**Versioning in S3**

Amazon S3 Versioning to keep multiple versions of an object in one bucket.

**Ex:** you could store my-image.jpg (version 111111) and my-image.jpg (version 222222) in a single bucket.

S3 Versioning protects you from the consequences of unintended overwrites and deletions.

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

**Note:** You must explicitly enable S3 Versioning on your bucket. By default, S3 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 S3 Versioning is enabled, Amazon S3 assigns a version ID value for the object. This value distinguishes it from other versions of the same key.

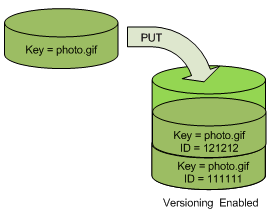
Enabling and suspending versioning is done at the bucket level.

When you enable versioning on an existing bucket, objects that are already stored in the bucket are unchanged. The version IDs (null), contents, and permissions remain the same.

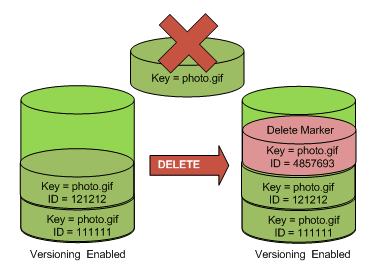
After you enable S3 Versioning for a bucket, each object that is added to the bucket gets a version ID, which distinguishes it from other versions of the same key.

**Note:** Only Amazon S3 generates version IDs, and they can’t be edited. Version IDs are Unicode, UTF-8 encoded, URL-ready, opaque strings that are no more than 1,024 bytes long.

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.

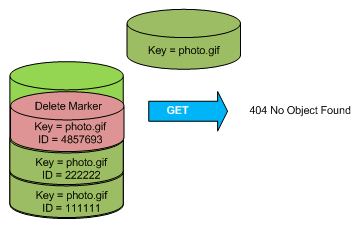


When you DELETE an object, all versions remain in the bucket and Amazon S3 inserts a delete marker, as shown in the following figure.



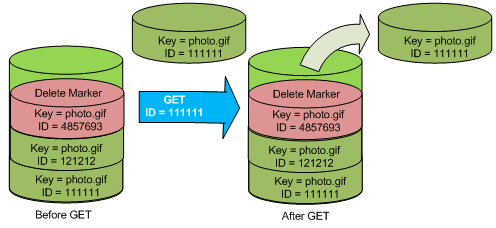
The delete marker becomes the current version of the object. By default, GET requests retrieve the most recently stored version.

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.



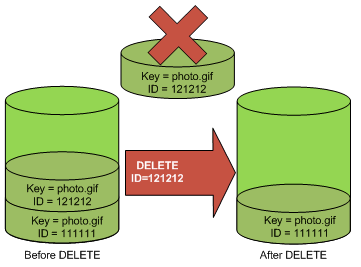
However, you can GET a noncurrent version of an object by specifying its version ID.

In the following figure, you 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.

The following figure shows how DELETE versionId permanently deletes an object from a bucket and that Amazon S3 doesn't insert a delete marker.



**Note:** Only the owner of an Amazon S3 bucket can permanently delete a version.

**Note:** You can add additional security by configuring a bucket to enable MFA (multi-factor authentication) Delete. When you do, the bucket owner must include two forms of authentication in any request to delete a version or change the versioning state of the bucket.