are private by default
- Event notifications for specific actions, can send alerts or trigger actions, and it can be sent to:
    - SNS Topics.
    - SQS Queue.
    - Lambda functions.
    - Need to configure SNS/SQS/Lambda before S3.
    - No extra charges from S3 but you pay for SNS, SQS and Lambda.

S3 Storage Classes

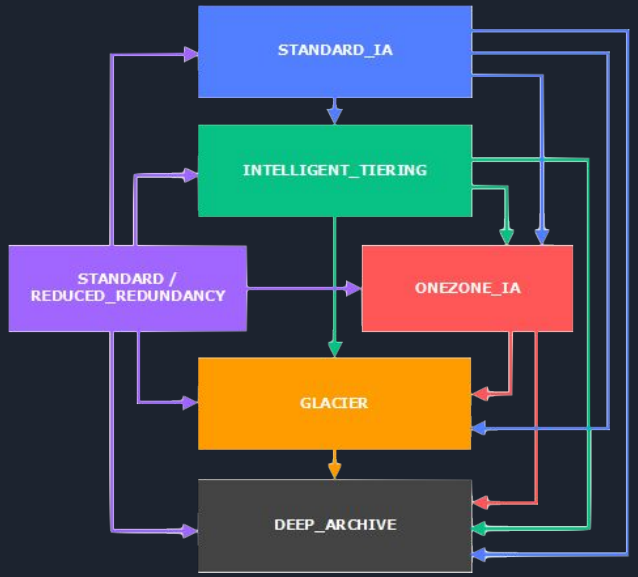


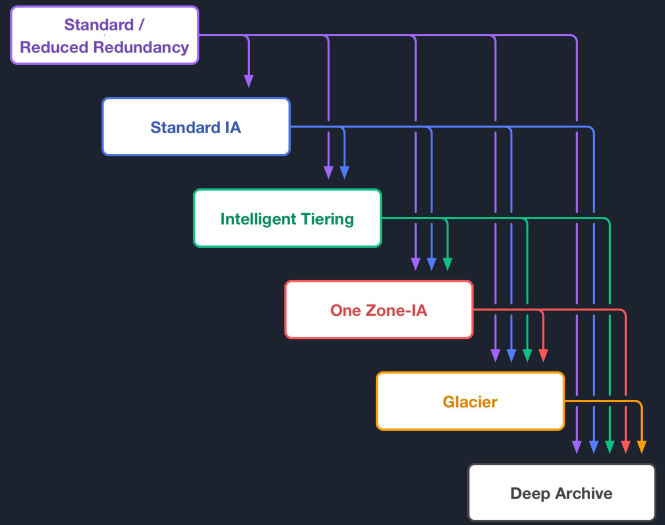
https://aws.amazon.com/s3/storage-classes-infographic/

S3 – Moving between storage classes

You can transition objects between storage classes via S3 Lifecycle Rules

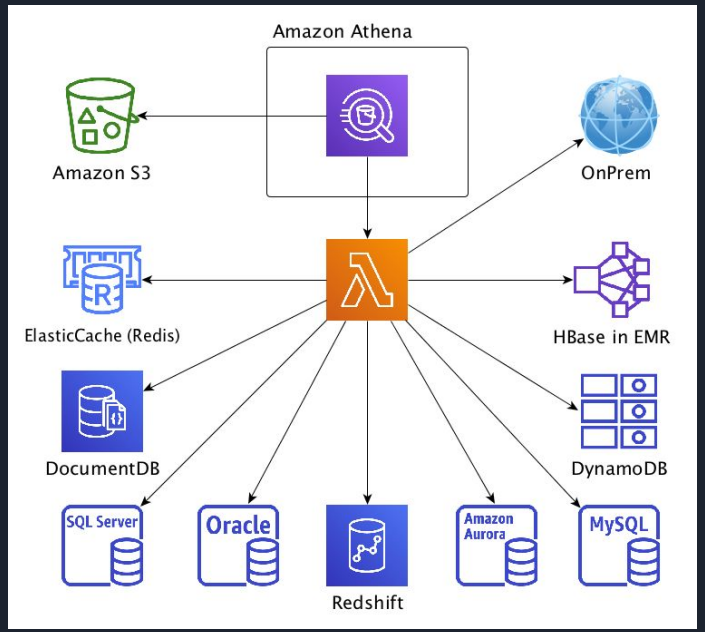
- Moving objects can be automated using a lifecycle configuration
- We can define Transition actions or Expiration actions in S3 Lifecycle Rules



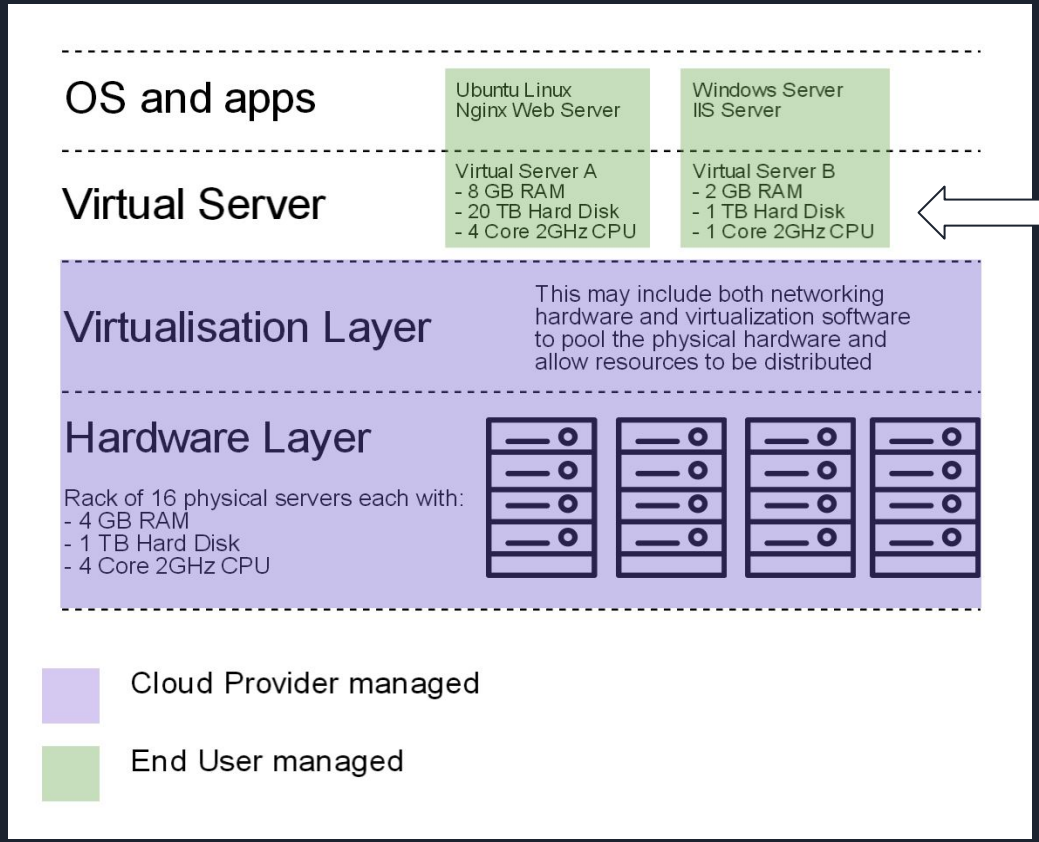


Amazon Athena

- Serverless query service to perform analytics against S3 objects
- Uses standard SQL language to query the files
- Supports CSV,JSON,ORC,Avro,and Parquet(builtonPresto)
- Pricing: $5.00 per TB of data scanned
- Use compressed or columnar data for cost-savings (less scan)
- Use cases: Business intelligence / analytics, analyze & query VPC Flow Logs, ELB Logs, CloudTrail trails, etc...
- Exam Tip: analyze data in S3 using serverless SQL, use Athena

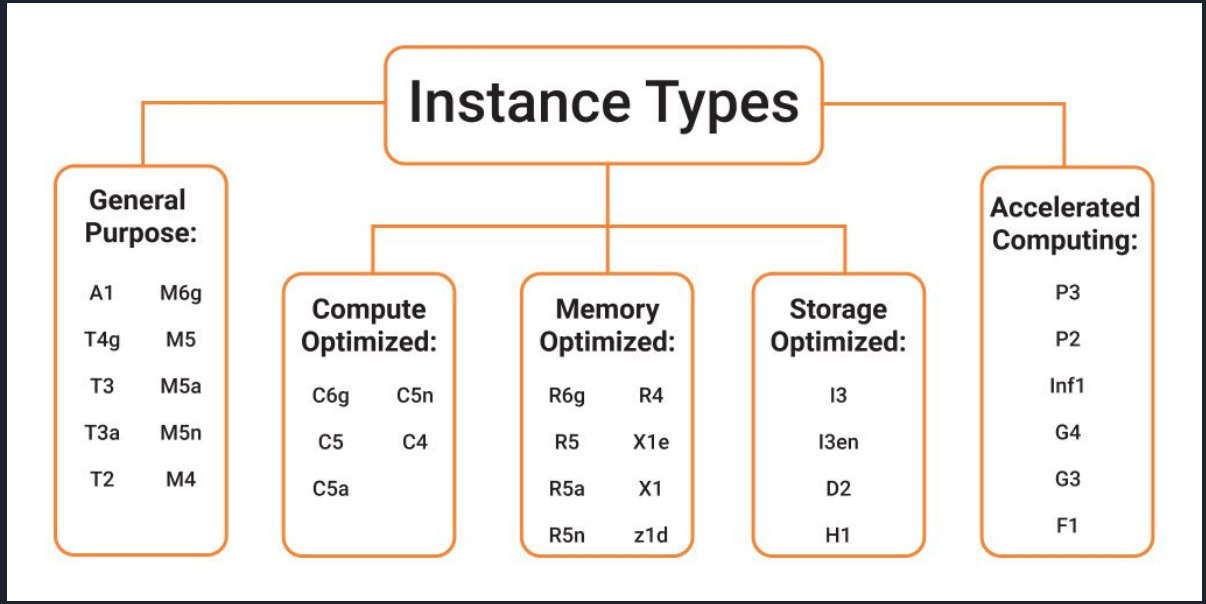


Amazon EC2


OS and apps
Ubuntu Linux
Nginx Web Server
Windows Server
IIS Server
Virtual Server
Virtual Server A
- 8 GB RAM
- 20 TB Hard Disk
- 4 Core 2GHz CPU
Virtual Server B
- 2 GB RAM
- 1 TB Hard Disk
- 1 Core 2GHz CPU
Virtualisation Layer
This may include both networking hardware and virtualization software to pool the physical hardware and allow resources to be distributed
Hardware Layer
Rack of 16 physical servers each with:
- 4 GB RAM
- 1 TB Hard Disk
- 4 Core 2GHz CPU
Cloud Provider managed
End User managed


Amazon EC2

Amazon EC2 Instance Families



EC2 Instance Billing Model

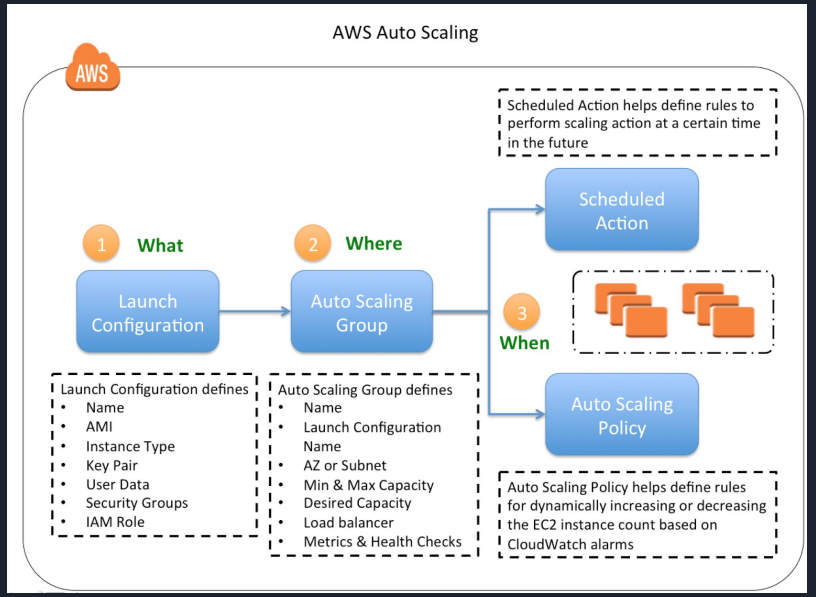
Types of EBS

	Solid-State Drives (SSD)		Hard Disk Drives (HDD)	
API Name	gp2	io1	st1	sc1
Volume Size	1 GiB – 16 TiB	4 GiB – 16 TiB	500 GiB – 16 TiB	500 GiB – 16 TiB
Max. IOPS/Volume	16,000	64,000	500	250
Max. Throughput/Volume	250 MiB/s	1,000 MiB/s	500 MiB/s	250 MiB/s
Max. IOPS/Instance	80,000	80,000	80,000	80,000
Max. Throughput/Instance	1,750 MiB/s	1,750 MiB/s	1,750 MiB/s	1,750 MiB/s
Dominant Performance Attribute	IOPS	IOPS	MiB/s	MiB/s

Amazon EC2

Ec2 AutoScaling Overview

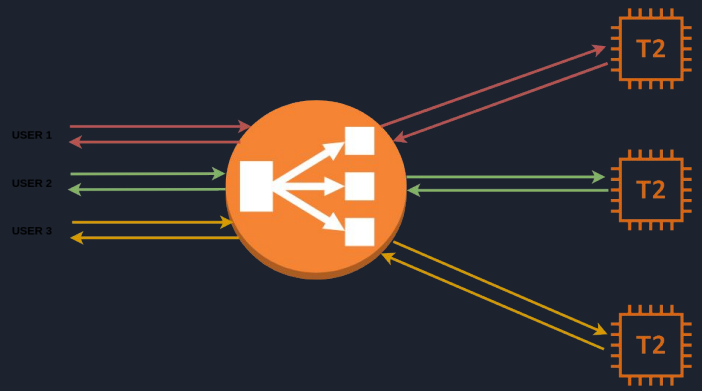
- In real-life, the load on your websites and application can change
- In the cloud, you can create and get rid of servers very quickly
- The goal of an Auto Scaling Group (ASG) is to:
  - Scale out (add EC2 instances)
  - Scale in (remove EC2 instances)
  - Ensure we have a min and a max number of machines running
  - Automatically register new instances to a load balancer
  - Replace unhealthy instances



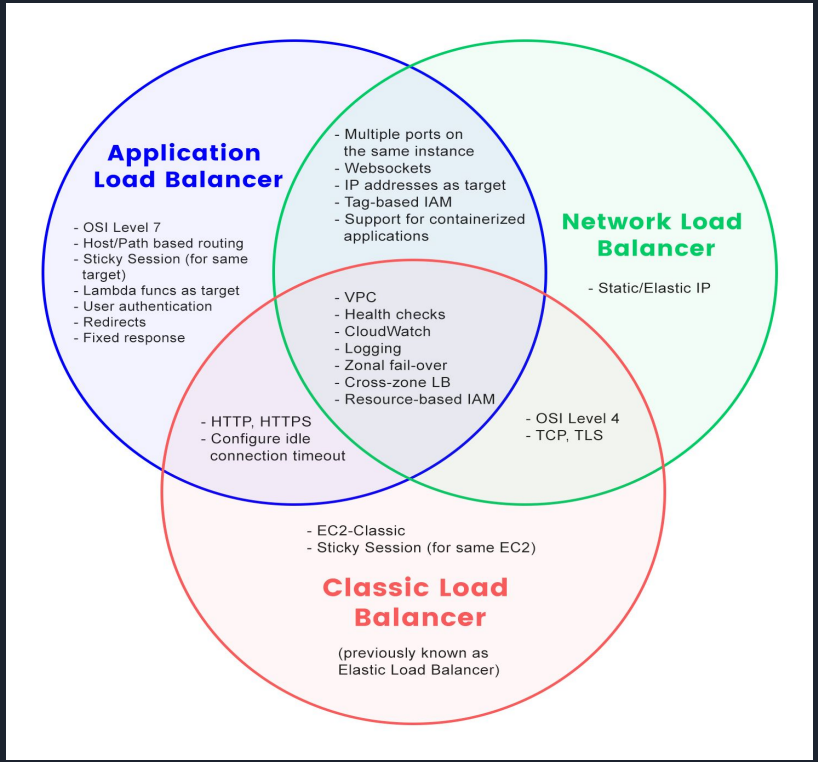
- Cost Savings: only run at an optimal capacity (principle of the cloud)

What is Elastic Load Balancer?

- Load balancers are servers that forward internet traffic to multiple servers (EC2 Instances) downstream
- An ELB (Elastic Load Balancer) is a managed load balancer
- AWS takes care of upgrades, maintenance, high availability
- AWS provides only a few configuration knobs
- It costs less to setup your own load balancer but it will be a lot more effort on your end (maintenance, integrations)

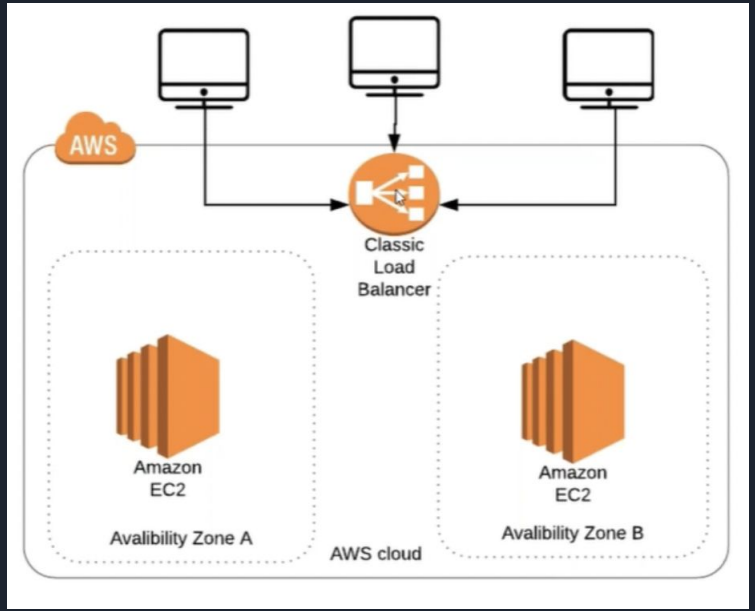


Type of Elastic Load Balancer



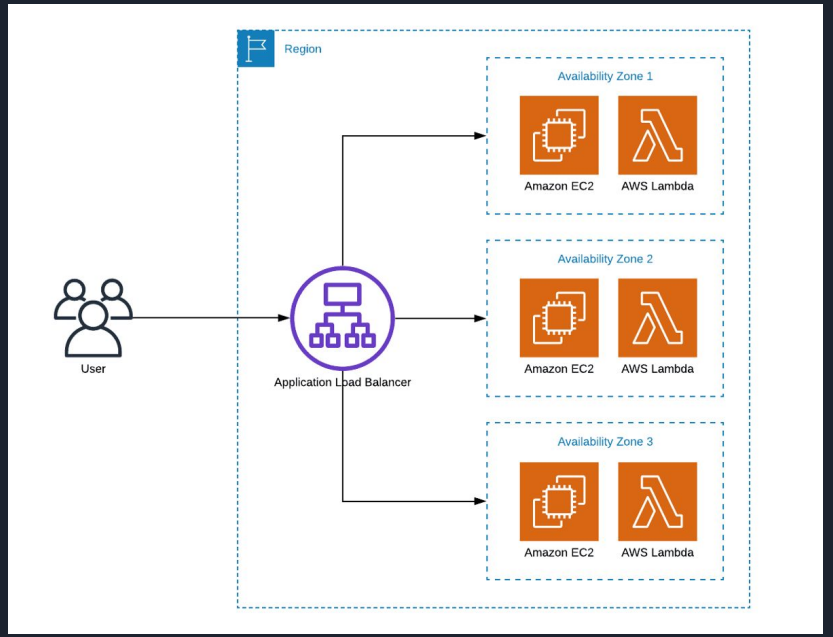
Classical Load Balancer

- Provides basic load balancing across multiple EC2 instances and operates at both the request level and connection level.
- is intended for applications that were built within the EC2-Classic network.
- is ideal for simple load balancing of traffic across multiple EC2 instances.



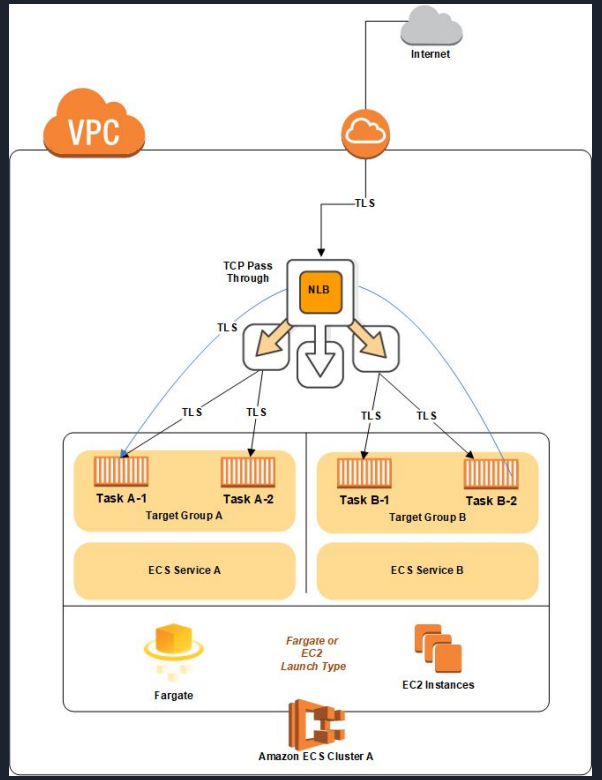
Application Load Balancer

- Ideal for microservices or container-based architectures where there is a need to route traffic to multiple services or load balance across multiple ports on the same EC2 instance.
- operates at the request level (layer 7), routing traffic to targets - EC2 instances, containers, IP addresses, and Lambda functions based on the content of the request.
- is ideal for advanced load balancing of HTTP and HTTPS traffic, and provides advanced request routing targeted at delivery of modern application architectures, including microservices and container-based applications.
- simplifies and improves the security of the application, by ensuring that the latest SSL/TLS ciphers and protocols are used at all times



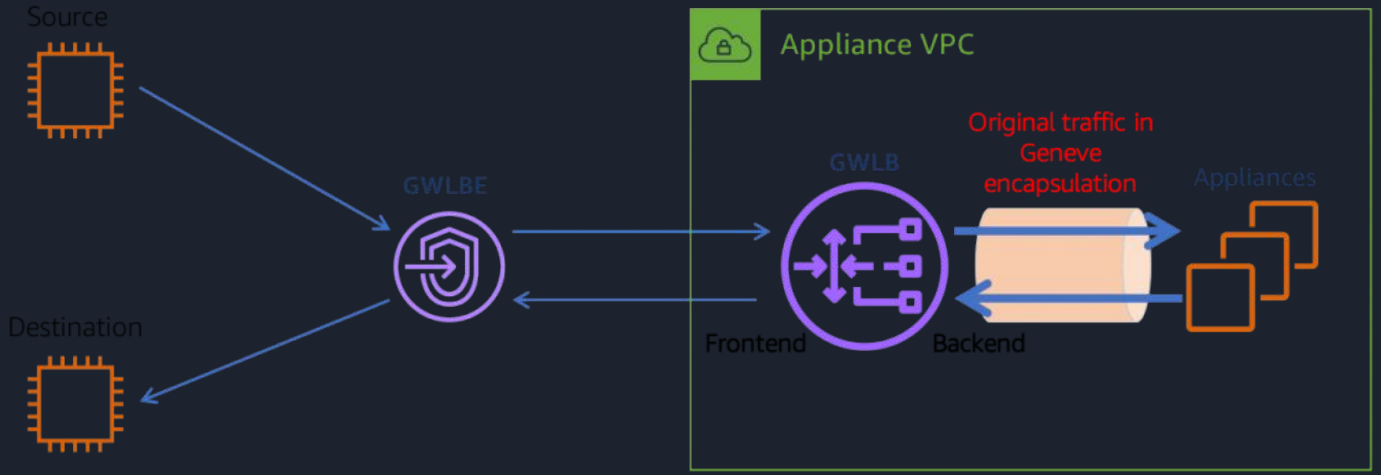
Network Load Balancer

- operates at the connection level (Layer 4), routing connections to targets – EC2 instances, microservices, and containers – within VPC based on IP protocol data.
- is ideal for load balancing of both TCP and UDP traffic,
- is capable of handling millions of requests per second while maintaining ultra-low latencies.
- is optimized to handle sudden and volatile traffic patterns while using a single static IP address per AZ
- is integrated with other popular AWS services such as Auto Scaling, ECS, CloudFormation, and AWS Certificate Manager (ACM).
- AWS recommends using Application Load Balancer for Layer 7 and Network Load Balancer for Layer 4 when using VPC



Gateway Load Balancer

AWS Gateway Load Balancer (GWLB), a new service that helps customers deploy, scale, and manage third-party virtual network appliances such as firewalls, intrusion detection and prevention systems, analytics, visibility and others



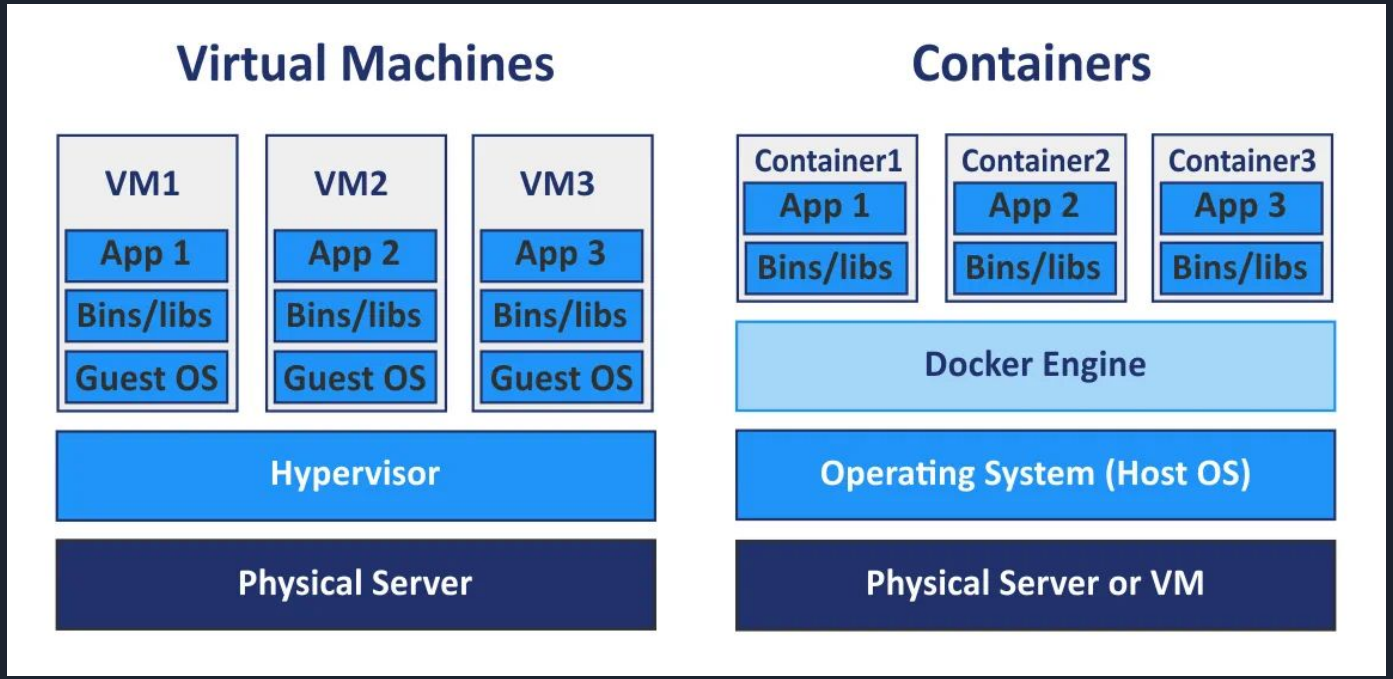
Type of Elastic Load Balancer

	ALB	**NLB**	**GLB**	**CLB**
OSI model layer	7	4	3	7 and 4
Protocol supported	HTTP, HTTPS	TCP, UDP, TLS	GENEVE	HTTP, HTTPS, TCP
Supports static IP for ELB	No, only DNS name	Yes	No	No, only DNS name
SSL offloading	Yes	Yes (TLS termination)	No	Yes
SNI support	Yes	Yes	No	No
Authentication offloading	Yes	No	No	No
End to end encryption	No if using SSL offloading	Yes	No	Yes
Sticky sessions	Yes	Yes	Yes	Yes
Path patterns	Yes			
Cross zone load balancing	Enabled by default	Yes	Yes	Disabled. Enable it manually
Type of registered targets	Instance, Lambda, IP	Instance, IP	Instance, IP	Instance,IP
Use cases	Websites, web applications	Application requiring low latency load balancing	Load balancing or scaling virtual appliances for IDP, firewall etc.	Web applications.

Amazon ECS

Amazon EKS

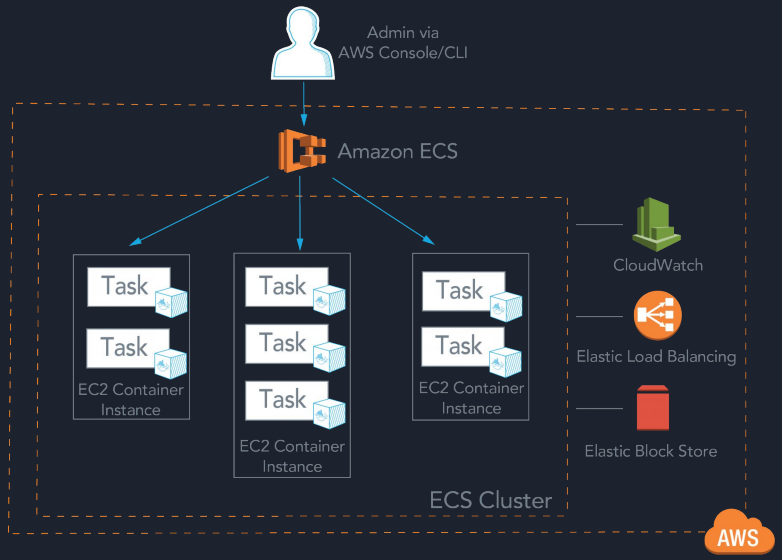
VMs vs. Containers



Overview AWS ECS

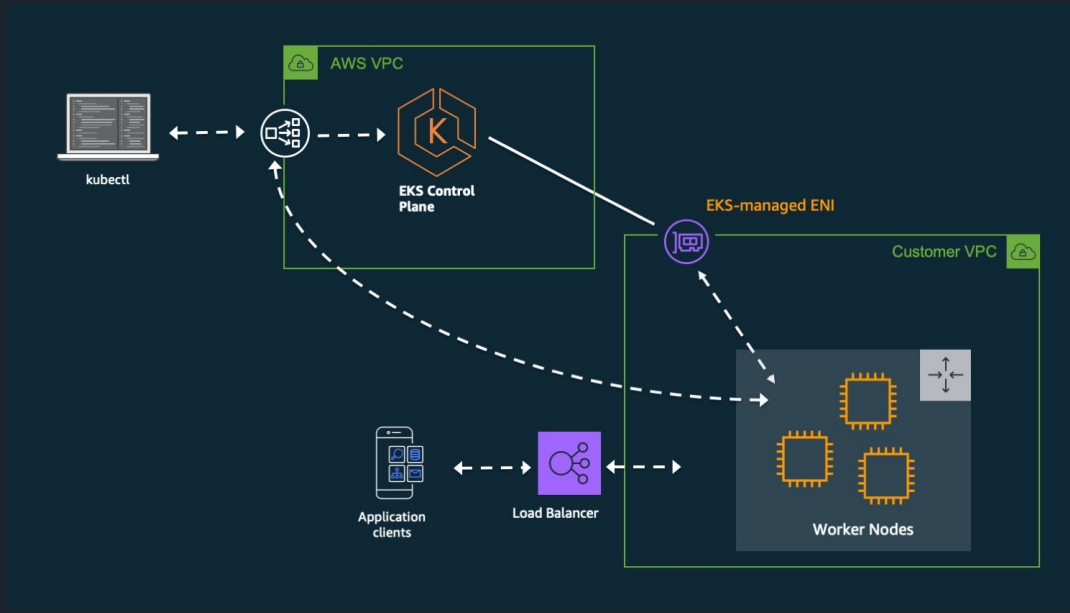
ECS, which is provided by Amazon as a service, is composed of multiple built-in components which enable administrators to create clusters, tasks and services

- **State Engine**: The state engine is designed to keep track of available hosts, running containers, and other functions of a cluster manager.
- **Schedulers**: These components use information from the state engine to place containers in the optimal EC2 container instances.
- **Cluster**: This is a logical placement boundary for a set of EC2 container instances within an AWS region.
- **Tasks**: A task is a unit of work. Task definitions, written in JSON, specify containers that should be co-located (on an EC2 container instance).
- **Services**: This component specifies how many tasks should be running across a given cluster.

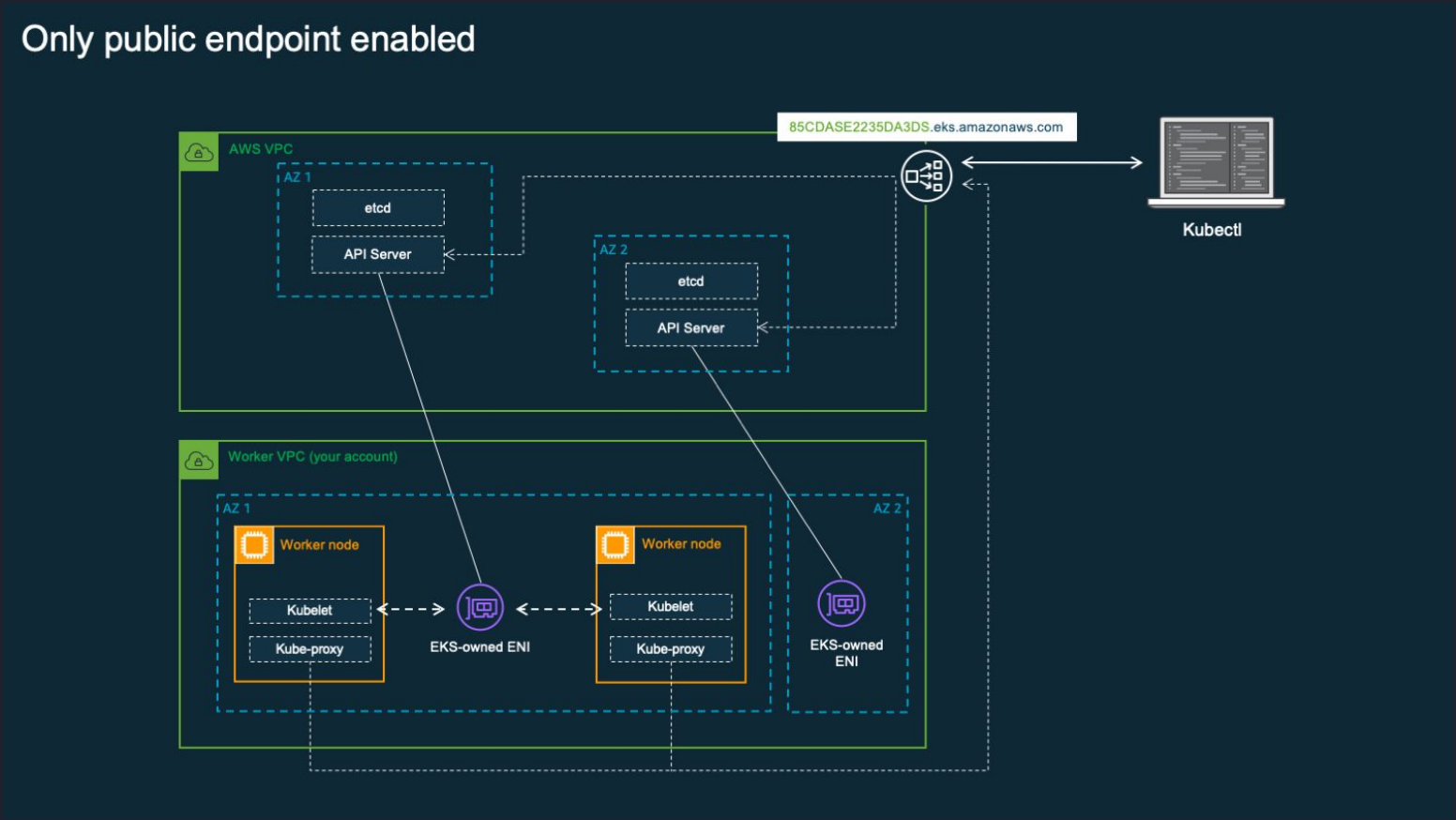


Overview AWS EKS

- Amazon Elastic Kubernetes Service (Amazon EKS) is a managed service that you can use to run Kubernetes on AWS without needing to install, operate, and maintain your own Kubernetes control plane or nodes.
- Kubernetes is an open-source system for automating the deployment, scaling, and management of containerized applications. Amazon EKS:
- Automatically scales control plane instances based on load, detects and replaces unhealthy control plane instances.



De-mystifying AWS EKS



AWS ECS or AWS EKs

	Amazon ECS	Amazon EKS
Open source	No - AWS proprietary	Yes - Kubernetes
Atomic Container Term	Task	Pod
Deployment Effort	Easy (AWS Dashboard)	Medium (AWS plus Kubernetes knowledge required)
Security (IAM)	Comes with the service	Requires addon software and additional configuration
Security (ENI support)	Yes - per task (single container)	Yes - per Pod (which can serve multiple containers)
Per VM Container Limit	Up to 120	Up to 750 Pods (which can host multiple containers)
System Service Cost	Used resources	Used resources plus ~$75 per cluster per month
Multi-cloud integration	No - AWS specific	Yes - Public and Private cloud integration.

Amazon Lambda

What is serverless?

What is Serverless?

a cloud-native platform

for

short-running, stateless computation

and

event-driven applications

which

scales up and down instantly and automatically

and

charges for actual usage at a millisecond granularity

Greater Agility

Less Overhead

Better Focus

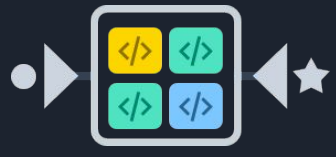
Increased Scale

More Flexibility

Faster Time To Market

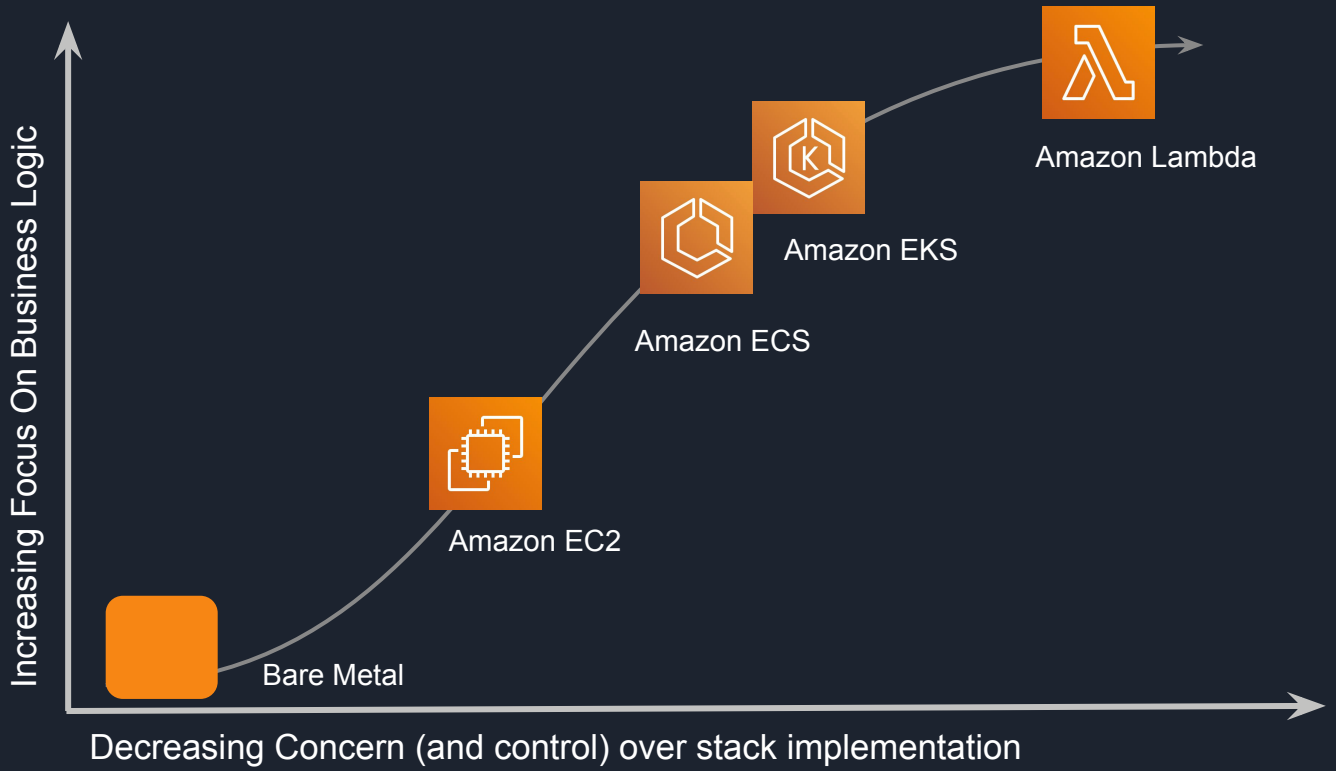
Why is Serverless attractive?

- Server-less means no servers? Or worry-less about servers?
- Runs code **only** on-demand on a per-request basis
- Making app development & ops dramatically faster, cheaper, easier
- Drives infrastructure cost savings



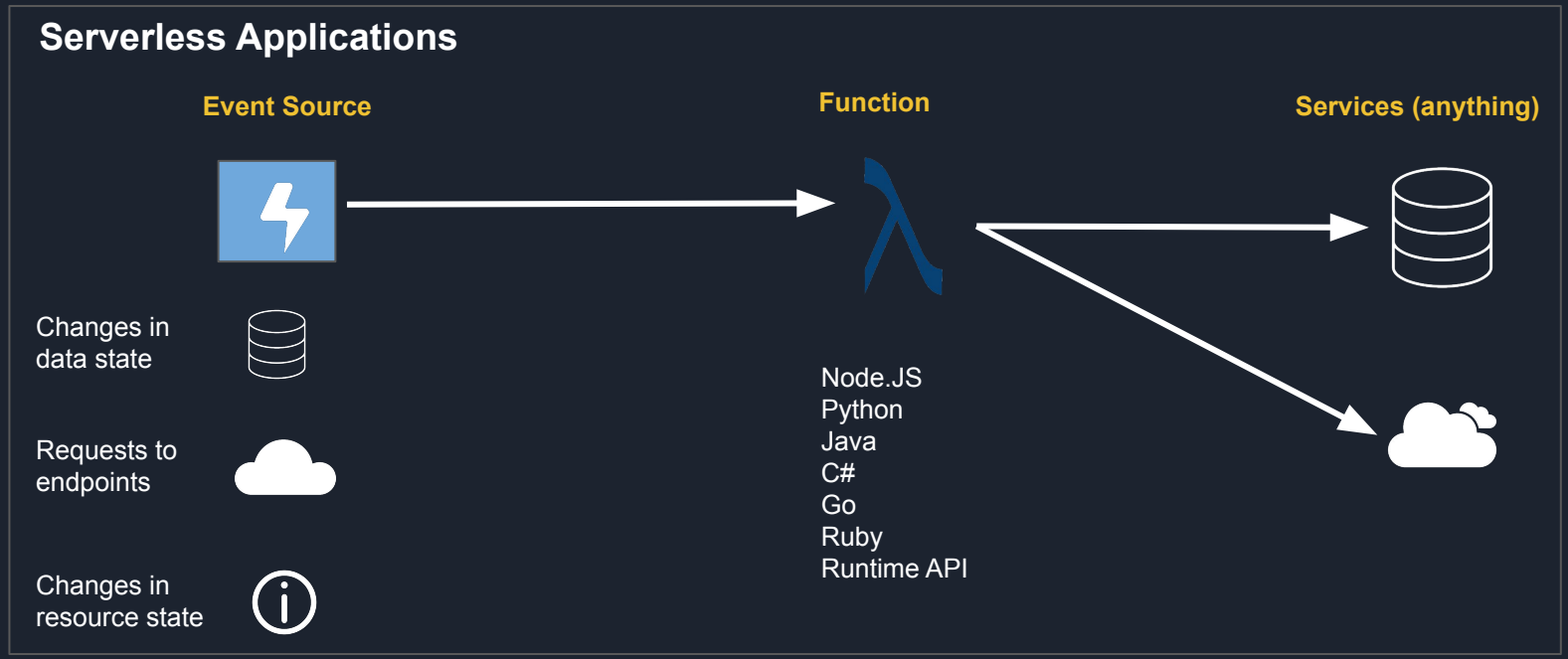
	On-prem	VMs	Containers	Serverless
Time to provision	Weeks-months	Minutes	Seconds-Minutes	Milliseconds
Utilization	Low	High	Higher	Highest
Charging granularity	CapEx	Hours	Minutes	Blocks of milliseconds

Where Serverless Stands?



What triggers code execution?

- Runs code in response to events
- Event-programming model



What is Serverless good for?

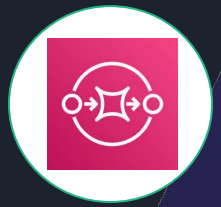
Serverless is **good** for short-running stateless event-driven

- Microservices
- Mobile Backends
- Bots, ML Inferencing
- IoT
- Modest Stream Processing
- Service integration

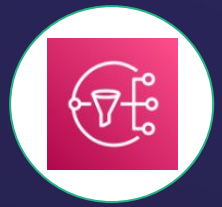
Serverless is **not good** for Long-running stateful number crunching

- Databases
- Deep Learning Training
- Heavy-Duty Stream Analytics
- Numerical Simulation
- Video Streaming

Application Integration



AWS SQS

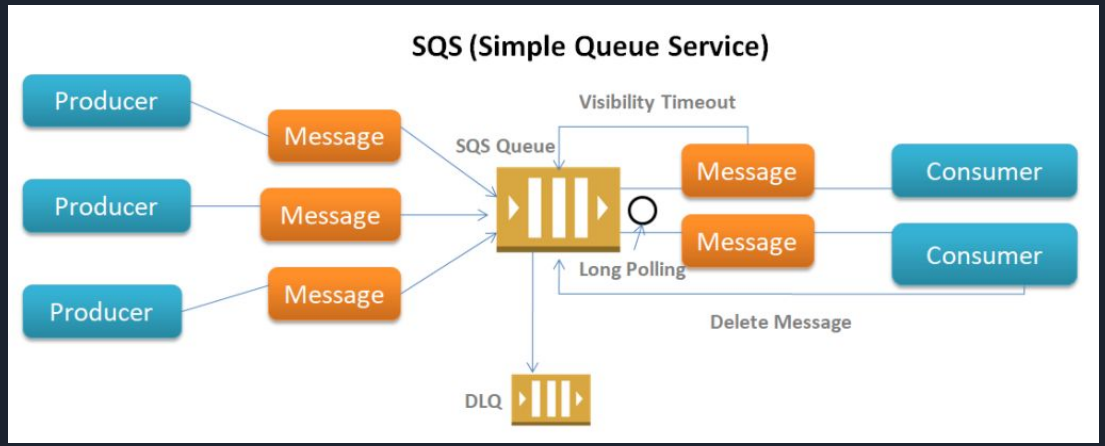


AWS SNS

AWS Step Function

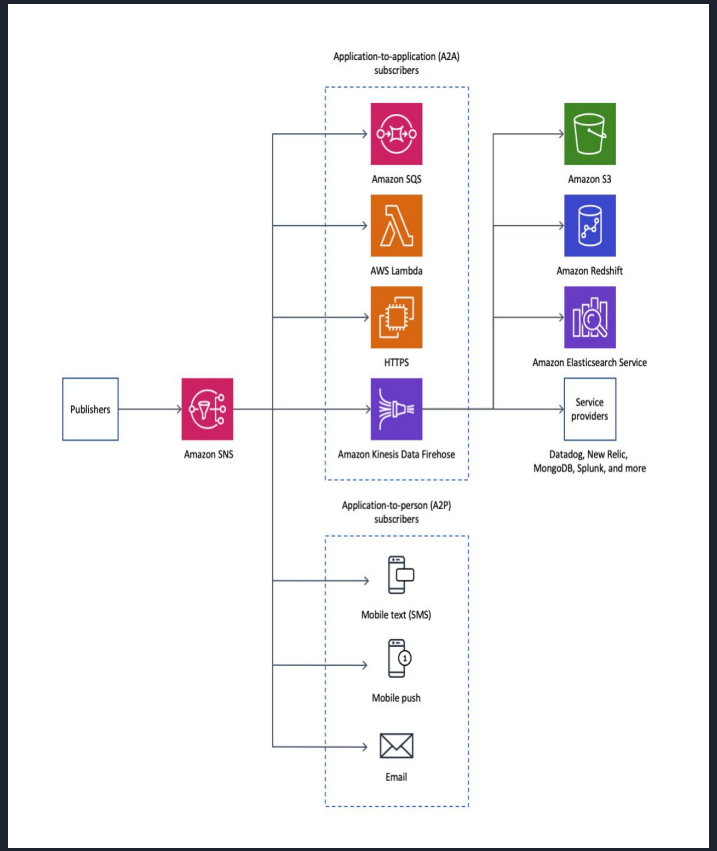
What is AWS SQS?

- Oldest AWS offering (over 10 years old)
- Fully managed service, use to decouple applications
- Scales from 1 message per second to 10,000s per second, and Low latency (<10 ms on publish and receive)
- Default retention of messages: 4 days, maximum of 14 days
- No limit to how many messages can be in the queue, and messages are deleted after they’re read by consumers
- Consumers share the work to read messages & scale horizontally



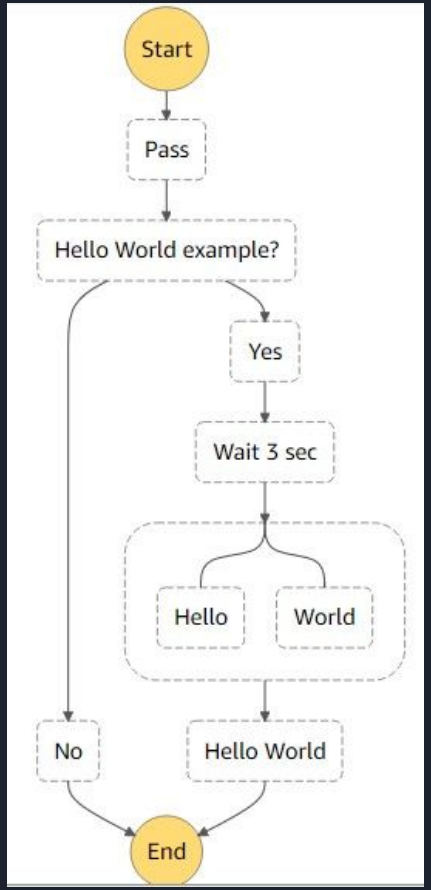
What is AWS SNS?

- Publishers sends message to a SNS topic and multiple subscribers listen to the SNS topic notifications
- Each subscriber to the topic will get all the messages
- Up to 10,000,000 subscriptions per topic, 100,000 topics limit
- SNS Subscribers can be:
  - HTTP / HTTPS (with delivery retries – how many times)
  - Emails,SMS messages,Mobile Notifications
  - SQS queues (fan-out pattern), Lambda Functions (write-your-own integration)



What is AWS Step Functions?

- AWS Step Function is a serverless orchestration service that allows integrating multiple AWS services to collate & design an enterprise-critical application or workflow with advance conditional branching and error handling
- ASL consist of three things
  - **State Machine Structure** - State machines are declared using JSON text and represents a structure consists of Comment, TimeoutSeconds, Version, StartAt, States
  - **Intrinsic functions** - Intrinsics are constructs like in programming languages, and can be leveraged to manipulate the data going to and from Task Resources
  - **Common State Fields** - Common State Fields consists of Comment, InputPath, OutputPath, Type, Next, End



Thank you!

aws USER GROUP
BANGLADESH

AWS
User Groups