# **S3 Object Lifecycle Management**

* S3 Object lifecycle can be managed by using a lifecycle configuration, which defines how S3 manages objects during their lifetime.
* Lifecycle configuration enables simplification of object lifecycle management, for e.g. moving of less frequently access objects, backup or archival of data for several years or permanent deletion of objects,
* S3 controls all transitions automatically
* Lifecycle Management rules applied to a bucket are applicable to all the existing objects in the bucket as well as the ones that will be added anew
* S3 Object lifecycle management allows 2 types of behaviour
  + **Transition** - the storage class for the objects change
  + **Expiration** - the objects expire and are permanently deleted
* Lifecycle Management can be configured with [Versioning](https://jayendrapatil.com/aws-s3-object-versioning/), which allows storage of one current object version and zero or more non current object versions
* Object’s lifecycle management applies to both Non Versioning and Versioning enabled buckets
* For Non Versioned buckets
  + Transitioning period is considered from the **object’s creation date**
* For Versioned buckets,
  + Transitioning period for current object is calculated for the object creation date
  + Transitioning period for non current object is calculated for the date **when the object became a noncurrent versioned object**
  + S3 uses the number of days since its successor was created as the number of days an object is noncurrent.

**Different Storage Classes on S3 Objects**

Storage Classes for Frequently accessed objects

**S3 Standard -** The default storage class.

**Reduced Redundancy** - The Reduced Redundancy Storage (RRS) storage class is designed for noncritical, reproducible data that can be stored with less redundancy than the S3 Standard storage class.

Storage Class for automatically Optimized Data

**S3 Intelligent-Tiering -** Amazon S3 storage class designed to optimize storage costs by automatically moving data to the most cost-effective access tier, without operational overhead. These objects are stored automatically in the Frequent Access tier

Storage Class for Infrequently accessed Objects

These storage classes are designed for long-lived and infrequently accessed data. IA stands for Infrequent access. Used for Storing backups

**S3 Standard-IA -** Amazon S3 stores the object data redundantly across multiple geographically separated Availability Zones (similar to the S3 Standard storage class). S3 Standard-IA objects are resilient to the loss of an Availability Zone. This storage class offers greater availability and resiliency than the S3 One Zone-IA class.

**S3 One Zone-IA -** Amazon S3 stores the object data in only one Availability Zone, which makes it less expensive than S3 Standard-IA. However, the data is not resilient to the physical loss of the Availability Zone resulting from disasters, such as earthquakes and floods. The S3 One Zone-IA storage class is as durable as Standard-IA, but it is less available and less resilient.

Storage Class for archiving Objects

There storage classes are designed for low-cost data archiving. These storage classes offer the same durability and resiliency as the S3 Standard storage class

**S3 Glacier** – Use for archives where portions of the data might need to be retrieved in minutes. Data stored in the S3 Glacier storage class has a minimum storage duration period of 90 days and can be accessed in as little as 1-5 minutes using expedited retrieval. If you have deleted, overwritten, or transitioned to a different storage class an object before the 90-day minimum, you are charged for 90 days

**S3 Glacier Deep Archive -** Use for archiving data that rarely needs to be accessed. Data stored in the S3 Glacier Deep Archive storage class has a minimum storage duration period of 180 days and a default retrieval time of 12 hours. If you have deleted, overwritten, or transitioned to a different storage class an object before the 180-day minimum, you are charged for 180 days

Supported lifecycle transitions for objects stored in S3 are listed below

1. S3 Standard storage class –> other storage class.
2. Any storage class –> S3 Glacier or S3 Glacier Deep Archive storage classes.
3. S3 Standard-IA –> S3 Intelligent-Tiering or S3 One Zone-IA
4. S3 Intelligent-Tiering storage class –> S3 One Zone-IA storage class.
5. S3 Glacier storage class –> S3 Glacier Deep Archive storage class.

### Unsupported lifecycle transitions

1. Any storage class X S3 Standard
2. Any storage class X Reduced Redundancy
3. S3 Intelligent-Tiering X S3 Standard-IA
4. S3 One Zone-IA  X S3 Standard-IA or S3 Intelligent-Tiering

Lifecycle transitions Constraints

1. STANDARD -> (128 KB & 30 days) -> STANDARD-IA or One Zone-IA or S3 Intelligent-Tiering
   * **Larger Objects** – Only objects with size more than 128 KB can be transitioned, as cost benefits for transitioning to STANDARD-IA or One Zone-IA can be realized only for larger objects
   * **Minimum 30 days** – Objects must be stored for at least 30 days in the current storage class before being transitioned to the STANDARD-IA or One Zone-IA, as younger objects are accessed more frequently or deleted sooner than is suitable for STANDARD-IA or One Zone-IA
2. GLACIER -> (90 days) -> Permanent Deletion OR GLACIER Deep Archive -> (180 days) -> Permanent Deletion
   * Deleting data that is archived to Glacier is free, if the objects deleted are archived for three months or longer.
   * S3 charges a prorated early deletion fee, if the object is deleted or overwritten within three months of archiving it.
3. Archival of objects to Glacier by using object lifecycle management is performed asynchronously and there may be a delay between the transition date in the lifecycle configuration rule and the date of the physical transition. However, AWS charges Glacier prices based on the transition date specified in the rule
4. For a versioning-enabled bucket
   * Transition and Expiration actions apply to current versions.
   * NoncurrentVersionTransition and NoncurrentVersionExpiration actions apply to noncurrent versions and works similar to the non versioned objects except the time period is from the time the objects became noncurrent
5. Expiration Rules

For Non Versioned bucket

* + Object is permanently deleted

For Versioned bucket

* + Expiration is applicable to the Current object only and does not impact any of the non current objects
  + S3 will insert a Delete Marker object with unique id and the previous current object becomes a non current version
  + S3 will not take any action if the Current object is a Delete Marker
  + If the bucket has a single object which is the Delete Marker (referred to as expired object delete marker), S3 removes the Delete Marker

For Versioned Suspended bucket

* + S3 will insert a Delete Marker object with version ID null and overwrite the any object with version ID null

1. When an object reaches the end of its lifetime, S3 queues it for removal and removes it asynchronously. There may be a delay between the expiration date and the date at which S3 removes an object. Charged for storage time associated with an object that has expired are not incurred.
2. Cost is incurred if objects are expired in STANDARD-IA before 30 days, GLACIER before 90 days and GLACIER\_DEEP\_ARCHIVE before 180 days.

## Creating a lifecycle configuration

An S3 Lifecycle configuration is an XML file that consists of a set of rules with predefined actions that you want Amazon S3 to perform on objects during their lifetime.

You can also configure the lifecycle by using the Amazon S3 console, REST API, AWS SDKs and AWS CLI. For more information, see [Setting lifecycle configuration on a bucket](https://docs.aws.amazon.com/AmazonS3/latest/userguide/how-to-set-lifecycle-configuration-intro.html).

Amazon S3 provides a set of REST API operations for managing lifecycle configuration on a bucket. Amazon S3 stores the configuration as a lifecycle subresource that is attached to your bucket. For details, see the following:

[PUT Bucket lifecycle](https://docs.aws.amazon.com/AmazonS3/latest/API/RESTBucketPUTlifecycle.html)

[GET Bucket lifecycle](https://docs.aws.amazon.com/AmazonS3/latest/API/RESTBucketGETlifecycle.html)

[DELETE Bucket lifecycle](https://docs.aws.amazon.com/AmazonS3/latest/API/RESTBucketDELETElifecycle.html)