Volume States:

1. **In-use:** Volume is already attached to a resource.

2. **Available:**

3. **Creating:** Volume is being created for use.

Attaching EBS volume to a Windows Instance:

1. Select the volume which we want to attach.
2. Click on 'Actions'.
3. Click on 'Attach Volume'.
4. Select the instance in the same AZ.
5. Select the device name (sdf/sdg).
6. Connect to Windows Instance using RDP.
7. Now in instance, search for 'Server Manager'.
8. At top-right, select 'Tools' and them select 'Computer Management'.
9. Within it, in 'Storage', select 'Disk Management'.
10. Select 'GPT'.

Disk 0 -> Root EBS volume (30 GB)

Disk 2-> Additional EBS volume (25 GB)

Disk 1 -> Instance Store (1200 GB)

R&D on MBR & GPT

R&D on Disc RAID

Drawbacks of EBS volume:

1. EBS volumes are AZ specific.
2. One volume can be connected to only one instance at a time.
3. Maximum storage of EBS is limited to 16 TB.

S3 (Simple Storage Service)

Types of Data Storage:

2. Object Storage:

Object: Data, metadata, key

- Data: The actual content & each object is a data
     eg: The video itself.

-Metadata: Data about the data.
     eg: video type, length of video, size, etc.

- Key: Unique identifier
     eg: reference address to the video/object.

>> In object storage, data is stored as an object, not as files or blocks on disks.
>> Storing the data as it is. Eg: a movie is stored as a movie, same goes with a mp3 file.
>> They can have:
     — Structured data: Tabular data
     — Unsructured data: audio, video,images,etc.
     — Unsructured data: log files.

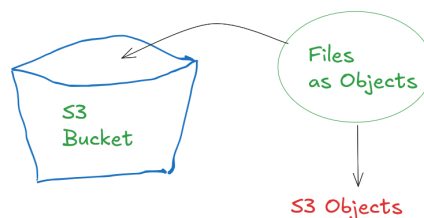>> It is very easy to scale and can store unlimited data virtually.

S3

>> S3 is an object storage service that offers scalability, data availability, security, and performance.

>> It belongs to storage category. .docx,.jpeg images, .pem,.java,.etc,

>> It was launched in 2006,

>> Using S3, we can keep any types of file & can retrieve them when needed.

EBS	S3
1. Limited scalability as maximum storage capacity cap at 16 TB.	1. Unlimited storage to store billions of objects.
2. Data is stored in blocks.	2. Data is stored as obejcts..
3. Data is only available in that particular AZ.	3. Data is accessible across AZ.
4. High performance it acts as a local disk & it will available immediately to store & retrieve the data.	4. Not suited for performance and will have higher latency.
5. Use cases: when the app requires frequent read/write.	5. Use cases: storing videos, backup files, log files, etc.

S3
Bucket

Files
as Objects

S3 Objects

S3 Bucket:

>> In S3, we store the data in the form of bucker, i.e, whatever storage we
   are creating in S3, it will be called as 'Bucket'.

>> Buckets will be created in a region.

    S3 bucket -> Mumbai   S3 bucket -> ap-south-1a;
                                              ap-south-1b;
                                              ap-south-1c.
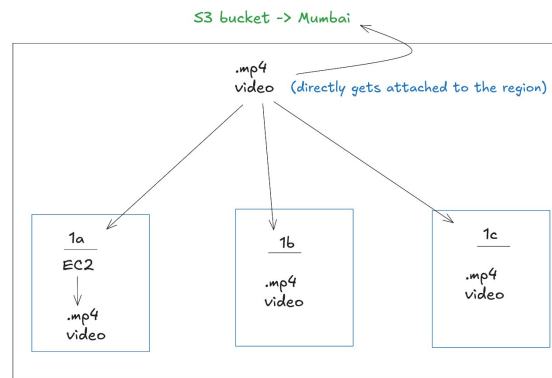
>> Buckets will be created in a region.
>> S3 service is a regional service and not AZ-specific (like volumes).
>> S3 is a global service.

                    In AWS, a 'global service is one that:
                    —— doesn't depend on a region.
                    —— can be used & managed at world-wide level.
                    —— doesn't depend on an AZ.
                    —— e: S3, CDN, IAM.
                    —— non-global services: EC2,EBS, RDS, Lambda,etc.

S3 bucket -> Mumbai

.mp4
video   (directly gets attached to the region)

1a	1b	1c
EC2	.mp4 video	.mp4 video
.mp4 video		

Bucket naming convention:

>> Buckets name should have:

        min: 3 characters
        max : 63 characters

>> Only consists of lowercase characters, numbers, dots(.) & hyphen (-).
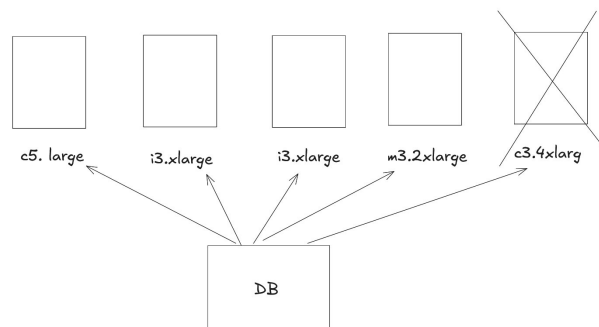
>> Bucket's name should begin & end with either a letter or name.

>> Bucket's name must not contain 2 adjacent periods.

>> Bucket's name must be unique across all AWS account.

    https://docs.aws.amazon.com/AmazonS3/late
    st/userguide/bucketnamingrules.html

Spot Request

| c5. large | i3.xlarge | i3.xlarge | m3.2xlarge | c3.4xlarg |

DB

-> 5 instance running for complete 72 hours.

-> Spot fleet request

-> if instances are reclaimed by the user, then that instance
    is interrupted BUT, AWS will replinsh/replace that instance
    with another unused instance.