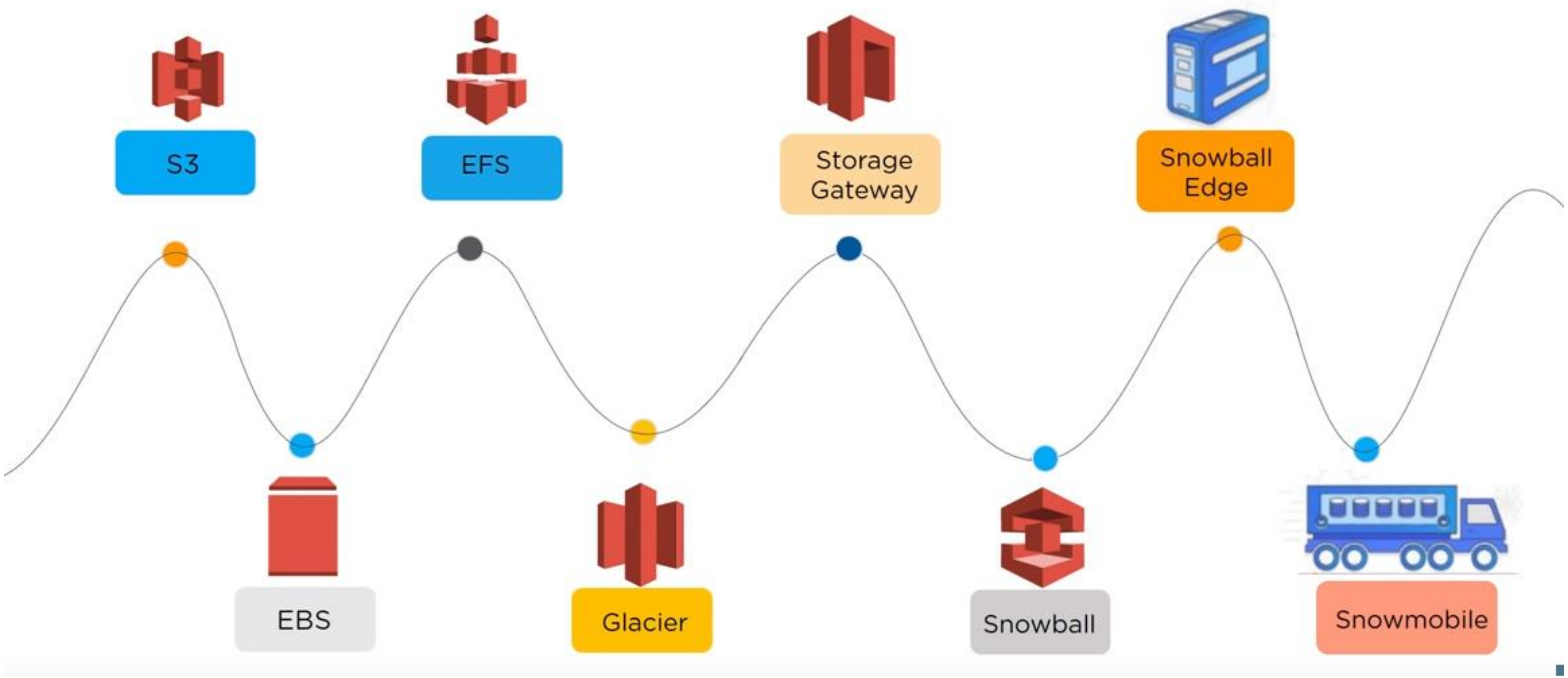
Simple Storage Service(S3)

Need more Space? Storage?

- The need for **storage** is increasing every day, so building and maintaining your own repositories, therefore, becomes a tedious and tiresome job because knowing the amount of capacity you may need in the future is difficult to predict.
- You may either over-utilize it leading to an application failure because of not having sufficient space or you may end up buying stacks of storage which will then be under-utilized.
- Keeping all these hassles in mind, Amazon came up with an internet storage service called *AWS S3.*

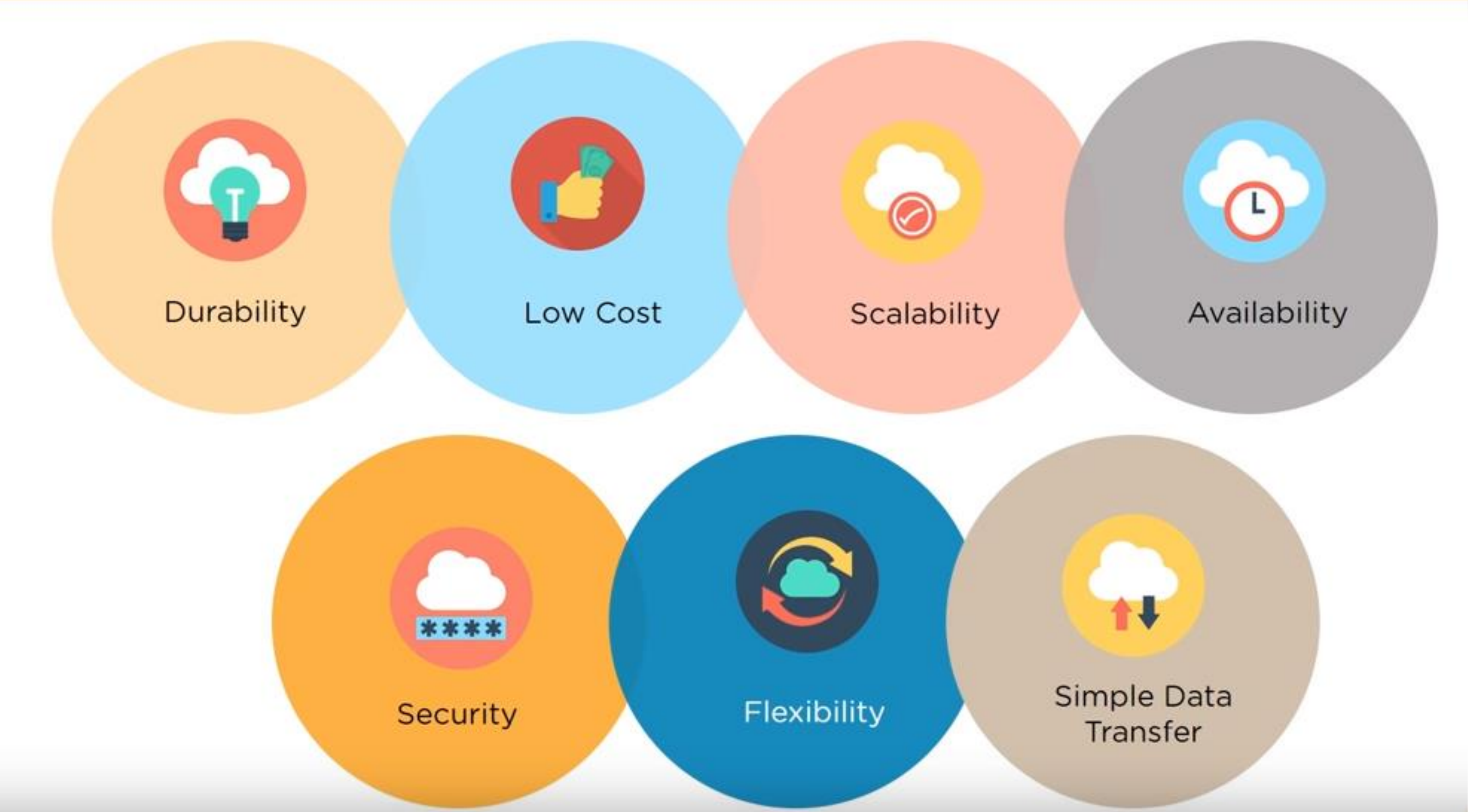
Types of Storages on AWS



Amazon Simple Storage Service (S3)

- **Amazon S3** (Simple Storage Service) is a scalable, high-speed, low-cost web-based service designed for online backup and archiving of data and application programs.
- It allows to upload, store, and download any type of files from 0 byte up to 5 TB in size.
- This service allows the subscribers to access the same systems that Amazon uses to run its own web sites. The subscriber has control over the accessibility of data, i.e. privately/publicly accessible.

Benefits


Durability
Low Cost
Scalability
Availability
Security
Flexibility
Simple Data Transfer

Amazon S3 Concepts

S3
Scalable Storage in the Cloud

This section describes key concepts and terminology you need to understand to use Amazon S3 effectively. They are presented in the order you will most likely encounter them.

- Buckets
- Objects
- Keys
- Regions

Buckets

S3
Scalable Storage in the Cloud

- A bucket is a container for objects stored in Amazon S3. Every object is contained in a bucket.
- For example, if the object named photos/puppy.jpg is stored in the johnsmith bucket, then it is addressable using the URL ***http://johnsmith.s3.amazonaws.com/photos/puppy.jpg***
- You can configure buckets so that they are created in different regions.
- You can also configure a bucket so that every time an object is added to it, Amazon S3 generates a unique version ID and assigns it to the object.

Objects

S3

Scalable Storage in the Cloud

- Objects are the fundamental entities stored in Amazon S3.
- Objects consist of object data and metadata.
- The metadata is a set of name-value pairs that describe the object. These include some default metadata, such as the date last modified, and standard HTTP metadata, such as Content-Type. You can also specify custom metadata at the time the object is stored.
- An object is uniquely identified within a bucket by a key (name) and a version ID

Keys

- A key is the unique identifier for an object within a bucket.(simply name of the object.)

- Every object in a bucket has exactly one key. Because the combination of a bucket, key, and version ID uniquely identify each object.

- Amazon S3 can be thought of as a basic data map between "bucket + key + version" and the object itself. Every object in Amazon S3 can be uniquely addressed through the combination of the web service endpoint, bucket name, key, and optionally, a version.

- For example, in the URL http://doc.s3.amazonaws.com/2006-03-01/AmazonS3.wsdl, "doc" is the name of the bucket and "2006-03-01/AmazonS3.wsdl" is the key.

Regions

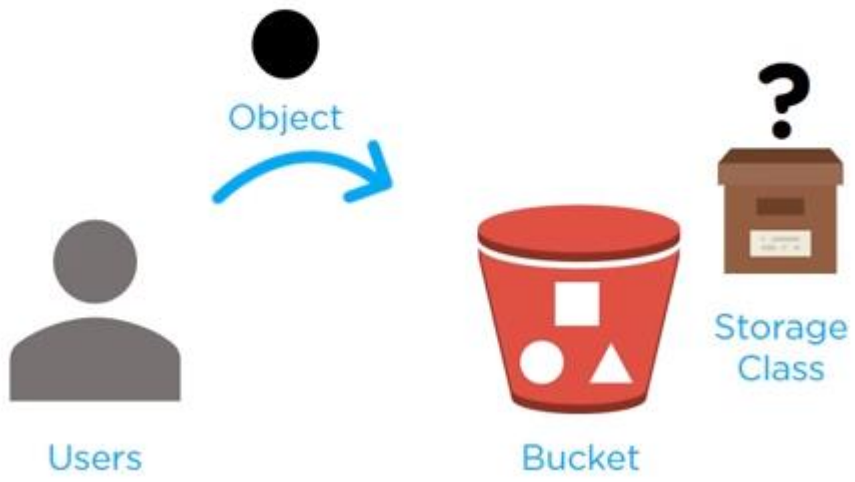
S3
Scalable Storage in the Cloud

- You can choose the geographical region where Amazon S3 will store the buckets you create.

- You might choose a region to optimize latency, minimize costs, or address regulatory requirements. Objects stored in a region never leave the region unless you explicitly transfer them to another region. For example, objects stored in the EU (Ireland) region never leave it.

How does Amazon S3 Works ?

When files are uploaded to the bucket, the user will specify the types of s3 storage class to be used for those specific objects



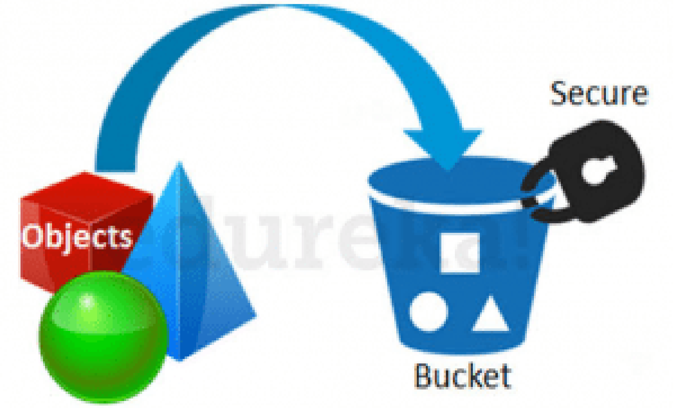
How is data organized in S3?

Data in S3 is organized in the form of buckets.

- A Bucket is a logical unit of storage in S3.

- A Bucket contains objects which contain the data and metadata.

Before adding any data in S3 the user has to create a bucket which will be used to store objects

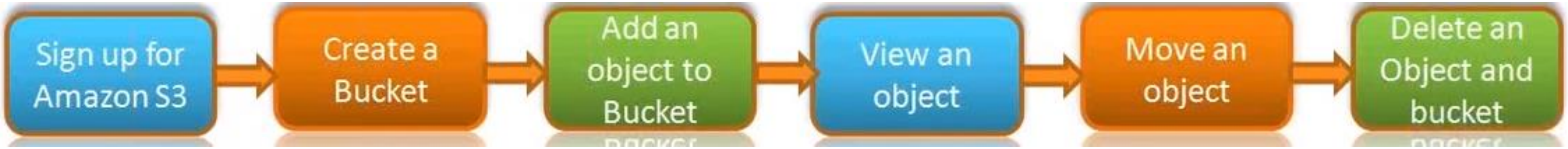


Advantages of S3

Amazon S3 is intentionally built with a minimal feature set that focuses on simplicity and robustness. Following are some of advantages of the Amazon S3 service:

- **Create Buckets** – Create and name a bucket that stores data. Buckets are the fundamental container in Amazon S3 for data storage.
- **Store data in Buckets** – Store an infinite amount of data in a bucket. Upload as many objects as you like into an Amazon S3 bucket. Each object can contain up to 5 GB of data. Each object is stored and retrieved using a unique developer-assigned key.
- **Download data** – Download your data or enable others to do so. Download your data any time you like or allow others to do the same.
- **Permissions** – Grant or deny access to others who want to upload or download data into your Amazon S3 bucket. Grant upload and download permissions to three types of users. Authentication mechanisms can help keep data secure from unauthorized access.
- **Standard interfaces** – Use standards-based REST and SOAP interfaces designed to work with any Internet-development toolkit.

What can we do with S3 ?



How much data can I store in Amazon S3?

- The total volume of data and number of objects you can store are unlimited.
- Individual Amazon S3 objects can range in size from a minimum of 0 bytes to a maximum of 5 terabytes.
- The largest object that can be uploaded in a single PUT is 5 gigabytes.
- For objects larger than 100 megabytes, customers should consider using the Multipart Upload capability.
- For transferring large data use snowball

Amazon S3 Features

- **Low cost and Easy to Use −** Using Amazon S3, the user can store a large amount of data at very low charges.
- **Secure −** Amazon S3 supports data transfer over SSL and the data gets encrypted automatically once it is uploaded. The user has complete control over their data by configuring bucket policies using AWS IAM.
- **Scalable −** Using Amazon S3, there need not be any worry about storage concerns. We can store as much data as we have and access it anytime.
- **Durable-** It regularly verifies the integrity of data stored using checksums e.g. if S3 detects there is any corruption in data, it is immediately repaired with the help of replicated data.
- **Higher performance −** Amazon S3 is integrated with Amazon CloudFront, that distributes content to the end users with low latency and provides high data transfer speeds without any minimum usage commitments.
- **Integrated with AWS services −** Amazon S3 integrated with AWS services include Amazon CloudFront, Amazon CLoudWatch, Amazon Kinesis, Amazon RDS, Amazon Route 53, Amazon VPC, AWS Lambda, Amazon EBS, Amazon Dynamo DB, etc.

Bucket Restrictions

- **You can** create maximum of 100 buckets in each of your accounts
- You cant transfer ownerships of a bucket
- You can store unlimited number of objects in a bucket.
- You cant create a bucket within another bucket.

What kind of data can store in AWS S3?

You can store virtually any kind of data, in any format, in S3 and when we talk about capacity, the volume and the number of objects that we can store in S3 are **unlimited**.

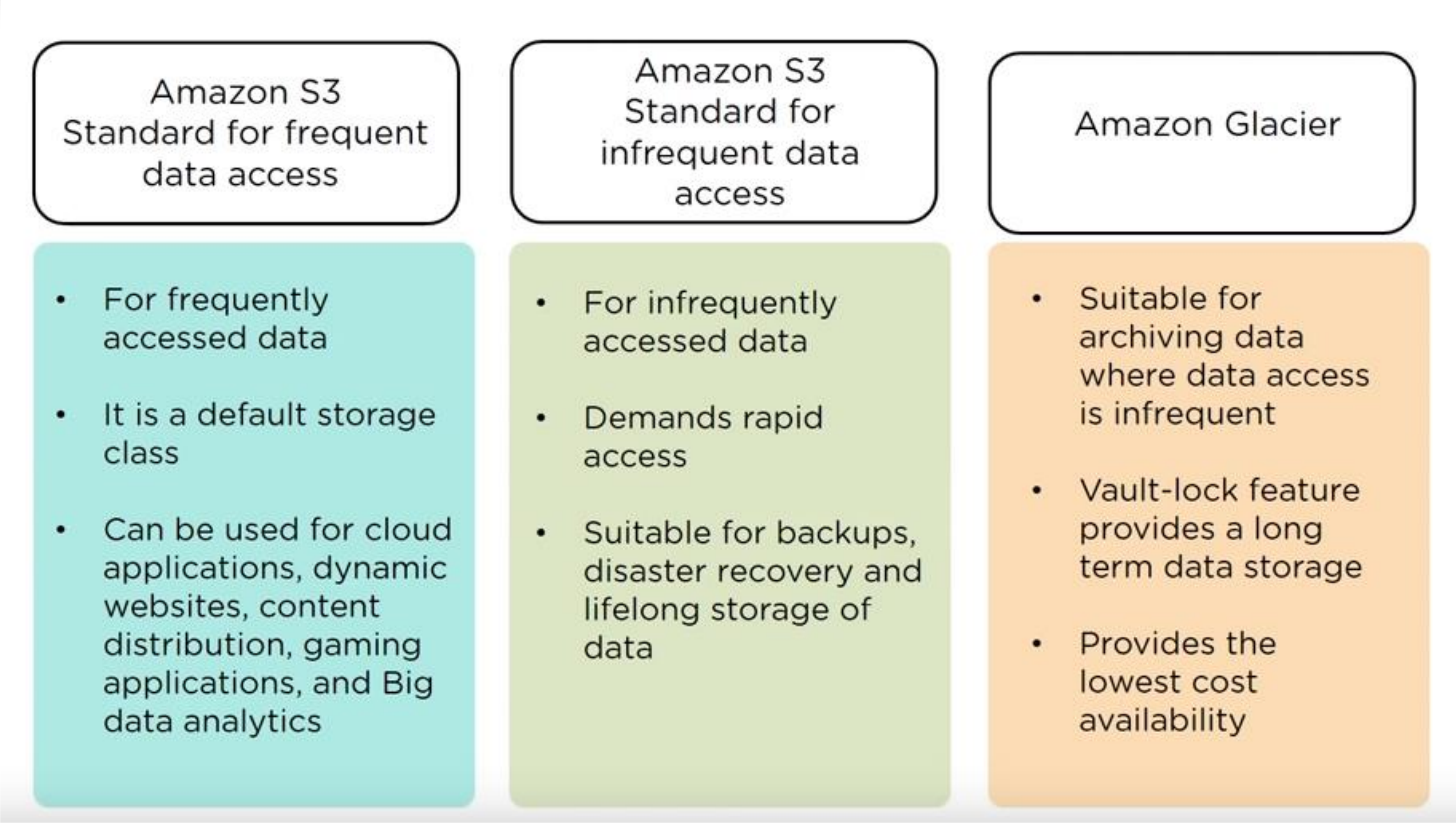
*An object is the fundamental entity in S3. It consists of data, key and metadata.

When we talk about data, it can be of two types-

- Data which is to be accessed frequently.
- Data which is accessed not that frequently.

Therefore, Amazon came up with 3 storage classes to provide its customers the best experience and at an affordable cost.

Storage Classes in S3



Amazon S3 Standard for frequent data access	Amazon S3 Standard for infrequent data access	Amazon Glacier
• For frequently accessed data • It is a default storage class • Can be used for cloud applications, dynamic websites, content distribution, gaming applications, and Big data analytics	• For infrequently accessed data • Demands rapid access • Suitable for backups, disaster recovery and lifelong storage of data	• Suitable for archiving data where data access is infrequent • Vault-lock feature provides a long term data storage • Provides the lowest cost availability

Let’s understand the 3 storage classes with a “health-care” use case:

Amazon S3 Standard for frequent data access

This is suitable for performance sensitive use cases where the latency should be kept low. e.g. in a hospital, frequently accessed data will be the data of admitted patients, which should be retrieved quickly.

Amazon S3 Standard for infrequent data access

This is suitable for use cases where the data is long lived and less frequently accessed, i.e for data archival but still expects high performance. e.g. in the same hospital, people who have been discharged, their records/data will not be needed on a daily basis, but if they return with any complication, their discharge summary should be retrieved quickly.

Amazon Glacier

Suitable for use cases where the data is to be archived, and high performance is not required, it has a lower cost than the other two services. e.g. in the hospital, patients’ test reports, prescriptions, MRI, X Ray, Scan docs etc. that are older than a year will not be needed in the daily run and even if it is required, lower latency is not needed.

S3 Storage Classes

S3 – Standard – Frequently accessed

- **Amazon S3** standard offers high durability, high availability, low latency and high performance object storage for general purpose use.
- 99.99 % availability
- 99.9999999 % durability
- Stored redundantly across multiple devices in multiple facilities and is designed to sustain the loss of 2 facilities concurrently.

S3 Storage Classes

S3 – IA(Infrequently accessed)

- **For data** that is infrequently accessed, but required rapid access when needed. Lower fee than s3, but you are charged for retrieval fee.
- Min Object size is 128 KB
- Designed for durability of 99.999999999 % of objects. 11 9’s
- Designed for 99.9 % availability over a given year.
- Lower price then s3 standard
- Designed for storing less frequently accessed data.
- Minimum duration 30 days
- Retrieval charges applicable.

S3 Storage Classes - RRS

S3 – Reduced Redundancy Storage (RRS)

Designed to provide 99.99 % durability and 99.99% availability of objects over a given year.

- Frequently accessed data but not critical

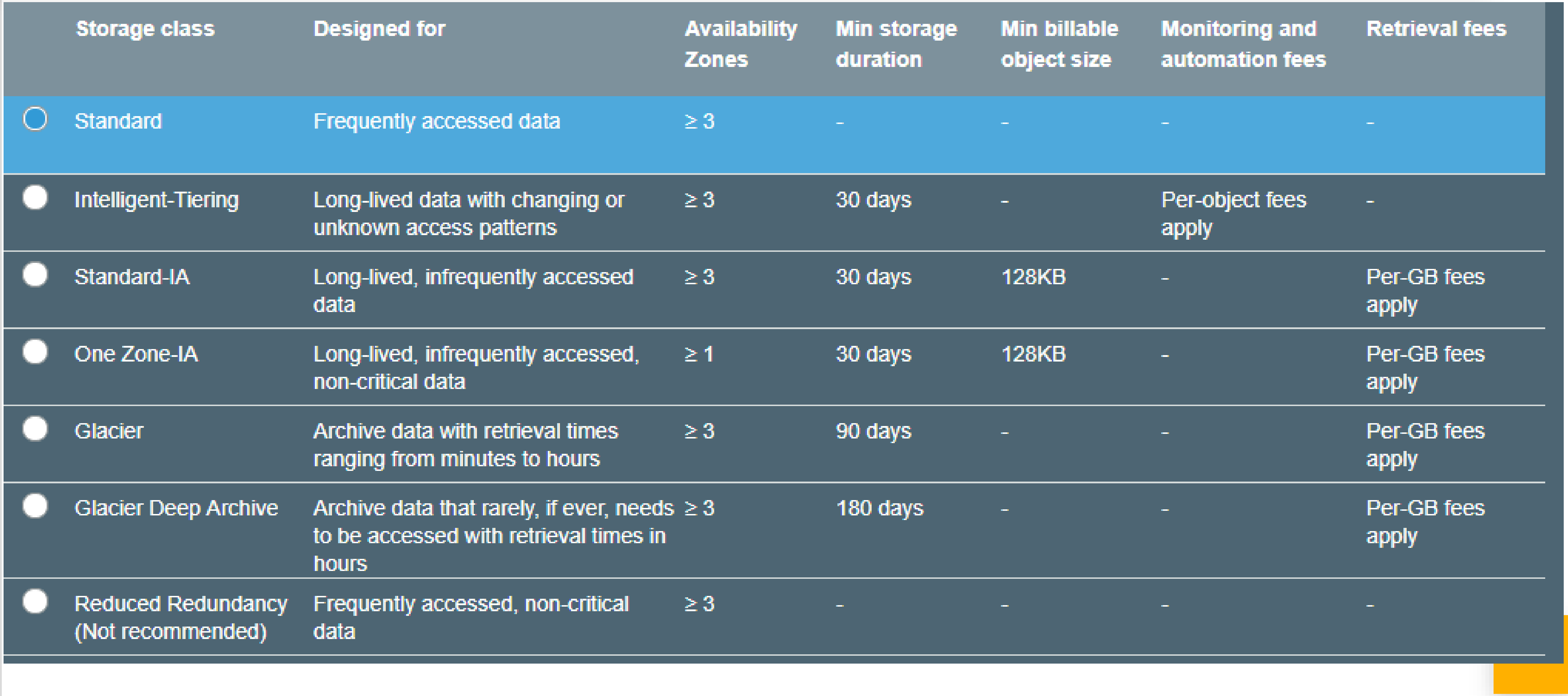
S3 Storage Classes - GLACIER

Glacier

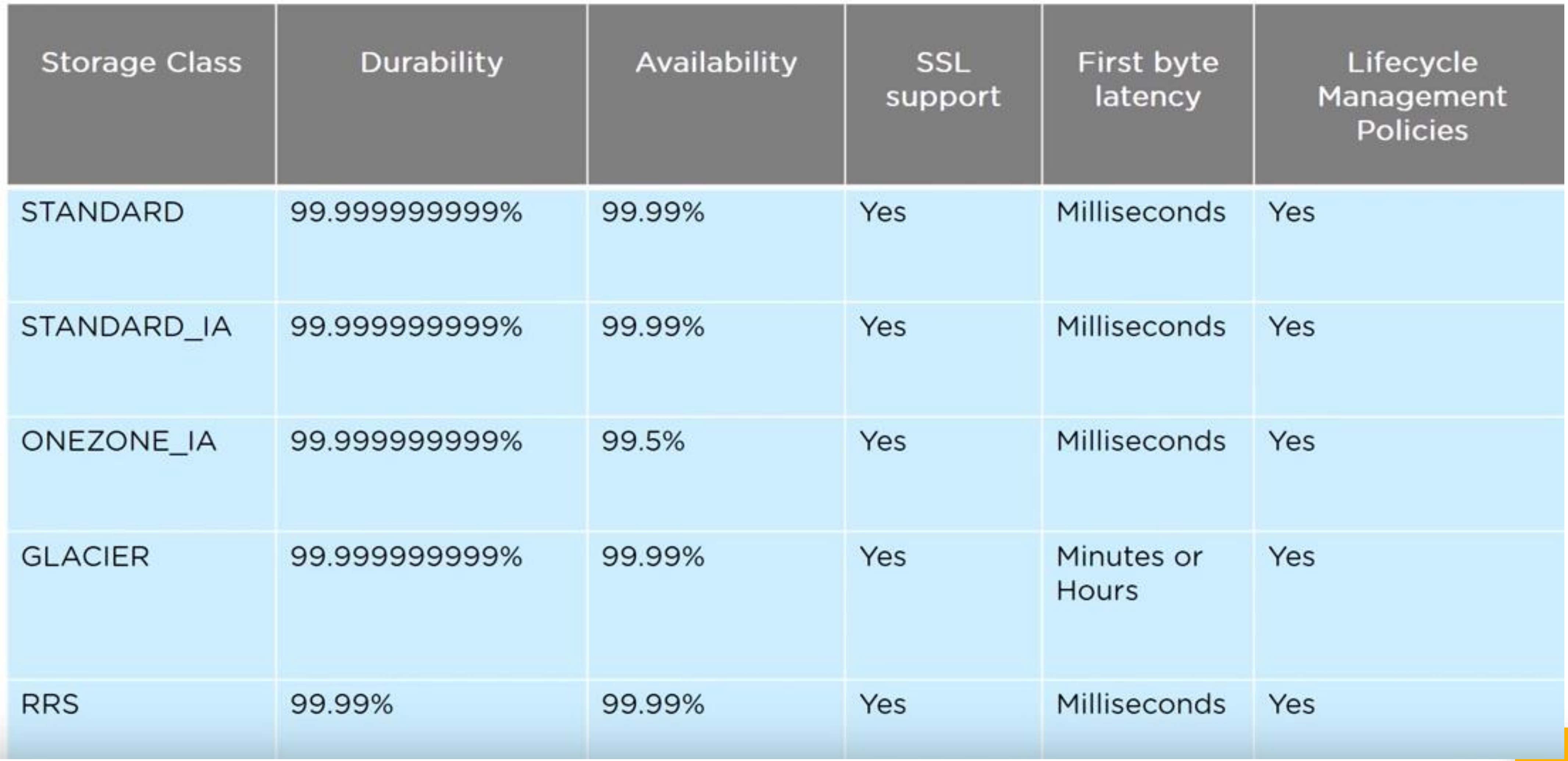
Amazon Glacier is an extremely low cost storage service that provides durable, secure, and flexible storage for data archiving and online backup. Storage class offers secure, durable and extremely low cost cloud storage for data that does not require real time access, such as archives and long term backups

- **Archives**: In Glacier, data is stored in archives. An archive can contain up to 40 TB of data, and you can have unlimited number of archives.
- **Vaults**: Vaults are containers for archives. Each AWS account can have up to 1000 vaults.
- After requesting for data, three to five hours later, the glacier object is copied to S3 RRS.
- Glacier allows you to retrieve up to 5% of the Amazon S3 data stored in Glacier for free each month.

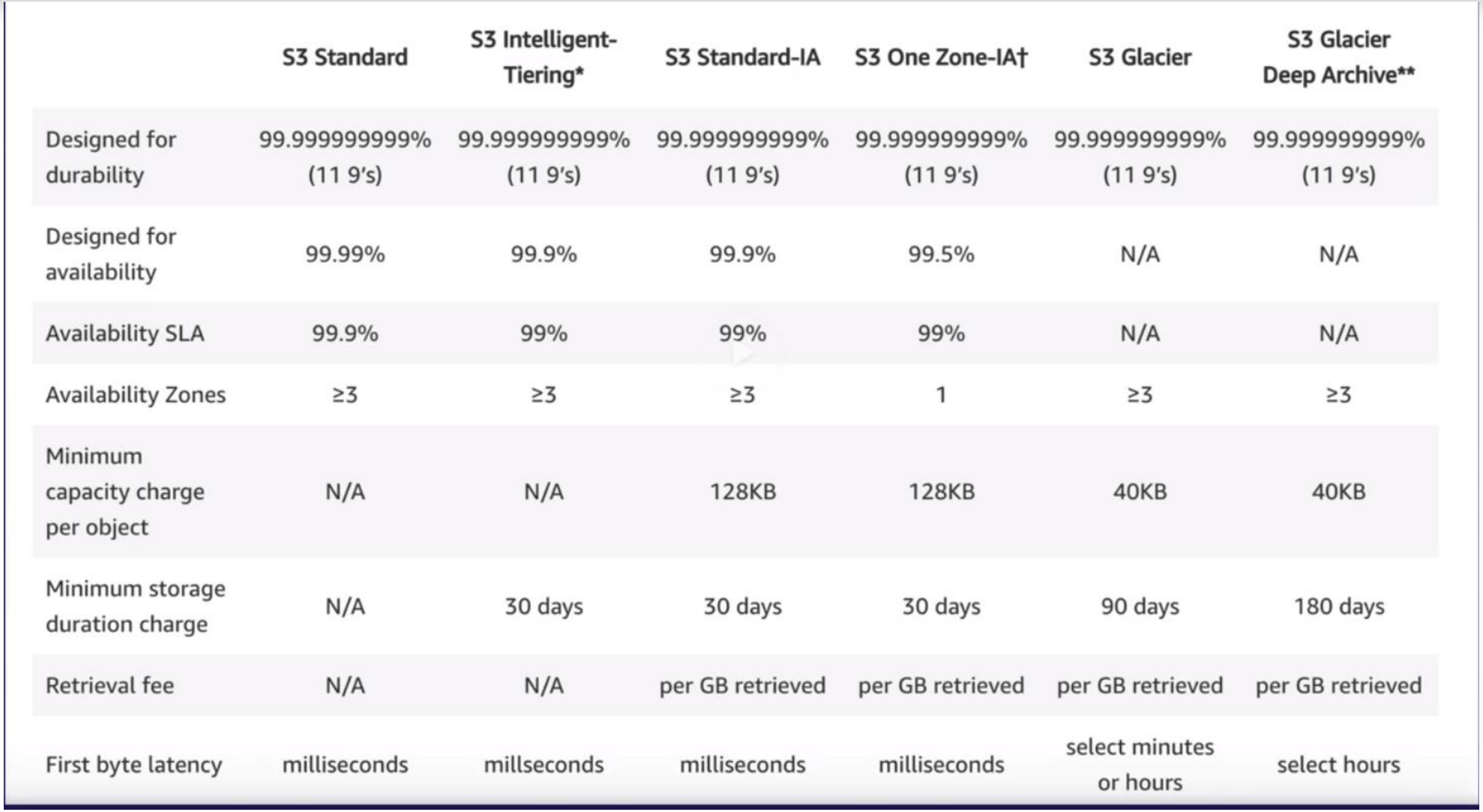
S3 Storage Classes



	Storage class	Designed for	Availability Zones	Min storage duration	Min billable object size	Monitoring and automation fees	Retrieval fees
○	Standard	Frequently accessed data	≥ 3	-	-	-	-
○	Intelligent-Tiering	Long-lived data with changing or unknown access patterns	≥ 3	30 days	-	Per-object fees apply	-
○	Standard-IA	Long-lived, infrequently accessed data	≥ 3	30 days	128KB	-	Per-GB fees apply
○	One Zone-IA	Long-lived, infrequently accessed, non-critical data	≥ 1	30 days	128KB	-	Per-GB fees apply
○	Glacier	Archive data with retrieval times ranging from minutes to hours	≥ 3	90 days	-	-	Per-GB fees apply
○	Glacier Deep Archive	Archive data that rarely, if ever, needs to be accessed with retrieval times in hours	≥ 3	180 days	-	-	Per-GB fees apply
○	Reduced Redundancy (Not recommended)	Frequently accessed, non-critical data	≥ 3	-	-	-	-



Storage Class	Durability	Availability	SSL support	First byte latency	Lifecycle Management Policies
STANDARD	99.999999999%	99.99%	Yes	Milliseconds	Yes
STANDARD_IA	99.999999999%	99.99%	Yes	Milliseconds	Yes
ONEZONE_IA	99.999999999%	99.5%	Yes	Milliseconds	Yes
GLACIER	99.999999999%	99.99%	Yes	Minutes or Hours	Yes
RRS	99.99%	99.99%	Yes	Milliseconds	Yes



	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier	S3 Glacier Deep Archive**
Designed for durability	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
Designed for availability	99.99%	99.9%	99.9%	99.5%	N/A	N/A
Availability SLA	99.9%	99%	99%	99%	N/A	N/A
Availability Zones	≥3	≥3	≥3	1	≥3	≥3
Minimum capacity charge per object	N/A	N/A	128KB	128KB	40KB	40KB
Minimum storage duration charge	N/A	30 days	30 days	30 days	90 days	180 days
Retrieval fee	N/A	N/A	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved
First byte latency	milliseconds	millseconds	milliseconds	milliseconds	select minutes or hours	select hours

Bucket Policies

- Bucket policies provide centralized access control to buckets and objects based on a variety of conditions,

- Including Amazon S3 operations, requesters, resources, and aspects of the request (e.g., IP address). The policies are expressed in our access policy language and enable centralized management of permissions.

- The permissions attached to a bucket apply to all of the objects in that bucket.

Versioning

Versioning helps to protect your data against accidental or malicious deletion by keeping multiple versions of each object in the bucket, identified by a unique version ID

- Versioning is a great backup tool.
- Versioning is turned on at the bucket level
- Once enabled, versioning cannot be removed from a bucket, it can only be suspended
- If you enable versioning you will get current version files and previous version files in your bucket.
- If you delete current version file, it will overwrite with a Delete Marker, if you want to get that object back to your s3 bucket, you can delete the delete marker.
- Versioning’s MFA Delete capability

AWS S3 Versioning

Object Versioning

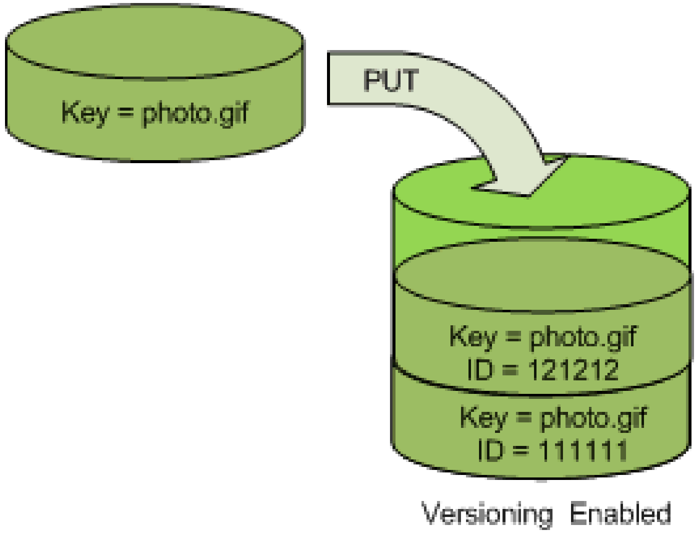
Use versioning to keep multiple versions of an object in one bucket. For example, you could store my-image.jpg(version 1111111) and my-image.jpg(version 222222) in a single bucket. Versioning protect you from the consequences of unintended overwrites and deletions. You can also use versioning to archive objects so you have access to previous versions.

- You must explicitly enable versioning on your bucket. By default, versioning is disabled. Regardless of whether you have enabled versioning, each object in your bucket has a version ID.
- If you have not enabled versioning, Amazon S3 sets the value of the version ID to null. If you have enabled versioning, Amazon S3 assigns a unique version ID value for the object. When you enable versioning on a bucket, objects already stored in the bucket are unchanged. The version IDs (null), contents, and permissions remain the same.

AWS S3 Versioning

Object Versioning

You PUT an object in a versioning-enabled bucket, the noncurrent version is not overwritten. The following figure shows that when a new version of photo.gif is PUT into a bucket that already contains an object with the same name, the original object (ID = 111111) remains in the bucket, Amazon S3 generates a new version ID (121212), and adds the newer version to the bucket

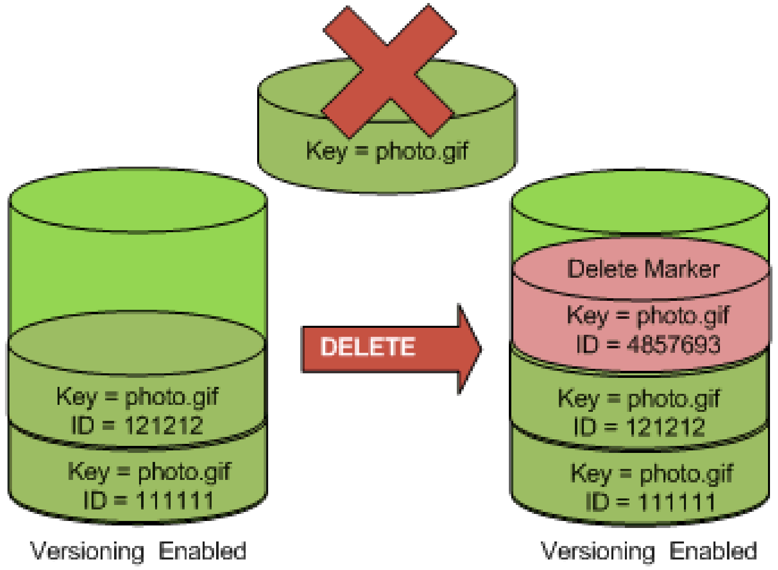


AWS S3 Versioning

Object Versioning

This functionality prevents you from accidentally overwriting or deleting objects and affords you the opportunity to retrieve a previous version of an object.

When you DELETE an object, all versions remain in the bucket and Amazon S3 inserts a delete marker, as shown in the following figure.



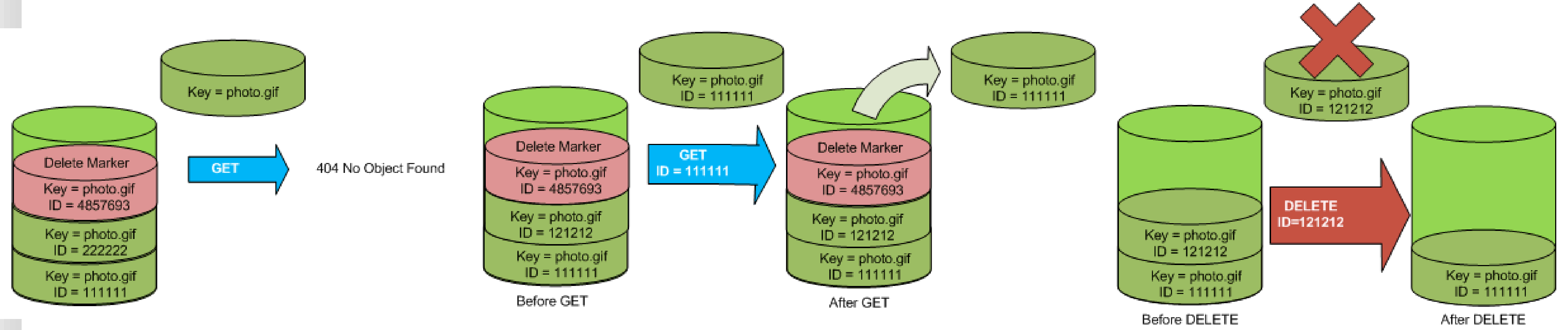
AWS S3 Versioning

Object Versioning

The delete marker becomes the current version of the object. By default, GET requests retrieve the most recently stored version. Performing a simple GET Object request when the current version is a delete marker returns a 404 Not Found error, as shown in the following figure.

You can, however, GET a noncurrent version of an object by specifying its version ID. In the following figure, we GET a specific object version, 111111. Amazon S3 returns that object version even though it's not the current version.

You can permanently delete an object by specifying the version you want to delete. Only the owner of an Amazon S3 bucket can permanently delete a version. The following figure shows how DELETE versionId permanently deletes an object from a bucket and that Amazon S3 doesn't insert a delete marker.



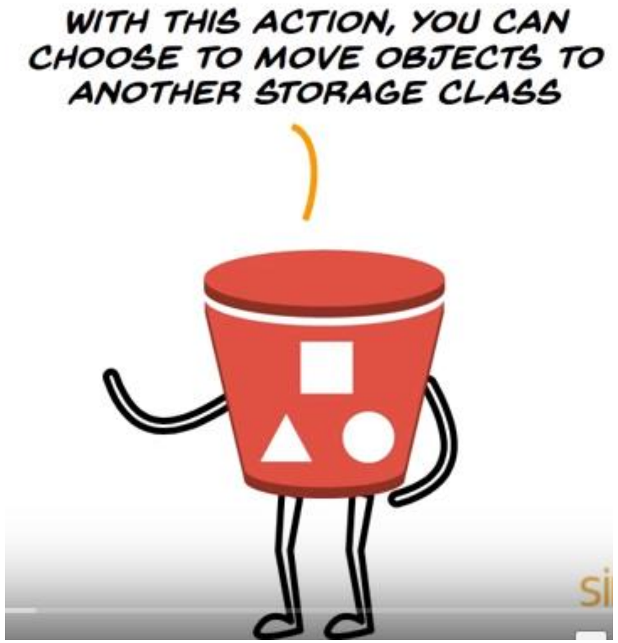
Life Cycle Management

In Life Cycle management, Amazon S3 applies a set of rules that define actions to a group of objects.

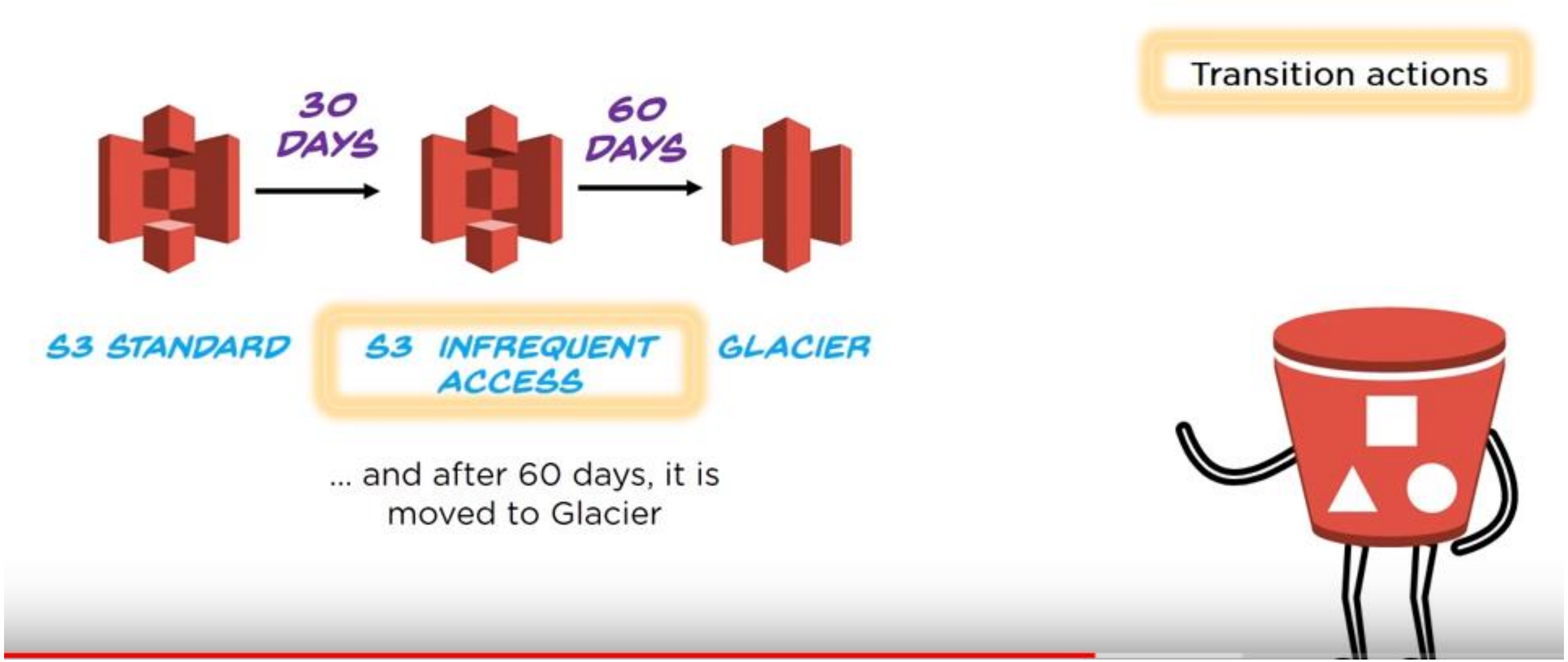
By using Life Cycle management we can automate the storage tiers in s3 buckets. We can move objects from one storage class/tier to another storage class/tier based on our business requirement

There are 2 features

1. **Transition actions**: from one storage tier to another
2. **Expiration actions**: delete after certain time.


WITH THIS ACTION, YOU CAN CHOOSE TO MOVE OBJECTS TO ANOTHER STORAGE CLASS
Si

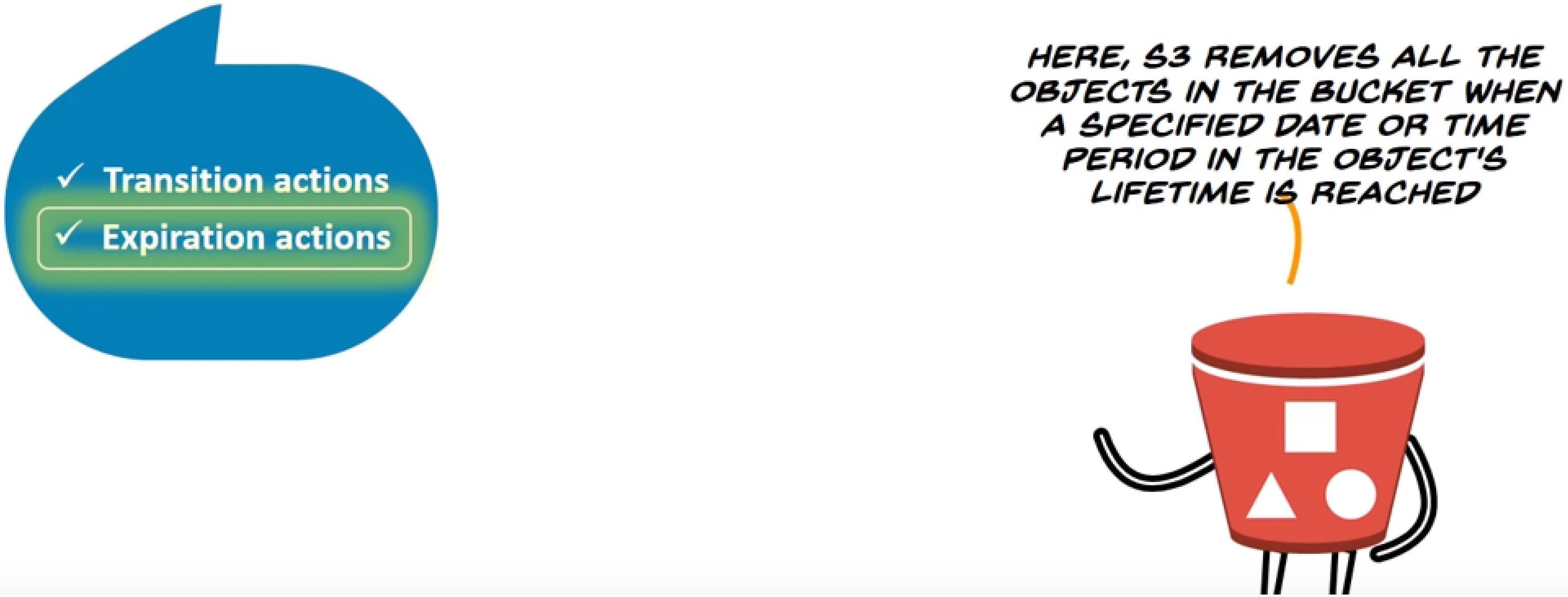
Life Cycle Management



... and after 60 days, it is moved to Glacier

Life Cycle Management

In Lifecycle management, Amazon S3 applies set of rules to a group of objects


✓ Transition actions
✓ Expiration actions
HERE, S3 REMOVES ALL THE OBJECTS IN THE BUCKET WHEN A SPECIFIED DATE OR TIME PERIOD IN THE OBJECT'S LIFETIME IS REACHED

Logging

By enabling logs we can track requests on your Amazon S3 bucket. Logging is off by default. You can enable it from bucket properties.

Every log will contain the below information

- Requestor account and IP address
- Bucket Name
- Request time
- Action (GET, PUT, LIST etc)
- Response status or error code.

Object Lock

Amazon S3 Object Lock enables you to store objects using a "Write Once Read Many" (WORM) model. Using S3 Object Lock, you can prevent an object from being deleted or overwritten for a fixed amount of time or indefinitely.

S3 Object Lock provides two ways to manage object retention: **retention periods** and **legal holds**.

- **A retention period** specifies a fixed period of time during which an object remains locked. During this period, your object will be WORM-protected and can't be overwritten or deleted.
- **A legal hold provides** the same protection as a retention period, but has no expiration date. Instead, a legal hold remains in place until you explicitly remove it.

S3 Object Lock works only in versioned buckets.

Cross-Origin Resource Sharing (CORS)

Enabling one S3 bucket to access another S3 bucket when you want to share assets requires you to setup Cross-origin resource sharing (CORS).

For example, you can configure a website of a particular domain to load content from another bucket of a different domain. This enables you to build rich client-side web applications.

Cross Region Replication

With cross region replication Amazon S3 allows you to asynchronously replicate all new objects in the source bucket in one AWS region to a target bucket in another region.

- Versioning must be enabled on both the source and destination buckets.
- Regions must be unique
- Files in an existing bucket are not replicated automatically, All subsequent/Future updated files will be replicated automatically.
- You cannot replicate to multiple buckets or user daisy chaining(at this time).
- Delete markers are not replicated
- Deleting individual versions or delete markers will not be replicated.
- Cross-region replication is used to reduce the latency required to access objects in Amazon S3 by placing objects close to a set of users or to meet requirements to store backup data at a certain distance from the original source data.
- Amazon s3 must have permission to replicate objects from that source bucket to the destination bucket on your behalf.

Amazon S3 Transfer Acceleration

Amazon S3 Transfer acceleration enables fast, easy and secure transfers of files over long distances between your client and an s3 bucket.

Transfer acceleration takes advantage of Amazon CloudFront’s globally distributed edge locations.

As the data arrives at an edge location , data is routed to Amazon S3 over an optimized network path. Charges apply

- By using the Amazon S3 Transfer Acceleration Speed Comparison tool we can compare the accelerated and non accelerated upload speeds across s3 regions
- The speed Comparison tool uses multipart uploads to transfer a file from your browser to various S3 regions with and without using Transfer acceleration.

Amazon S3 Events

Amazon S3 Event notifications can be sent in response to the actions taken on objects uploaded or stored in Amazon S3. The Amazon S3 notification feature enables you to receive notifications when certain events happen in your buckets.

- Notification messages can be sent through either Amazon SNS or SQS or directly to AWS Lambda functions.

Amazon S3 Inventory & Requester pays

Amazon S3 Inventory is one of the tools Amazon S3 provides to help manage your storage. Amazon S3 inventory provides a comma-separated values (CSV) flat file output of your objects and their corresponding metadata on a daily or weekly basis for an s3 bucket or a shared prefix.

Amazon S3 Requestor Pays

Generally, bucket owners pay for all Amazon S3 storage and data transfer costs associated with their bucket. If you enable Requester pays on the bucket, instead of bucket owner requested user will pay.

Anonymous access to that bucket is not allowed, if we want to enable the requester pays on bucket.

Amazon S3 Security & Encryption

BY Default , all newly created buckets are PRIVATE, you can setup access control to your buckets using

Bucket Policies

Access Control lists

S3 Bucket can be configured to create access logs which log all requests made to the bucket. This can be sent to another bucket and even another bucket in another account

Encryption IN TRANSIT is achieved by SSL/TLS

Encryption At REST is achieved by (NEXT SLIDE)

Amazon S3 Encryption

Amazon S3 Encryption has three types of encryptions.

Server-Side Encryption: All SSE performed by Amazon S3 and AWS Key Management Service uses the 256 bit Advanced Encryption Standard (AES)

SSE – S3 (AWS Managed Keys)

SSE – KMS (AWS KMS Keys)

SSE – C (Customer provided keys).

Client Side Encryption:

We cannot encrypt the data on the client before sending it to Amazon S3. We have to take care about the encryption and Decryption process.

In-Transit Encryption:

We can use SSL API endpoints, this ensure that all data send to and from Amazon S3 is encrypted while in transit using the HTTPS protocol.

AWS S3 Features

Common Operations

- **Create a Bucket** – Create and name your own bucket in which to store your objects.
- **Write an Object** – Store data by creating or overwriting an object. When you write an object, you specify a unique key in the namespace of your bucket. This is also a good time to specify any access control you want on the object.
- **Read an Object** – Read data back. You can download the data via HTTP or BitTorrent.
- **Deleting an Object** – Delete some of your data.
- **Listing Keys** – List the keys contained in one of your buckets. You can filter the key list based on a prefix.

How does data consistency work for S3?

- **Read** after write consistency for PUTS of new objects
- Eventual consistency for overwrites PUTS and DELETES (can take some time to propagate)

In Other Words

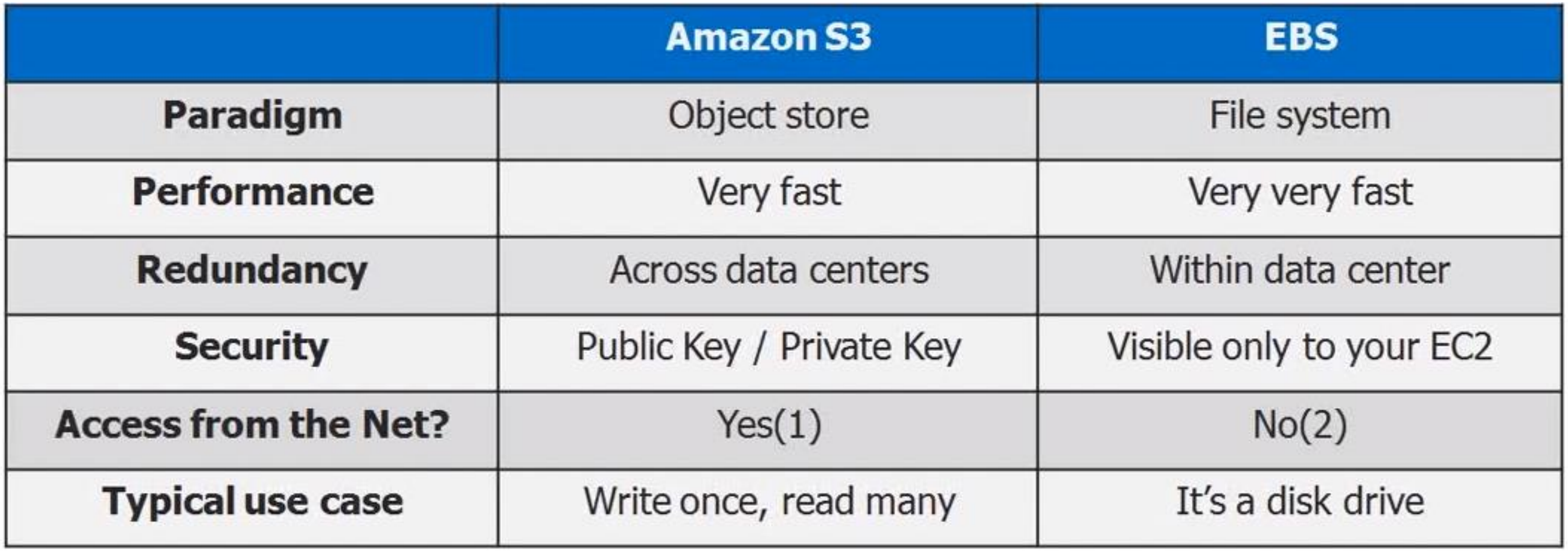
- If you write a new file and read it immediately afterwards, you will be able to view that data.

- If you update AN EXISTING file or delete a file and read it immediately, you may get the older version or you may not. Basically changes to a objects can take a little but of the time to propagate.

You are charged for S3 in following ways

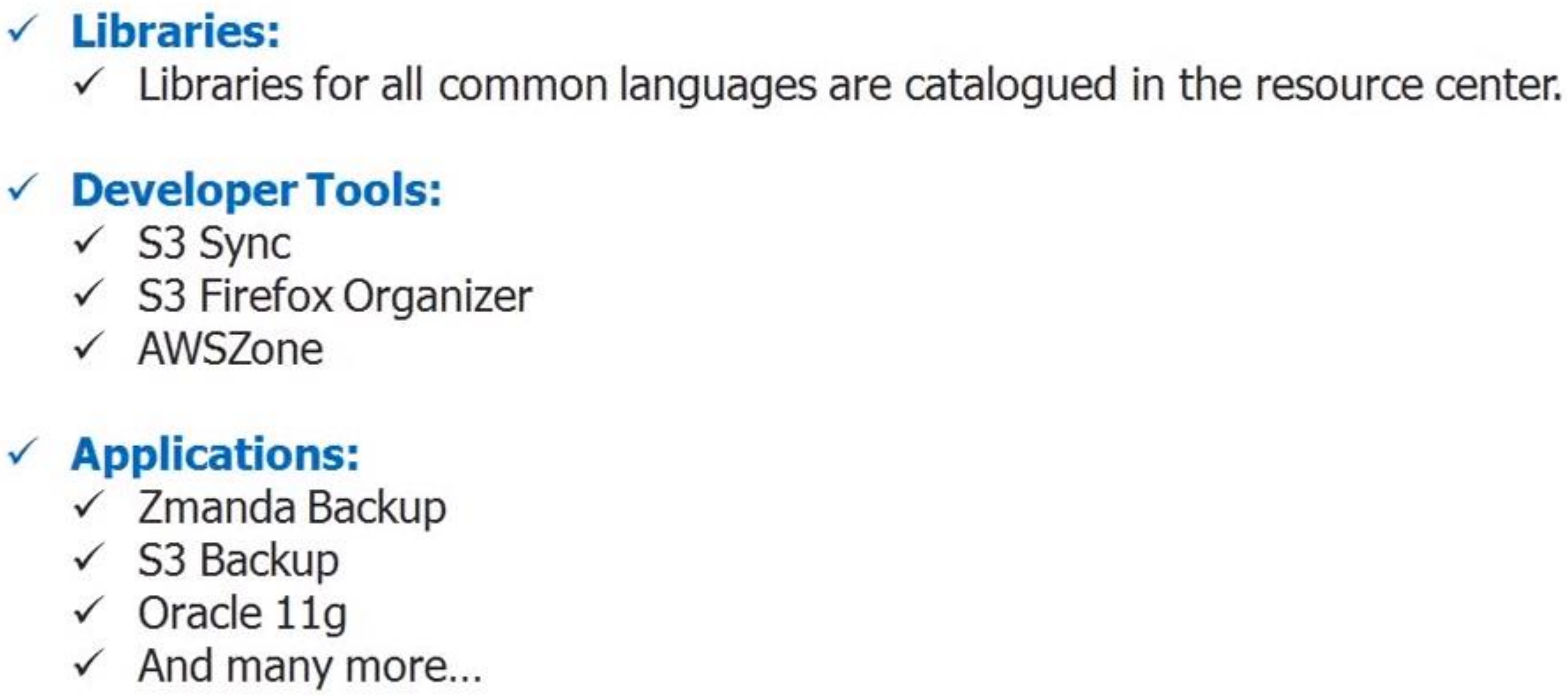
- **Storage**
- **Requests**
- **Storage Management pricing**
- **Data Transfer pricing**
- **Transfer acceleration**
- **Cross region replication**

S3 vs EBS



	Amazon S3	EBS
Paradigm	Object store	File system
Performance	Very fast	Very very fast
Redundancy	Across data centers	Within data center
Security	Public Key / Private Key	Visible only to your EC2
Access from the Net?	Yes(1)	No(2)
Typical use case	Write once, read many	It’s a disk drive

Backups and Tools



- ✓ **Libraries:**
  - ✓ Libraries for all common languages are catalogued in the resource center.
- ✓ **Developer Tools:**
  - ✓ S3 Sync
  - ✓ S3 Firefox Organizer
  - ✓ AWSZone
- ✓ **Applications:**
  - ✓ Zmanda Backup
  - ✓ S3 Backup
  - ✓ Oracle 11g
  - ✓ And many more…

Exam Tips

- **S3** is Object based i.e. allows you to upload, download, share files.
- All your objects reside in container called Buckets.
- S3 is universal namespace that means name of your bucket must be unique globally.
- Amazon s3 is cloud object storage, instead of being closely associated with a server, S3 storage is independent of a server and is accessed over the internet.
- You can create an use multiple buckets, you can have up to 100 per account by default, you can get it increased by creating ticket with AWS.
- File size can be from 0 Byte to 5TB.
- Single bucket can store an unlimited number of files.
- You can create buckets in your nearby region which is located close to a particular set of end user or customers in order to minimize latency.
- Bucket names must be at least 3 or no more than 63 chars long.
- Bucket names must not be formatted as an IP address.
- Successful uploads will generate a HTTP 200 status code.
- You can turn on MFA Delete.

Exam Tips

- **Key** (This is simply the name of the object)
- **Value**(This is simply the data and is made up of a sequence of bytes)
- **Version ID** (Important for versioning)
- **Metadata** (Data about data that you are storing)