
ondia

AWS EFS

AGENDA

- Introduction to EFS (Elastic File System)
- Features of EFS
- Comparison of Storage Systems

Introduction to EFS

What is File System?

- In computing, a file system or filesystem (often abbreviated to fs) is a method and data structure that the operating system uses to control how data are stored and retrieved.
- Every time you open a file on your computer or smart device, your operating system uses its file system internally to load it from the storage device.
- Or when you copy, edit, or delete a file, the file system handles it under the hood.

Why do we need a file system in the first place, you may ask?

- Well, without a file system, the storage device would contain a big chunk of data stored back to back, and the operating system wouldn't be able to tell them apart.
- The term file system takes its name from the old paper-based data management systems, where we kept documents as files and put them into directories.
- Imagine a room with piles of papers scattered all over the place. A storage device without a file system would be in the same situation - and it would be a useless electronic device.

What is File System?

It is a system used by an operating system to manage files. The system controls how data is saved or retrieved

What is File System?

It is a system used by an operating system to manage files. The system controls how data is saved or retrieved

What is File System?

Operating system stores files and directories in an organized and structured way

System configuration file = Folder A
User files = Folder B
Log files = Folder C
Commands or scripts = Folder D and so on

There are many different types of filesystems. In general, improvements have been made to filesystems with new releases of operating systems and each new filesystem has been given a different name
e.g. ext3, ext4, ext5, xfs, NTFS, FAT etc.

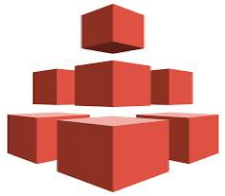
What is Network File System(NFS)?

The Network File System (NFS) is a mechanism for storing files on a network.

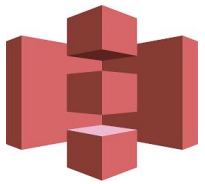
It is a distributed file system that allows users to access files and directories located on remote computers and treat those files and directories as if they were local.

Introduction to EFS

Recap of the Storage Options



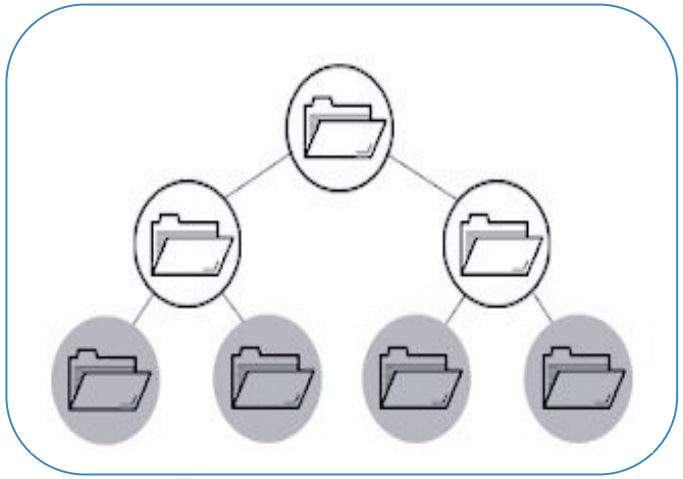
Amazon EFS



S3

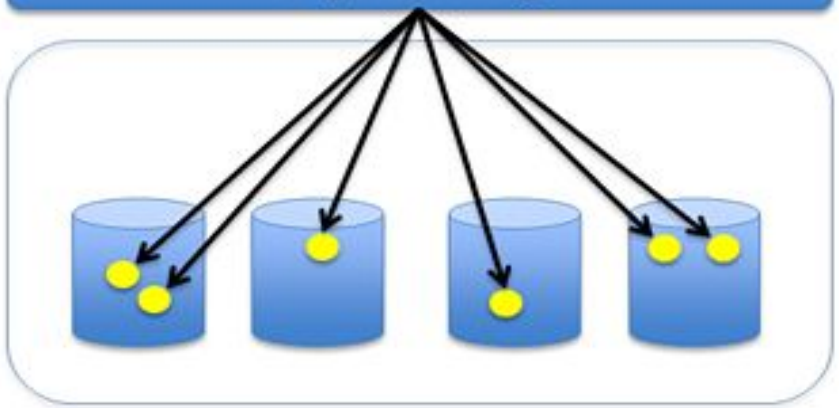
Amazon Elastic Block Storage (EBS)

File Storage

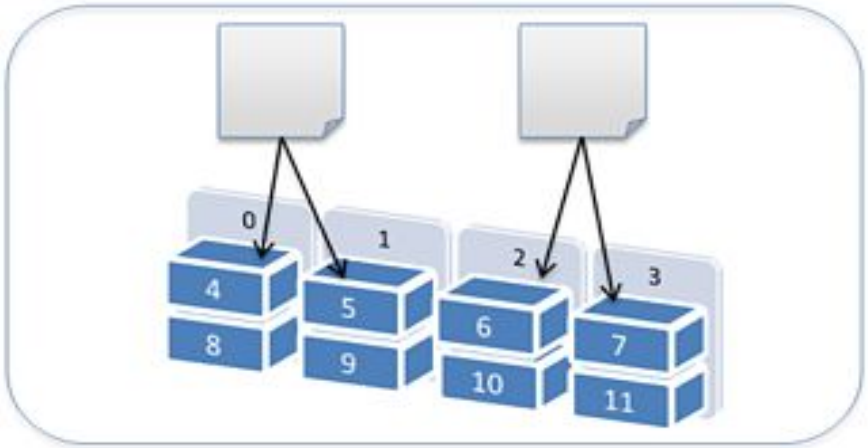


HTTP(S) Interface

Object Storage

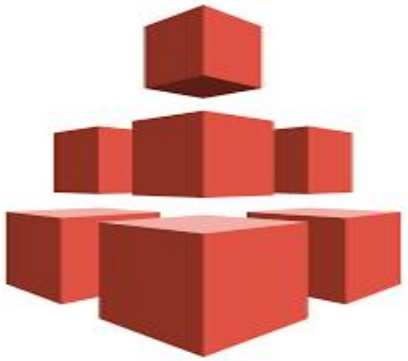


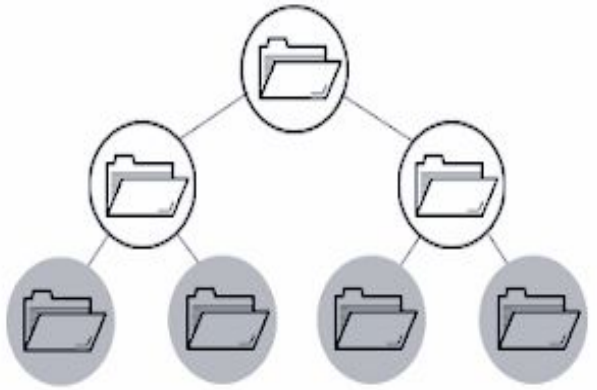
Block Storage



Introduction to EFS

What is EFS?





- Simple, scalable, fully managed Elastic NFS file system.
- It offers a traditional file storage concept, with data organized into directories and subdirectories.

Introduction to EFS

An EFS is a Network File System (NFS) that organizes data in a logical file hierarchy. Data is stored in a path-based system, where data files are organized in folders and sub-folders.

EFSs are ideal candidates for storing:

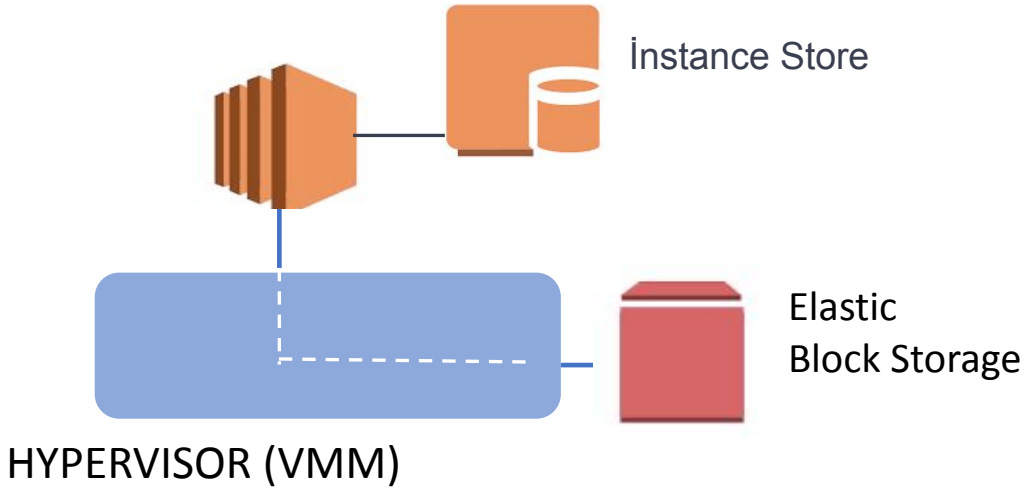
- Organizational data
- File server
- Individual data
- Application data

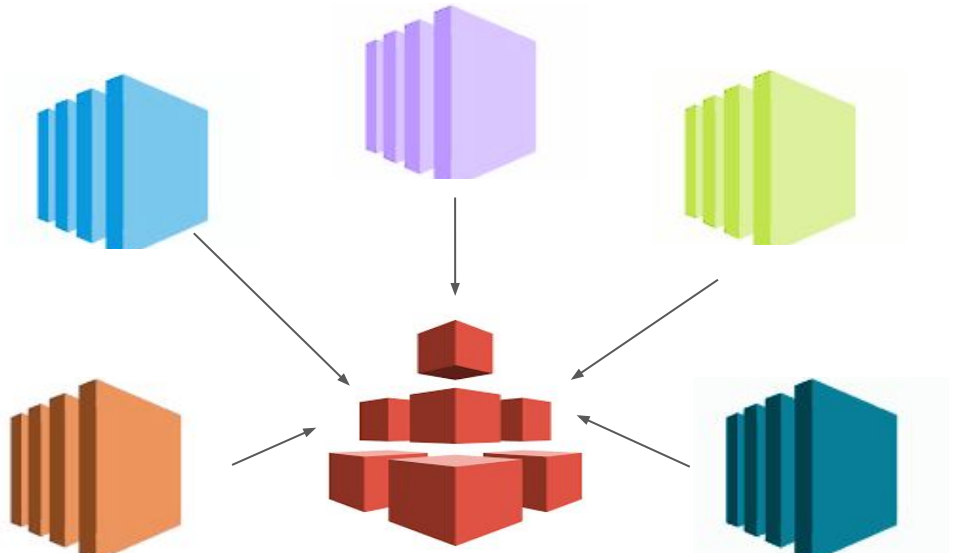
Amazon states that a single EFS can be simultaneously connected to thousands of Elastic Compute Cloud (EC2) instances or on-premise resources, allowing you to share EFS data with as many resources as needed. Access to shared EFS folders and data is provided through native operating system interfaces.

Features of EFS

Features of EFS

Attaching



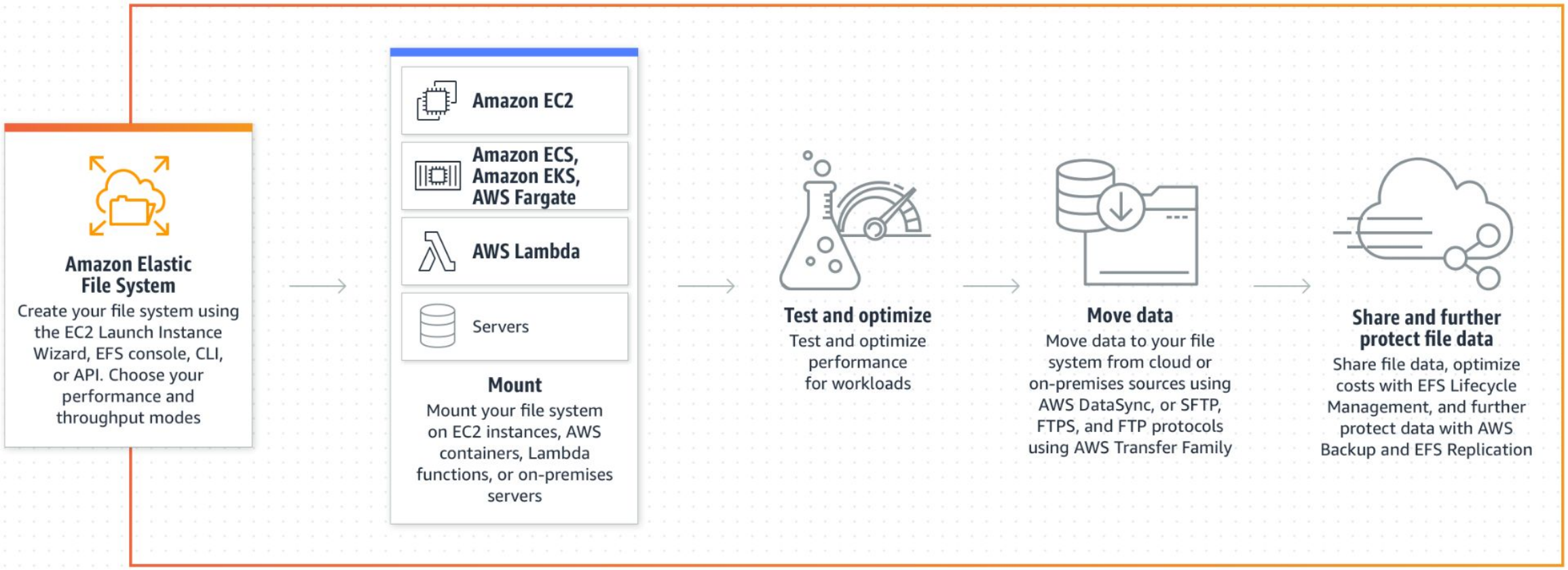


- Unlike *EBS, **multiple Amazon EC2** instances (**Linux only**) even **in different AZ's** can be attached Amazon **EFS** file system **at the same time**.
-

*Except Nitro-based instances in the same Availability Zone.

Features of EFS

Attaching



Features of EFS

Scalability-Cost

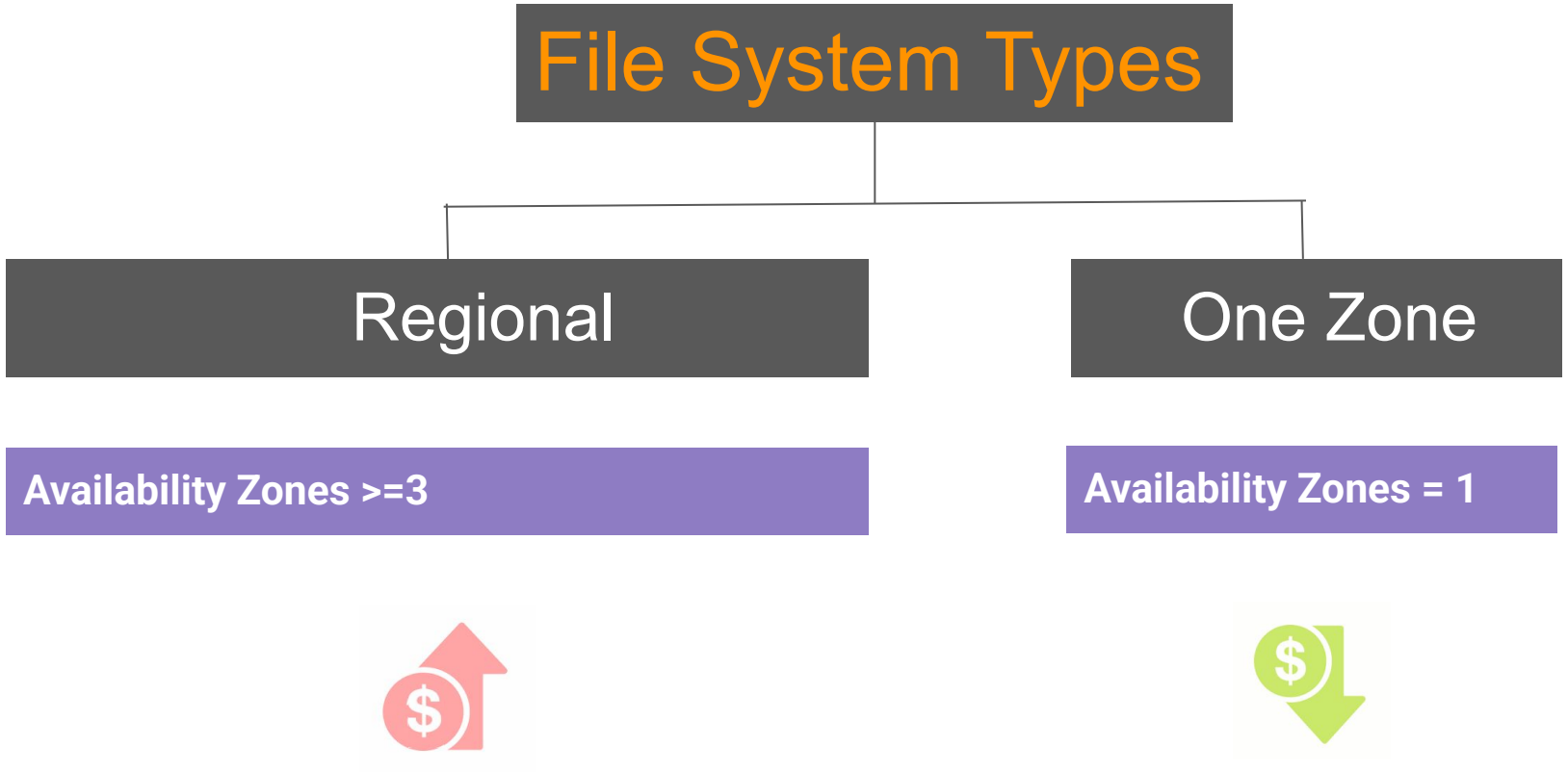
- Since EFS is scalable, it increases and decreases the storage capacity automatically as you add and delete files,

- There is no minimum fee or setup cost.

Free Tier : 5 GB of EFS Standard storage with a Regional file system for 12 months.

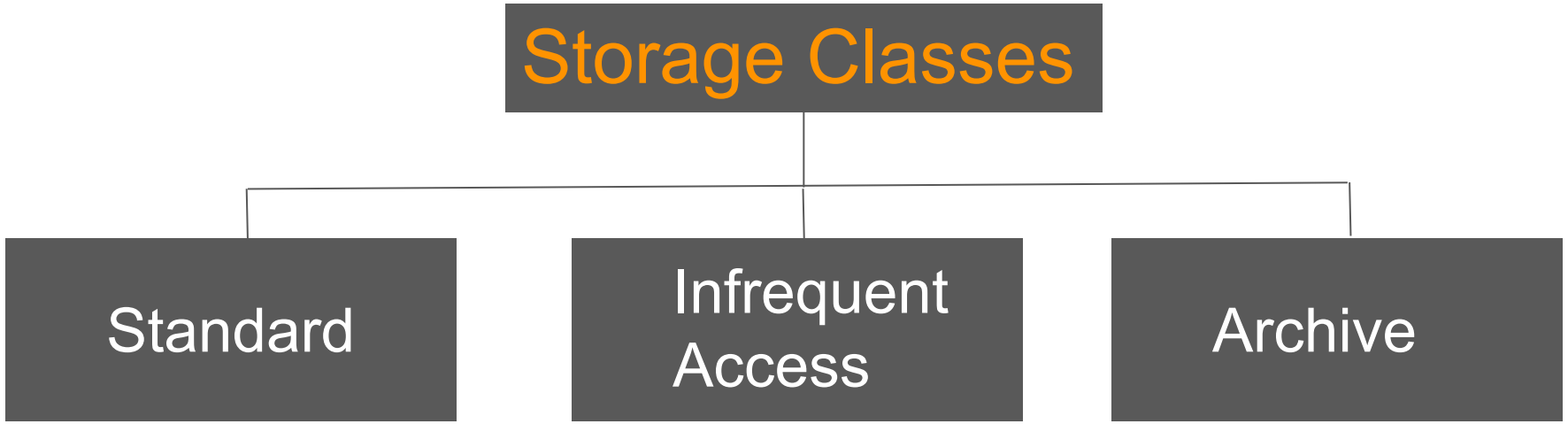
Features of EFS

EFS File System Types



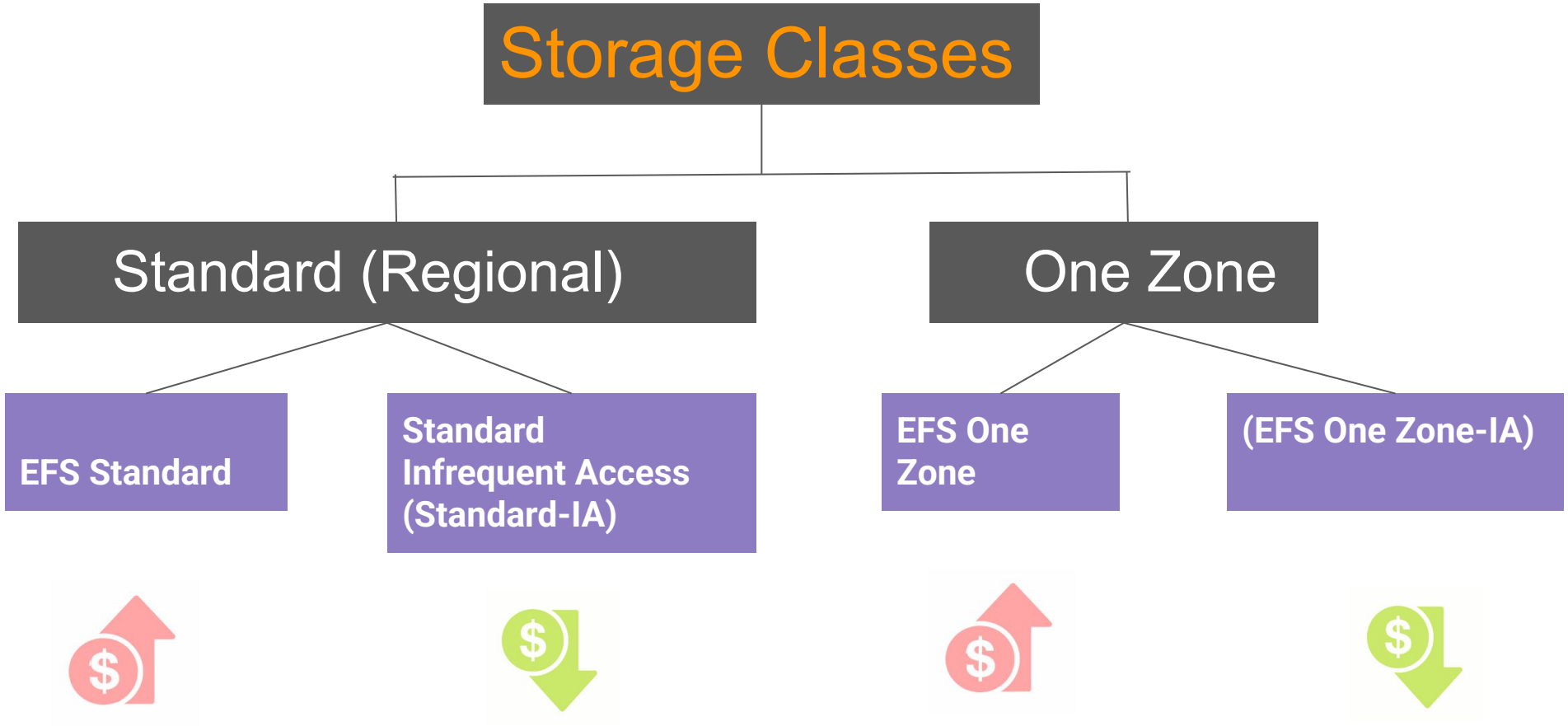
Features of EFS

Storage Classes

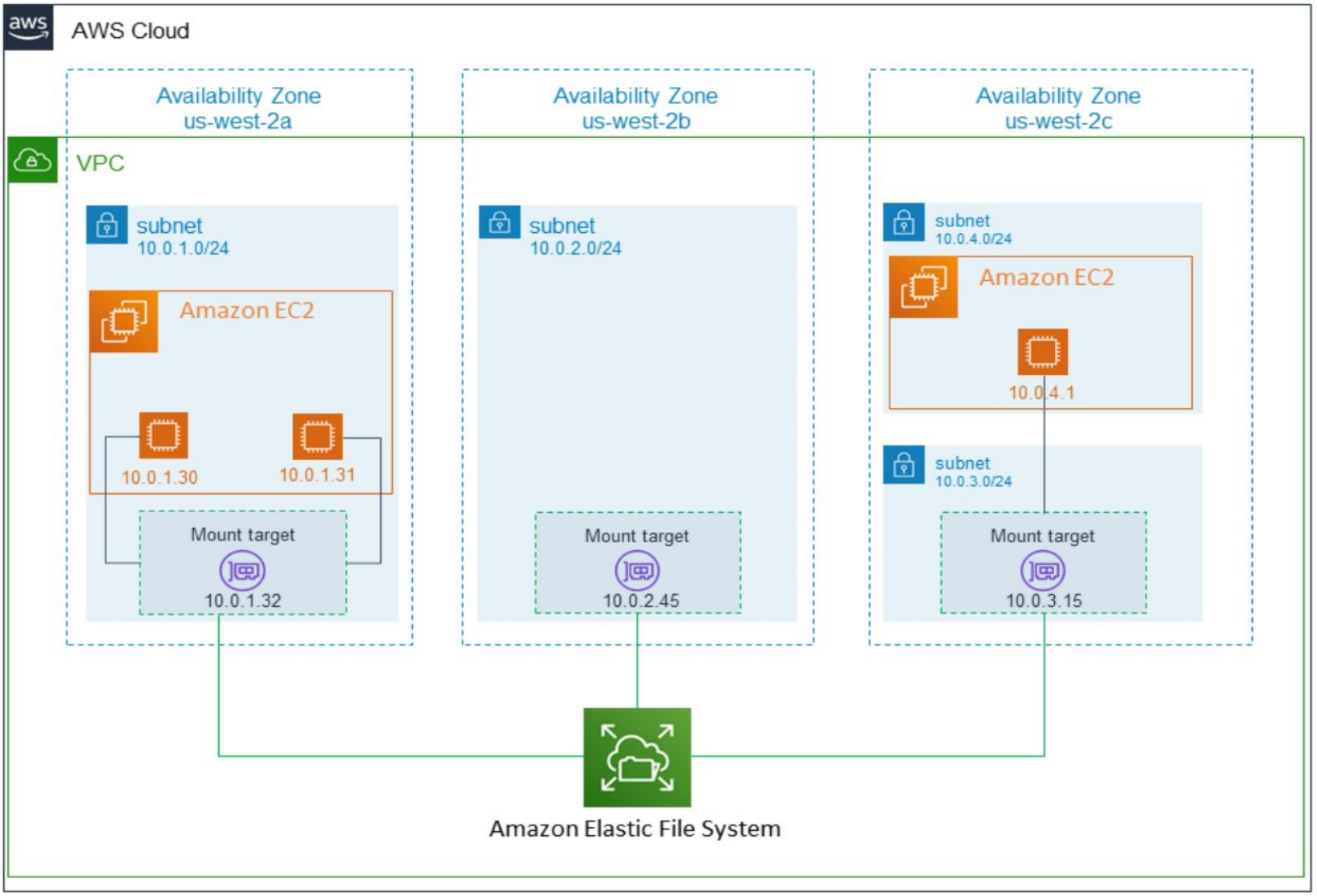


Features of EFS

Storage Classes

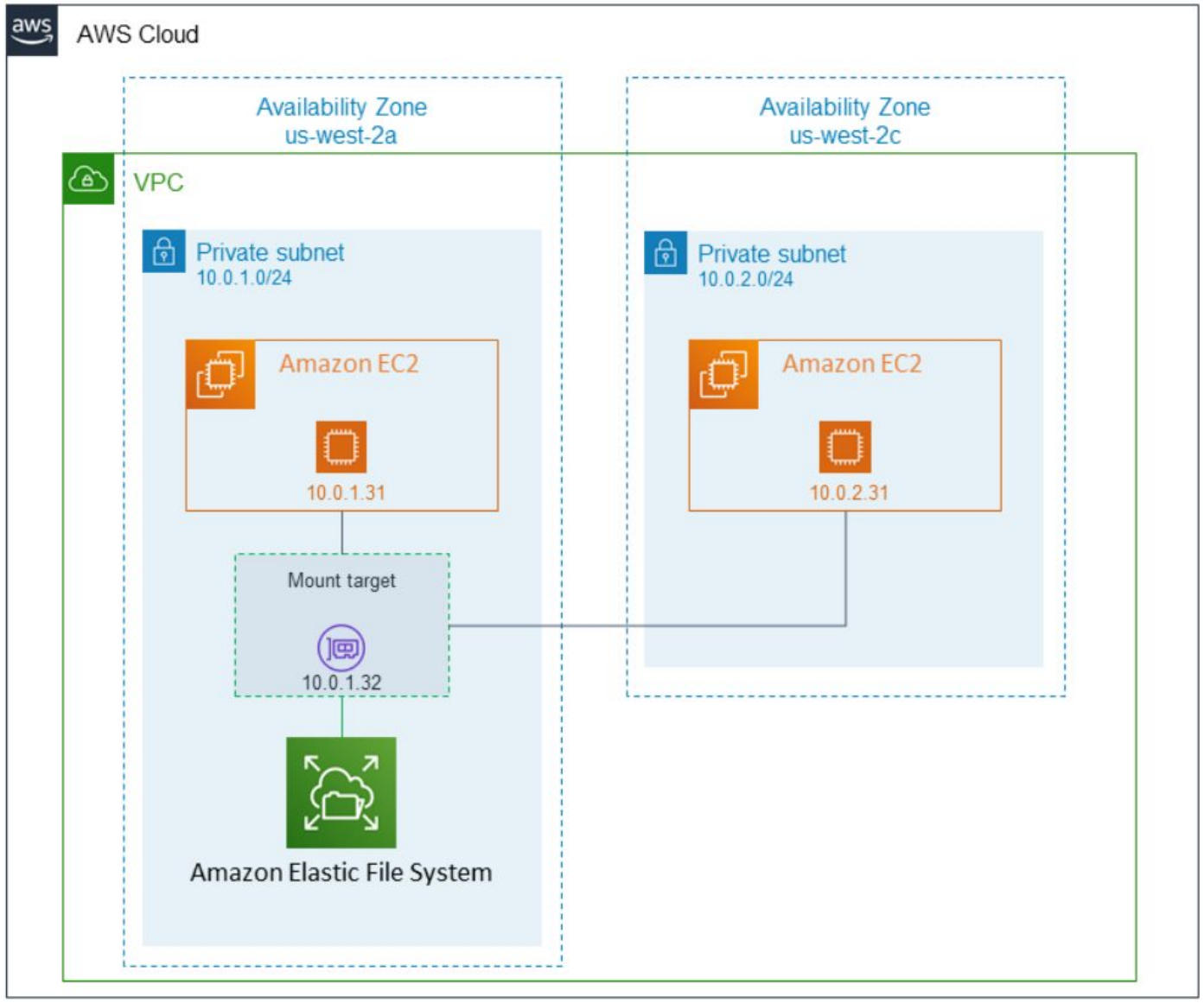


EFS Structure : Mount Target (for Regional Storage Class)



- Mount Target is a AZ based component.
- You can create only one Mount Target in a AZ
- It will be located only in one subnet of the relevant AZ.
- EFS mount targets are not "servers", but elastic network interfaces which expose EFS to our VPC. Our client access EFS by connecting to the mount targets (i.e. ENIs).
- If our VPC spans 3 AZs, to ensure high availability of access to our file system, a good practice is to create a mount target for each AZ.

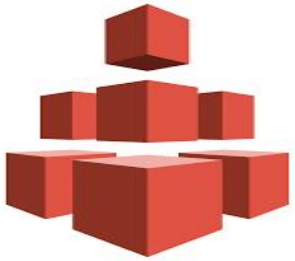
EFS Structure : Mount Target (for One-Zone Storage Class)



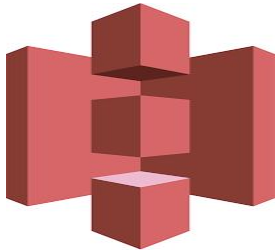
- Mount Target is created only in one subnet in relevant AZ.
- Other AZs also uses this Mount Target to communicate with EFS
- Amazon EFS file systems that use One Zone storage classes support only a single mount target which is located in the same Availability Zone as the file system.

Comparison of Storage System

Comparison of Storage Systems



Amazon EFS

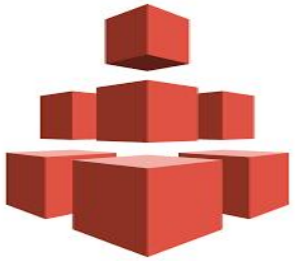


S3

EBS

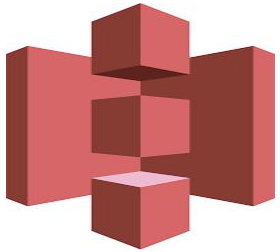
- Cost Optimized : EFS > EBS > S3
- Speed : EBS , EFS >S3
- EC2 mount : S3 : No
  EBS : Single*
  EFS : Multiple
- Storage Capacity : S3, EFS = ∞ vs. EBS = 16 - 64 TiB

Comparison of Storage Systems



Amazon EFS

- Large quantities of data,
- Large analytic workloads.
- Global content management



S3

-Website images and videos,
-Data analytics of mobile/web applications.
- Data which is needed to be accessed from anywhere.

EBS

- High IOPS required data ,
- Database management.

Comparison of Storage Systems

		File Amazon EFS	Object Amazon S3	Block Amazon EBS
Performance	Per-operation latency	Low, consistent	Low, for mixed request types, and integration with CloudFront	Lowest, consistent
	Throughput scale	Multiple GBs per second	Multiple GBs per second	Single GB per second
Characteristics	Data Availability/Durability	Stored redundantly across multiple AZs	Stored redundantly across multiple AZs	Stored redundantly in a single AZ
	Access	One to thousands of EC2 instances or on-premises servers, from multiple AZs, concurrently	One to millions of connections over the web	Single EC2 instance in a single AZ
	Use Cases	Web serving and content management, enterprise applications, media and entertainment, home directories, database backups, developer tools, container storage, big data analytics	Web serving and content management, media and entertainment, backups, big data analytics, data lake	Boot volumes, transactional and NoSQL databases, data warehousing & ETL

THANKS!

Any questions?