AWS Foundation

Introduction to EC2, EBS & EFS

Agenda

1. Introduction to EC2
2. Regions and Availability Zones
3. Pre-EC2
4. EC2 Instance Types
5. Demo 1: Launching an Instance
6. What is an AMI?
7. Demo 2: Creating and Copying an AMI
8. Introduction to EBS
9. EBS Snapshots
10. Demo 3: EBS
11. Introduction to EFS
12. Demo 4: EFS
13. Instance Tenancy and Reserved and Spot Instances
14. Pricing and Design Patterns

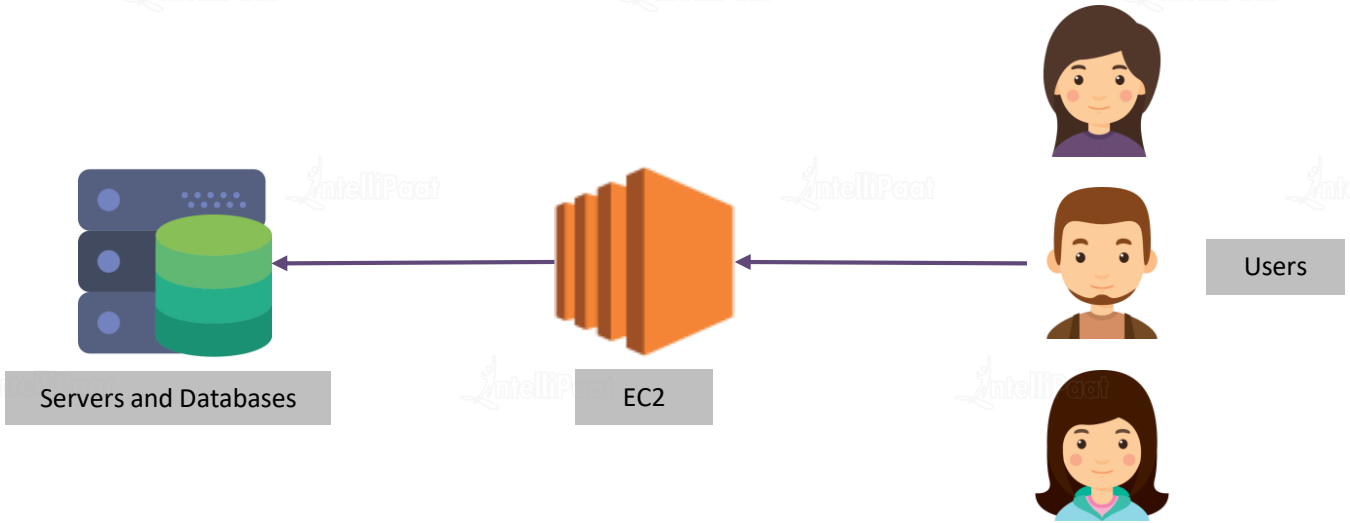
Introduction to EC2

Introduction to EC2

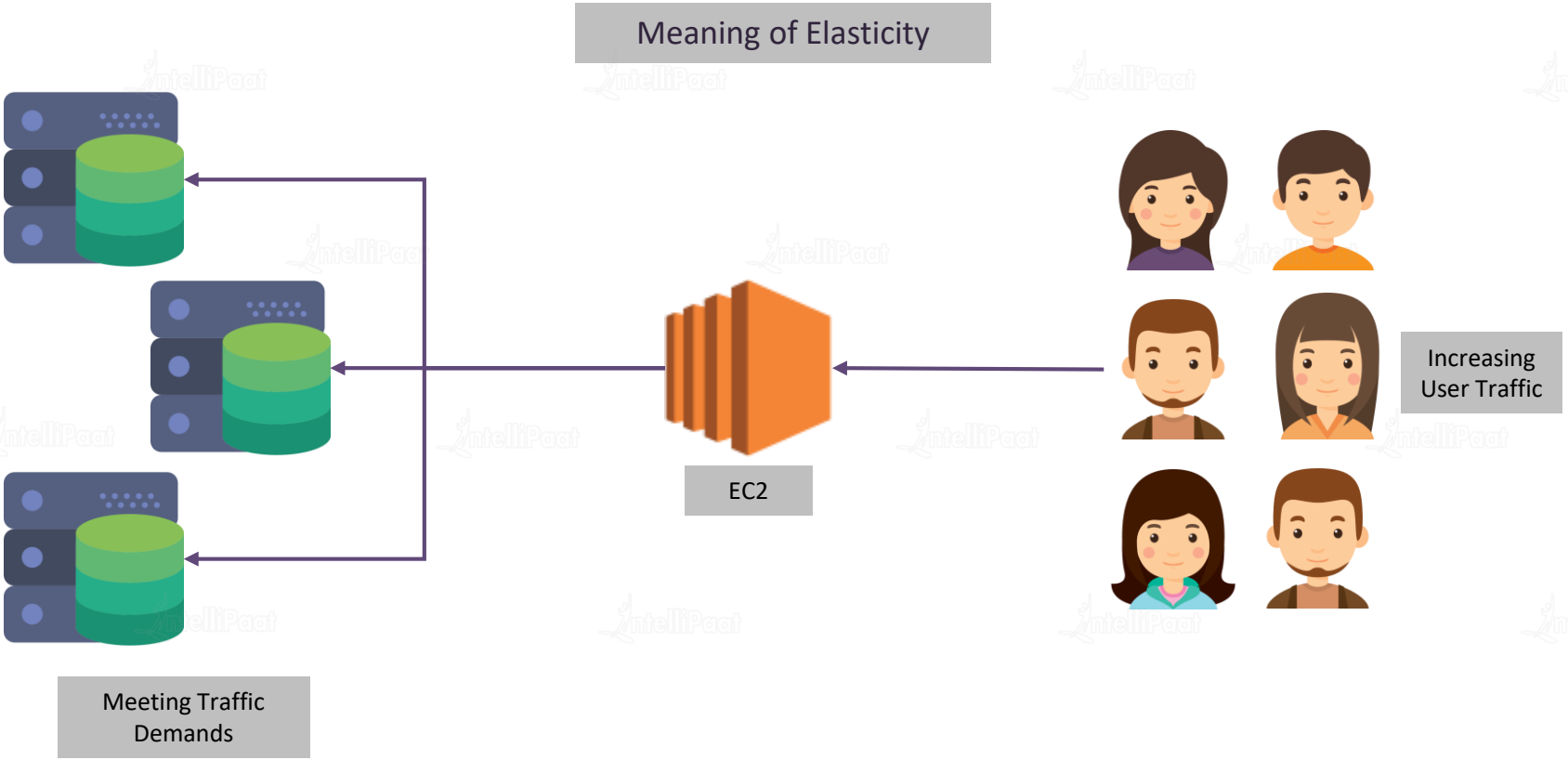
Elastic Compute Cloud

Elastic: It is the level at which a system is able to adapt to workload changes by provisioning and de-provisioning resources such that the resources meet the current demand as closely as possible

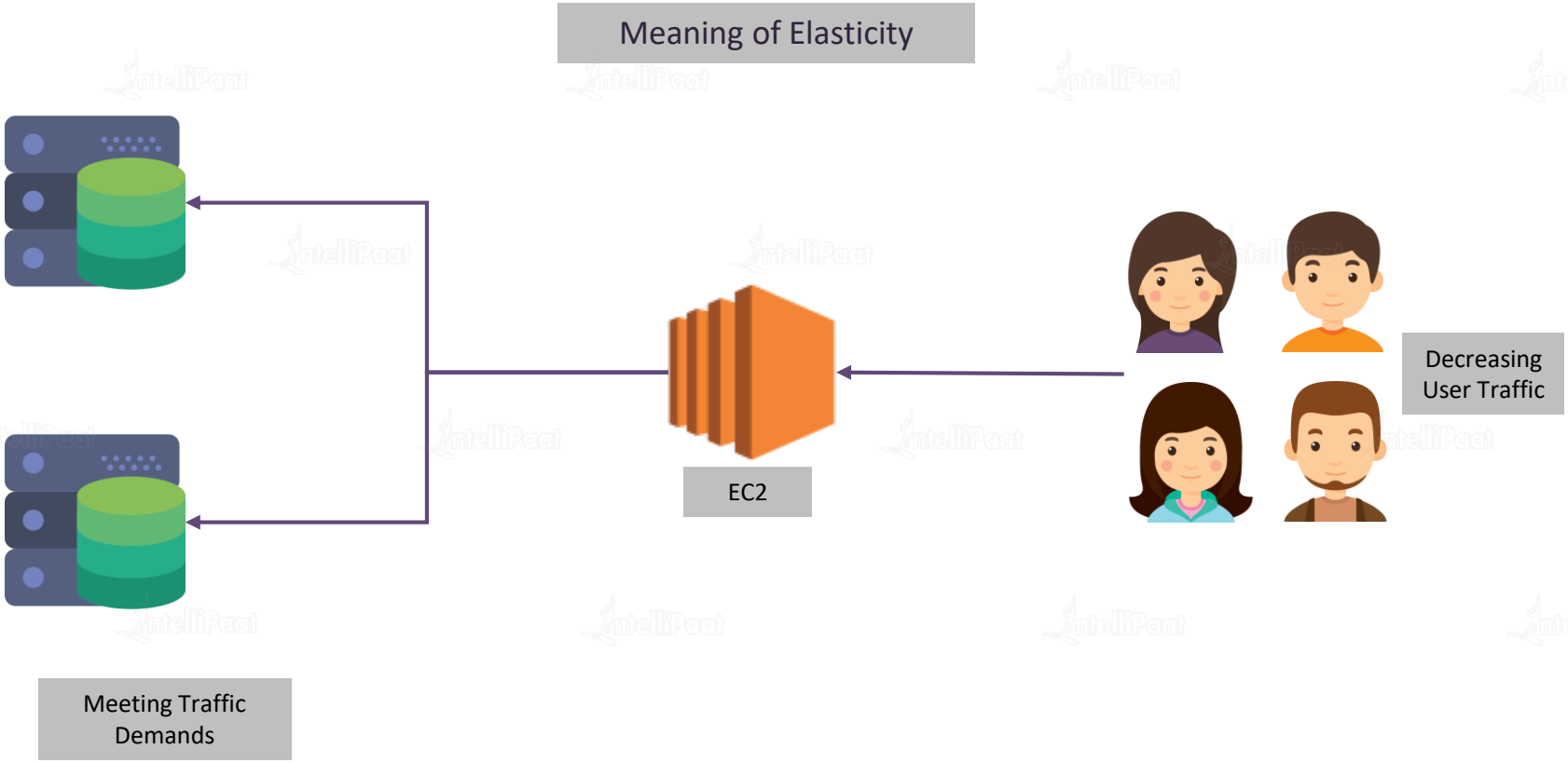
Meaning of Elasticity



Introduction to EC2

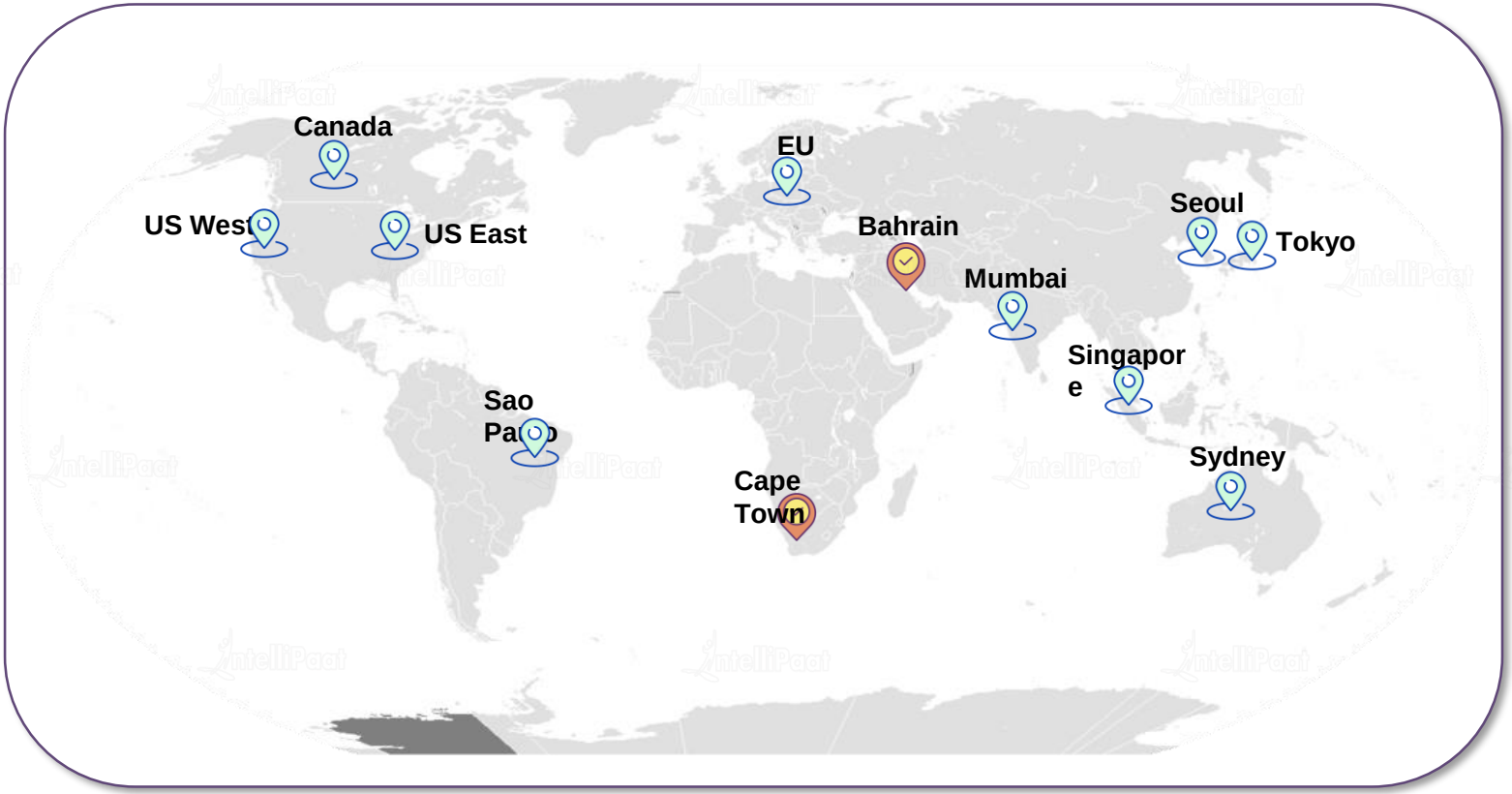


Introduction to EC2



Regions and Availability Zones

EC2: Regions and Availability Zones



EC2: Regions and Availability Zones

Regions are geographical locations where AWS data centers reside. Following are AWS region names and their subdivisions:

- US East: N. Virginia (us-east-1), Ohio (us-east-2)
- US West: N. California (us-west-1), Oregon (us-west-2)
- Asia Pacific: Mumbai (ap-south-1), Seoul (ap-northeast-2), Singapore (ap-southeast-1)
- EU: Frankfurt (eu-central-1), Ireland (eu-west-1), London (eu-west-2), Paris (eu-west-3)

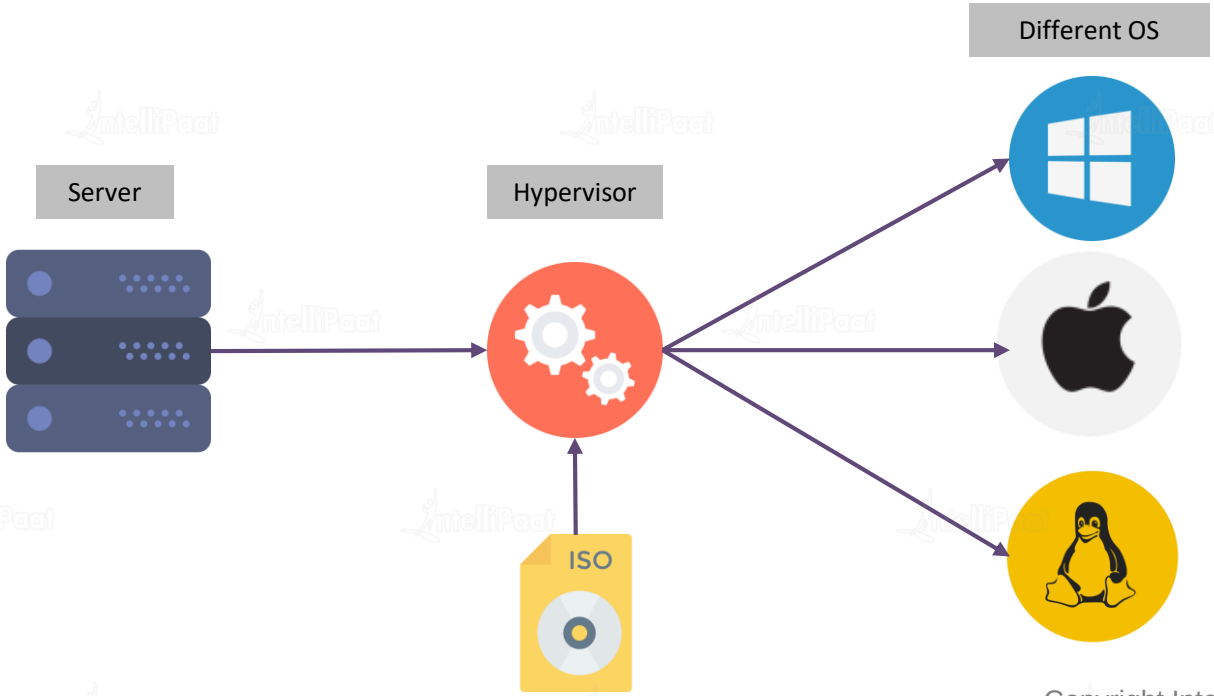
For instance, 'us-east-1' contains 6 data centers or availability zones:

- us-east-1a
- us-east-1b
- us-east-1c
- us-east-1d
- us-east-1e
- us-east-1f

Pre-EC2

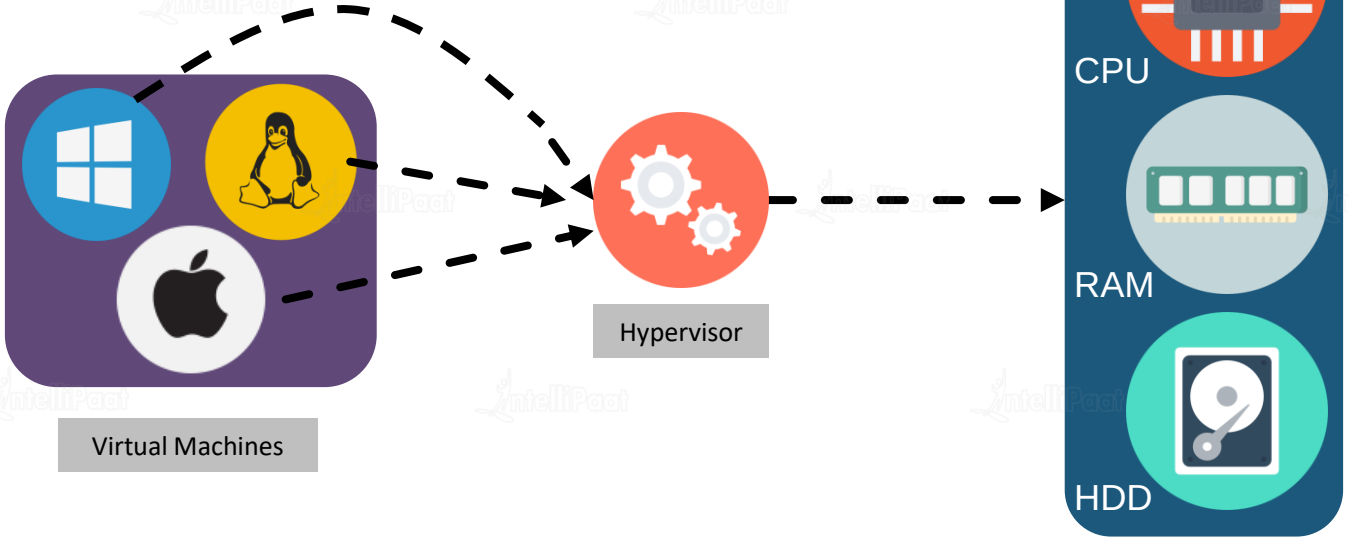
Pre-EC2

A virtual machine is an emulation of a computer system, having an OS, RAM, and CPU or compute capacity



Pre-EC2

- In simple terms, it is running a virtual operating system inside an operating system
- Suppose, we want to run Ubuntu in our Windows OS, we could easily install and use it as a virtual OS



Pre-EC2

Intel Processor Generation

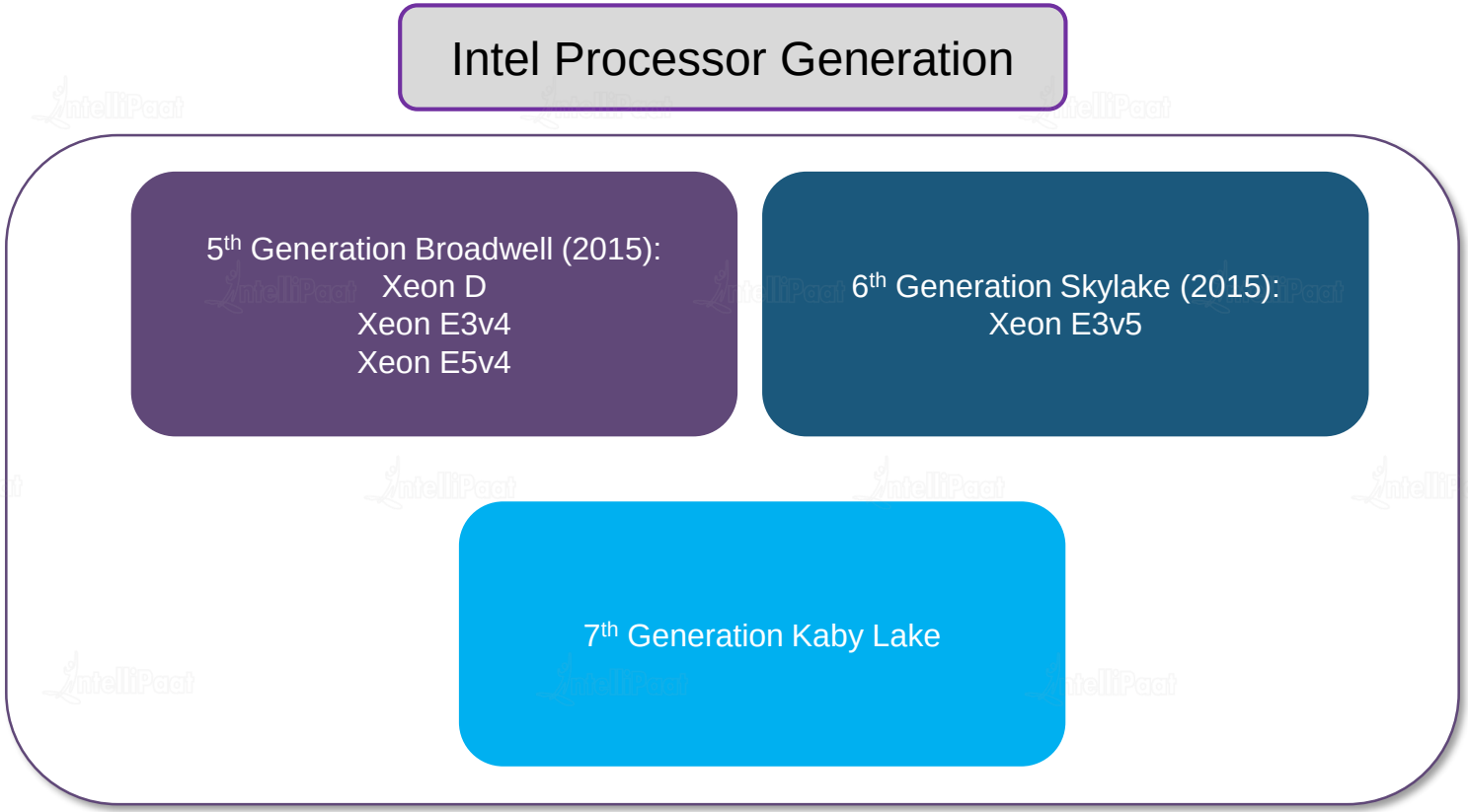
1st Generation Nehalem (2006):
Introduced hyper-threading

2nd Generation Sandy Bridge (2011):
Pentium
Xeon E3
Xeon E5

3rd Generation Ivy Bridge (2012):
Pentium
Xeon E3v2
Xeon E5v2
Xeon E7v2

4th Generation Haswell (2013):
Xeon E3v3
Xeon E5v3
Xeon E7v3

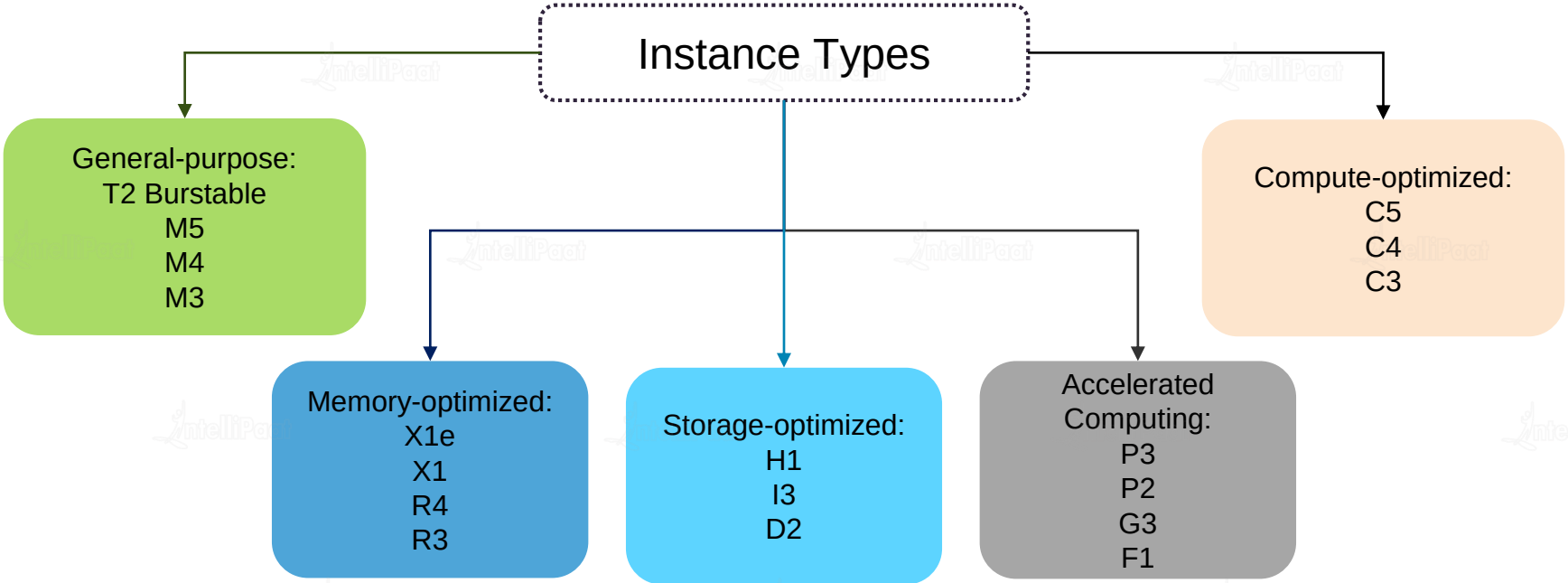
Pre-EC2



EC2 Instance Types

EC2 Instance Types

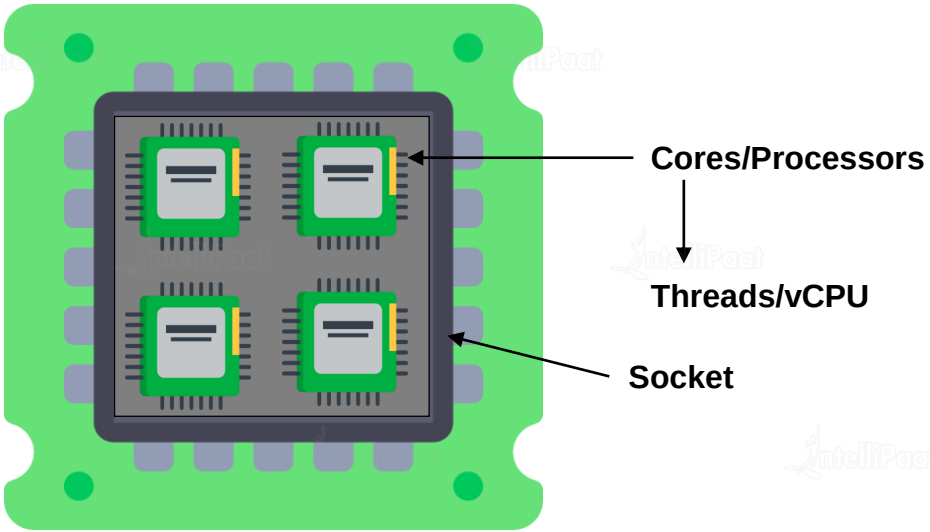
The instance type determines the hardware of the underlying host computer on which EC2 instances are launched



The vCPU and the Root Device Volume

Each vCPU is a hyper-thread of an Intel Xeon core except for t2 and m3.medium instances (AWS Definition)

The Root Device Volume contains the image using which the instance is booted



Demo 1: Launching an Instance

Demo 1: Launching an Instance

Steps for Launching an EC2 Instance

1. Open AWS Management Console; click on **Services** drop-down, and choose **EC2**
2. Click on **Launch Instance**, and choose an AMI (i.e., here, Ubuntu 18.04)
3. Choose **Instance type** (Free tier eligible), and click on **Next**
4. Configure instances, and add storage and unique tags
5. Configure the network group (choose Create new group), and then review once and launch
6. Next, choose **Create a new key pair**; give a name, and download
7. Click on **Launch instances** and wait until it initializes

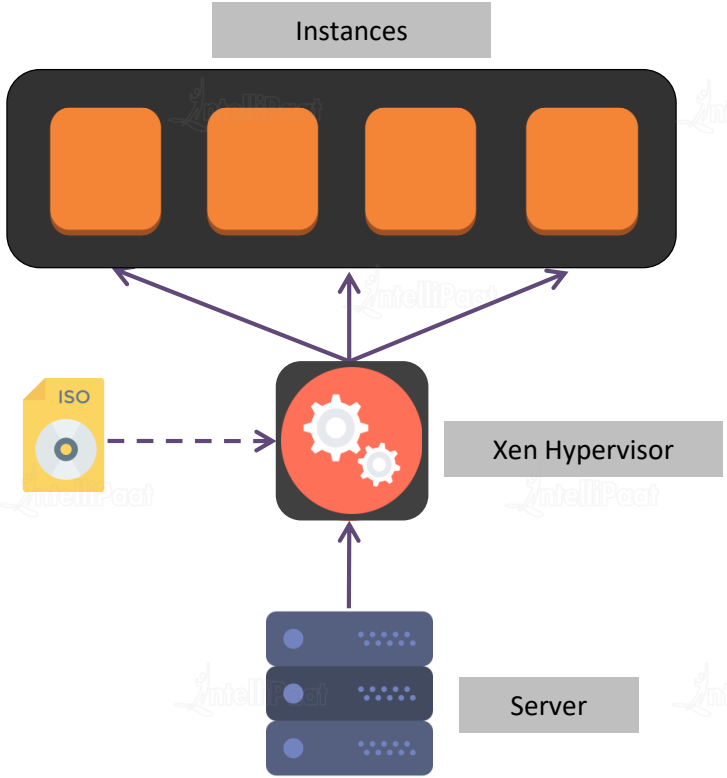
Now, we have successfully created a EC2 instance!

What is an AMI?

What is an AMI?

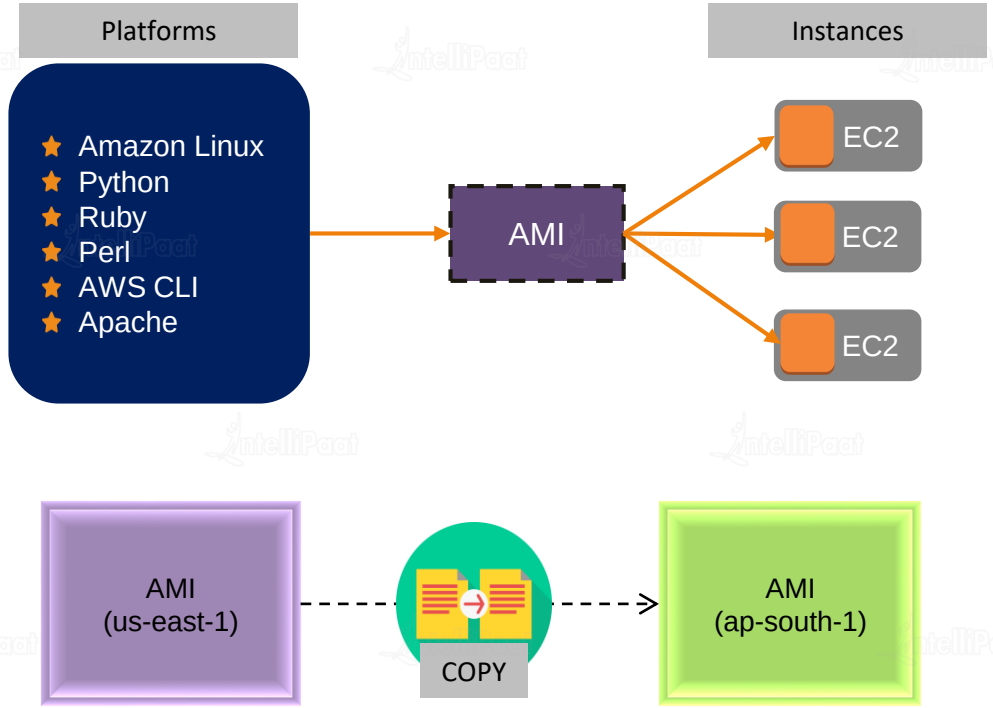
Amazon Machine Image (AMI) contains the information required to launch an instance

- Operating system
- Architecture
- Storage for the root device (Instance store or EBS-backed)
- Virtualization type (HVM or PV)



Creating and Copying an AMI

- Create an AMI from an instance
- Launch multiple instances from it
- Copy the AMI
- AMI permissions



Demo 2: Creating and Copying an AMI

Demo 2: Creating and Copying an AMI

Creating an AMI

1. Select the instance we created in the last demo
2. Click on the **Actions** button, and choose: Image → Create image
3. Provide a name and a small description for the image, and then click on **Create AMI**
4. Now, click on **AMIs** under the Images group in the left-side scroll bar

We will see that the AMI has been created!

Copying an AMI to Another Region

1. Select the created AMI, and click on the Actions button (Actions → Copy AMI)
2. Choose the destination region, and click on **Copy AMI**
3. Go back to the AMIs view, and wait until it is available

We have now successfully created and copied an AMI!

Public IP vs Elastic IP

Public IP

- It is not associated with an AWS account
- No charges for the public IP, even if it is not being used while the instance is running
- Whenever the instance is re-launched, the public IP changes

Elastic IP

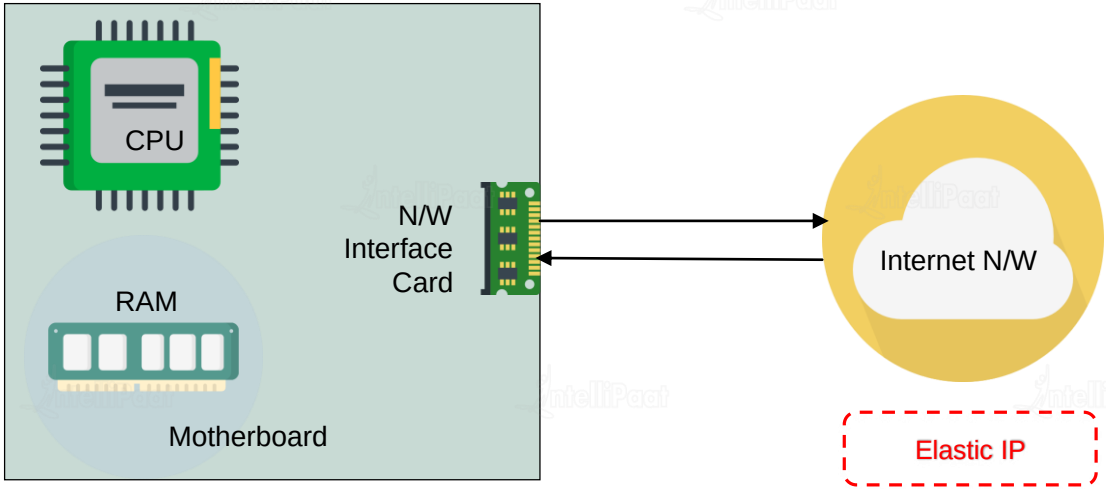
- It is associated with the AWS account
- Charges will be applied if the same is done with the elastic IP
- The elastic IP is the same and static for every launch until we manually release it

Elastic Network Interface

A network interface is the interface between a computer and an Internet network. The network IO happens through n/w interface cards

N/W interfaces contain:

- Elastic IP
- Public IP
- Private IP
- Security Groups



Introduction to EBS

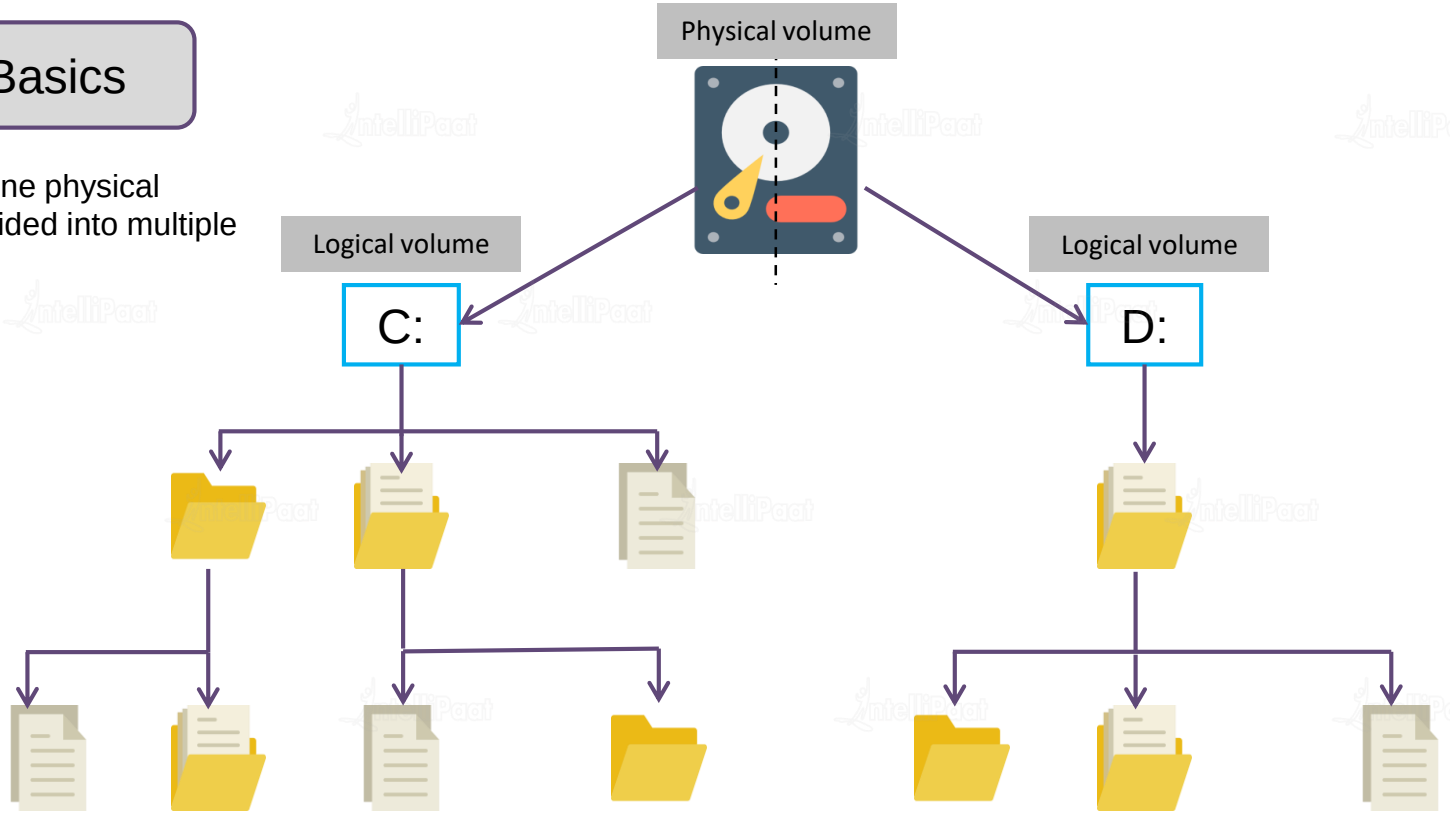
Introduction to EBS

File System Basics

- In simple terms, one physical volume will be divided into multiple logical volumes

Directory

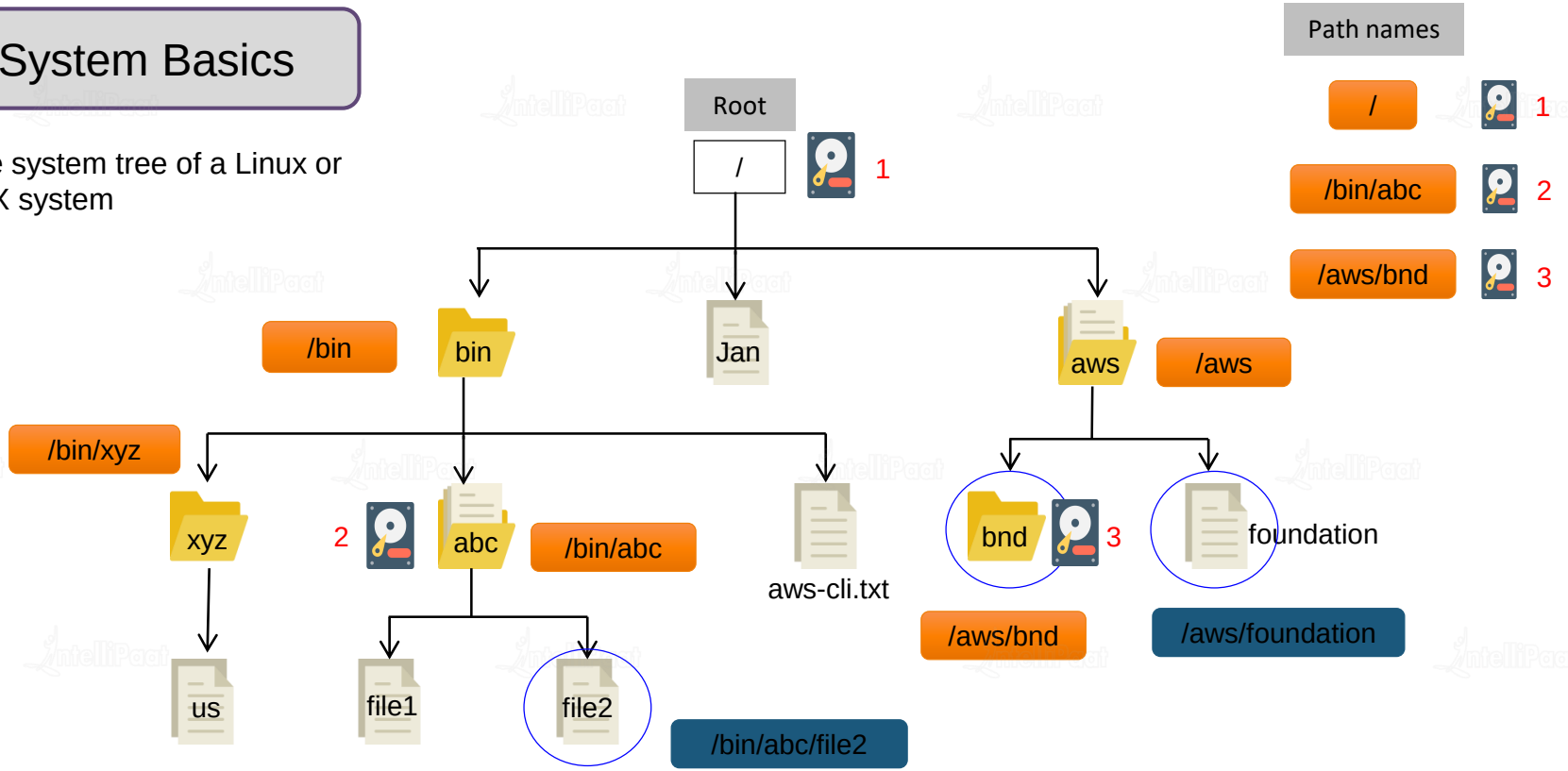
File



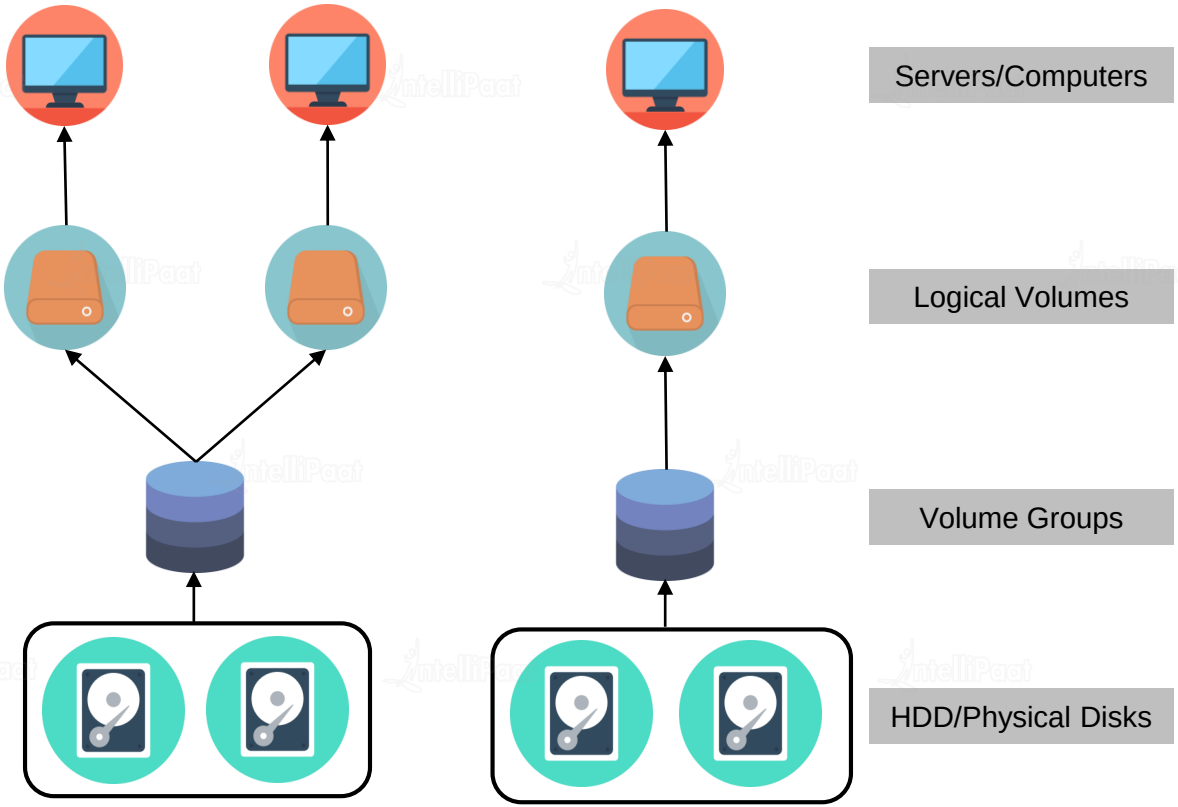
Introduction to EBS

File System Basics

- A file system tree of a Linux or UNIX system

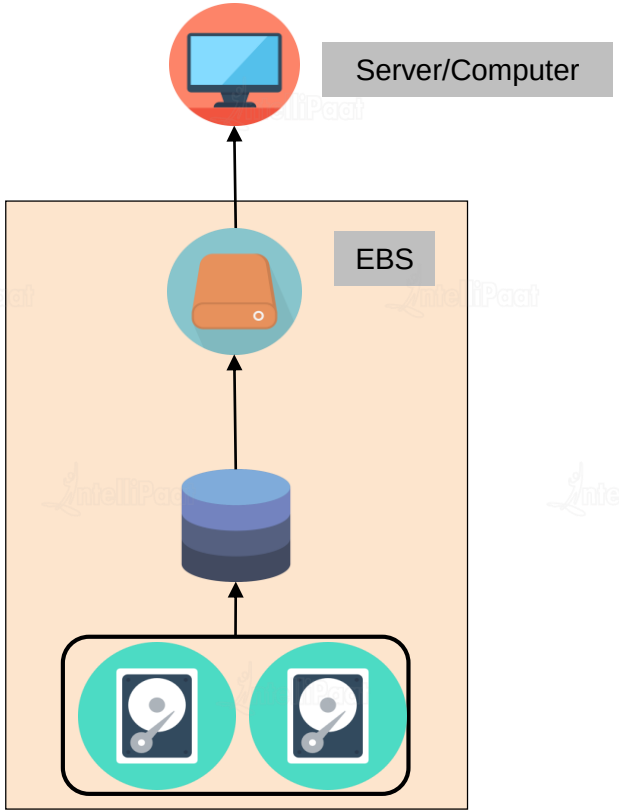


Pre-EBS Storage Layers



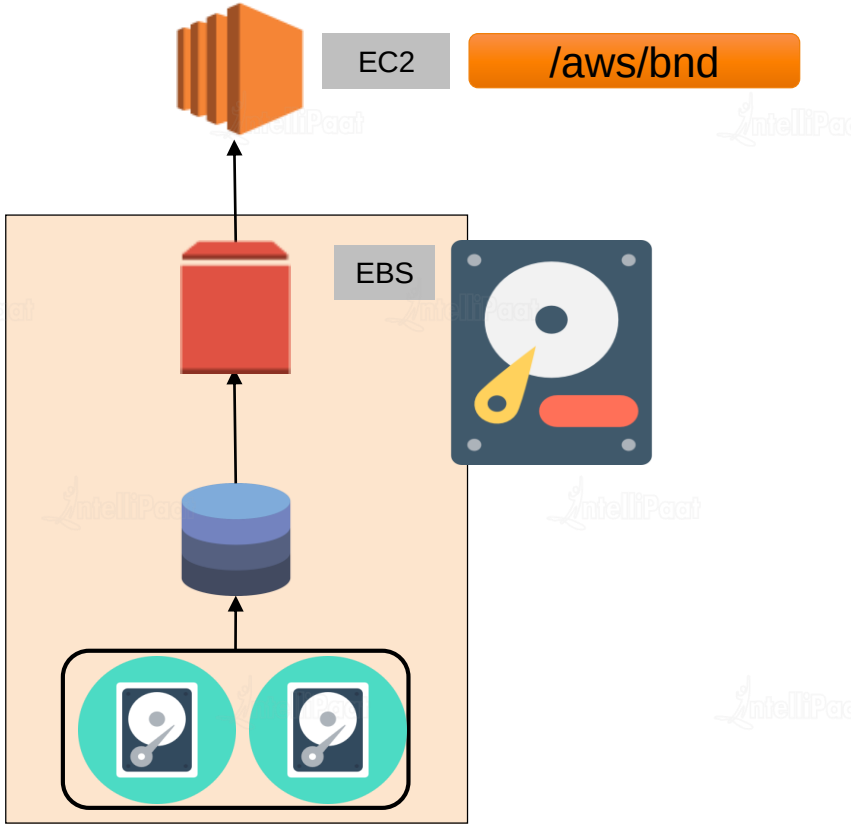
Elastic Block Store

- In an EBS block-level storage, the server-base operating system connects with the raw volumes that are created through a fiber channel
- Then, they are used as individual disks, and if it is very versatile, it can be used as file storage, database storage, and virtual machine volumes



Elastic Block Store

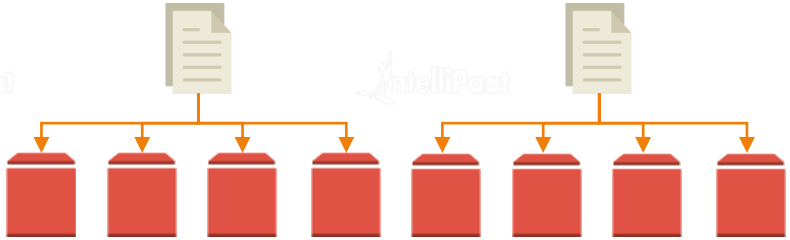
- An EC2 instance is directly connected to EBS
- While the instance is running, a volatile memory called ephemeral storage will be attached to the instance
- If the instance is stopped, the ephemeral memory will be detached



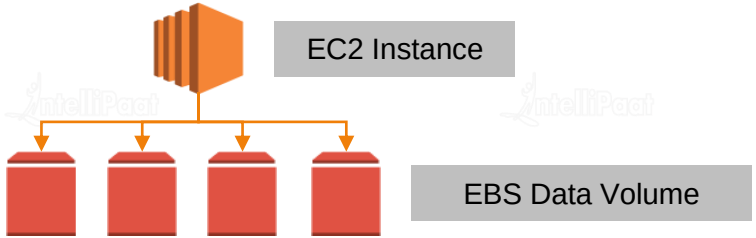
EBS Concepts

EBS Concepts

- It is the raw unformatted block-level storage; it is exposed as raw device to the EC2 instance
- EBS volumes persist independently from the life of the EC2 instance
- An EBS volume is automatically replicated within an availability zone
- **Throughput**: It is the sequential transfer rate that an SSD or HDD will maintain continuously



- **IOPS**: It is the measure of the number of I/O operations a drive, SSD, or HDD can handle per second with each block being read from or written to a RANDOM location in the disk



EBS Concepts

Volume Types

GP2: General-purpose SSD

- Baseline performance is 3 IOPS/GB with a min. of 100 IOPS and a max. of 10000 IOPS
- Max. burst performance is 3000 IOPS
- Max. throughput per volume is 160 MB/s (16 KB IO size)

IO1: Provisioned SSD

- From 100 to 32000 IOPS can be provisioned
- Max. throughput per volume is 500 MB/s

ST1: Throughput-optimized HDD

- Baseline performance is 40 MB/s per TB with a max. of 500 MB/s per volume
- Burst performance is 250 MB/s per TB with a max. of 500 MB/s per volume

SC1: Cold Storage HDD

- Baseline performance is 12 MB/s per TB with a max. of 192 MB/s per volume
- Burst performance is 80 MB/s per TB with a max. of 250 MB/s per volume

EBS Concepts

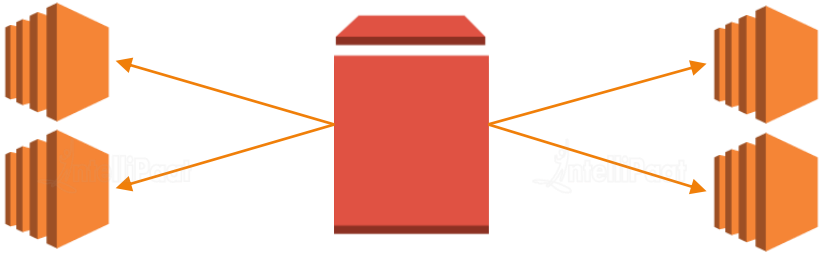
Volume Types

Volume Type	Size Limit	Maximum IOPS	Maximum Throughput	Maximum Burst
GP2	1 GB – 16 TB	10000	160 MB/s	3000
IO1	4 GB – 16 TB	32000	500 MB/s	NA
ST1	500 GB – 16 TB	500	500 MB/s	500 MB/s
SC1	500 GB – 16 TB	250	192 MB/s	250 MB/s

EBS Concepts

New Feature: EBS Multi-Attach

Amazon EBS Multi-Attach is now available on Provisioned IOPS io1 volumes

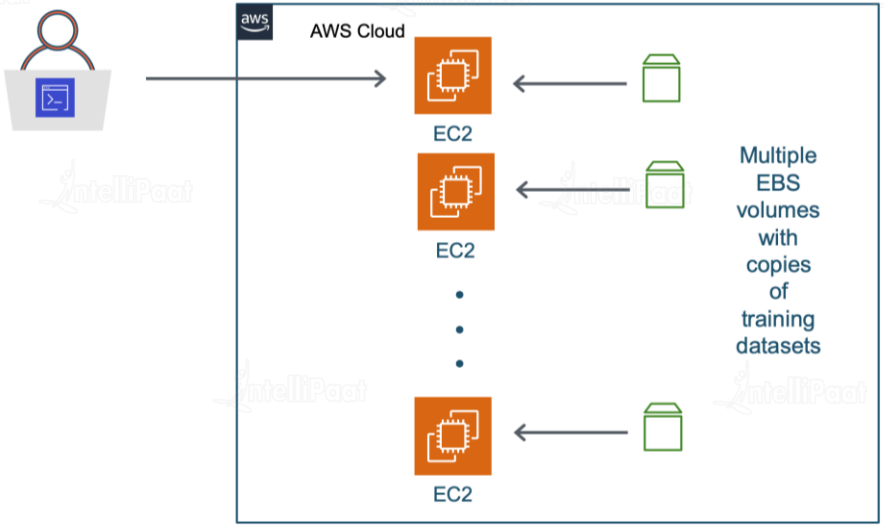


We can now enable Multi-Attach on Amazon EBS Provisioned IOPS io1 volumes to allow a single volume to be concurrently attached to up to 16 AWS Nitro System-based Amazon EC2 instances within the same availability zone

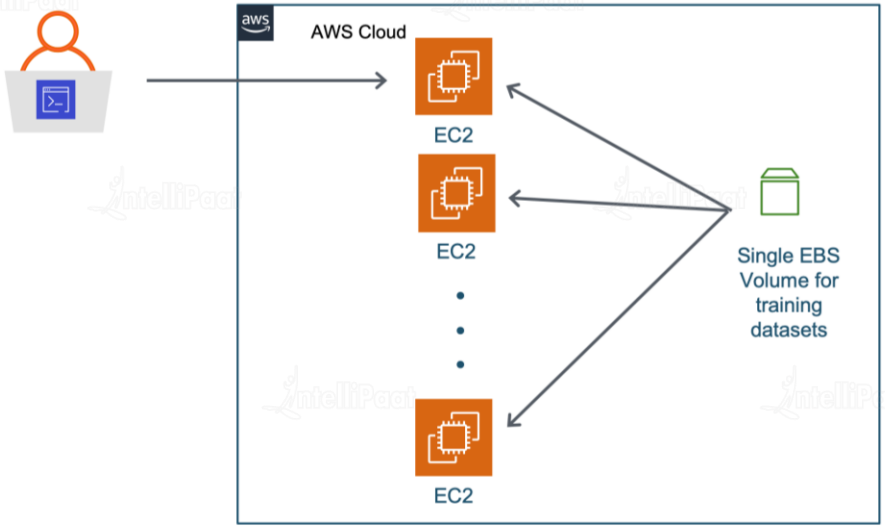
EBS Concepts

New Feature: EBS Multi-Attach

Without Multi-Attach



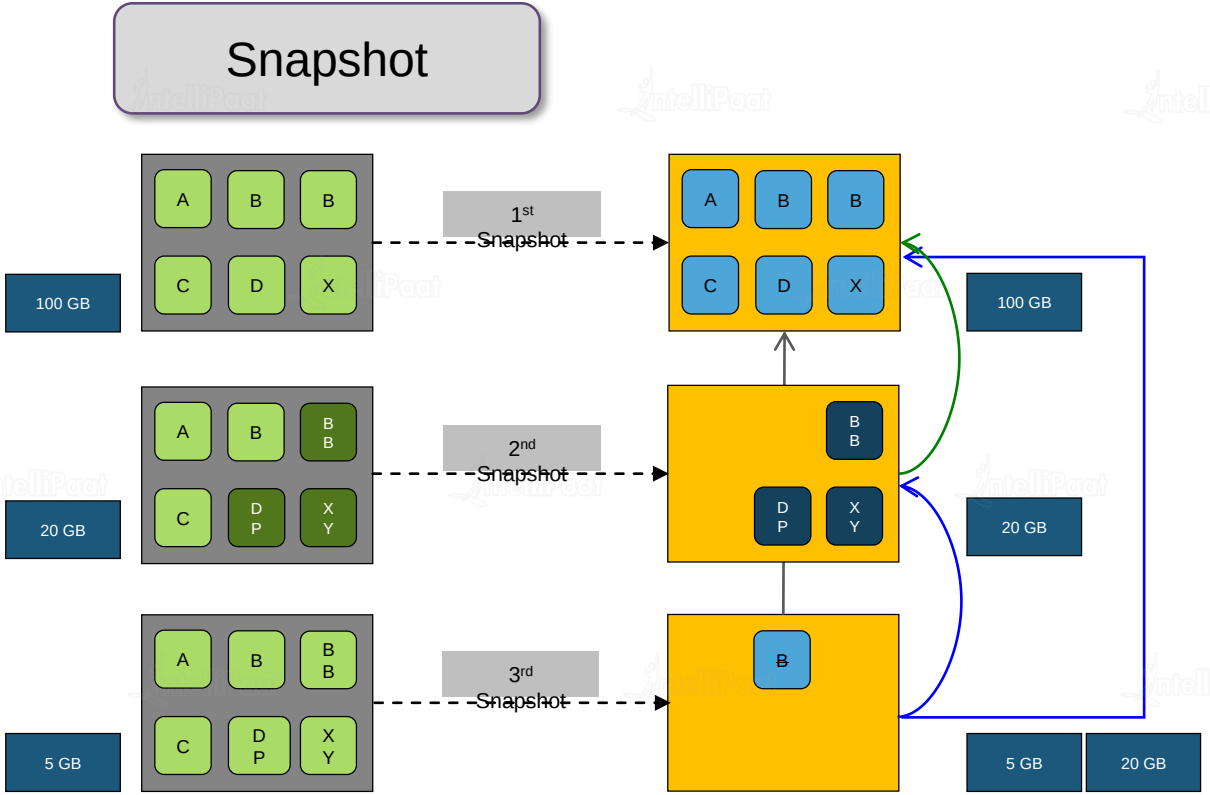
With Multi-Attach



EBS Snapshots

EBS Snapshot

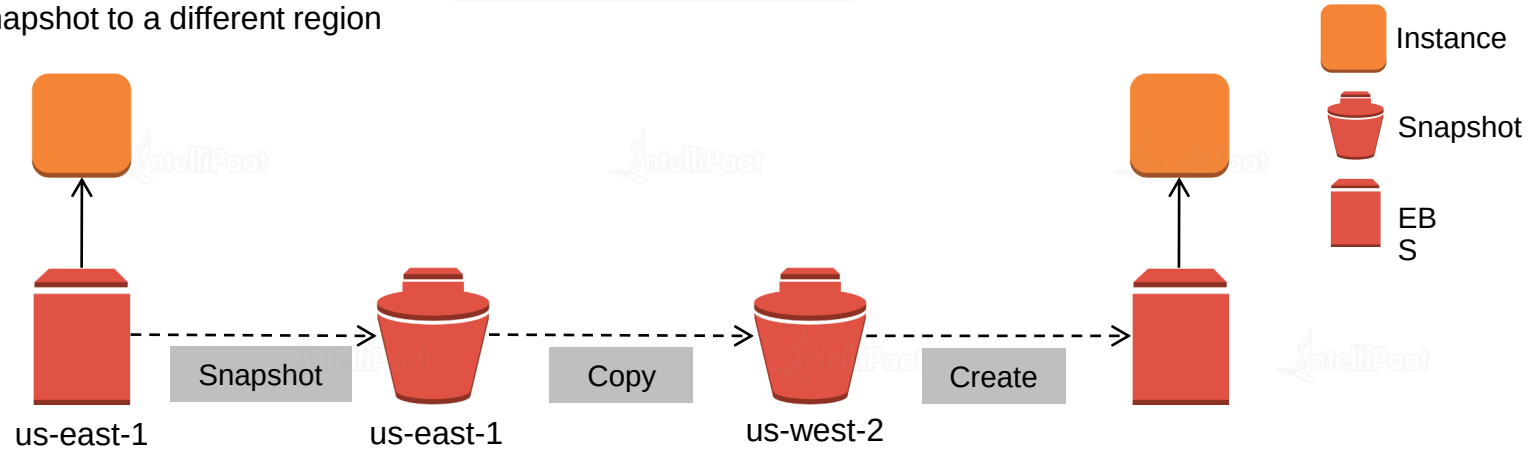
- Snapshots are used to backup data on EBS volumes
- All snapshots are incremental backups except for the first one
- Snapshots are copied to Amazon S3



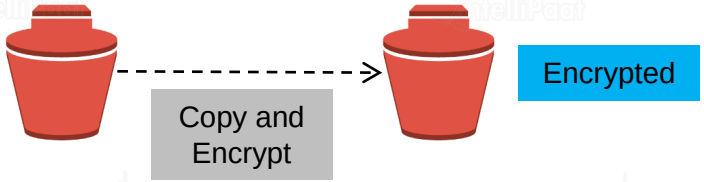
EBS Snapshot

Snapshot Copy

- Copy snapshot to a different region

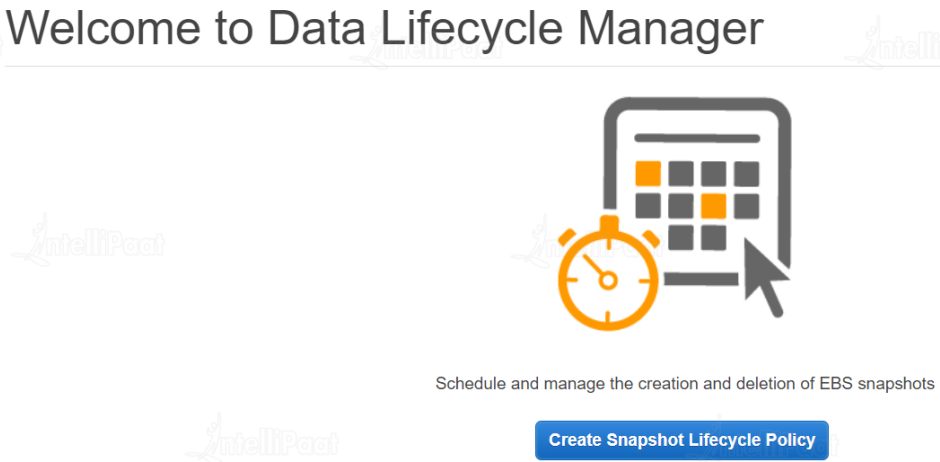


- Encrypt during copying



EBS Snapshot

New Feature: Data Lifecycle Manager for Snapshots



- Amazon DLM supports Amazon EBS volumes and snapshots
- We can define backup and retention schedules for EBS snapshots by creating lifecycle policies based on tags
- It is free to use
- We no longer need to create custom scripts for backup and restore

EBS Snapshot

Automating the snapshot cycle helps with:

- Protecting valuable data by enforcing a regular backup schedule
- Retaining backups as required by auditors or internal compliance
- Reducing storage costs by deleting outdated backups

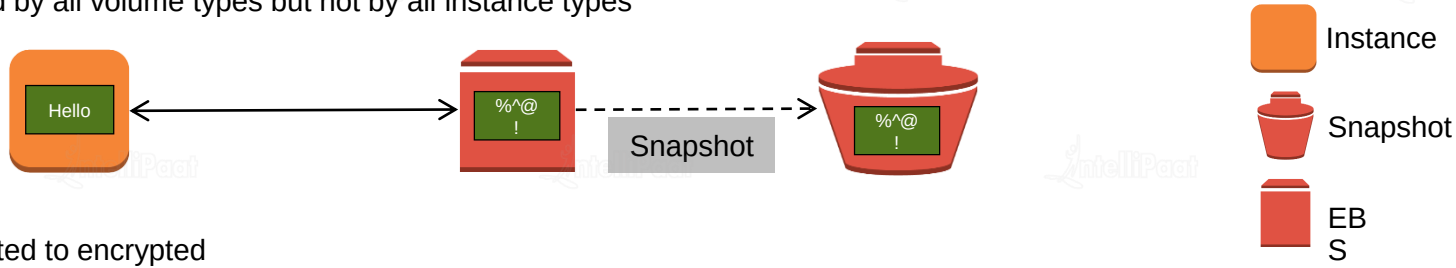
Quotas for AWS DLM:

- We can create up to 100 lifecycle policies per region
- We can add up to 45 tags per resource
- We can create one schedule per lifecycle policy

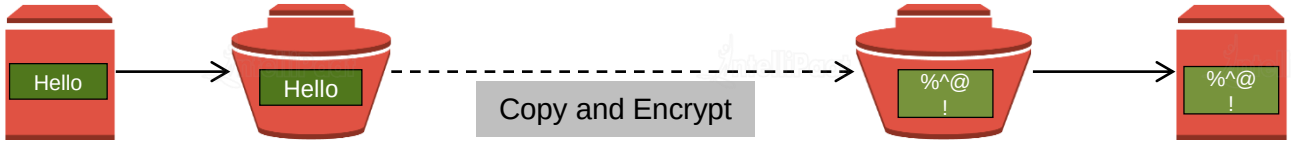
EBS Snapshot

EBS Encryption

- Supported by all volume types but not by all instance types



- Unencrypted to encrypted



- Encrypted to unencrypted

Demo 3: Creating and Mounting an EBS Volume

Demo 3: Creating and Mounting an EBS Volume

Creating an EBS volume

1. Choose EBS under Volumes from the EC2 dashboard
2. Click on Create volume
3. Reduce the size to 8 GB, and choose the availability zone the same as the zone of the created Ubuntu EC2 instance
4. Create the volume, and click on Actions → Attach Volume
5. Click on the instance field; choose the available instance, and proceed to create

A volume is created and is ready to be mounted!

Mounting the EBS volume onto the instance

Reconnect our Ubuntu instance and follow the commands below one by one to mount the created EBS volume to it

- lsblk (to get the device name)
- sudo mkfs -t ext4 <device-name>
- sudo mount <device-name> <file-system-name>
- sudo file –s <device-name> (to get file system type)

Creating an volume which can be attached with multiple EC2 instances with the Multi-attach feature.

Introduction to EFS

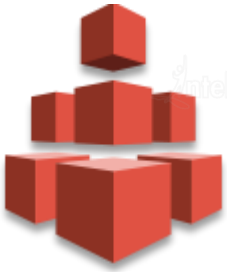
Introduction to EFS

Amazon Elastic File System

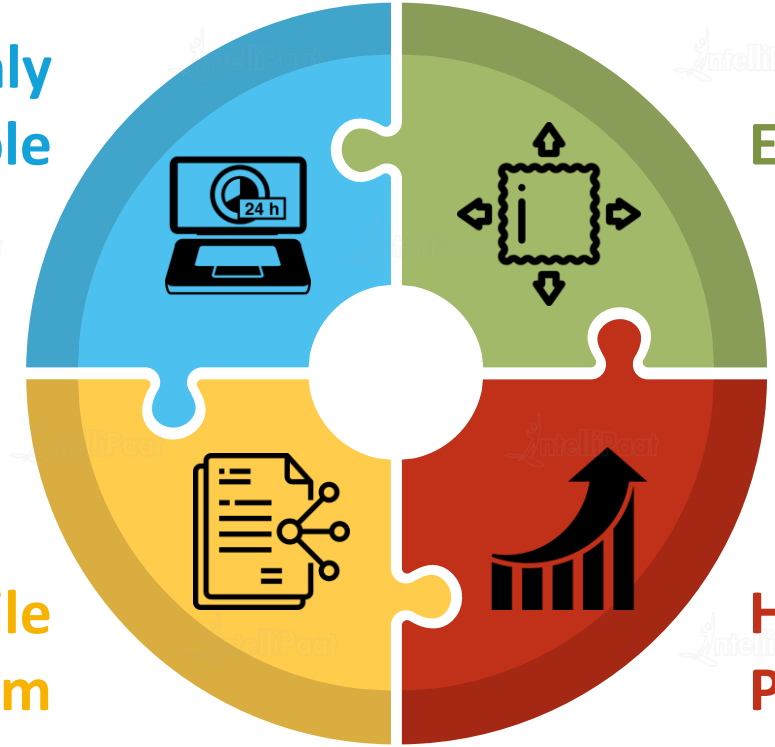
Amazon EFS (**Elastic File System**) is a cloud-based file storage service for applications and workloads that run in the **Amazon Web Services** (**AWS**) public cloud

Why do we need EFS?

If our application is running on Amazon EC2 and needs a file system or in any use case where a file system is needed



EFS Benefits



Demo: Creating and Mounting an EFS

Demo 4: Creating and Mounting an EFS

Creating an Amazon EFS

1. Open AWS Management Console; click on Services drop down, and choose EFS
2. Choose **Create File System**
3. Choose the **default VPC** from the VPC list
4. Tick off all checkboxes for all availability zones, and then click on Next
5. Name our file system, and add tags if needed
6. Select **General Purpose** and **Bursting** for high performance
7. Review the file system properties once, and then choose **Create File System**
8. Note down the **File System ID** value for further use

Mounting the EFS onto an EC2 Instance

1. Connect our Ubuntu EC2 instances using PuTTY
2. Install the NFS client using the following command:

```
sudo apt-get -y install nfs-common
```

3. Now, proceed with the commands one by one as mentioned below:

```
sudo mkdir efs
```

```
sudo mount -t efs fs-12345678:/ /mnt/efs
```

```
cd efs
```

```
sudo mkdir getting-started
sudo chown ec2-user getting-started
cd getting-started
```

```
touch test-file.txt
```

```
ls -al
```

Demo 5: Connecting to Elastic File System

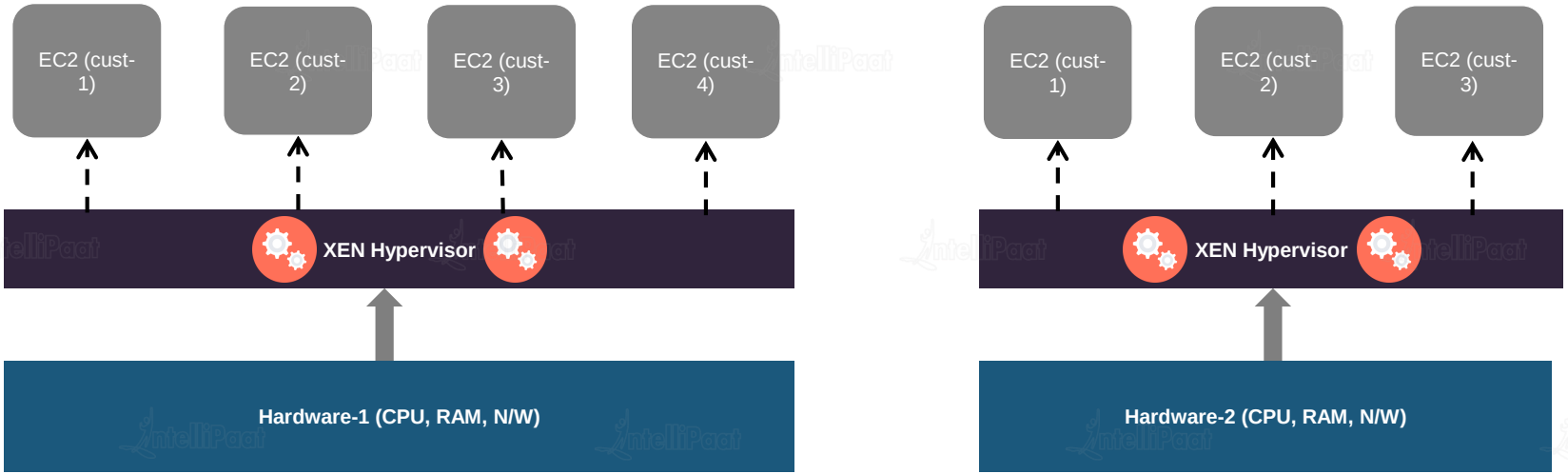
Connecting Multiple Instances with a Shared EFS

1. Create another EC2 instance (Ubuntu)
2. Mount the previously created EFS onto this instance
3. Create a file in the EFS directory in the second instance
4. Verify in the first instance whether the file that was created in the second instance is available

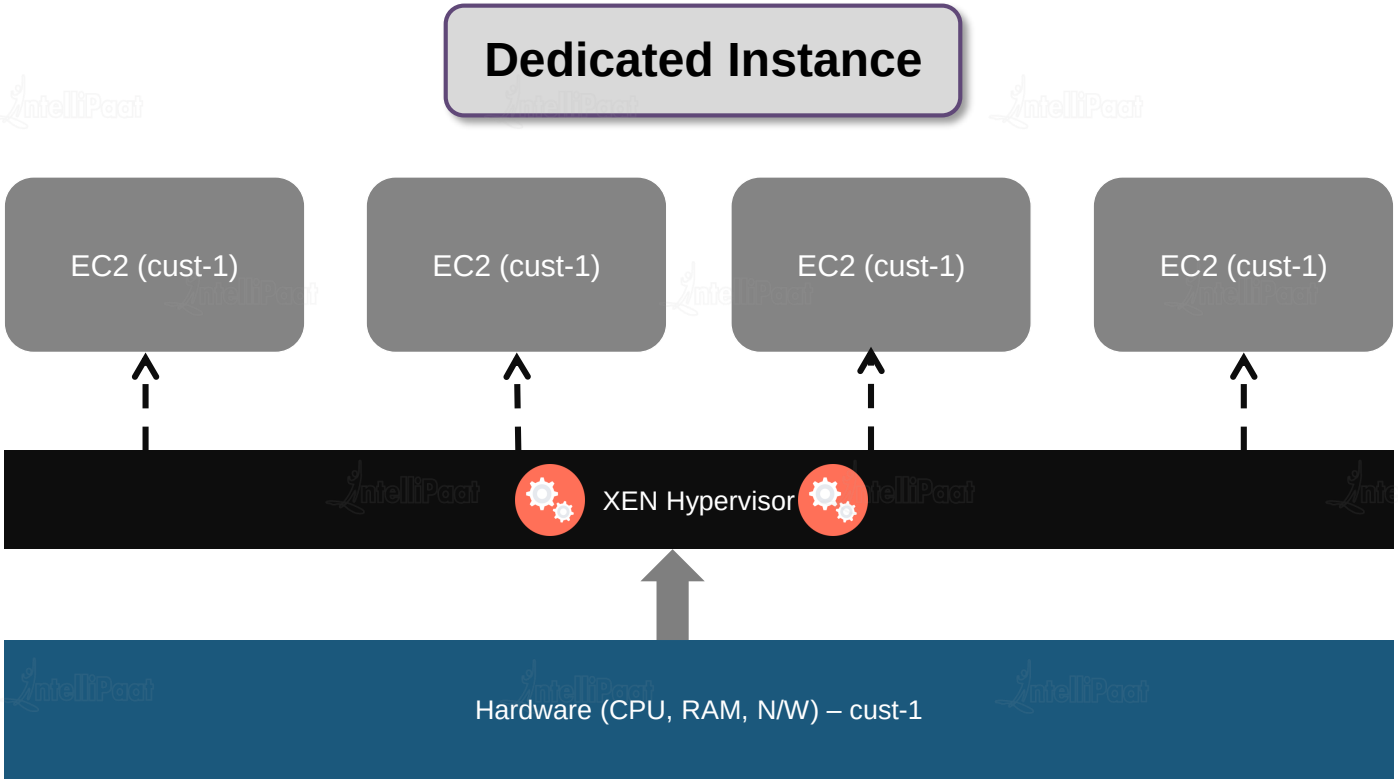
Instance Tenancy and Reserved and Spot Instances

Instance Tenancy

Shared/Default Instance

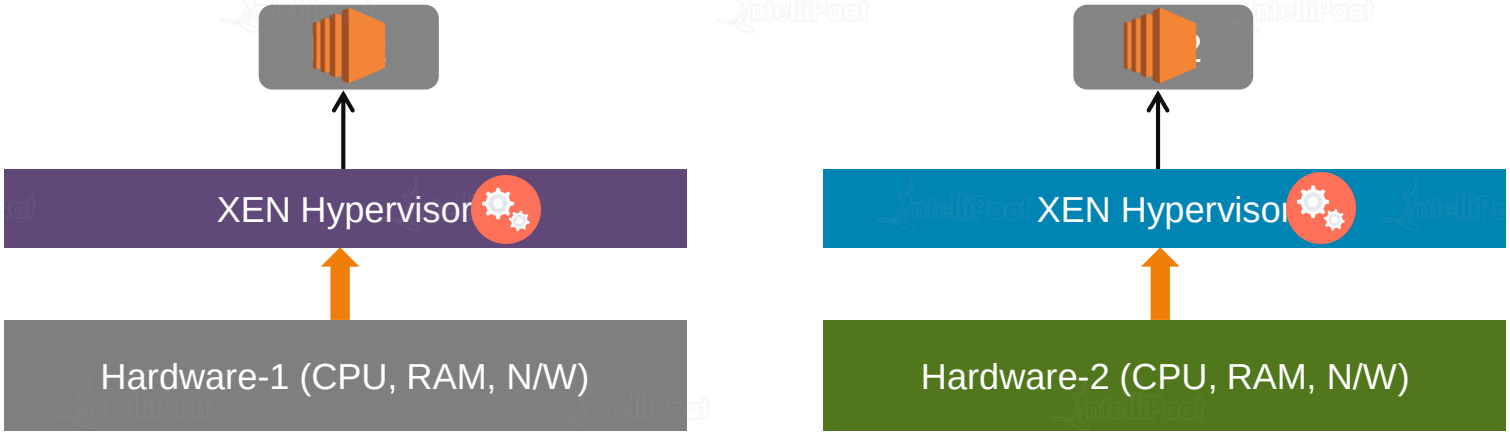


Instance Tenancy



Instance Restart

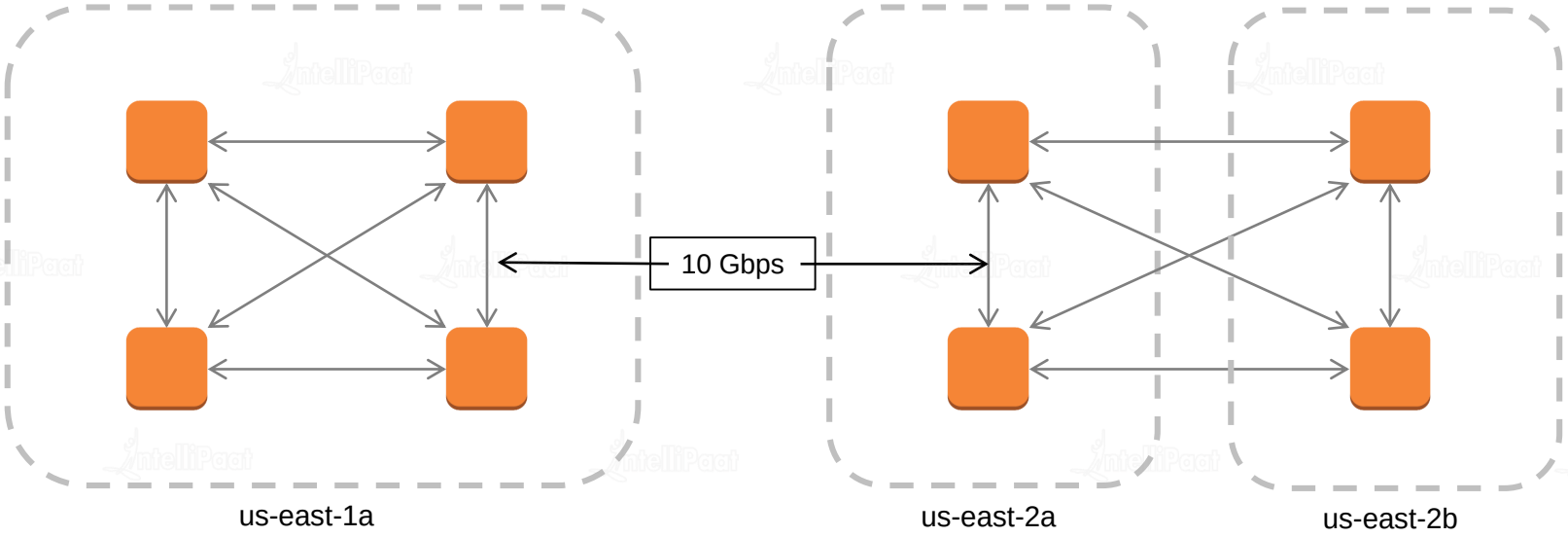
What happens during a restore?



Placement Group

Cross-platform PG

EC2 instances should support enhanced N/W



Reserved and Spot Instances

Reserved Instances

- Regional RI – AZ and Instance Size Flexibility (Both default and dedicated tenancy)
- Resources and capacity is reserved until the contract period ends
- Scheduled RI

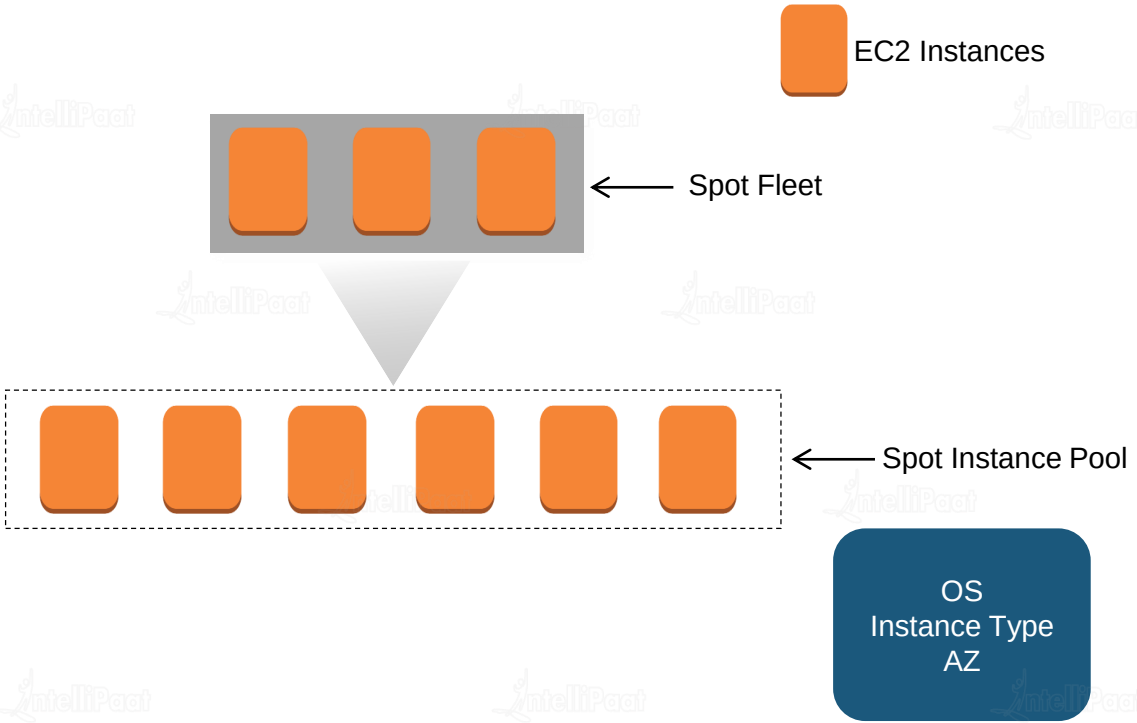
Running Instance	RI bought
4 m3.large Linux, default tenancy in AZ us-east-1a	4 m3.large, Linux, default tenancy, AZ us-east-1a
2 m4.4xlarge Amazon Linux, default tenancy in us-east-1b	4 m4.large, Amazon Linux, default tenancy, region us-east-1
c4.xlarge RHEL dedicated tenancy in AZ us-east-1c	C4.large, RHEL, default tenancy, region us-east-1

Instance size	Normalization factor
nano	0.25
micro	0.5
small	1
medium	2
large	4
xlarge	8
2xlarge	16
4xlarge	32
8xlarge	64
9xlarge	72
10xlarge	80
12xlarge	96
16xlarge	128
18xlarge	144
24xlarge	192
32xlarge	256

Reserved and Spot Instances

Spot Instances

- Unused EC2 instances available for lesser price than the on-demand price
- Instances are terminated if the spot price increases than the bid price
- Significant price reduction



Pricing

Pricing

EC2 Pricing (us-east-1)

- Pay as you use
- Free Tier: 750 hours per month of Amazon Linux, RHEL, SLES, Windows t2.micro single instance usage

On-demand price:

- m5.large = US$0.096/hour
- c5.large = US$0.085/hour
- r4.large = US$0.133/hour

Data Transfer IN:

FREE from anywhere

SLA = 99.99% Uptime

Data Transfer OUT:

From EC2 to

- S3, Glacier, DynamoDB, SES, and SQS in same region = FREE
- S3, Glacier, DynamoDB, SES, and SQS in different region = US$0.020/GB
- EC2, RDS, Redshift, Elasticache, ELB, and ENI in same AZ = FREE with private IP and US$0.010/GB with public IP
- EC2, RDS, Redshift, Elasticache, ELB, and ENI in different AZ = US$0.010/GB

EC2 Purchasing Options (RI)

- Reserved Instance: 1 to 3-year terms
- Pricing (on-demand us-east-1 region)

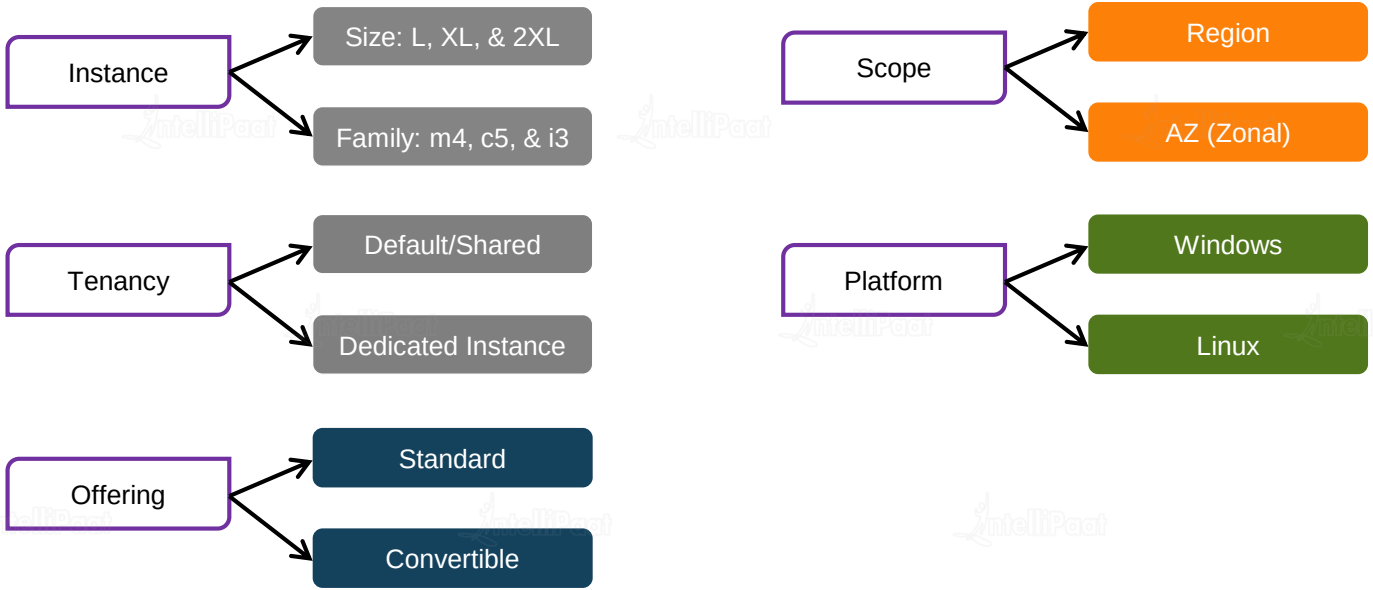
M5.XLARGE = US$0.192/hr

Yearly = US$1681.92

Payment Type	One Time Payment	Total Yearly Cost	Savings
No Upfront	US$0	US$89.79*12 = US$1077.48	36%
Partial Upfront	US$512	US512 + (42.34*12) = US$1020.08	39%
Full Upfront	US$1003	US$1003	40%

EC2 Purchasing Options (RI)

Reserved Instances



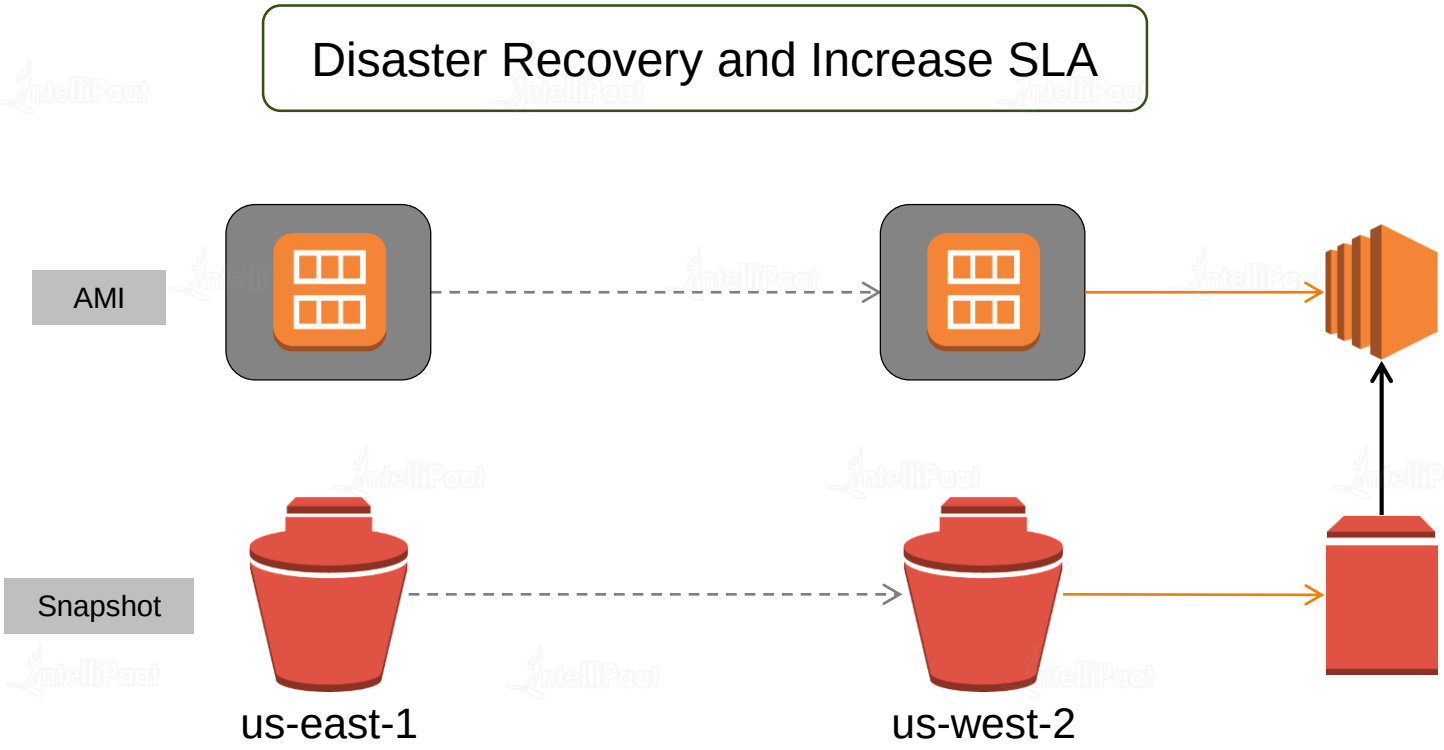
EBS Pricing

- gp2: US$0.1 per GB per month
- io1: US$0.125 per GB per month and US$0.065 per provisioned IOPS per month
- st1: US$0.045 per GB per month
- sc1: US$0.025 per GB per month
- EBS snapshot to Amazon S3: US$0.05 per GB per month
- Free Tier: 30 GB/month, a combination of gp2 and magnetic. 2,000,000 IO with magnetic, 1 GB of snapshot storage
- Visit https://aws.amazon.com/ebs/pricing/ for details

Uptime SLA: 99.99%

Design Patterns

Design Patterns



India : +91-7847955955

US : 1-800-216-8930 (TOLL FREE)

support@intellipaat.com

24/7 Chat with Our Course Advisor