Pantech e Learning
DIGITAL LEARNING SIMPLIFIED

Amazon Web Services

MLOps with AWS

Masterclass

aws

Machine Learning Operations with AWS

Day -4

Amazon Web Services

MLOps with AWS

Masterclass

Pantech e Learning

DIGITAL LEARNING SIMPLIFIED

AWS Lambda

AWS Lambda

AWS Lambda is a serverless compute service

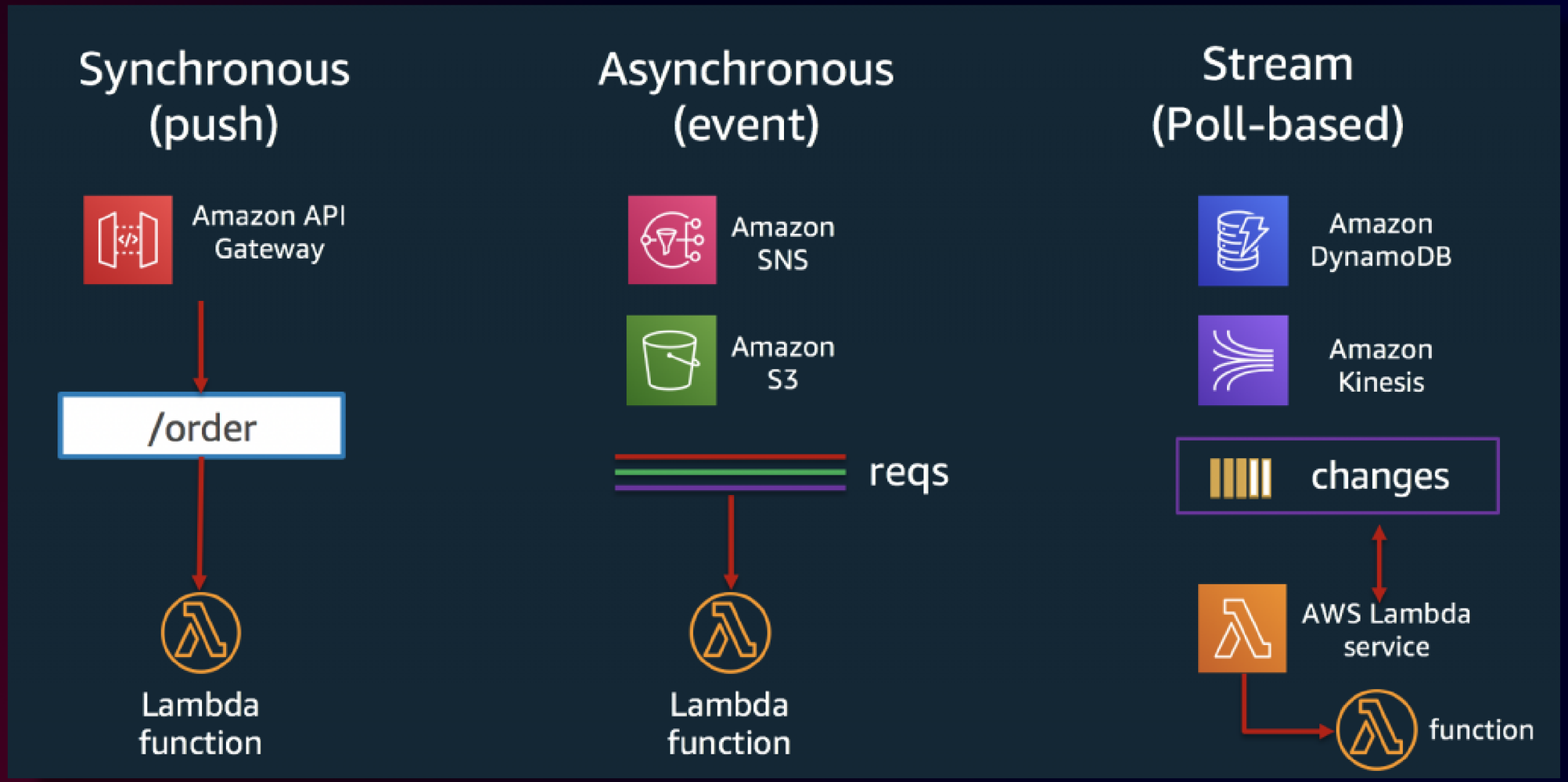
Serverless ?

In Serverless computing, the cloud provider allocates machine resources on demand, taking care of the servers on behalf of their customers.

AWS Lambda

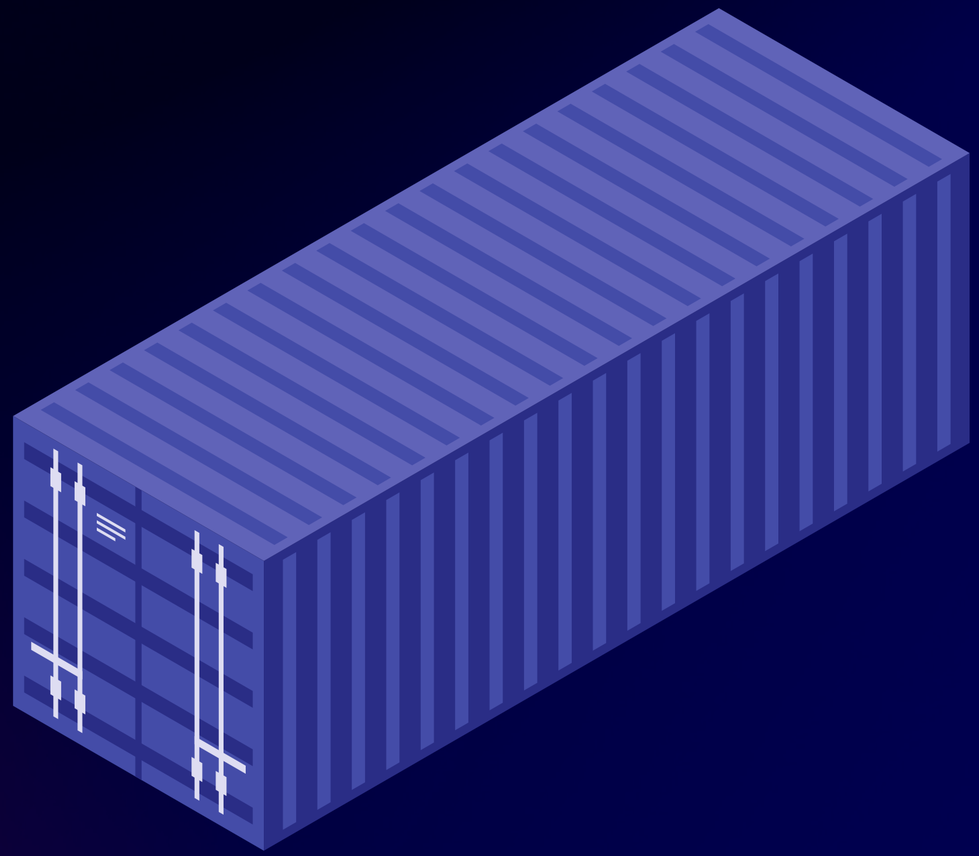
- AWS Lambda is a serverless compute service that runs your code in response to trigger and automatically manages the underlying compute resources for you.
- These trigger may include changes in state or an update, such as a user placing an item in a shopping cart on an ecommerce website.

AWS Lambda Triggers



AWS Lambda use case example

Containers

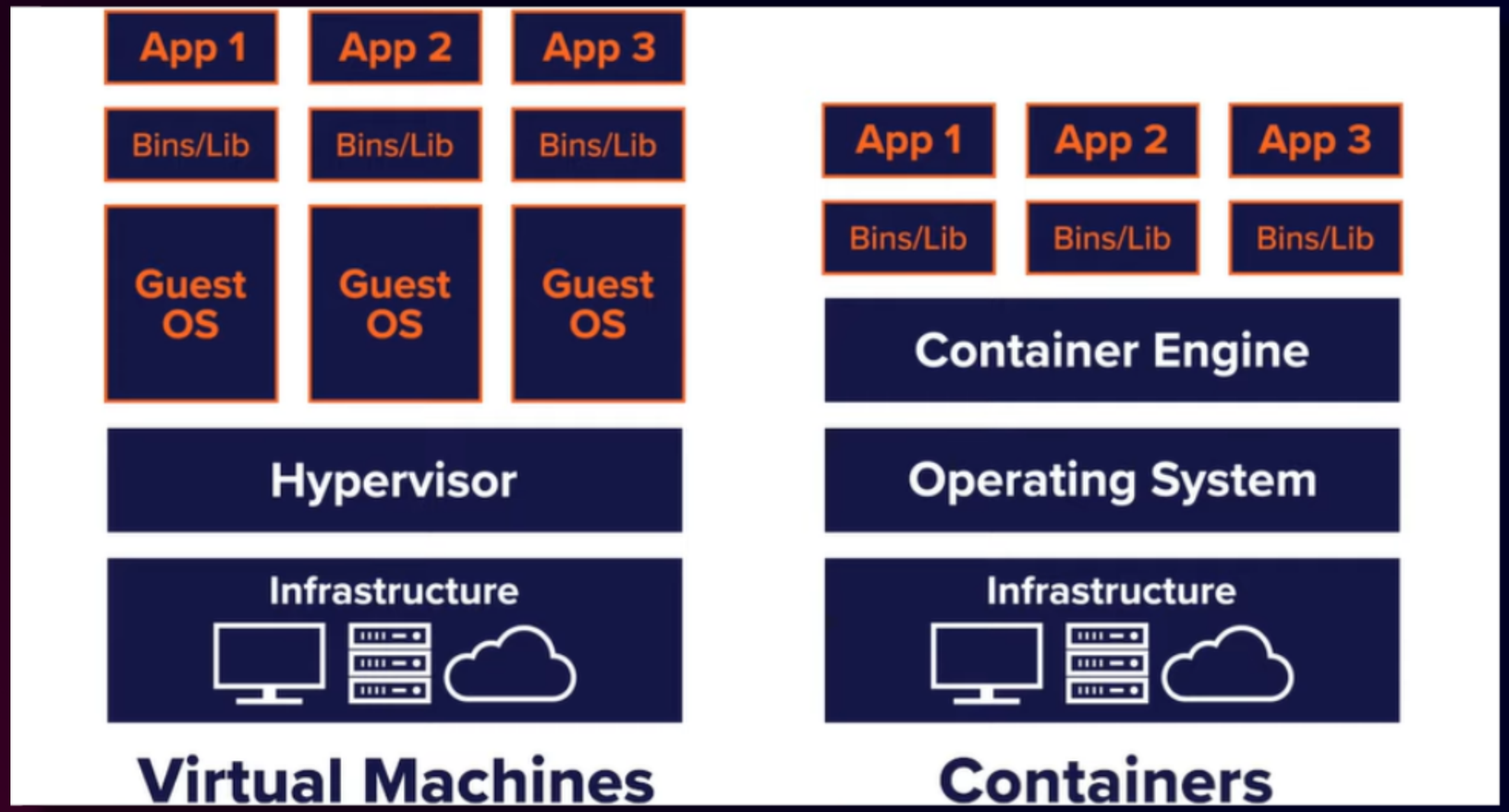


Containers

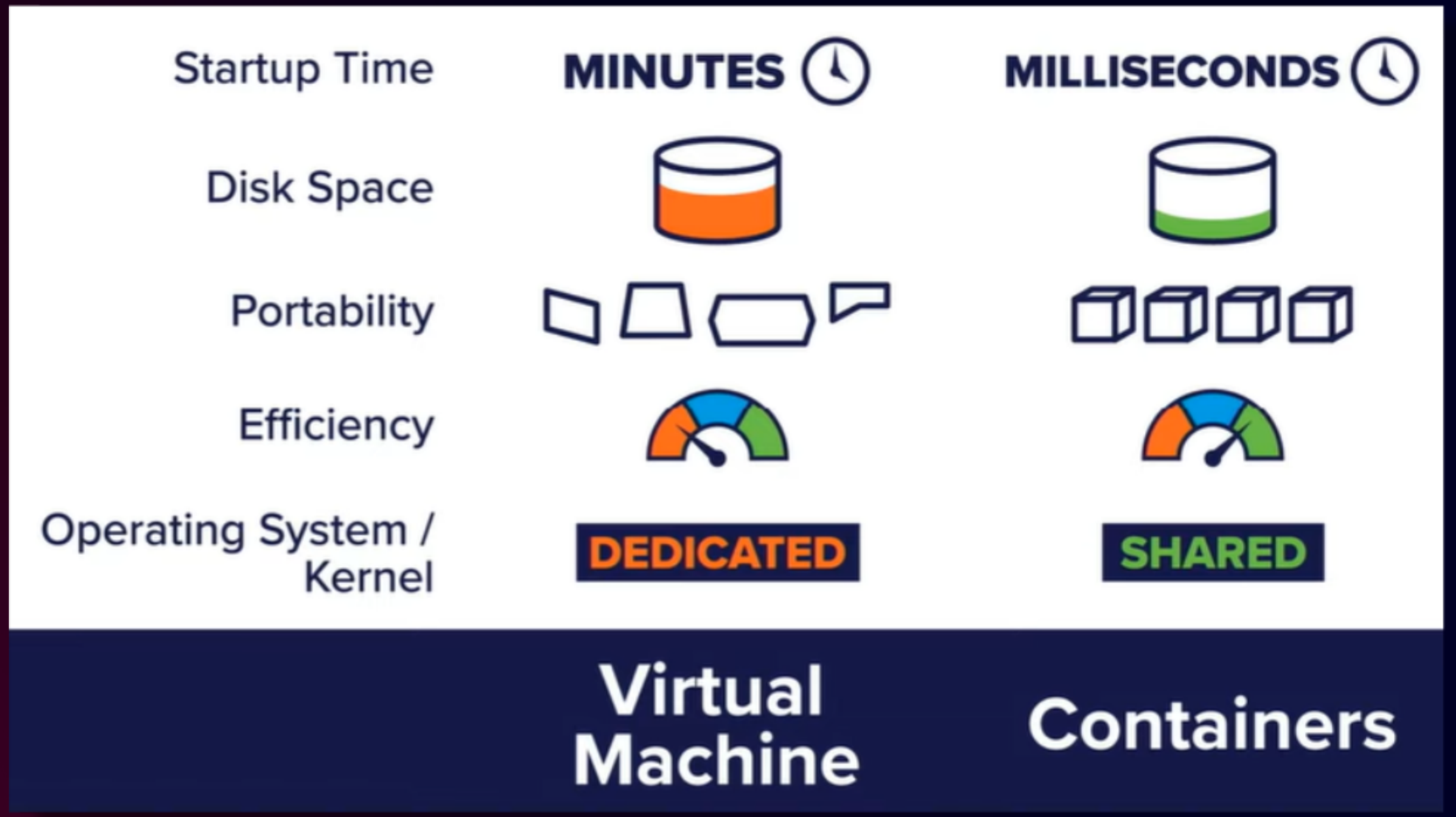
- Containers are a lightweight and isolated unit of software packaging that include everything needed to run an application, such as code, runtime, system tools, and libraries.

- Containers provide a consistent and predictable environment for software to run, regardless of the underlying infrastructure.

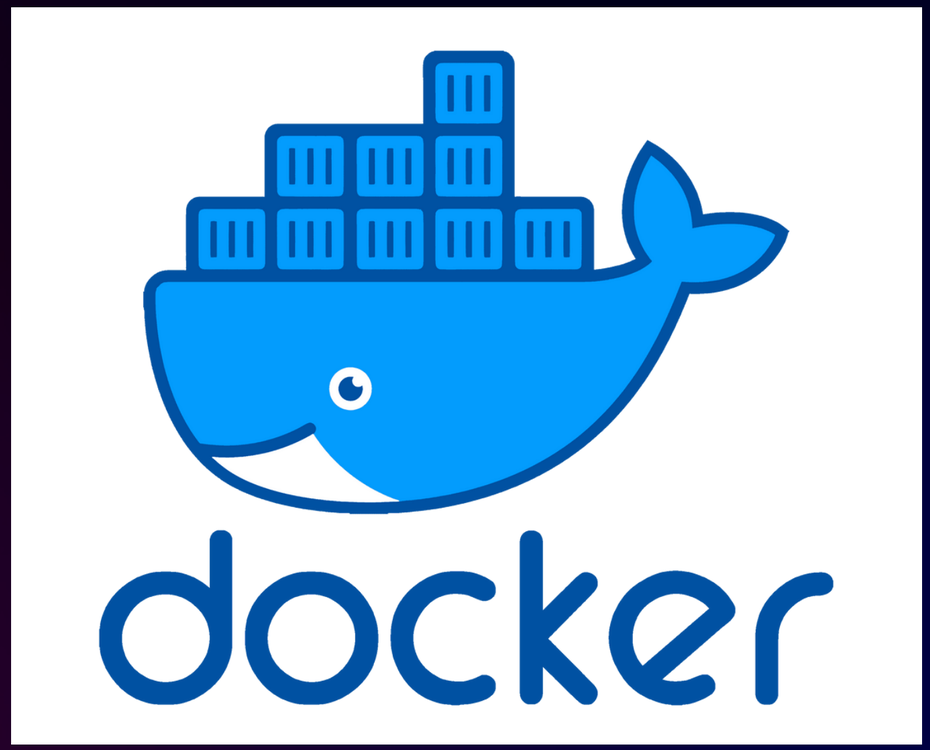
Containers vs Virtual Machines



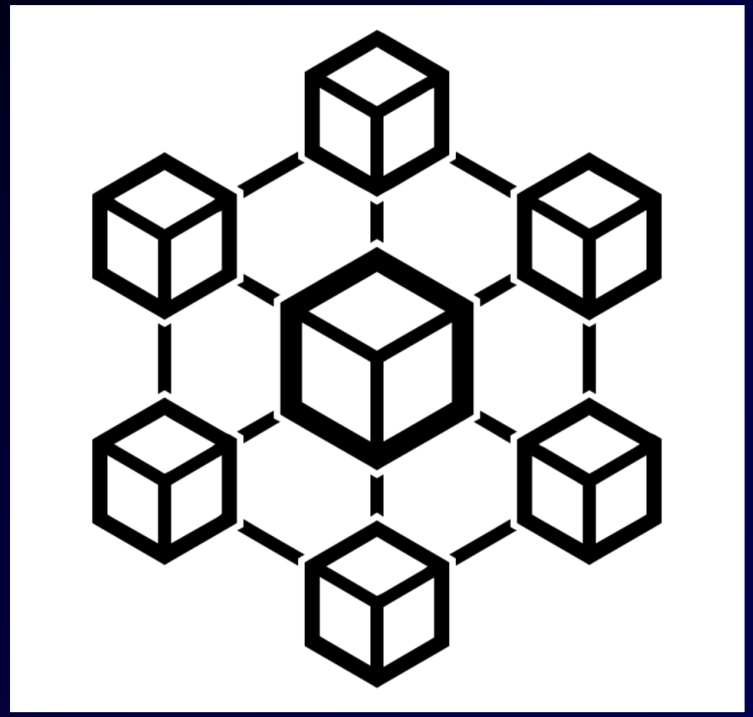
Containers vs Virtual Machines



Top Container Tool



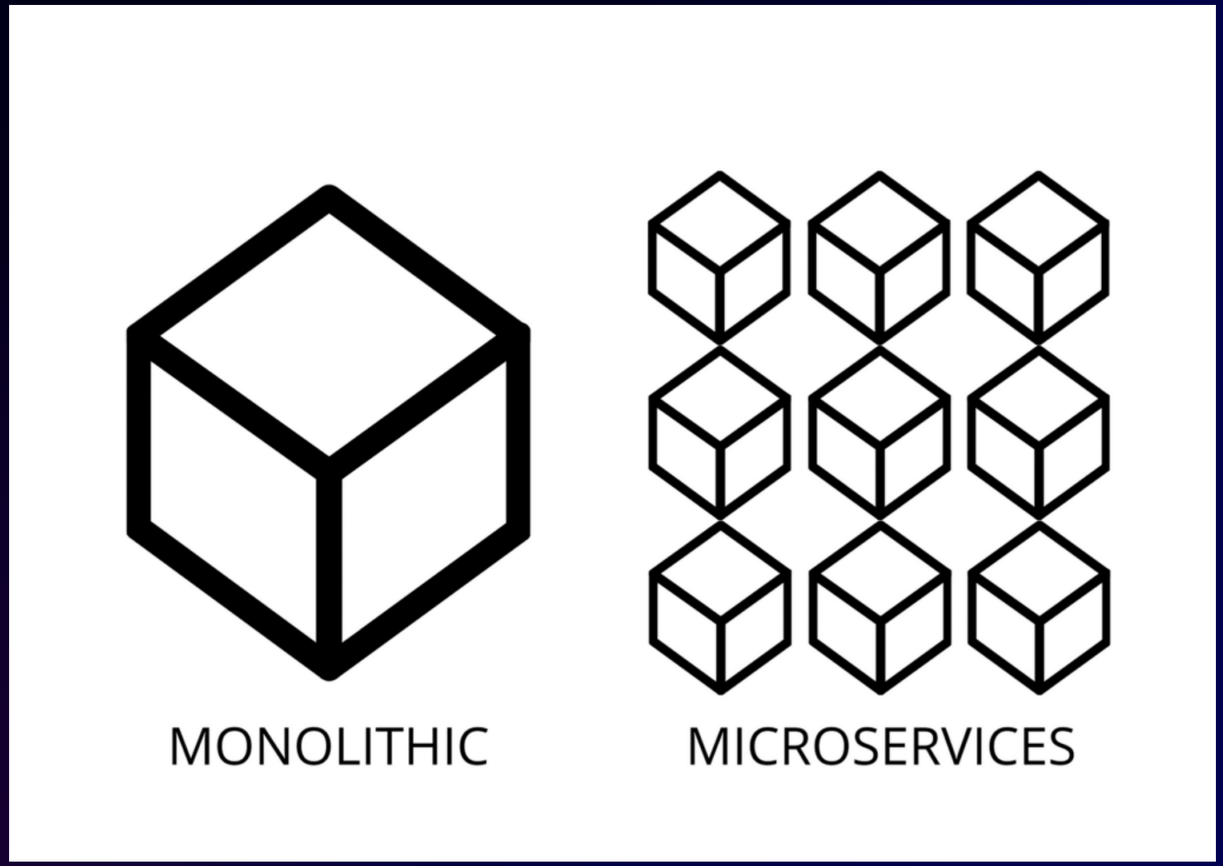
Microservices



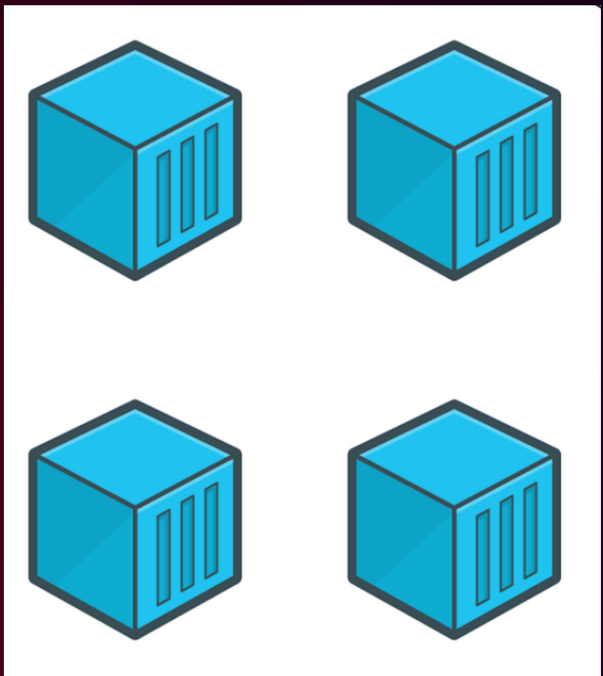
Microservices

- Microservices refer to a software architecture pattern where an application is built as a collection of small, independent services.
- Since a microservices architecture consists of units that run independently, each service can be developed, updated, deployed, and scaled without affecting the other services.

Monolithic vs Microservices



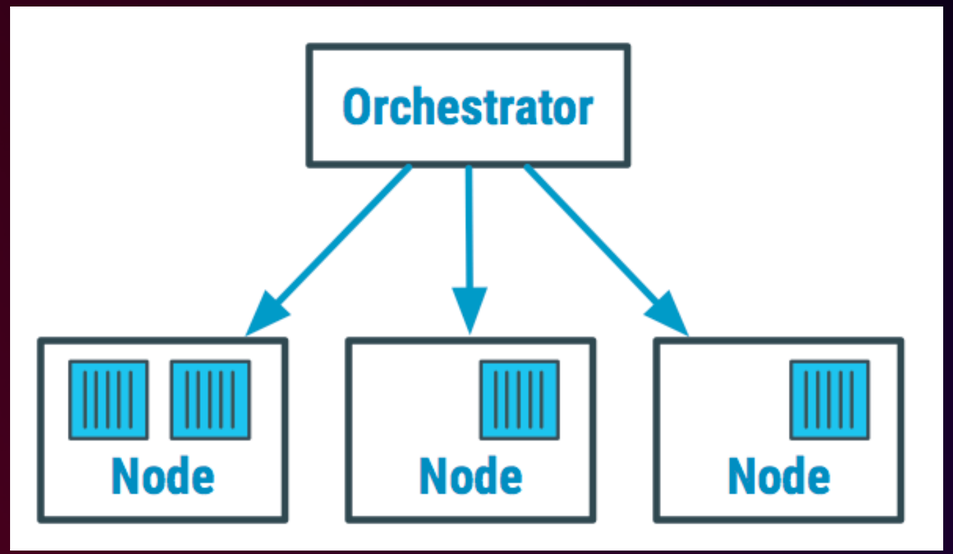
Monolithic vs Microservices



- Microservices can be achieved through containers by packaging each microservice and its dependencies into separate containers.
- Each container represents a self-contained unit that provides the necessary isolation and portability to deploy and manage microservices effectively.

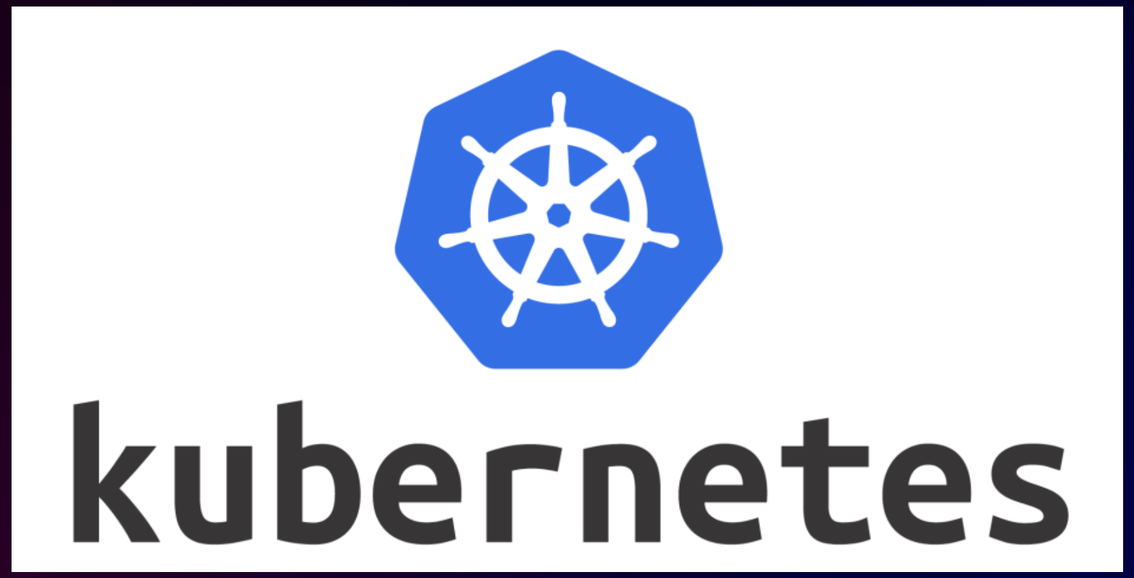
Container Orchestration

Container Orchestration



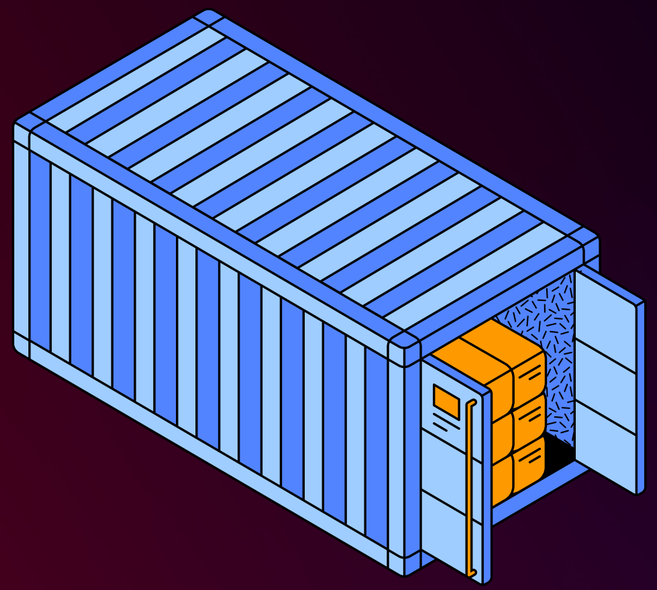
- Container orchestration refers to the management and coordination of containerized applications across a cluster of machines or a distributed infrastructure.
- Container orchestration automates the deployment, management, scaling, and networking of containers.

Top Orchestration Tool



Amazon
Elastic Container Service (ECS)

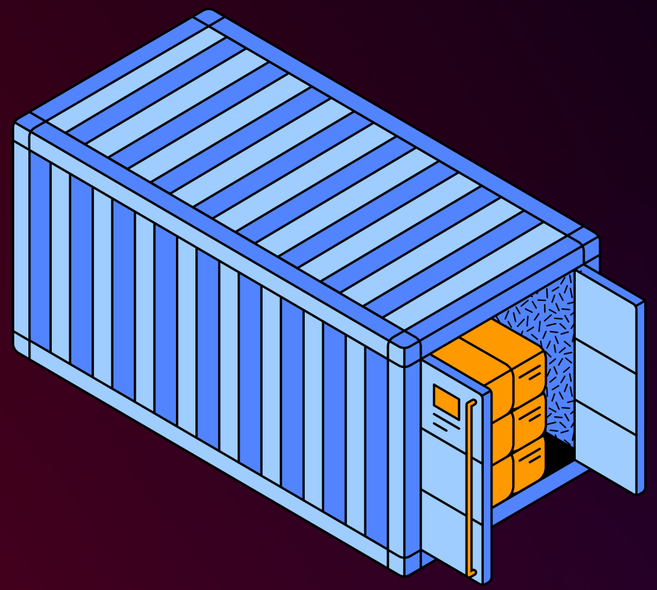
Elastic Container Service (ECS)



Amazon Elastic Container Service (ECS), also known as Amazon EC-2 Container Service, is a managed service that allows users to run Docker-based applications packaged as containers across a cluster of EC2 instances.

Amazon
Elastic Kubernetes Service (EKS)

Elastic Kubernetes Service (EKS)



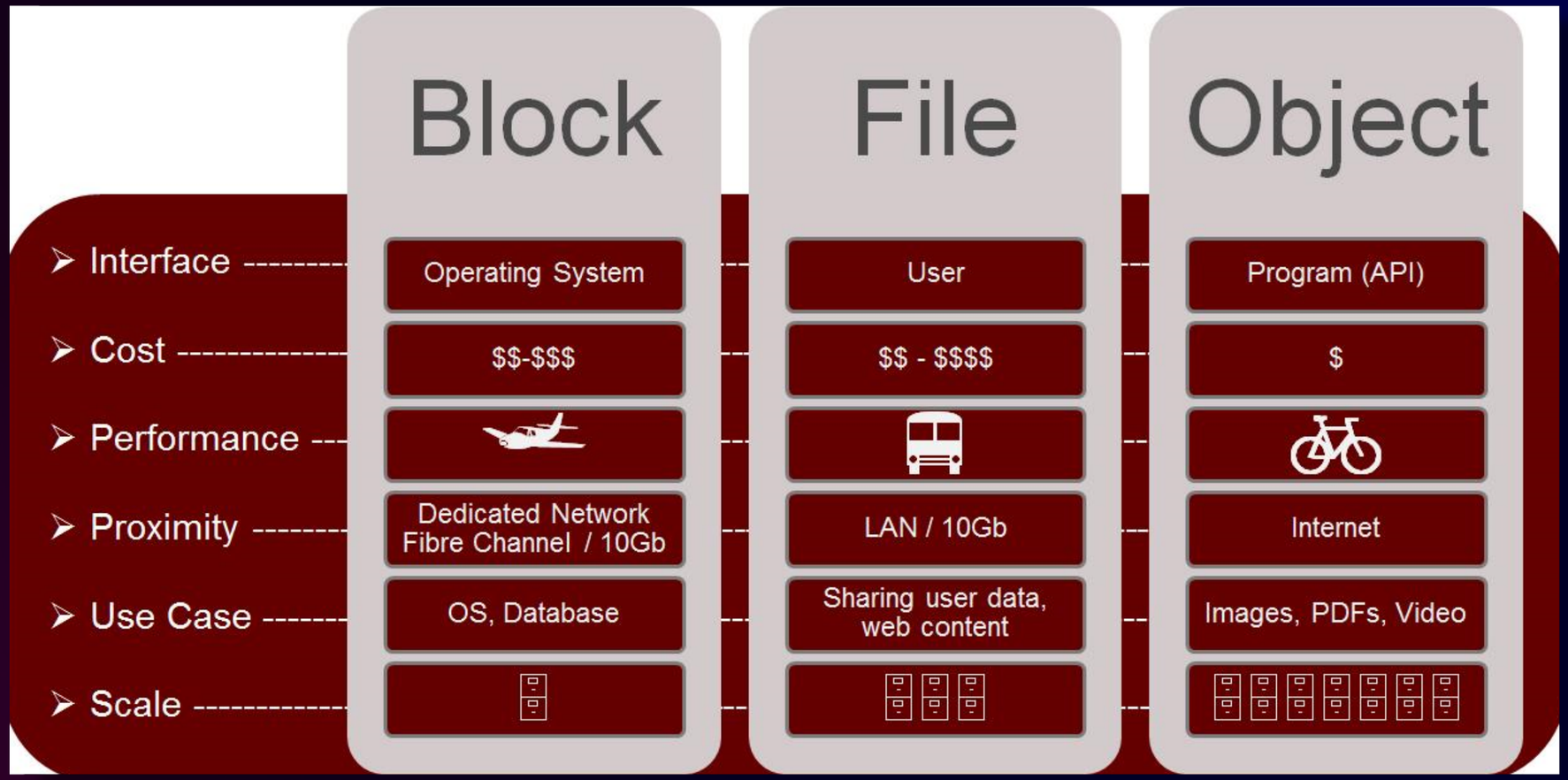
Amazon EKS automatically manages the availability and scalability of the Kubernetes control plane nodes responsible for scheduling containers, managing application availability, storing cluster data, and other key tasks.

Recap ⏮

- Containers
- Microservices
- Container Orchestration

Data Storage in AWS

Types of Storage



Block storage in AWS

File storage in AWS

Object storage in AWS

Data Storage Terminology

- Availability
- Durability
- Consistency
- Scaling

Availability

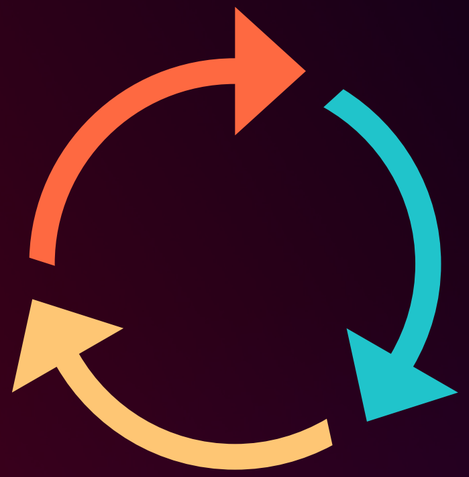
Availability	Maximum Unavailability (per year)	Application Categories
99%	3 days 15 hours	Batch processing, data extraction, transfer, and load jobs
99.9%	8 hours 45 minutes	Internal tools like knowledge management, project tracking
99.95%	4 hours 22 minutes	Online commerce, point of sale
99.99%	52 minutes	Video delivery, broadcast **workloads**
99.999%	5 minutes	ATM transactions, telecommunications **workloads**

Durability

99.999999999% (11 nines) durability

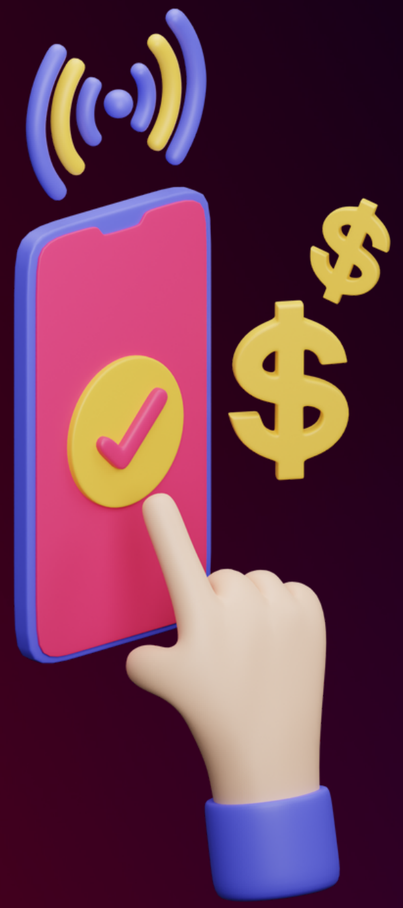
If you store one million objects for ten million years, you would expect to lose one file in that time.

Consistency



- Consistency refers to the state of data being coherent and synchronized across multiple distributed systems or replicas.
- When data is stored and accessed in a distributed environment, maintaining consistency becomes a critical factor.

Strong Consistency



- Strong Consistency simply means the data must be strongly consistent at all times.
- All the server nodes across the world should contain the same value as an entity at any point in time.
- Example : Transactional Applications

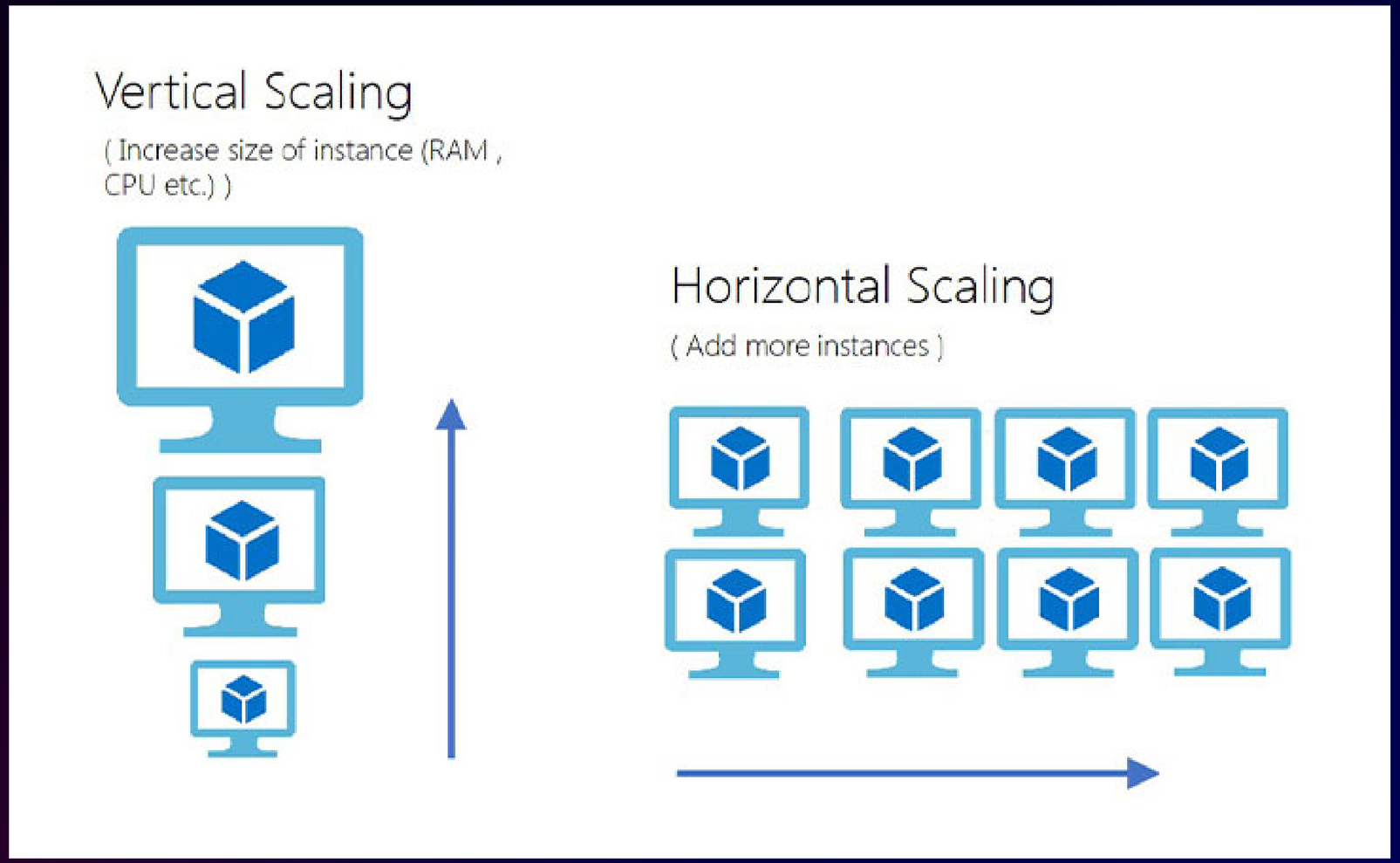
Eventual Consistency

- In Eventual consistency, different copies of the data may not be immediately in sync.
- There can be temporary differences between replicas, a little lag, but over time, they will all become consistent.
- Example : Social Media

Atomicity

- Atomicity is a property of a transaction in a database or a system that ensures that all the operations within the transaction are treated as a single, indivisible unit.
- It guarantees that either all the operations in the transaction are successfully completed and applied to the database, or none of them are applied at all.
- In other words, an atomic transaction is all or nothing.

Scaling


Vertical Scaling
(Increase size of instance (RAM , CPU etc.))
Horizontal Scaling
(Add more instances)

Types of Data formats

- Structured data - Table(rows & columns) - Relational SQL databases
- Semi-Structured data - Key, Values - NoSQL databases
- Unstructured data - Photos, videos - Object,block,file strorage

Relational Database in AWS

Amazon RDS

- Amazon Relational Database Service (RDS) is a managed SQL database service provided by Amazon Web Services (AWS).
- It also helps with relational database management tasks, such as data migration, backup, recovery and patching

SCALABILITY

Amazon RDS allows you to easily adjust compute and storage resources to handle changing workloads efficiently.

ATOMICITY

RDS ensures that transactions are treated as indivisible units, either fully committed or completely rolled back in case of errors or failures.

DURABILITY

RDS automatically replicates data across multiple availability zones to ensure data is protected against failures and disruptions.

AVAILABILITY

RDS provides high availability through features like Multi-AZ deployments, which replicate data synchronously to standby instances, ensuring resilience and minimizing downtime.

NoSQL Database in AWS

Amazon DynamoDB

- Amazon DynamoDB is a fully managed, serverless, key-value NoSQL database.
- DynamoDB is primarily a key-value store in the sense that its data model consists of key-value pairs in a schemaless, very large, non-relational table of rows (records).

SCALABILITY

DynamoDB offers seamless and automatic scalability, allowing you to handle any amount of traffic

ATOMICITY

DynamoDB guarantees atomicity for individual item writes. Each write operation is treated as an atomic unit, ensuring that it is either fully completed or not applied at all.

DURABILITY

DynamoDB provides built-in durability by replicating data across multiple Availability Zones within a region, ensuring high availability and data protection.

AVAILABILITY

DynamoDB is designed to provide high availability with automatic multi-AZ replication and failover capabilities, ensuring that your applications can access the database without disruptions.

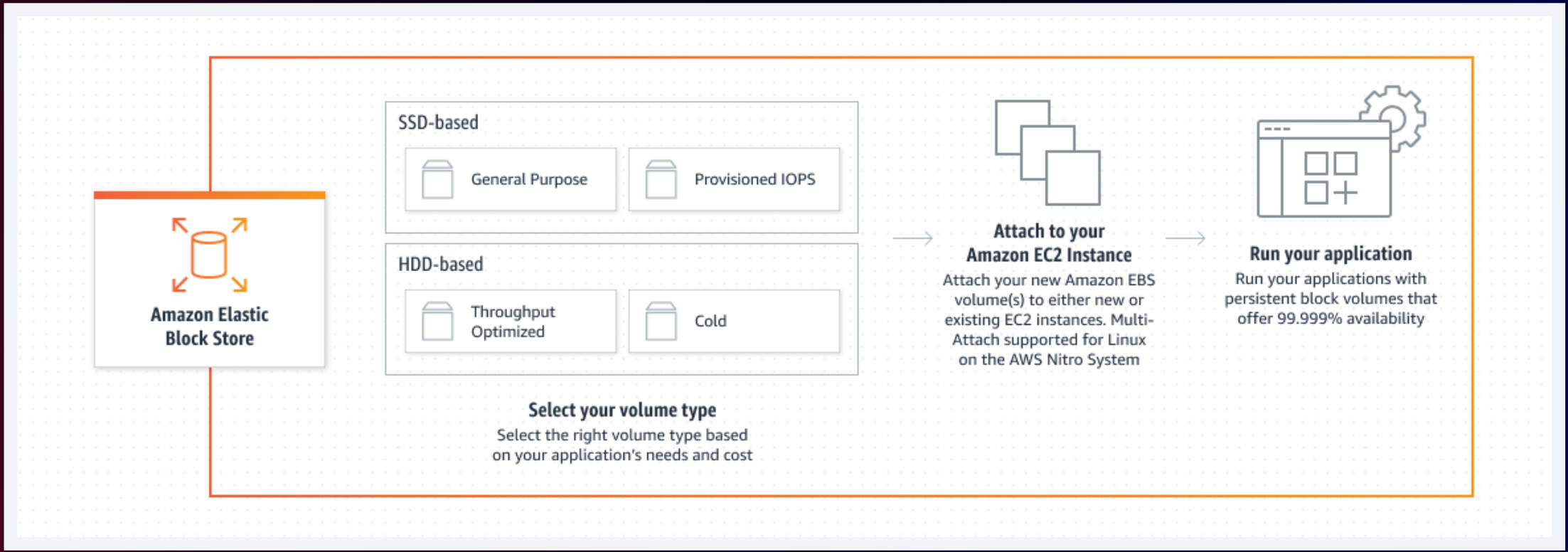
Unstructured data storage in aws

Block storage

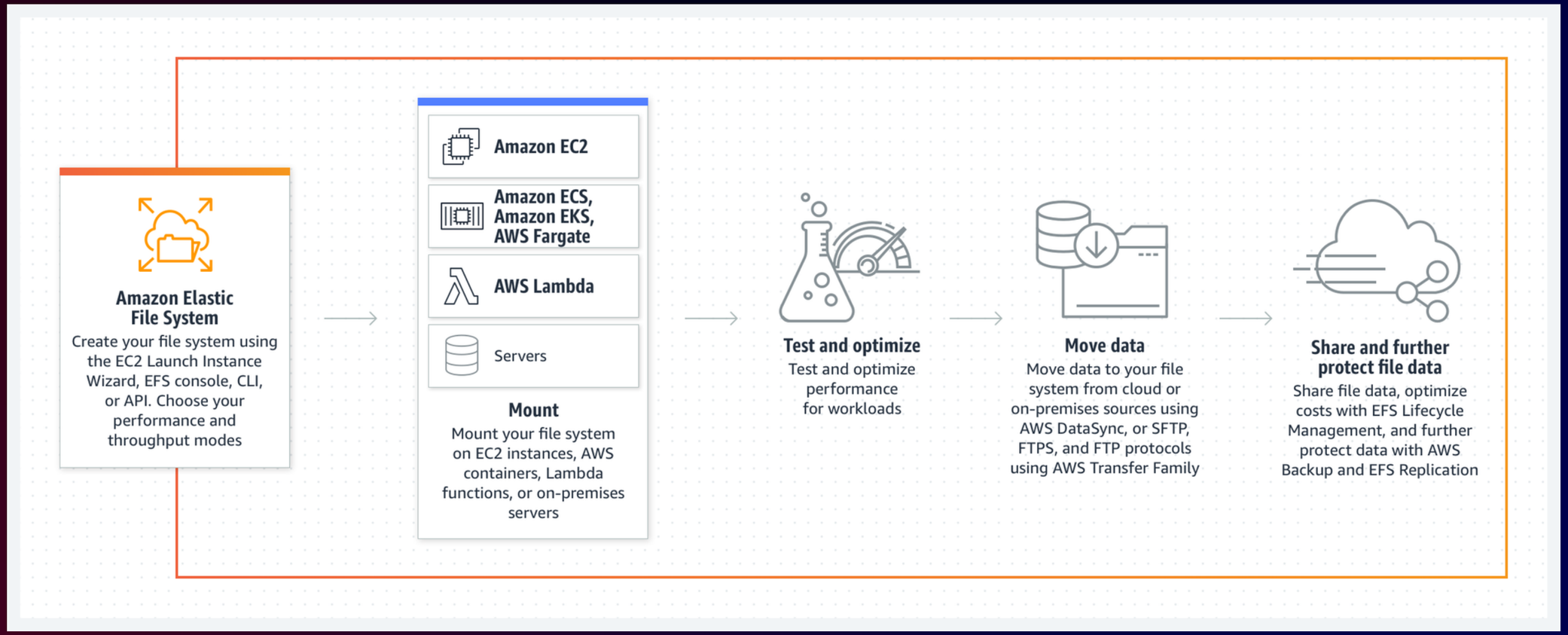
File storage

Object storage

Amazon EBS


Amazon Elastic Block Store
SSD-based
General Purpose
Provisioned IOPS
HDD-based
Throughput Optimized
Cold
Select your volume type
Select the right volume type based on your application's needs and cost
Attach to your Amazon EC2 Instance
Attach your new Amazon EBS volume(s) to either new or existing EC2 instances. Multi-Attach supported for Linux on the AWS Nitro System
Run your application
Run your applications with persistent block volumes that offer 99.999% availability

Amazon EFS


Amazon Elastic File System
Create your file system using the EC2 Launch Instance Wizard, EFS console, CLI, or API. Choose your performance and throughput modes
Amazon EC2
Amazon ECS, Amazon EKS, AWS Fargate
AWS Lambda
Servers
Mount
Mount your file system on EC2 instances, AWS containers, Lambda functions, or on-premises servers
Test and optimize
Test and optimize performance for workloads
Move data
Move data to your file system from cloud or on-premises sources using AWS DataSync, or SFTP, FTPS, and FTP protocols using AWS Transfer Family
Share and further protect file data
Share file data, optimize costs with EFS Lifecycle Management, and further protect data with AWS Backup and EFS Replication

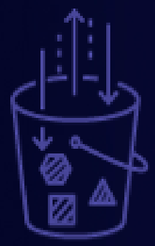
Amazon S3


Analytics data
Log files
Application data
Videos and pictures
Backup and archival
Move data
Move your data to Amazon S3 from wherever it lives – in the cloud, in applications, or on-premises
Amazon S3
Object storage built to store and retrieve any amount of data from anywhere
Store data
Create bucket, specify the Region, access controls, and management options. Upload any amount of data
Control access to data
Optimize cost with storage classes
Replicate data to any Region
Access from on-premises or VPC
Protect and secure your data
Gain visibility into your storage
Artificial Intelligence (AI)
Advanced analytics
Machine Learning (ML)
Analyze data
Use AWS and 3rd party services to analyze your data to gain insights

Amazon S3

- Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.
- Store and protect any amount of data for a range of use cases, such as data lakes, websites, cloud-native applications, backups, archive, machine learning, and analytics.
- Amazon S3 is designed for 99.999999999% (11 9's) of durability, and stores data for millions of customers all around the world.

Amazon S3 Types



S3 Standard

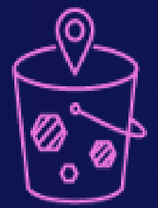
- Active, frequently access data
- Milliseconds access
- ≥ 3 AZ
- $0.023/GB

S3 Intelligent-Tiering

- Data with changing access patterns
- Milliseconds access
- ≥ 3 AZ
- $0.025 to $0.0125/GB
- Monitoring fee per object
- Min storage duration

S3 Standard-IA

- Infrequently accessed data
- Milliseconds access
- ≥ 3 AZ
- $0.0125/GB
- Retrieval fee per GB
- Min storage duration
- Min object size



S3 One Zone-IA

- Re-creatable, less accessed data
- Milliseconds access
- 1 AZ
- $0.01/GB
- Retrieval fee per GB
- Min storage duration
- Min object size

S3 Glacier Instant Retrieval

- Archival data with fast access
- Milliseconds access
- ≥ 3 AZ
- $0.004/GB
- Retrieval fee per GB
- Min storage duration
- Min object size

S3 Glacier Flexible Retrieval

- Archival data
- Minutes or hours
- ≥ 3 AZ
- $0.0036/GB
- Retrieval fee per GB
- Min storage duration

S3 Glacier Deep Archive

- Long-term archival data
- Select hours
- ≥ 3 AZ
- $0.000099
- Retrieval fee per GB
- Min storage duration

Frequent ← *Access Frequency* → *Infrequent*

Thank you