Hitachi Content Platform for Cloud Scale

S3 Console Guide

This document contains information and guidance on using the S3 Console to manage buckets and objects stored through Hitachi Content Platform for cloud scale (HCP for cloud scale).

MK-HCPCS009-03
September 2021

Hitachi reserves the right to make changes to this Material at any time without notice and assumes no responsibility for its use. The Materials contain the most current information available at the time of publication.

Some of the features described in the Materials might not be currently available. Refer to the most recent product announcement for information about feature and product availability, or contact Hitachi Vantara LLC at https://support.hitachivantara.com/en_us/contact-us.html.

Notice: Hitachi products and services can be ordered only under the terms and conditions of the applicable Hitachi agreements. The use of Hitachi products is governed by the terms of your agreements with Hitachi Vantara LLC.

By using this software, you agree that you are responsible for:

1. Acquiring the relevant consents as may be required under local privacy laws or otherwise from authorized employees and other individuals; and
2. Verifying that your data continues to be held, retrieved, deleted, or otherwise processed in accordance with relevant laws.

Notice on Export Controls. The technical data and technology inherent in this Document may be subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. Reader agrees to comply strictly with all such regulations and acknowledges that Reader has the responsibility to obtain licenses to export, re-export, or import the Document and any Compliant Products.

Contents

Preface

This document describes and provides instructions for using the S3 Console software on Hitachi Content Platform for cloud scale systems.

Please read this document carefully to understand how to use these products, and maintain a copy for your reference.

Intended audience

This document is intended for consumers who use HCP for cloud scale as a way to store and retrieve objects in S3 buckets.

Product version

This document revision applies to S3 Console 2.3.0.

Document conventions

This document uses the following typographic conventions:

Convention	Description
Bold	▪ Indicates text in a window, including window titles, menus, menu options, buttons, fields, and labels. Example: Click **OK**. ▪ Indicates emphasized words in list items.
Italic	▪ Indicates a document title or emphasized words in text. ▪ Indicates a variable, which is a placeholder for actual text provided by the user or for output by the system. Example: `pairdisplay -g` *`group`* (For exceptions to this convention for variables, see the entry for angle brackets.)

Convention	Description
`Monospace`	Indicates text that is displayed on screen or entered by the user. Example: `pairdisplay -g oradb`
< > angle brackets	Indicates variables in the following scenarios: ▪ Variables are not clearly separated from the surrounding text or from other variables. Example: `Status-<report-name><file-version>.csv` ▪ Variables in headings.
[ ] square brackets	Indicates optional values. Example: [a | b] indicates that you can choose a, b, or nothing.
{ } braces	Indicates required or expected values. Example: { a | b } indicates that you must choose either a or b.
| vertical bar	Indicates that you have a choice between two or more options or arguments. Examples: [a | b] indicates that you can choose a, b, or nothing. { a | b } indicates that you must choose either a or b.

This document uses the following icons to draw attention to information:

Icon	Label	Description
🗎	Note	Calls attention to important or additional information.
💡	Tip	Provides helpful information, guidelines, or suggestions for performing tasks more effectively.
⚠	Caution	Warns the user of adverse conditions and/or consequences (for example, disruptive operations, data loss, or a system crash).
⚠	WARNING	Warns the user of a hazardous situation which, if not avoided, could result in death or serious injury.

Accessing product documentation

Product user documentation is available on the Hitachi Vantara Support Website: https://knowledge.hitachivantara.com/Documents. Check this site for the most current documentation, including important updates that may have been made after the release of the product.

Accessing product downloads

Product software, drivers, and firmware downloads are available on Hitachi Vantara Support website: https://support.hitachivantara.com.

Log in and select Product Downloads to access the most current downloads, including important updates that may have been made after the release of the product.

Getting help

The Hitachi Vantara Support Website is the destination for technical support of products and solutions sold by Hitachi Vantara. To contact technical support, log on to the Hitachi Vantara Support Website for contact information: https://support.hitachivantara.com/en_us/contact-us.html.

Hitachi Vantara Community is a global online community for Hitachi Vantara customers, partners, independent software vendors, employees, and prospects. It is the destination to get answers, discover insights, and make connections. **Join the conversation today!** Go to community.hitachivantara.com, register, and complete your profile.

Comments

Please send us your comments on this document to doc.comments@hitachivantara.com. Include the document title and number, including the revision level (for example, -07), and refer to specific sections and paragraphs whenever possible. All comments become the property of Hitachi Vantara LLC.

Thank you!

Chapter 1: About the S3 Console

The S3 Console provides Hitachi Content Platform for cloud scale (HCP for cloud scale) users with a place to manage and browse their buckets and objects. In addition, it can be used to manage bucket policies, such as expiration lifecycle, sync-to replication, and sync-from replication. In addition, metrics are provided for each bucket.

Chapter 2: Getting started

All completable actions have predefined roles, which are controlled by a system admin who assigns them to bucket owners.

In order to use the S3 Console, bucket owners must first generate their S3 credentials.

Logging in as an admin

A system admin is a user under the local admin account.

> **Important:** The local admin user cannot log in to the S3 Console directly like basic users. The admin user can only login via the admin port.

To log in to the S3 Console as an admin:

Procedure

1. Connect to the admin port:

   `http://`*`clustername`*`:8000`

2. Select **S3 Console**.

Logging in as a user

> **Note:** The local admin user is required to log in through the admin port and can not access the S3 Console through the user login page, as it is a realmless acount. To log in as an admin, see Logging in as an admin (on page 10).

To log in to the S3 Console:

Procedure

1. Enter your **Username** and **Password**.
2. If **Security Realm** is presented, select the appropriate realm.
3. Click **Log in**.

Generating S3 credentials

S3 credentials are used to connect to the S3 gateway for S3 operations. They are the credentials assigned to a bucket owner, allowing you to create and manage buckets and objects from within HCP for cloud scale.

> ⚠ **WARNING:** Generating S3 credentials (*Access Key* and *Secret Key*) will invalidate any previously created keys. Additionally, these keys are only viewable to the user when they are generated. If lost, new keys will need to be created.

To generate new S3 credentials:

Procedure

1. Click the user icon at the top right corner of the page and select **Generate credentials**.
2. To create credentials, click **Generate**.
   A warning screen appears.
3. Click **Continue**.
   New values for **Access Key** and **Secret Key** appear.
4. To copy one of these values, click **Copy**.
5. Click **Done**.

Logging out

To log out of the S3 Console:

Procedure

1. In the upper right corner of the screen, click the user icon (👤).
2. Click **Log out**.

Permissions

In the S3 Console app, the following rules apply to permissions:

- You can view and browse buckets and objects that you are given access to.
- Only bucket owners have the ability to view policies or assign them to buckets, even if you are provided access to those buckets.
- Only an admin can assign roles to bucket owners to provide the proper privileges for them to be able to configure and view bucket policies.

The following HCP for cloud scale roles can be applied and allow/disallow bucket owners to:

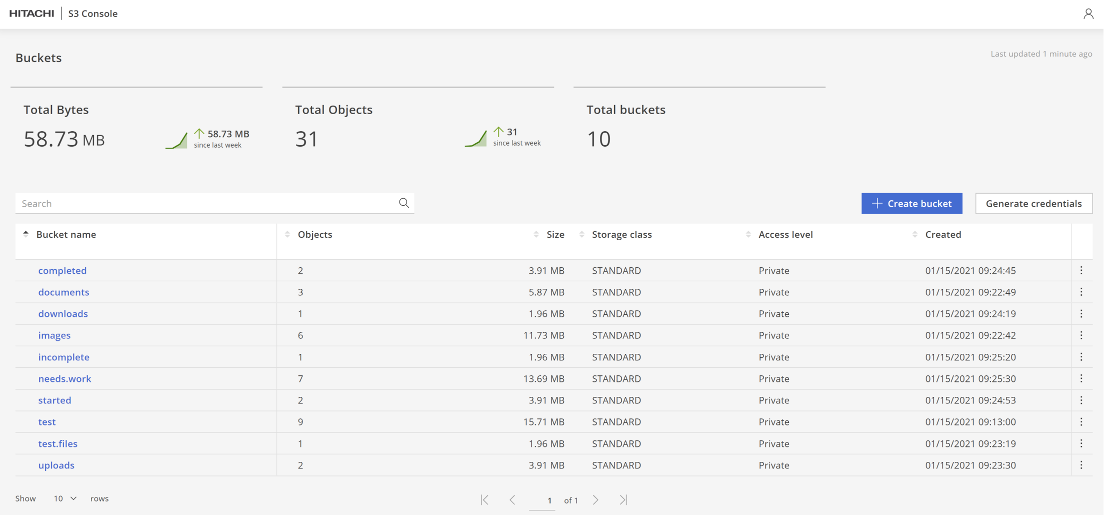
- Set sync-to replication policies: `data:bucket:sync:to:set`
- Set sync-from replication policies: `data:bucket:sync:from:set`
- View sync-to and sync-from replication policies: `data:bucket:sync:get`

Note: You are required to have `s3:user:generate_credentials` permission to log into the S3 Console.

For more information on assigning roles, see the Administrator Help.

Chapter 3: Bucket management

Buckets are containers that store your data on HCP for cloud scale. Contained within buckets are objects, which are the documents and files that you intend on storing in the cloud. Each bucket can also be assigned its own custom configuration and can be set with a unique set of permissions.



Bucket name	Objects	Size	Storage class	Access level	Created
completed	2	3.91 MB	STANDARD	Private	01/15/2021 09:24:45
documents	3	5.87 MB	STANDARD	Private	01/15/2021 09:22:49
downloads	1	1.96 MB	STANDARD	Private	01/15/2021 09:24:19
images	6	11.73 MB	STANDARD	Private	01/15/2021 09:22:42
incomplete	1	1.96 MB	STANDARD	Private	01/15/2021 09:25:20
needs.work	7	13.69 MB	STANDARD	Private	01/15/2021 09:25:30
started	2	3.91 MB	STANDARD	Private	01/15/2021 09:24:53
test	9	15.71 MB	STANDARD	Private	01/15/2021 09:13:00
test.files	1	1.96 MB	STANDARD	Private	01/15/2021 09:23:19
uploads	2	3.91 MB	STANDARD	Private	01/15/2021 09:23:30

To further define objects in a bucket, policies can be added. Rules (where both tags and prefixes can be added as filters) define which objects in the bucket that the policy applies to.

For more information about the different available policies, see Policy management (on page 33). For more information about tags, see Tags (on page 21). For more information about prefixes, see Prefixes (on page 20).

On the **Buckets** page, performance metrics are displayed regarding general bucket utilization and usage. You can also view the following information:

- **Bucket name**: The name of the bucket
- **Objects**: The number of objects and object versions stored in the bucket
- **Size**: The size of the bucket and all of its objects and object versions
- **Storage class**: The selected storage class for the bucket

  > **Note:** S3 Standard is currently the only supported storage class.

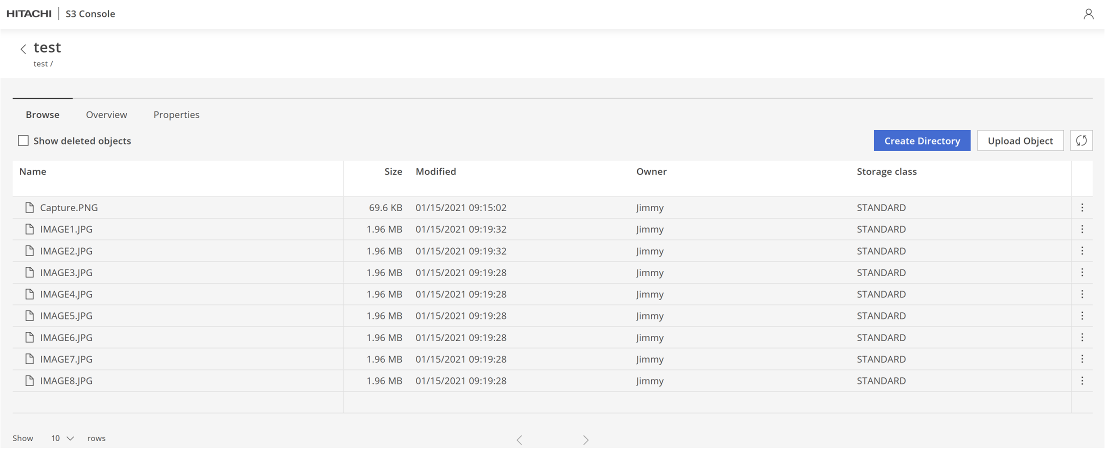
- **Access type**: The level of authentication required to use and view the bucket
  - **Private**: Only you have access to this bucket
  - **Authenticated**: Lets you grant access to this bucket for any user with an account on the system
  - **Unauthenticated**: Lets you grant public access to this bucket for anyone and allows you to assign read or read/write privileges
- **Created**: The date and time the bucket was made

By clicking a bucket's more icon (⋮), you can find additional options for interacting with your bucket:

- **Browse**: Takes you directly to the bucket's **Browse** page
- **Overview**: Takes you directly to the bucket's **Overview** page
- **Properties**: Takes you directly to the bucket's **Properties** page
- **Delete**: Deletes the bucket

Additionally, you can search for a specific bucket by using the search field.

Buckets



☐ Show deleted objects

Name	Size	Modified	Owner	Storage class	
Capture.PNG	69.6 KB	01/15/2021 09:15:02	Jimmy	STANDARD	⋮
IMAGE1.JPG	1.96 MB	01/15/2021 09:19:32	Jimmy	STANDARD	⋮
IMAGE2.JPG	1.96 MB	01/15/2021 09:19:32	Jimmy	STANDARD	⋮
IMAGE3.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD	⋮
IMAGE4.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD	⋮
IMAGE5.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD	⋮
IMAGE6.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD	⋮
IMAGE7.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD	⋮
IMAGE8.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD	⋮

Show 10 rows

From a bucket's main page, you are provided with the following options for interacting with it:

- **Browse**: Where you can view the objects stored within the bucket. Selected by default.
- **Overview**: Where you can view a graphical representation of the bucket's data usage.
- **Properties**: Where you can set the access level and policies for the bucket.

Creating a bucket

To create a bucket:

Procedure

1. From the **Buckets** page, click **+ Create Bucket**.
   The **Create bucket** page appears.
2. In the **Name** field, enter a name for your bucket.

   > **Tip:** Bucket names can only contain lowercase letters, numbers, periods, and hyphens.

3. In the **Access level** section, select your required level of security.
   - **Private**: Only you have access to this bucket.
   - **Authenticated**: Lets you grant access to this bucket for any user with an account on the system.
   - **Unauthenticated**: Lets you grant public access to this bucket for anyone. You can choose to assign **Read** or **Read/Write** privileges.
4. In the **Bucket policies** section, chose your preferred policy using the selection slider.
   - **Expiration Lifecycle policy**: Lets you define when objects expire.
   - **Sync-to Replication**: Lets you enable automatic copying of objects to remote buckets. You cannot apply both Sync-to and Sync-from replications to the same bucket.
   - **Sync-from Replication**: Lets you enable automatic copying of objects from remote buckets. You cannot apply both Sync-to and Sync-from replications to the same bucket.
5. To add rules to your selected policy:
   a. On your selected policy, click **Configure**.
   b. On the **Configure** page, click **+ Add rule**.
   c. To add the rule to a subset of your objects, click **Filter objects** and add tags and prefixes. Additionally, you can have the rule apply to all objects in a bucket by selecting **All objects**.
   d. When you are finished configuring your rule, click **Done**.
6. When you are finished configuring your policy, click **Create**.
   You are returned to the **Buckets** page and a message confirming the creation of the new bucket is displayed.
7. To view your new bucket, select it by clicking its name from the **Bucket name** column.
   The bucket page is displayed and an overview of your bucket is provided.

Creating a bucket with object lock

Note: Object lock on a bucket can only be enabled when the bucket is created.

When applying object lock to a bucket, you can then enable compliance mode to set a retention period to its contents. Additionally, when object lock is applied, legal hold can be set on a version of an object. See Setting legal hold on a version of an object (on page 23).

To create a bucket with object lock:

Procedure

1. From the **Buckets** page, click **+ Create Bucket**.
   The **Create bucket** page appears.
2. In the **Name** field, enter a name for your bucket.
3. Click the **Object lock** toggle to enable it.
4. In the **Access level** section, select your required level of security.
   - **Private**: Only you have access to this bucket.
   - **Authenticated**: Lets you grant access to this bucket for any user with an account on the system.
   - **Unauthenticated**: Lets you grant public access to this bucket for anyone. You can choose to assign Read or Read/Write privileges.
5. In the **Bucket policies** section, choose **Object Lock**.
6. Click **Configure** on the **Object Lock** policy to set retention.
7. Click the **Default retention** toggle to enable it and set the retention period.
8. Click **Update**.
9. When you are finished configuring your bucket, click **Update**.
10. To view your new bucket, select it by clicking its name from the **Bucket name** column.
    The bucket page is displayed and an overview of your bucket is provided.

Deleting a bucket

A bucket can only be deleted if it is emptied of all objects contained within it.

⚠ **WARNING:** Bucket deletion is permanent.

To delete a bucket:

Procedure

1. From the **Buckets** page, navigate to the bucket you want to delete.
2. Delete all objects from the bucket. See Deleting an object (on page 27).
3. From the **Buckets** page, click the more icon for your bucket and then select **Delete**.
   A confirmation message appears.
4. Click **Confirm Delete**.

Directories

Directories are folders within a bucket that house objects and help to provide organization.

Creating a directory

To create a directory for a bucket:

Procedure

1. From the **Buckets** page, select the bucket you want to create a directory in.
2. Click **Create Directory**.
3. Give your new directory a name and click **Save**.

Deleting directories

When you delete a directory, all of the files contained within it are also deleted. A delete marker is created for the directory, which can be used to restore it and all of its contents at a later time. See Displaying deleted objects and directories (on page 28).

Deleting a directory

To delete a directory:

Procedure

1. From the **Buckets** page, select the bucket containing the directory you want to delete.
2. Click the more icon at the righthand side of the directory.
3. Click **Delete**.
   The directory is deleted.

Deleting multiple directories

To delete multiple objects:

Procedure

1. From the **Buckets** page, select the bucket containing the directories you want to delete.
2. Click the checkboxes associated with the directories.
3. From the righthand side of the object selection bar, click **Delete**.
   The directories are deleted from the bucket.

Deleting an empty directory

An empty directory is characterized as a folder that doesn't have any listed objects, even though some objects may have the delete marker as the current version.

After deleting an empty directory, the object list is refreshed to show the change.

To delete an empty directory:

Procedure

1. From the **Buckets** page, navigate to the directory you want to delete.
2. Click the more icon at the righthand side of the directory.
   A warning message appears regarding the deletion of the bucket.
3. To confirm the delete, click **Confirm**.
   The directory is deleted.

Restoring directories

To restore a deleted directory, you must first enable it to view its delete marker. See Displaying deleted objects and directories (on page 28). When you restore a directory, all of the files contained within it are also restored.

Restoring a deleted directory

To restore a deleted directory:

Procedure

1. From the **Buckets** page, select the bucket containing the directory you want to restore.
2. Click the **Show deleted objects** box.
   Your deleted directories now display with a trashcan icon next to them.
3. Click the more icon at the righthand side of the directory, noted by its trashcan icon.
4. Click **Restore**.
   The directory is restored to the bucket.

Restoring multiple deleted directories

To restore multiple deleted directories:

Procedure

1. From the **Buckets** page, select the bucket containing the directories you want to restore.
2. Click the **Show deleted objects** box.
   Your deleted directories now display with a trashcan icon next to them.
3. Click the checkboxes associated with the deleted directories.
4. From the righthand side of the object selection bar, click **Restore**.
   The directories are now restored to your bucket.

Destroying directories

When you destroy a directory, all versions of the directory are completely removed. All objects in the directory must first be deleted and its object count must be at 0 in order to take this action.

Destroying a directory

⚠ **WARNING:** Destroying a directory cannot be undone.

To destroy a deleted directory:

Procedure

1. From the **Buckets** page, select the bucket containing the directory you want to destroy.
2. Click the **Show deleted objects** box.
   Your deleted directory is now displayed with a trashcan icon next to it.
3. Click the more icon at the righthand side of the directory.
4. Click **Destroy**.
5. Type **YES** and then click **Confirm Destroy**.
   The directory is permanently removed from the bucket.

Rules

Rules can be applied to the objects in a bucket using either prefixes, tags, or a combination of both. They can also be helpful if want to set object expiration for current and non-current versions of objects. Policies can support up to 1,000 rules at a time.

Adding rules to policies

To add a new rule to a policy:

Procedure

1. From the **Buckets** page, select your bucket.
2. Click the **Properties** tag.
3. Click **Configure** on the policy.
4. Click **+ Add rule**.
   The **Add Rule** page appears.
5. To add tags to your rule, click **+ Add tag**.
6. To apply actions (for the expiration lifecycle policy) or to change configuration settings for your rule, scroll to the bottom of the page.
7. When finished, click **Done**.
   The rule is added to the selected policy's **Configure policy** page.
8. From the **Rules** table, select the rule(s) you want to add to your policy using the checkbox column.
9. Click **Done**.
   You are returned to the **Create bucket** page and your rules are noted in the **Configured rules** section of your selected policy.

Editing a rule

To edit a rule:

Procedure

1. From the **Buckets** page, select your bucket.
2. Click the **Properties** tag.
3. Click **Configure** on the policy.
4. Click the more button for your rule and then select **Edit**.
5. Click **Done**.

Deleting rules from a policy

To delete a rule from a policy:

Procedure

1. From the **Buckets** page, select your bucket.
2. Click the **Properties** tag.
3. Click **Configure** on the policy.
4. Click the more button for your rule and then select **Delete**.
5. Click **Okay**.

Filters

All rules can be assigned a filters. Filters help you identify which objects in a bucket the rule applies to.

Prefixes

Prefixes are an S3 concept that represent the path to a virtual folder. They are a way to help you visualize the concept of cloud storage and storage, given that no actual folders truly exist, and allow you to group objects by using common strings.

For example, setting a rule that applies to the `/foo` prefix would apply to all objects that start with `/foo`, such as `/foo/bar`, `/foo/bar1/bar2`, etc.

Note: A rule can only be assigned a single prefix.

Adding a prefix filter to a policy

Important: When adding a prefix filter to an expiration lifecycle policy, you must enable at least one action on the policy.

To add a prefix filter to a policy:

Procedure

1. From the **Buckets** page, select your bucket.
2. Click the **Properties** tag.
3. Click **Configure** on the policy.
4. Click **+ Add rule**.
   The **Add Rule** page appears.
5. Click **Filter objects**.
   The **Prefix** field and **Tags** list appear.
6. In the **Prefix** field, enter your prefix.
7. Click **Done**.

Editing a prefix filter

To edit a prefix filter on a policy:

Procedure

1. From the **Buckets** page, select your bucket.
2. Click the **Properties** tag.
3. Click **Configure** on the policy.
   The **Rules** page appears.
4. Click the more icon for your prefix filter and select **Edit**.

Removing a prefix filter from a policy

To remove a prefix filter from a policy:

Procedure

1. From the **Buckets** page, select your bucket.
2. Click the **Properties** tag.
3. Click **Configure** on the policy.
   The **Rules** page appears.
4. Click the more icon for your prefix filter and select **Delete**.

Tags

Tags are independent of prefixes and are represented as key-value pairs. You can create a rule only using tags without adding a prefix. A rule with tags means that it only applies to objects with matching values.

Adding tags to rules

To add a tag to a rule:

Procedure

1. From the **Buckets** page, select your bucket.

2. Click the **Properties** tag.
3. Click **Configure** on your selected policy.
4. Click **+ Add rule**.
5. Click **Filter objects**.
   The **Prefix** field and **Tags** list appear.
6. To add tags to your rule, click **+ Add tag**.
   The **Add tag** window appears.
7. In the **Key** and **Value** fields, enter your tagging information.
8. When finished, click **Save**.
9. (Optional) To add additional tags, click the **Add another tag** box and then click **Save**.
10. On the **Add rule** page, click **Done**.

Editing a tag

To edit a tag which is part of a current rule:

Procedure

1. From the **Buckets** page, select your bucket.
2. Click the **Properties** tag.
3. Click **Configure** on your selected policy.
4. On the **Configure policy** page, click the more icon for the rule containing your tag and then select **Edit**.
   The **Add Rule** page appears.
5. Click **Filter objects**.
   The **Prefix** field and **Tags** list appear.
6. To edit a tag, click its more icon from the **Tags** list and then select **Edit**.
7. In the **Key** and **Value** fields, update your tagging information.
8. When finished, click **Save**.
9. To finalize your changes, on the **Add rule** page, click **Done**.

Deleting a tag filter

To delete a tag from a rule:

Procedure

1. From the **Buckets** page, select your bucket.
2. Click the **Properties** tag.
3. Click **Configure** on your selected policy.
4. On the **Configure policy** page, click the more icon for the rule containing your tag and then select **Delete**.

Bucket retention management

Setting retention on an object version protects it from deletion for a set amount of time. Once the retention period ends, that version of the object can then deleted.

Setting retention on a version of an object

> **Note:** Retention can only be set on a version of an object when an object lock policy has been applied to a bucket. See Adding an object lock policy (on page 39).

To set legal hold on a version of an object within a bucket:

Procedure

1. From the **Buckets** page, click the name of the bucket in which the object resides.
2. Navigate to the object you want to view and then click its more icon.
3. Select **View versions**.
   All versions of the selected object are displayed.
4. Click the more icon at the righthand side of the object you want to add retention to.
5. To set retention on an object, click **Set Retention**.
6. In the **Date** and **Time** fields, set your values for your object's retention length.
7. Click **Save**.
   Retention is set on that version of the object.

Setting legal hold on a version of an object

> **Note:** Legal hold can only be set on a version of an object when an object lock policy has been applied to a bucket. See Adding an object lock policy (on page 39).

To set legal hold on a version of an object within a bucket:

Procedure

1. From the **Buckets** page, select the bucket containing the object.
2. Navigate to the object and then click its more icon.
3. Select **View versions**.
   All versions of the selected object are displayed.
4. Click the more icon at the righthand side of the object you want to add legal hold to.
5. Click **Set legal hold**.
   The **Set legal hold** window displays.
6. Click the toggle to enable legal hold.
7. Click **Save**.
   Legal hold is set on that version of the object.

Restoring a version of an object

> **Important:** When recovering a version of an object, the object's tags are preserved but it will lose its metadata in the recovery.

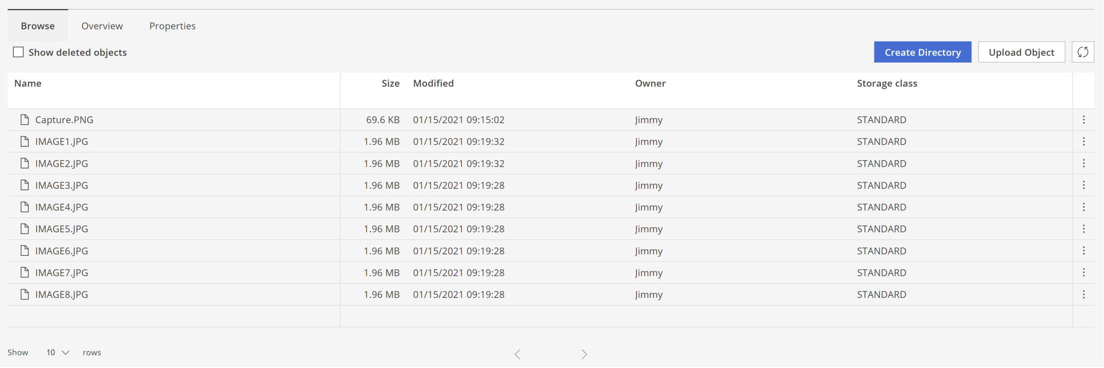
To restore a version of an object:

Procedure

1. From the **Buckets** page, select the bucket containing the object version you want to restore.
2. Navigate to the object and then click its more icon.
3. Select **View versions**.
   All versions of the selected object are displayed.
4. Click the more icon at the righthand side of the version you want to restore and select **Restore Data**.
   The object version is now restored to your bucket.

Chapter 4: Object management

An object is any file or document contained within a bucket. All objects that are currently stored in a bucket can be viewed from the **Browse** tab, which is selected by default.



☐ Show deleted objects

Name	Size	Modified	Owner	Storage class
Capture.PNG	69.6 KB	01/15/2021 09:15:02	Jimmy	STANDARD
IMAGE1.JPG	1.96 MB	01/15/2021 09:19:32	Jimmy	STANDARD
IMAGE2.JPG	1.96 MB	01/15/2021 09:19:32	Jimmy	STANDARD
IMAGE3.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD
IMAGE4.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD
IMAGE5.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD
IMAGE6.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD
IMAGE7.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD
IMAGE8.JPG	1.96 MB	01/15/2021 09:19:28	Jimmy	STANDARD

Show 10 rows

On the object page, you can view the following information:

- **Bucket name**: The name of the object
- **Size**: The size of the object (in MB, GB, TB, or PB)
- **Storage class**: The selected storage class for the bucket

  Note: Currently, the S3 Standard class is the only supported storage class.

- **Owner**: The user that owns the object

Using objects

You are provided basic functionality with your bucket to manage the the objects stored inside of it. Similar to file storage on the internet, objects can be uploaded to the bucket, downloaded from the bucket, and have authentication applied to them. You can also view information specific to the file or versions of the file.

Uploading an object to a bucket

When uploading an object, you can add files up to 5GB in size. The changes may take up to 60 seconds to display in your bucket.

To upload an object:

Procedure

1. From the **Buckets** page, select the bucket containing the object.

2. Click **Upload Object**.
   The **Upload Object** screen appears.
3. To upload an object, drag and drop a file or browse for one by clicking **Select files**.
   The object appears listed beneath the drag and drop window and a checkmark confirms when it has finished uploading to the bucket.
4. When you are finished uploading, close the **Upload Object** window.
   The object appears in your selected bucket.

Downloading an object

To download an object:

Procedure

1. From the **Buckets** page, select the bucket containing the object.
2. Click the more icon at the righthand side of the object you want to download.
3. Click **Download**.

Generating an authenticated link for an object

Authenticated links can be generated that provide you with direct access to an object, with date and time fields that can be manually set to expire. All authenticated links expire after a maximum of seven days.

Procedure

1. From the **Buckets** page, select the bucket containing the object.
2. Click the more icon at the righthand side of the object you want to link to.
3. To create a link, click **Generate link**.
4. Enter your **Access key** and **Secret key** and then click **Next**.
5. In the **Link expiration** fields, set your values for the retention length of the link.
6. Click **Copy link**.
   The link is copied to your clipboard.

Viewing the details of an object

To display the details of an object:

Procedure

1. From the **Buckets** page, select the bucket containing the object.
2. Click the more icon at the righthand side of the object with properties you want to view.
3. Click **View properties**.
   Object properties are displayed.
4. When finished, click **Done**.

Viewing the versions of an object

When viewing different versions of an object, you can see which objects have object lock or retention set on them.

To display the different versions of an object.

Procedure

1. From the **Buckets** page, select the bucket containing the object.
2. Click the more icon at the righthand side of the object with different versions you want to view.
3. Click **View versions**.
   All versions of the selected object are displayed.
4. When finished, click **Done**.

Deleting objects

When you delete an object or a version of an object, it is removed from the object list but a delete marker is created, allowing for the data associated with it to be restored to full functionality at any point.

> **Tip:** When deleting objects, it can take up to 60 seconds for the change to display in your bucket.

Deleting an object

When you delete an object, the object is removed from the bucket and a delete marker is created, allowing you to restore it at a later time. See Displaying deleted objects and directories (on page 28) and Restoring a deleted object (on page 28).

To delete an object:

Procedure

1. From the **Buckets** page, select the bucket containing the object you want to delete.
2. Click the more icon at the righthand side of the object.
3. Click **Delete**.
   The object is deleted from the bucket.

Deleting multiple objects

When you delete objects, delete markers are created for each one, which can be used to restore them at a later time.

To delete multiple objects:

Procedure

1. From the **Buckets** page, select the bucket containing the object you want to delete.

2. Select the objects you want to delete by using their checkboxes. Additionally, you can select all objects in a bucket by clicking **All** on the object selection bar.
3. From the righthand side of the object selection bar, click **Delete**.
   The objects are deleted from the bucket.

Displaying deleted objects and directories

When an object or a directory is deleted, it is not permanently removed from the bucket. A delete marker is created which maintains a copy of it until it is destroyed. At any point during your usage of the S3 Console, deleted objects and directories can be renabled and displayed.

To display deleted objects and directories:

Procedure

1. From the **Buckets** page, navigate to the bucket containing the deleted objects and directories you want to view.
2. Click the **Show deleted objects** box.
   Your deleted objects and directories now display with a trashcan icon next to them.

Restoring objects

When you restore an object or a version of an object, the delete marker that was created when it was deleted is fully restored, returning it back to its full functionality.

Restoring a deleted object

To restore a deleted object, you must first enable it to view its delete marker. See Displaying deleted objects and directories (on page 28).

To restore a deleted object:

Procedure

1. From the **Buckets** page, select the bucket containing the object you want to restore.
2. Click the more icon at the righthand side of the object, noted by its trashcan icon.
3. Click **Restore**.
   The object is restored to the bucket.

Restoring multiple deleted objects

To restore multiple deleted objects, you must first enable them to view their delete markers. See Displaying deleted objects and directories (on page 28).

To restore multiple deleted objects:

Procedure

1. From the **Buckets** page, select the bucket containing the objects you want to restore.

2. Click the checkboxes associated with the deleted objects.
3. From the righthand side of the object selection bar, click **Restore**.
   The objects are restored to the bucket.

Restoring a version of an object

> **Important:** When recovering a version of an object, the object's tags are preserved but it will lose its metadata in the recovery.

To restore a version of an object:

Procedure

1. From the **Buckets** page, select the bucket containing the object version you want to restore.
2. Navigate to the object and then click its more icon.
3. Select **View versions**.
   All versions of the selected object are displayed.
4. Click the more icon at the righthand side of the version you want to restore and select **Restore Data**.
   The object version is now restored to your bucket.

Destroying objects

Destroying objects and their versions is different from deleting them in that it completely removes them from your bucket. It destroys all delete markers associated with the object or the specific version of an object, preventing any future recovery over the lifetime of the bucket. When you destroy an object, it is completely removed from your cloud storage.

Destroying an object

> **WARNING:** Destroying an object cannot be undone.

To destroy an object:

Procedure

1. From the **Buckets** page, select the bucket containing the object you want to destroy.
2. Click the **Show deleted objects** box.
   Your deleted object is now displayed with a trashcan icon next to it.
3. Click the more icon at the righthand side of the object.
4. Click **Destroy**.
5. Type **YES** and then click **Confirm Destroy**.
   The object is permanently removed from the bucket.

Destroying multiple objects

⚠ **WARNING:** Destroying objects cannot be undone.

To destroy multiple objects:

Procedure

1. From the **Buckets** page, select the bucket containing the objects you want to destroy.
2. Select the objects you want to destroy by using their checkboxes. Additionally, you can select all objects in a bucket by clicking **All** on the object selection bar.
3. From the righthand side of the object selection bar, click **Destroy**.
4. Type **YES** and then click **Confirm Destroy**.
   The objects are permanently removed from the bucket.

Destroying a version of an object

When you destroy a version of an object, that specific version is completely removed from your cloud storage. All other versions remain unaffected.

⚠ **WARNING:** Destroying a version of an object cannot be undone.

To destroy a specific version of an object:

Procedure

1. From the **Buckets** page, click the name of the bucket in which the object resides.
2. Navigate to the object you wish to view and then click its more icon.
3. Select **View versions**.
   All versions of the selected object are displayed.
4. Click the more icon at the righthand side of the version you want to delete.
5. Click **Destroy**.
6. Type **YES** and then click **Confirm Destroy**.
   The version is permanently removed from the bucket.

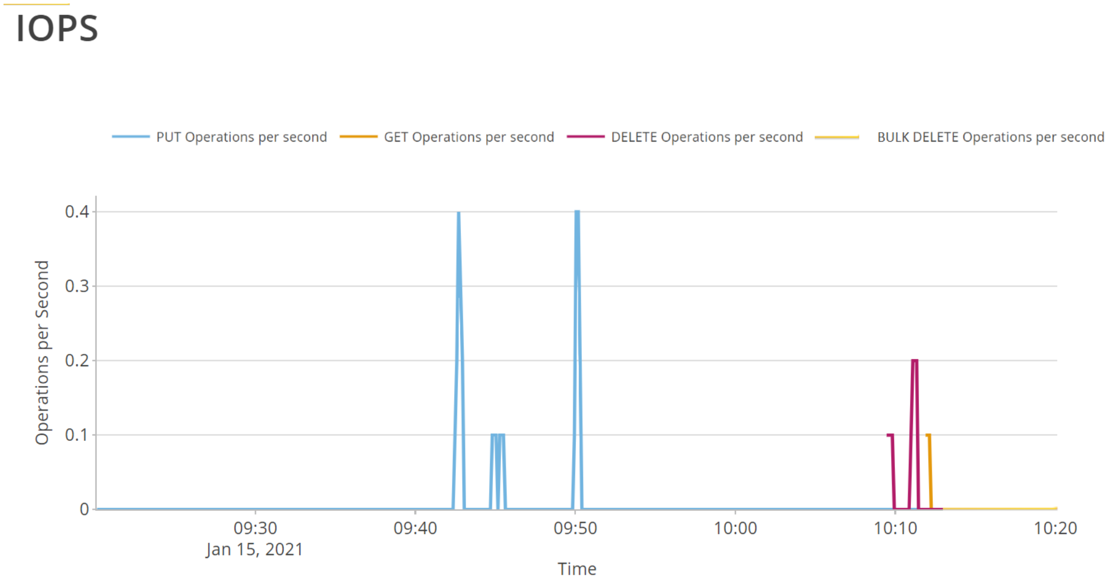
Chapter 5: Monitoring

The S3 Console provides powerful metrics that let you track input/output operations, the loading of objects (ingest), the number of objects stored, and the disk usage of stored objects.

All available buckets provide you with a dashboard to view its performance. They can be found on the Overview tab of any bucket, which is automatically displayed upon selecting a bucket. The dashboard can be customized to display different ranges of time by clicking Range.

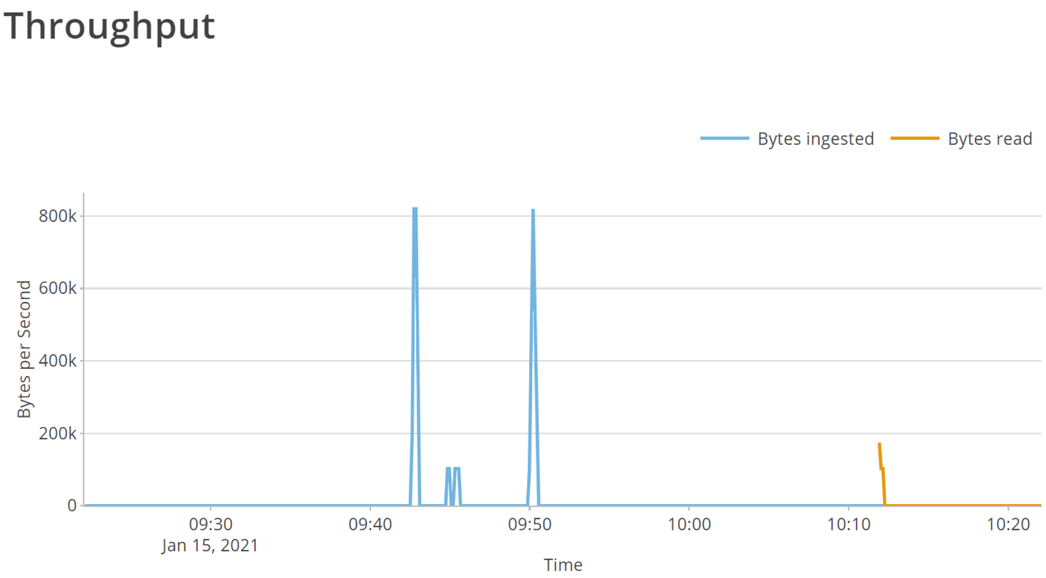
IOPS dashboard

This dashboard displays the input/output operations per second (IOPS). You can hover over the PUT Operations per second, GET Operations per second, DELETE Operations per second, and BULK DELETE Operations per second lines to be provided with metrics for specific data points.



Throughput dashboard

This dashboard displays the size (in bytes) of files and documents that have been taken into the bucket. You can hover over the Bytes ingested and Bytes read lines to be provided with metrics for specific data points.


Throughput
Bytes ingested
Bytes read
Bytes per Second
800k
600k
400k
200k
0
09:30
Jan 15, 2021
09:40
09:50
10:00
10:10
10:20
Time

Chapter 6: Policy management

Policies allow you to apply specific retention and permissions to buckets and the objects contained within.

The S3 Console supports the following policies:

- Expiration lifecycle
- Sync-from replication
- Sync-to replication
- Object lock

Tip: When navigating through policies, you can use the breadcrumbs found under the bucket's name to quickly navigate back to previous screens.

Expiration lifecycle policy

The *expiration lifecycle* policy sets an expiration date on the objects within a bucket.

A set of rules is applied to this policy that define actions across groups of objects. These rules can apply to current versions, non-current versions, incomplete multi-part uploads, and expired delete markers.

Each policy can contain up to 1,000 rules. Additionally, each rule contains filters (such as prefixes and tags), as well as actions.

Note: Actions are applied as to your expiration lifecycle policy as rules and can be implemented from the **Add rule** page of a given bucket.

Actions apply to all objects in the bucket and are specific to the expiration lifecycle policy. They can be added as individually set rules and do not require tags or a prefix. Currently, the following four expiration actions are supported:

- **Current versions**: Permanenetly deletes an object after a set number of days from object creation, or on a specific date. The default is 365 days.
- **Non-current versions**: Permanently deletes an object after a set number of days from having been made a previous version, or on a specific date. The default is 30 days.
- **Incomplete multi-part uploads**: Removes partial MPU uploads if they are not successfully completed within a set number of days.
- **Expired delete markers**: Retains an expired delete marker in the event that all previous versions of an object expire after the deletion of a versioned object. The default is 7 days.

> **Important:** The **Expired delete markers** policy cannot be set if the **Current versions policy** is enabled.

Adding an expiration lifecycle policy to a new bucket

To add an expiration lifecycle policy to a new bucket:

Procedure

1. From the **Buckets** page, click **Create bucket**.
2. Enable the **Expiration Lifecycle** policy by clicking its selection toggle.
3. Configure your policy by clicking **Configure**.
4. When you are finished editing, click **Done**.
5. Click **Create**.

Adding an expiration lifecycle policy to a pre-existing bucket

To add an expiration lifecycle policy to a pre-existing bucket:

Procedure

1. From the **Buckets** page, navigate to your bucket and click its name in the **Bucket name** column to select it.
2. Enable the **Expiration Lifecycle** policy by clicking its selection toggle.
3. Edit your policy by clicking **Configure**.
4. When you are finished editing, click **Done**.
5. Click **Create**.

Adding actions to an expiration lifecycle policy

To add actions to an expiration lifecycle policy:

Procedure

1. From the **Buckets** page, select your bucket.

2. Click the **Properties** tag.
3. Click **Configure** on the policy.
4. Click **+ Add rule**.
5. From the **Actions** section, select your preferred action by clicking its selection slider.
   - To place an expiration lifecycle policy on current versions, enable **Current versions**. You can then set a number of days to hold these files or a specific date by which they will be deleted.
   - To place an expiration lifecycle policy on previous versions, enable **Non-current versions**. You can then set a number of days to hold these files.
   - To place an expiration lifecycle policy on incomplete multi-part uploads, enable **Incomplete multi-part uploads**. You can then set a number of days to hold these partially uploaded files until they are deleted.
   - Optionally, you can enable **Expired delete markers** to automatically remove expired objects.
6. Once selected, configure your action.
7. When you are finished editing, click **Done**.

Editing an expiration lifecycle policy

To edit an expiration lifecycle policy:

Procedure

1. From the **Buckets** page, navigate to your bucket and click its name in the **Bucket name** column to select it.
2. On the bucket's page, click the **Properties** tab.
3. Edit the **Expiration Lifecycle** policy by clicking **Configure**.
4. When you are finished editing, click **Done**.
5. Click **Update**.

Removing an expiration lifecycle policy

To remove an expiration lifecycle policy from a bucket:

Procedure

1. From the **Buckets** page, navigate to your bucket and click its name in the **Bucket name** column to select it.
2. On the bucket page, click the **Properties** tab.
3. Remove the **Expiration Lifecycle** policy by clicking its selection toggle.
   The policy is greyed out.
4. Click **Update**.
   The policy is removed from the bucket.

Sync-from replication policy

The *sync-from replication* policy provides information about replicated objects, their remote buckets, and information from the remote queue.

A set of rules that define asynchronous replication *from* remote buckets is applied. Each rule defines the objects to be replicated, the remote bucket these objects are replicated from, and the corresponding AWS SQS queue. The queue is used for notifications about the changes in the remote bucket.

Each policy can contain up to 1,000 rules and each rule contains filters (such as prefixes and tags). If a filter is not applied to a sync-from replication policy on bucket, then the policy applies to all objects.

As you set up your policy, all required fields are highlighted to make configuration easier.

> **Important:** When adding a rule to this policy, the All objects filter is selected by default. To add tags, a prefix, or both, click Filter objects.

Adding a sync-from replication policy to a new bucket

To add a sync-from replication policy to a new bucket:

Procedure

1. From the **Buckets** page, click **Create bucket**.
2. Enable the **Sync-from Replication** policy by clicking its selection toggle.
3. Edit your policy by clicking **Configure**.
    a. Add your S3 access information to the **Remote bucket configuration** section.
    b. *Optional*: To test your S3 connection, click the **Test bucket connection** button.
    c. Add your AWS SQS credentials to the **AWS SQS queue** section.

    > **Important:** The **Queue** and **Region** fields are the *from* of the sync-from replication policy.

    d. *Optional*: To test your AWS SQS queue connection, click the **Test queue connection** button.
4. When you are finished editing, click **Done**.
5. Click **Create**.

Adding a sync-from replication policy to a pre-existing bucket

To add a replication sync-from replication policy to a pre-existing bucket:

Procedure

1. From the **Buckets** page, navigate to your bucket and click its name in the **Bucket name** column to select it.
2. Enable the **Sync-from Replication** policy by clicking its selection toggle.

3. Edit your policy by clicking **Configure**.
4. When you are finished editing, click **Done**.
5. Click **Update**.

Editing a sync-from replication policy

To edit a bucket's sync-from replication policy:

Procedure

1. From the **Buckets** page, navigate to your bucket and click its name in the **Bucket name** column to select it.
2. On the bucket's page, click the **Properties** tab.
3. Edit the **Sync-from Replication** policy by clicking **Configure**.
4. When you are finished editing, click **Done**.
5. Click **Update**.

Removing a sync-from replication policy

To remove a sync-from replication policy from a bucket:

Procedure

1. From the **Buckets** page, navigate to your bucket and click its name in the **Bucket name** column to select it.
2. On the bucket page, click the **Properties** tab.
3. Remove the **Sync-from Replication** policy by clicking its selection toggle.
   The policy is greyed out.
4. Click **Update**.
   The policy is removed from the bucket.

Sync-to replication policy

The *sync-to replication* policy provides information about replicated objects and their remote buckets.

A set of rules that define asynchronous replication *to* remote buckets is applied. Each rule defines the objects to be replicated and the remote bucket these objects are to be replicated in.

Each policy can contain up to 1,000 rules and each rule contains filters (such as prefixes and tags). If a filter is not applied to a sync-to replication policy on bucket, then the policy applies to all objects.

As you set up your policy, all required fields are highlighted to make configuration easier.

> **Important:** When adding a rule to this policy, the All objects filter is selected by default. To add tags, a prefix, or both, click Filter objects.

Adding a sync-to replication policy to a new bucket

To add a sync-to replication policy to a new bucket:

Procedure

1. From the **Buckets** page, click **Create bucket**.
2. Enable the **Sync-to Replication** policy by clicking its selection toggle.
3. Edit your policy by clicking **Configure**.
   a. Add your S3 access information to the **Remote bucket configuration** section.

      > **Important:** The **S3 hostname** field is the *to* of the sync-to replication policy.

   b. *Optional*: To test your S3 connection, click the **Test bucket connection** button.
4. When you are finished editing, click **Done**.
5. Click **Create**.

Adding a sync-to replication policy to a pre-existing bucket

To add a sync-to replication policy to a pre-existing bucket:

Procedure

1. From the **Buckets** page, navigate to your bucket and click its name in the **Bucket name** column to select it.
2. Enable the **Sync-to Replication** policy by clicking its selection toggle.
3. Configure your policy by clicking **Configure**.
4. When you are finished editing, click **Done**.
5. Click **Update**.

Editing a sync-to replication policy

To edit a bucket's sync-to replication policy:

Procedure

1. From the **Buckets** page, navigate to your bucket and click its name in the **Bucket name** column to select it.
2. On the bucket's page, click the **Properties** tab.
3. Edit the **Sync-to Replication** policy by clicking **Configure**.
4. When you are finished editing, click **Done**.
5. Click **Update**.

Removing a sync-to replication policy

To remove a sync-to replication policy from a bucket:

Procedure

1. From the **Buckets** page, navigate to your bucket and click its name in the **Bucket name** column to select it.
2. On the bucket page, click the **Properties** tab.
3. Remove the **Sync-to Replication** policy by clicking its selection toggle.
   The policy is greyed out.
4. Click **Update**.
   The policy is removed from the bucket.

Object lock policy

An *object lock policy* allows you to set a retention period on an object or bucket, allowing them to prevent its deletion for a set period of time.

The objects are stored using a write-once-read-many (WORM) model.

Adding an object lock policy

Note: The object lock policy can only be enabled when a bucket is created.

Adding an object lock policy to a bucket provides you with the ability to add retention and legal hold to the objects contained within.

You can also enable compliance mode to set a retention period to all of the contents within a bucket by default. Additionally, when object lock is applied, legal hold can be set on a version of an object within the bucket. See Setting legal hold on a version of an object (on page 23).

To create a bucket with an object lock policy:

Procedure

1. From the **Buckets** page, click **+ Create Bucket**.
   The **Create bucket** page appears.
2. In the **Name** field, enter a name for your bucket.
3. Click the **Object lock** toggle to enable it.
4. In the **Access level** section, select your required level of security.
   - **Private**: Only you have access to this bucket.
   - **Authenticated**: Lets you grant access to this bucket for any user with an account on the system.
   - **Unauthenticated**: Lets you grant public access to this bucket for anyone. You can choose to assign Read or Read/Write privileges.
5. In the **Bucket policies** section, choose **Object Lock**.
6. Click **Configure** on the **Object Lock** policy to set retention.
7. Click the **Default retention** toggle to enable it and set the retention period.
8. Click **Done**.

9. When you are finished configuring your bucket, click **Create**.
   You are returned to the **Buckets** page and a message confirming the creation of the new bucket is displayed.
10. To view your new bucket, select it by clicking its name from the **Bucket name** column.
    The bucket page is displayed and an overview of your bucket is provided.

Editing an object lock policy

To edit an object lock policy:

Procedure

1. From the **Buckets** page, navigate to your bucket and click its name in the **Bucket name** column to select it.
2. On the bucket's page, click the **Properties** tab.
3. Edit the **Object Lock** policy by clicking **Configure**.
4. When you are finished editing, click **Done**.
5. Click **Update**.
   Your changes are applied to your policy.

Deleting a bucket with an object lock policy

Once an object lock policy is applied to a bucket, it cannot be disabled.

To remove a bucket with an object lock policy:

1. Delete the bucket. See Deleting a bucket (on page 16).

Chapter 7: Bucket synchronization

Hitachi Content Platform for cloud scale (HCP for cloud scale) lets you configure and manage bucket synchronization.

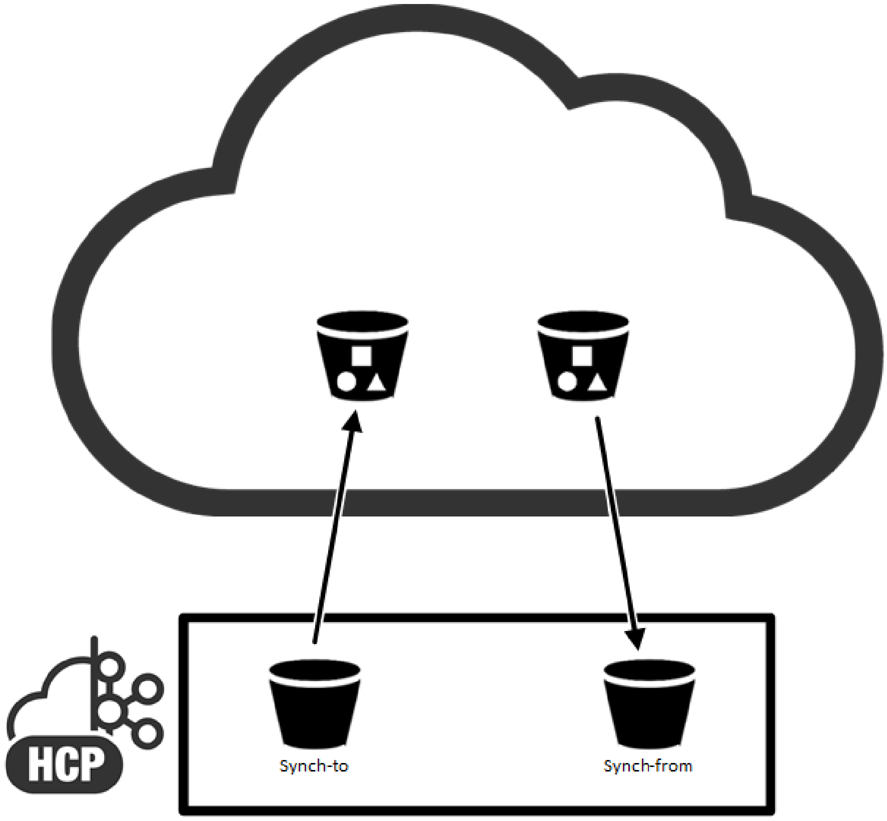
To configure bucket synchronization, use S3 `put bucket replication` API requests. Scripts are available to simplify the process.

About bucket synchronization

HCP for cloud scale can synchronize the following kinds of data in buckets:

- Object data
- All user metadata (that is, anything that can be returned in the header `x-amz-meta-*`)
- Tags
- `Content-Type` system metadata
- Objects that the owner of the source bucket doesn't have permission to read

This diagram illustrates the concept of bucket synchronization.



Limitations on bucket synchronization

Objects that existed before synchronization functions are configured are not synchronized.

HCP for cloud scale verifies the rules that are valid at the time an object is synchronized, not at the time the object is ingested.

Objects that are marked as deleted are not synchronized.

Most system metadata is not synchronized, specifically:

- Owner ID and Name
- Timestamps (when last modified)
- Metadata returned in `x-amz-grant-*`
- Metadata returned in `x-amz-acl`
- Metadata returned in `x-amz-grant-*`
- Metadata returned in `x-amz-acl`
- Metadata returned in `x-amz-storage-class`
- Metadata returned in `x-amz-replication-status`
- Metadata returned in `x-amz-server-side-encryption-*`
- Metadata returned in `x-amz-restore-*`
- Metadata returned in `x-amz-version-id-*`

- Metadata returned in `x-amz-website-redirect-location`
- Metadata returned in `x-amz-object-lock-*`

The bucket sync-from function only supports one rule for the same external SQS queue and external bucket. If a bucket has multiple sync-from rules for the same external queue, objects might not be synchronized. To use multiple rules for an external bucket, use one SQS queue for each rule.

Comparing synchronization to replication

Unlike AWS replication, HCP for cloud scale can synchronize with buckets on storage systems outside of AWS.

AWS determines the destination bucket using rules, but only applies one rule to each new object. In contrast, HCP for cloud scale can apply multiple rules to each new object so long as the destination buckets are different. This is how one-to-many synchronization is implemented.

AWS does not replicate, but HCP for cloud scale synchronizes, objects that the owner of the source bucket doesn't have permission to read.

In contrast with AWS replication, HCP for cloud scale does not synchronize the following:

- Access control lists (ACLs)
- Lock retention information
- Objects that are encrypted using Amazon S3 managed keys (SSE-S3) and AWS KMS managed keys (SSE-KMS)

If an object being synchronized has the same name as an object in the target bucket, the result depends on whether the target bucket uses versioning:

- If versioning is used, the old object is kept as an old version.
- If versioning is not used, the old object is replaced by the new object.

HCP for cloud scale buckets always use versioning. The best practice is to use versioning in all target buckets.

Best-effort ordering

HCP for cloud scale guarantees that operations are applied in the order of their arrival (*strong consistency*). However, synchronizing multiple operations applied in a short period of time to the same object presents the following difficulties:

- In a distributed system, especially when many systems are involved, synchronizing all operations in correct order is complex.
- Even if HCP for cloud scale synchronizes all operations in correct order to an external storage component, that component might not guarantee that the operations are applied with strong consistency. In particular, AWS guarantees only "eventual consistency."
- For bucket sync-from, the external queue service might not guarantee that messages are provided in correct order. In particular, AWS Simple Queue Service (SQS) does not support first-in, first-out (FIFO) queues for S3 notifications.

Therefore, HCP for cloud scale makes its best effort to synchronize only the latest state of an object, not each version or operation for the object. For example:

- Assume that a client sends three operations to an object and that they are all committed: (1) PUT, (2) PUT, (3) DEL. The latest state of the object is (3) DEL. HCP for cloud scale only synchronizes DEL.
- Assume that a client sends three operations to an object and that they are all committed: (1) PUT, (2) DEL, (3) PUT. The latest state of the object is (3) PUT. HCP for cloud scale only synchronizes (3) PUT.

This approach does not guarantee that the latest state of an object will be in the external storage for all situations. Partly because of the "eventual consistency" offered by AWS S3 API, corner cases still exist.

Synchronization to an external bucket: high-level tasks

Synchronization to an external bucket involves assigning roles and permissions to users, creating and synchronizing the buckets, and then reading from and writing to the buckets.

This description of high-level tasks assumes three classes of user:

1. An HCP for cloud scale system administrator to create roles and assign them to users using an IdP
2. An HCP for cloud scale bucket administrator, who could be a tenant administrator, to create and configure an HCP for cloud scale bucket
3. An Amazon Web Services (AWS) user, who could be a customer, to create a remote bucket using AWS S3 and then read and write data

Note: The default HCP for cloud scale account has full permissions and can perform the tasks assigned to the first two user classes.

Procedure

1. The system administrator assigns permissions to the bucket administrator to configure bucket synchronization.
   a. In the System Management application, create a role with the permission group **bucket_sync**.
   b. In the IdP server, set up two groups: bucket administrators and bucket users.
   c. In the IdP server, register users in these groups.
   d. In the System Management application, assign the role to the bucket administrator group.
2. The bucket administrator creates local and remote buckets.
   a. In the S3 User Credentials application, generate S3 credentials.

      Tip: Use the base64 utility to encode S3 credentials.

   b. Using the S3 credentials, use an S3 API to create an HCP for cloud scale (local) bucket.
   c. Use an AWS S3 API to create an S3 (remote) bucket.

3. The bucket administrator configures bucket synchronization between the HCP for cloud scale bucket and the S3 bucket using an S3 PUT Bucket Replication method, replacing the bucket's Amazon Resource Name (ARN) with configuration settings. By using multiple rules and filters, the bucket administrator can specify what objects are synchronized to the S3 bucket.
4. The bucket administrator sets access control lists to let the bucket user write data to the HCP for cloud scale bucket.
    a. Using a management API, get the user ID of the bucket user.
    b. Using an S3 API, assign write permission to the bucket user for the HCP for cloud scale bucket.
5. The AWS user is now free to write objects to the HCP for cloud scale bucket, which is now synchronized with the remote bucket.

Synchronization from an external bucket: high-level tasks

Synchronization from an external bucket involves assigning roles and permissions to users, creating and synchronizing buckets, and then reading from and writing to the buckets.

This description of high-level tasks assumes three classes of user:

1. An HCP for cloud scale system administrator to create roles and assign them to users using an IdP
2. An HCP for cloud scale bucket administrator, who could be a tenant administrator, to create and configure an HCP for cloud scale bucket
3. An AWS user, who could be a customer, to create a remote bucket using AWS S3, create an AWS SQS queue, and then configure S3 notifications to SQS

Note: The default HCP for cloud scale account has full permissions and can perform the tasks assigned to the first two user classes.

Procedure

1. The system administrator assigns permissions to the bucket administrator to configure bucket synchronization.
    a. In the System Management application, create a role with the permission group **bucket_sync**.
    b. In the IdP server, set up two groups: bucket administrators and bucket users.
    c. In the IdP server, register users in these groups.
    d. In the System Management application, assign the role to the bucket administrator group.
2. The bucket administrator creates local and remote buckets.
    a. In the S3 User Credentials application, generate S3 credentials.

    Tip: Use the base64 utility to encode S3 credentials.

    b. Using the S3 credentials, use an S3 API to create an HCP for cloud scale (local) bucket.

   c. Use an AWS S3 API to create an S3 (remote) bucket.

3. The AWS user creates a standard queue in SQS.
   a. Using an AWS account, create a queue of the type **Standard Queue**.
   b. Create a policy document.

4. The AWS user configures the remote bucket to send S3 notifications to the AWS SQS queue.
   a. Add a notification for all object creation events to the remote bucket.

5. The bucket administrator configures bucket synchronization between the S3 bucket and the HCP for cloud scale bucket using an S3 PUT Bucket Replication method, replacing the bucket ARN with configuration settings. By using multiple rules and filters, the bucket administrator can specify what objects are synchronized to the local bucket.

6. The bucket administrator sets access control lists to let the bucket user read data from the HCP for cloud scale bucket.
   a. Using a management API, get the user ID of the bucket user.
   b. Using an S3 API, assign write permission to the bucket user for the HCP for cloud scale bucket.

7. The AWS user is now free to read objects from the HCP for cloud scale bucket, which is now synchronized with the remote bucket.

Bucket synchronization configuration

Bucket synchronization is configured using S3 `PUT bucket replication` API requests that define rules. Each bucket can have up to 1,000 rules, but all rules must be sync-to or sync-from rules. Each rule defines the following:

- External bucket settings
- A set of one or more prefixes; an object with one of the prefixes is mirrored
- A set of one or more tags; an object with all, or any, of the tags is mirrored
- For sync-from, external queue settings

Because you can configure multiple rules with multiple tags, you have flexibility in selecting objects to mirror. For example:

- To mirror all objects that contain Tag_1 and Tag_2, you can configure one rule that includes both tags.
- To mirror all objects that contain Tag_1 or Tag_2, you can configure two rules, one for each tag.

For information on `PUT bucket replication` see Configure bucket synchronization (PUT bucket replication) (on page 47).

Visibility of new buckets and objects

After they are created, buckets and objects are not immediately visible. Some client applications (such as Cloudberry Explorer) immediately retrieve the list of buckets to display the new bucket or object, which is not visible. If you create a new bucket or object and it's not immediately visible, update the list manually.

Rule collisions

HCP for cloud scale can apply multiple bucket synchronization rules to each new object so long as the destination buckets are different. This is how one-to-many synchronization is implemented.

A rule collision is when two or more rules that apply to an object have the same destination (that is, the same external host, port, and bucket). HCP for cloud scale does not allow rule collisions, so `PUT bucket replication` requests are rejected if they contain rule collisions. To avoid rule collisions, you can define as many tags in a rule as necessary, so that multiple rules with the same destination are not needed.

Effect of configuration changes

After an object operation is performed, the policy engine asynchronously checks if that object needs to be copied according to the sync-to rules. When bucket synchronization rules are created, updated, or deleted, the changes only apply to new objects, object operations, and to objects that have not been yet processed by the policy engine. Objects that existed before the rules were configured are not synchronized. If an object exists in the `PENDING` state when a rule is created, updated, or deleted, the rule change might not be applied.

Synchronizing to the same source and destination

You cannot set up bucket synchronization with the same bucket as both the source and the destination.

Configure bucket synchronization (PUT bucket replication)

You can configure S3 bucket sync-to and sync-from settings.

> **Note:**
>
> - If you use the AWS command-line interface to configure bucket synchronization, use at least `aws-cli` v1.16.211 and `aws-sdk` 1.11.610.
> - Configuration rules should be provided to AWS CLI from a file, rather than inline. This is to avoid problems with double quote characters in some terminals.

HTTP request syntax (URI)

```
aws --endpoint-url https://10.08.1019 s3api put-bucket-replication --
bucket "hcpcs_bucket" --replication-configuration file://rules.json
```

Request structure

A rule consists of up to 1000 prefixes and tag-value pairs. You can configure up to 1000 rules per bucket. Separate tag-value pairs in the rule using the keywords `"And":` or `"Or":`.

The content of the configuration JSON file is:

```
{
  "Role": "",
  "Rules": [{
    "ID": "string",
    "Filter": {
       "Prefix": "string",
       "Tag": {
         "Key": "string",
         "Value": "string"
      }
    },
    "Status": "boolean",
    "Destination": {
      "Bucket": "json"
    }
   }
   .
   .
   .
  }]
}
```

Note: S3 parameters not shown are not required, not supported, and if specified should be left empty.

Account Parameter	Required	Type	Description
Role	Yes	N/A	Not supported; leave empty.
ID	No	String	Unique identifier for rule, up to 255 characters. All rules must specify the same bucket.
Priority	Yes	Integer	Not supported; ignored.
DeleteMarkerReplication.Status	No	String	Not supported; if provided, leave as `Disabled`.
Prefix	No	String	Prefix (one per rule). Up to 1024 characters.

Account Parameter	Required	Type	Description
Key	No	String	Tag key (up to 1000 per rule). Up to 128 characters.
Value	No	String	Tag value. Up to 256 characters.
Rules.Status	Yes	Boolean	`Enabled` or `Disabled`. If `Disabled`, rule is ignored.
Rules.Destination.Bucket	Yes	Base64-encoded JSON	External S3 bucket access settings. ▪ For bucket sync-to, the settings to access the external bucket. ▪ For bucket sync-from, the settings to access the external bucket and the SQS queue settings. You can't specify the same bucket name and host as both source and destination.
Rules.Destination.Account	No	N/A	Not supported; leave empty.

Bucket sync-to structure

Bucket sync-to settings are defined by a set of parameters and passed in the value of `Rules.Destination.Bucket` as a Base64-encoded JSON structure.

The syntax inside the bucket parameter for the sync-to setting is:

```
{
  'version': 'version',
  'action': 'sync-from',
  'externalBucket': {
    'host': 'host',
    'type': 'type',
    'region': 'region',
    'remoteBucketName': 'bucket_name',
    'accessKey': 'B64_key',
    'secretKey': 'B64_key',
    'port': 'port',
    'authVersion': 'auth_version',
    'usePathStyleAlways': '[true|false]'
    },
  'notifications': {
    'type': 'type',
    'region': 'region',
    'queue': 'queue',
    'accessKey': 'B64_key',
    'secretKey': 'B64_key'
    }
}
```

Parameter	Required	Type	Description
version	Yes	String	`1.0`.
host	Yes	IP address	Host IP address.
type	Yes	String	Destination storage class: `AMAZON_S3` or `GENERIC_S3`.
region	Yes	String	The S3 region.
remoteBucketName	Yes	String	The name of the bucket, from 3 to 63 characters long, containing only lowercase characters (a-z), numbers (0-9), periods (.), or hyphens (-). The bucket must already exist.
accessKey	Yes	Base64 encoded string	The S3 access key credentials to the external S3 bucket.
secretKey	Yes	Base64 encoded string	The S3 secret key credentials to the external S3 bucket.
port	Yes	integer	Host port.
authVersion	Yes	String	AWS Signature version: `V2` or `V4`.
usePathStyleAlways	Yes	Boolean	Path-style URLs for bucket access: `true` or `false`.

Bucket sync-from structure

Bucket sync-from settings include both a bucket address and a notification queue. The settings are defined by a set of parameters and passed in the value of `Rules.Destination.Bucket` as a Base64-encoded string.

The syntax inside the bucket parameter for sync-from setting is:

```
"{
  'version': 'version',
  'action': 'sync-from',
  'externalBucket': {
    'host': 'host',
    'type': 'type',
    'region': 'region',
    'remoteBucketName': 'bucket_name',
    'accessKey': 'B64_key',
    'secretKey': 'B64_key',
    'port': 'port',
    'authVersion': 'auth_version',
    'usePathStyleAlways': '[true|false]'
    }
}"
```

Parameter	Required	Type	Description
version	Yes	String	Enter `1.0`.
host	Yes	IP address	Host IP address.
type	Yes	String	Destination storage class: `AMAZON_S3` or `GENERIC_S3`.
region	Yes	String	The S3 region.
remoteBucketName	Yes	String	The name of the bucket, from 3 to 63 characters long, containing only lowercase characters (a-z), numbers (0-9), periods (.), or hyphens (-). The bucket must already exist.
accessKey	Yes	Base64 encoded string	The S3 access key credentials to the external S3 bucket.
secretKey	Yes	Base64 encoded string	The S3 secret key credentials to the external S3 bucket.
port	Yes	integer	Host port.
authVersion	Yes	String	AWS Signature version: `V2` or `V4`.
usePathStyleAlways	Yes	Boolean	Path-style URLs for bucket access: `true` or `false`.
Destination.type	Yes	String	Always set as `AWS_SQS`.
Destination.region	Yes	String	Region of your AWS_SQS queue.
Destination.queue	Yes	String	Name of your AWS_SQS queue.
Destination.accessKey	Yes	Base64 encoded string	accessKey for permissions to read from your AWS_SQS queue.

Parameter	Required	Type	Description
Destination.secretKey	Yes	Base64 encoded string	secretKey for permissions to read from your AWS_SQS queue.

Response structure

None.

Example

Request example:

```
aws --endpoint-url https://10.08.1019 s3api put-bucket-replication --
bucket "hcpcs_bucket" --replication-configuration file://rules.json
```

Configuration rules.json:

```
{
    "ID": "sync_rule2_for_music",
    "Filter": {
      "Prefix": "/music/october/",
      "Tag": {
        "Key": "target",
        "Value": "cloud"
        }
      }
    },
    "Status": "Enabled",
    "Destination": {
      "Bucket": "{
        'version' : '1.0',
        'action' : 'sync_from',
        'externalBucket' : {
          'type' : 'AMAZON_S3',
          'region' : 'us-east-1',
          'remoteBucketName' : 'bluebucket',
          'authVersion' : 'V4',
          'usePathStyleAlways' : 'true',
          'accessKey' : 'access_key',
          'secretKey' : 'secret_key'
          },
        "notifications" : {
          "type" : "AMAZON_SQS",
          "region" : "us-east-1",
          "queue" : "testQueue",
          "accessKey" : "access_key",
          "secretKey" : "secret_key"
          }
        },
      }
    }
  }]
}
```

Get bucket synchronization rules (GET bucket replication)

You can retrieve the synchronization rules for a bucket.

HTTP request syntax (URI)

```
aws --endpoint -url https://host_ip s3api get-bucket-replication --bucket
"bucket"
```

Request structure

Not applicable.

Response structure

The response body is shown below:

```
{
  "ReplicationConfiguration": {
    "Role": "",
    "Rules": [
      {
        "Filter": {
          "And": {
            "Prefix": "string",
            "Tags": [
              {
              "Key": "string",
              "Value": "string"
        }
        .
        .
        .
      },
      "Status": "boolean",
      "Destination": {
        "Bucket": "access_settings",
      },
       "ID": "string",
     }
     ],
  }
}
```

Parameter	Required	Type	Description
Role	Yes	N/A	Not supported; empty.
Prefix	No	String	Prefix.
Key	No	String	Tag key.
Value	No	String	Tag value. Sets of prefixes and key-value pairs.
Status	Yes	Boolean	If `false`, rule is ignored.
Bucket	Yes	Base64-encoded JSON	Bucket access settings. S3 access and secret keys are masked.
ID	No	String	Unique identifier for rule, up to 255 characters.

HTTP status codes

Status code	HTTP name	Description
200	OK	The request was executed successfully.
401	Unauthorized	Access was denied due to invalid credentials.

Example

Request example:

```
aws --endpoint-url https://10.08.1019 s3api get-bucket-replication --
bucket "hcpcs_bucket"
```

JSON response:

```
{
  "ReplicationConfiguration": {
    "Role"": "",
    "Rules": [
      {
        "Filter": {
          "And": {
            "Prefix": "SQS",
            "Tags": [
              {
                "Value": "cloud",
                "Key": "target"
              }
            ]
          }
        },
        "Status": "Enabled",
        "Destination": {
          "Bucket": {
            'version': 'version',
            'action': 'sync-from',
            'externalBucket': {
              'host': 'host',
              'type': 'type',
              'region': 'region',
              'remoteBucketName': 'bucket_name',
              'port': 'port',
              'authVersion': 'auth_version',
              'usePathStyleAlways': '[true|false]'
              }
            }"
          },
        "ID": "mirrorBack_rule_for_images"
      }
    ]
  }
}
```

Get object synchronization status

The synchronization status of an object is returned in metadata as part of the response to a GET object or HEAD object request.

For a GET object or HEAD object request, the synchronization functions return a replication status header in addition to the standard response metadata. This information is useful before deletion from a source bucket to verify synchronization.

When an object is created, HCP for cloud scale evaluates the sync-to rules for the bucket. If the object matches the rules, it sets the object's sync state as `PENDING`. Most of the time, this sync state is accurate. However, it is never definitive because users may change the sync-to rules for the bucket before the policy engine starts processing the object, which happens asynchronously. The policy engine evaluates the sync-to rules again when processing an object to act according to the latest sync rules.

For example:

- An object was ingested that matches the sync-to rules, so its sync state is set as `PENDING`. Then, a user changes the sync-to rules. The object does not match the rules anymore so the object is actually not synced and that sync state is removed.
- An object was ingested that does not match the sync-to rules, so its sync state is not set. Then, a user changes the sync rules. The object now matches the rules so the object is actually synced and the sync state is set to `COMPLETED`.

Response header	Description
x-amz-replication-status	Status of synchronization: ▪ `COMPLETED`: For sync-to, all rules were successfully executed and the object was successfully synchronized. **Note**: This status is also returned for objects that match a sync-to rule but were skipped because they are not the most recent version. ▪ `PENDING`: For sync-to, one of the following: (1) a check is pending to see if the object needs synchronization; (2) the object needs synchronization, but the process is not complete. ▪ `FAILED`: For sync-to, the process has failed multiple times. To be synchronized, the object must be reloaded. ▪ `REPLICA`: For sync-from, the object is a replica created by Amazon S3.
(Header not in response)	The object did not match any rules.

Delete bucket synchronization rules (DELETE bucket replication)

You can delete S3 synchronization settings for buckets. This function is the same as in AWS S3.

HTTP request syntax (URI)

```
aws --endpoint -url https://host_ip s3api delete-bucket-replication --
bucket "bucket"
```

Request structure

None.

Response structure

None.

Example

Request example:

```
aws --endpoint-url https://10.08.1019 s3api delete-bucket-replication --
bucket "hcpcs_bucket"
```

Note: If a sync-from action fails it is retried and the SQS message about the failure is retained. To avoid a possible accumulation of SQS failure messages, the best practice is to define a suitable retention policy for SQS and to delete the sync-from rule once the desired results are obtained.

Hitachi Vantara

Corporate Headquarters
2535 Augustine Drive
Santa Clara, CA 95054 USA
HitachiVantara.com | community.HitachiVantara.com

Contact Information
USA: 1-800-446-0744
Global: 1-858-547-4526
HitachiVantara.com/contact