Object Storage Service

FAQ

Issue   14

Date   2022-12-12

HUAWEI TECHNOLOGIES CO., LTD.

Contents

1 Introduction to OBS Browser

1.1 OBS Browser Overview

☐ NOTE

> OBS Browser has been abandoned since April 15, 2020. Its functions are inherited by the new client tool OBS Browser+ that provides you with better user experience. Download the latest **OBS Browser+** We apologize for any inconvenience and appreciate your understanding.

OBS Browser is a client that runs on Windows systems. It can help you effortlessly manage object storage on your personal computers.

Table 1-1 provides personal computer (PC) specifications required by OBS Browser.

Table 1-1 PC specifications required by OBS Browser

Item	Specifications	Remarks
Operating system	- Windows 7 Pro SP1 32-bit - Windows 7 Pro SP1 64-bit - Windows 8 Pro 32-bit - Windows 10 Pro 64-bit - Windows Server 2008 R2 standard SP1 64-bit - Windows Server 2012 standard SP1 64-bit	For the login account of the Windows OS, its name can contain only digits and letters (uppercase and lowercase).
CPU	- Minimum: 2-core 1.6 GHz CPU - Standard: 2-core 2.4 GHz CPU	-

Item	Specifications	Remarks
Memory	- Minimum: 4 GB - Standard: 8 GB - Available memory: ≥ 512 MB	-
Available space of the installation CD-ROM	≥ 512 MB	Disk space required for installing OBS Browser
Available space of the data disk	≥ 3 GB	Disk space required for storing data files in the database of OBS Browser (The data files are stored in the system disk in the current version.)

OBS Browser has a flattened GUI, which is friendly to Windows users. For a better experience, use devices with a display resolution of 1280 x 1024 or higher.

You can click in the upper right corner of the page and click **About** to view the version information, or click **Online Help** to view the help document.

1.2 Function Description

Table 1-2 describes the functions of OBS Browser:

Table 1-2 Function description

Function	Description
Bucket management	Creates and deletes buckets of different storage classes in specific regions (service areas) and changes bucket storage classes.
Object management	Manages objects by uploading (including multipart uploading), downloading, deleting, sharing data through URLs, changing storage classes, and restoring Archived objects. Supports batch upload and download objects, as well as multipart upload.
Fragment management	Enables you to clear fragments that have been generated.
Permission control	OBS Browser supports permission control based on bucket policies, object policies, bucket ACLs, and object ACLs.

Function	Description
Server-side encryption	OBS allows user to encrypt objects using server-side encryption so that the objects can be securely stored in OBS.
CORS	CORS is a browser-standard mechanism provided by the World Wide Web Consortium (W3C). It defines the interaction methods between client-side web applications in one origin and resources in another origin. For general web page requests, website scripts and contents in one origin cannot interact with those in another origin because of Same Origin Policies (SOPs).
Logging	Records bucket access requests in logs for request analysis and log audit.
Lifecycle management	Supports the ability to set lifecycle rules for buckets to automatically delete expired objects.
External bucket adding	Allows you to manage data shared by other OBS users.
Task management	Allows you to upload, download, delete, and restore tasks.

2 Getting Started

2.1 Process Description

The follow-up sections describe how to complete the following tasks illustrated in **Figure 2-1**.
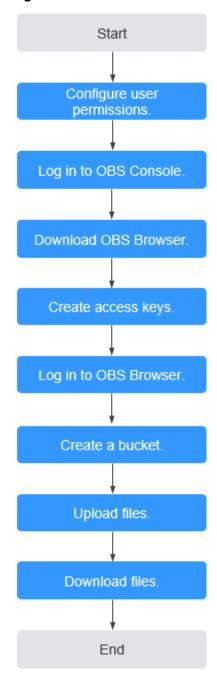
Figure 2-1 OBS Browser flowchart
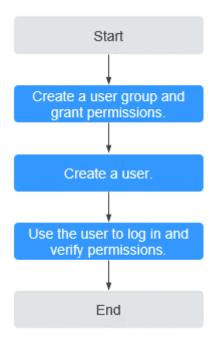
2.2 Configuring User Permissions

If your cloud service account does not need individual IAM users, then you may skip this section. Your permissions to use OBS functions are not affected.

If IAM users are required, you need to grant them access permissions on OBS, because OBS is separately deployed from other cloud resources.

Process

Figure 2-2 Process of granting an IAM user the OBS permissions

The below example describes how to grant an IAM user the **Tenant Guest** permission on OBS.

1. **Create a user group and assign permissions**.

   Create a user group on the IAM console, and assign the group the **Tenant Guest** permission.

2. **Create an IAM user and add it to the user group**.

   Create a user on the IAM console and add the user to the group created in **1**.

3. **Log in** and verify the permission granting.

   Log in to OBS Console using the newly created user, and verify that the assigned permission has taken effect:

   – Choose **Object Storage Service** from the service list to go to the OBS homepage. If the list of buckets is displayed and you can view the basic information about any bucket, but you cannot create or delete buckets, or perform any other operations, the **Tenant Guest** permission granted has already taken effect.

   – Go to an OBS bucket. If the list of objects is displayed and you can download objects, but you cannot upload or delete objects or perform any other operations, the **Tenant Guest** permission granted has already taken effect.

2.3 Logging In to OBS Console

This section describes how to log in to OBS Console using a web browser.

Procedure

Step 1 Log in to the **Huawei Cloud website**.

Step 2 Create a HUAWEI ID.

If you already have one, start from **Step 3**.

1. On the right of the top navigation bar, click **Register**.
2. Complete the creation as instructed.

   After the creation is complete, you will be navigated to your ID information page.

Step 3 On the right of the top navigation menu, click **Log In**, and enter the username and password.

Step 4 On the right of the top navigation bar, click **Console** to go to the management console.

Step 5 In the upper left corner of the navigation pane, click ☰ and choose **Storage** > **Object Storage Service**. The OBS Console page is displayed.

Step 6 It is recommended that you top up your account or subscribe to OBS by purchasing resource packages, so that the service can be used properly.

----End

2.4 Downloading OBS Browser

📖 **NOTE**

OBS Browser has been abandoned since April 15, 2020. Its functions are inherited by the new client tool OBS Browser+ that provides you with better user experience. Download the latest **OBS Browser+**. We apologize for any inconvenience and appreciate your understanding.

2.5 Creating Access Keys (AK and SK)

This section describes how to create access keys (AK and SK) in OBS Console. A pair of AK and SK is used to encrypt the signature of a request, ensuring that the request is secure and integral, and that identities of the request sender and receiver are correct.

Background Information

AKs and SKs support the authentication mechanism of Identity and Access Management (IAM).

- An Access Key ID (AK) defines a user that accesses the OBS system. An AK belongs to only one user, but one user can have multiple AKs. The OBS system recognizes the users who access the system by their AKs.

- A Secret Access Key (SK) is the key used by users to access OBS. It is the authentication information generated based on the AK and the request header. An SK and an AK group into a pair of access keys.

Limitations and Constraints

Each user can create a maximum of two valid AK/SK pairs.

Prerequisites

An account has been registered and activated.

Procedure

Step 1 Log in to OBS Console.

Step 2 On the top navigation menu, click the username and select **My Credentials**.

Step 3 On the **My Credentials** page, select **Access Keys** in the navigation pane on the left.

Step 4 On the **Access Keys** page, click **Add Access Key**.

> **NOTE**
>
> Each user can create a maximum of two valid access keys.

Step 5 In the **Add Access Key** dialog box that is displayed, enter the password and its verification code.

> **NOTE**
>
> ● If you have not bound an email address or mobile number, enter only the password.
> ● If you have bound an email address and a mobile number, you can verify through either one.

Step 6 Click **OK**.

Step 7 Save the key as prompted. The key is directly saved to the default download folder of the web browser.

> **NOTE**
>
> ● To prevent the access keys from being leaked, keep them secure. If you click **Cancel** in the dialog box, the access keys will not be downloaded, and you cannot download them later. You can re-create an access key if you need to use it.
> ● The access keys (AK and SK) need to be updated regularly.

Step 8 Open the downloaded **credentials.csv** file to obtain the access keys (AK and SK).

----End

2.6 Logging In to OBS Browser

This section describes how to log in to OBS Browser.

Procedure

Step 1 Double-click **obs.exe** to start OBS Browser.

If you are starting OBS Browser for the first time, the **Add Account** dialog box is automatically displayed. Go directly to **Step 4**.

Step 2 In the upper right corner of OBS Browser, click the account name, and select **Manage Account**.

Step 3 In the **Manage Account** dialog box that is displayed, click **Add Account**.

Step 4 In the **Add Account** dialog box, enter the account information and click **OK**. For details about account management, see **Managing Accounts**.

Information about the new account is displayed in the **Manage Account** dialog box.

Step 5 Click **OK**.

Step 6 In the upper right corner of OBS Browser, click the account name to log in the OBS Browser.

----End

Follow-up Procedure

For details about the system configuration of OBS Browser, see **Configuring the System**.

If you want to use OBS Browser to manage objects across accounts, click **Manage Account** to add account information. Then you can switch to the target account by clicking the name of that account. For details about account management, see **Managing Accounts**.

For details about the OBS Browser version update, see **Updating OBS Browser**.

2.7 Adding a Bucket

A bucket is a container that stores objects in OBS. Before you store data in OBS, you need to create a bucket first.

> 📖 **NOTE**
>
> An account (including all IAM users under this account) can create a maximum of 100 buckets and parallel file systems. You can use the fine-grained access control of OBS to properly plan and use buckets. For example, you can create folders in a bucket based on object prefixes and use **fine-grained permission control** to implement permission isolation between departments.

Procedure

Step 1 Log in to OBS Browser.
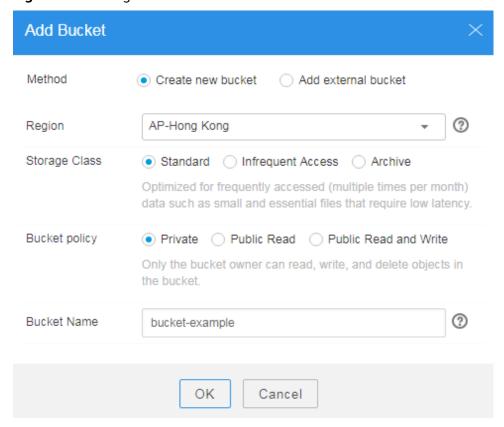
Step 2 Click **Add Bucket** on the upper left of the page. The **Add Bucket** dialog box is displayed.

Step 3 Select **Create new bucket**.

Step 4 Select a region.

Step 5 Select the Storage Class according to your actual needs.

Step 6 Bucket policy: You can select the **Private**, **Public Read**, or **Public Read and Write** policy for a bucket.

Step 7 Multiple AZs: If the multi-AZ mode is enabled, data is stored in multiple AZs. The multi-AZ mode does not apply to buckets in the Archive storage class.

Step 8 Enter a name in the text box of **Bucket Name**. You can move the pointer over next to the **Bucket Name** text box to view the naming rules for a bucket.

Figure 2-3 Adding a bucket

Step 9 Click **OK**.

Step 10 In the displayed dialog box, click **OK** to close the dialog box.

----End

2.8 Uploading a File or Folder

Background Information

- Files are uploaded in multiparts on OBS Browser. OBS Browser supports the upload of a single file with the maximum size of 48.8 TB.

- OBS Browser supports resumable transfer. If the upload task is suspended or fails, restart the task. According to the part information recorded in the task, the successfully uploaded parts will not be uploaded again, and other parts will be requested for uploading.

- If you want to classify files, you can create folders and upload files to the folders. The procedure is as follows:
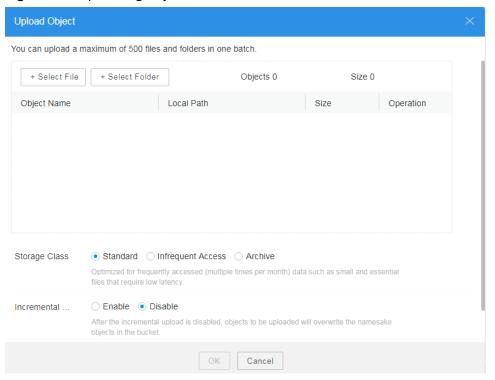
a.   Log in to OBS Browser.

b.   Click the bucket in which you want to create a folder. Click **Create Folder**.

c.   In the dialog box that is displayed, enter a folder name and click **OK**.

d.   In the displayed dialog box, click **OK** to close the dialog box.

Procedure

Step 1   Log in to OBS Browser.

Step 2   Click the bucket to which the file or folder will be uploaded.

Step 3   Click **Upload**. The **Upload Object** dialog box is displayed. For details, see **Figure 2-4**.

You can select either files or folders to upload. For details, see **Step 4** and **Step 5**.

Figure 2-4 Uploading objects

Step 4   Click **Select File**. The local file browser dialog box is displayed. Select the file that you want to upload and click **Open**.

You can upload a maximum of 500 files or folders at a time.

 NOTE

If the files that you want to upload to OBS are stored in Microsoft OneDrive, it is recommended that the names of these files contain a maximum of 32 characters to ensure compatibility.

Step 5   Click **Select Folder**, select a folder, and click **OK**.

Step 6   Select a storage class. If no storage class is selected, the object inherits the bucket storage class by default.

Step 7 Incremental Upload: If this function is enabled, an object is compared with its namesake object (if any) in the bucket before upload. The comparison includes object size, content, modification time, and other properties. The object is uploaded to overwrite the namesake object in the bucket only when any of these properties changes. Otherwise, the upload is skipped. If this function is disabled, no comparison is conducted before the upload, namesake objects are overwritten by the newly uploaded ones.

Step 8 Click **OK** to upload the file or folder.

----End

2.9 Downloading a File or Folder

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket from which you want to download a file or folder. Select a file or folder. Then right-click and select **Download**.

OBS Browser supports batch download. You can select multiple files or folders and click **Download** above the object list to download objects in batches.

📖 **NOTE**

● Before downloading **Archive** files, ensure that they are in the **Restored** state.

Step 3 In the dialog box that is displayed, select a path to save the file or folder and click **OK**.

If name-sake files exist in the target path, a message will be displayed indicating "Files with the same name exist in the target path. Select the saving method." You can choose to **Keep All** or **Overwrite**.

----End

2.10 Deleting a File or Folder

Deleting a folder will delete all files in the folder. Ensure that all files in a folder are not needed before deleting the folder.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Select the object that you want to delete and choose **More** > **Delete** on the right.

OBS Browser supports batch deletion. You can select multiple files or folders and click **Delete** above the list to delete them in batches.

Step 3 In the displayed dialog box, click **OK**.

----End

2.11 Deleting a Bucket

You can delete unwanted buckets on OBS Console to free up the quota of buckets.

Prerequisites

- All objects in the bucket have been deleted.
- This operation can be performed only by the bucket owner.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the blank area in the row of the bucket and choose **More** > **Delete**.

> 📖 **NOTE**
>
> The name of a deleted bucket can be reused for a bucket or a parallel file system at least 30 minutes after the deletion.

Step 3 In the displayed dialog box, click **OK**.

Step 4 In the displayed dialog box, click **OK** to close the dialog box.

----End

3 Storage Classes Overview

OBS supports tiered storage classes at the bucket level and object level.

OBS provides the following storage classes: Standard, Infrequent Access, and Archive.

Different storage classes meet different requirements for storage performance and costs.

- The Standard storage class features low access latency and high throughput. It is therefore suitable for storing a massive number of hot files (frequently accessed every month) or small files (less than 1 MB). The application scenarios include big data analytics, mobile apps, hot videos, and social apps.
- The Infrequent Access storage class is ideal for storing data that is semi-frequently accessed (less than 12 times a year), with requirements for quick response. The application scenarios include file synchronization, file sharing, and enterprise backup.
- The Archive storage class is suitable for archiving data that is rarely-accessed (averagely once a year). The application scenarios include data archiving and long-term data backups. The Archive storage class is secure, durable, and inexpensive, and can be used to replace tape libraries. To keep cost low, it may take hours to restore data from the Archive storage class.

Bucket Storage Classes vs. Object Storage Classes

When an object is uploaded, it inherits the storage class of the bucket by default, but you can change the default storage class when you upload the object.

Changing the storage class of a bucket does not change the storage classes of existing objects in the bucket, but newly uploaded objects will inherit the new storage class.

Comparison of Storage Classes

Compared Item	Standard	Infrequent Access	Archive
Feature	Top-notch performance, highly reliable and available	Reliable, inexpensive, and real-time storage access	Long-term storage for archived data at a very low cost
Application scenarios	Cloud application, data sharing, content sharing, and hot data storage	Web disk applications, enterprise backup, active archiving, and data monitoring	Archive, medical image storage, video material storage, and replacement of tape libraries
Designed durability	99.999999999%	99.999999999%	99.999999999%
Designed durability (multi-AZ)	99.9999999999%	99.9999999999%	-
Designed availability	99.99%	99%	99%
Designed availability (multi-AZ)	99.995%	99.5%	-
Minimum storage duration	Not required	30 days	90 days
Data retrieval	N/A	Billed for each GB retrieved.	Data can be restored at a standard or an expedited speed. Billed for each GB restored.
Image processing	Supported	Supported	Not supported

4 Managing Buckets
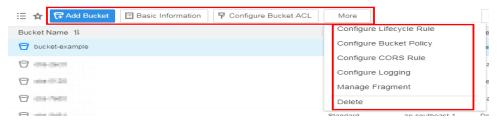
4.1 Bucket Management Methods

OBS Browser supports three methods of bucket management.

1. You can click the button above the bucket list to manage buckets.
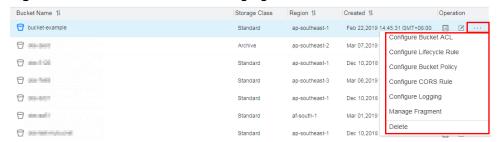
   Also, you can click the blank area in the row where the bucket to be managed resides, and you can manage the bucket using the buttons displayed above the bucket list. For details, see **Figure 4-1**.

   Figure 4-1 First method of managing buckets

2. Click the operation buttons in the row of the bucket to be managed.

   Click the operation buttons in the row of the bucket to be managed. For details, see **Figure 4-2**.

   Figure 4-2 Second method of managing buckets

3. Right-click the blank area in the row where the bucket to be managed resides.

   Right-click the blank area in the row where the bucket to be managed resides. A shortcut menu is displayed for managing the bucket. For details, see **Figure 4-3**.
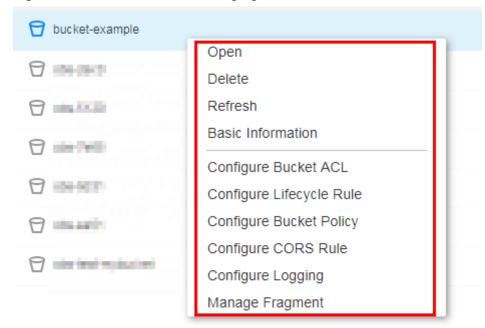
Figure 4-3 Third method of managing buckets

4.2 Adding a Bucket

Buckets are containers that store objects in OBS. Before you store data in OBS, you need to create buckets.

📖 **NOTE**

> An account (including all IAM users under this account) can create a maximum of 100 buckets and parallel file systems. You can use the fine-grained access control of OBS to properly plan and use buckets. For example, you can create folders in a bucket based on object prefixes and use **fine-grained permission control** to implement permission isolation between departments.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click **Add Bucket** on the upper left of the page. The **Add Bucket** dialog box is displayed.

Step 3 Set the bucket parameters, as listed in **Table 4-1**.

Figure 4-4 Adding a bucket
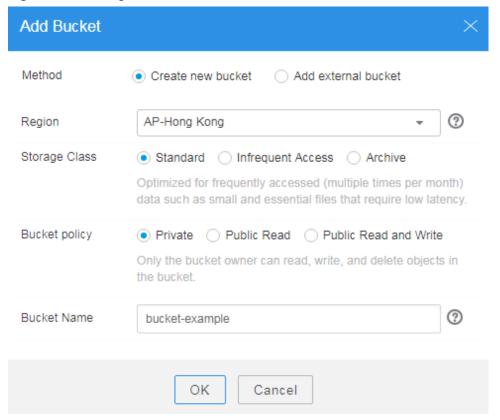
Table 4-1 Creating a Bucket

Parameter	Description
Method	Select **Create new bucket**.
Region	Region where the bucket to be created is located.
Storage Class	OBS offers three storage classes: standard, infrequent access, and archive.
	● Standard storage class: applies to storing frequently accessed (multiple times per month) hot or small files that require quick response.
	● Infrequent Access storage class: applies to storing semi-frequently accessed (less than 12 times a year) data requiring quick response.
	● Archive storage class: applies to archiving rarely-accessed (once a year) data without high requirements on response speed.
Bucket Policy	You can select the **Private**, **Public Read**, or **Public Read and Write** policy for the bucket.
Multi-AZ Mode	If the multi-AZ mode is enabled, data is stored in multiple AZs. The multi-AZ mode does not apply to buckets in the Archive storage class.

Parameter	Description
Bucket Name	Name the bucket according to the globally applied DNS naming regulation as follows: • The name must be globally unique in OBS. • The name must contain 3 to 63 characters. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. • The name cannot start or end with a period (.) or hyphen (-), and cannot contain two consecutive periods (.) or contain a period (.) and a hyphen (-) adjacent to each other. • The name cannot be an IP address. • If the name contains a period (.), certificate verification will fail when you access OBS through HTTPS using a virtual host.

 NOTE

- When a URL is used to access a bucket, the bucket name will become a part of the URL. According to the DNS rule, URLs do not support uppercase letters and cannot be used to access a bucket whose name contains uppercase letters. Therefore, a bucket name can contain only lowercase letters, digits, hyphens (-), and periods (.) For example, if you attempt to access bucket **MyBucket** using the URL, bucket **mybucket** will be accessed instead, causing an access error.

- DNS naming can standardize the bucket naming globally, facilitating the resolution when accessing a bucket. Users can benefit from new functions and optimized features, and static website hosting is then applicable to buckets.

- Once you create a bucket, you cannot change the name of it. Make sure that the bucket name you set is appropriate.

Step 4 Click **OK**.

Step 5 In the displayed dialog box, click **OK** to close the dialog box.

----End

Region Information Configuration

The **Region** information can be configured on OBS Browser. The following details the configuration procedure:

1. Open file **region** in folder **OBS Browser** in the decompression path of OBS Browser.

2. Change the value of parameter **options** in file **region**.

   Enter the region information to be added to the end of parameter **options** in the following format:

   {"key":"Region alias","value":"Region"}

   The newly added information must be in the JSON format. The following describes the parameters.

- **key**: indicates a user-defined region alias. Its value is in the **Region** drop-down list in the **Add Bucket** dialog box. For a convenient view, you are advised to enter not more than 25 characters.

- **value** indicates region. Enter the value based on regions supported by OBS.

  Each time when a **Region** is added, a group of values will be added to **options**, that is, **{"key":"***Region alias***","value":"***Region***"}**. Groups of values are separated by commas (,). The following provides two configuration examples of newly added **region01** and **region02**. Keep the values of other parameters in file **region** unchanged.

  "options":[{"key":"test_region01","value":"region01"},
  {"key":"test_region02","value":"region02"}]

3. After file **region** is successfully modified, restart OBS Browser so that the configurations can take effect.

Related Operations

You can specify its storage class when creating a bucket or change its storage class after bucket creation. The procedure is as follows:

Step 1 Log in to OBS Browser.

Step 2 Select a bucket from the bucket list and click on the right. The **Change Storage Class** dialog box is displayed.

Step 3 Select the desired storage class and click **OK**.

> **NOTE**
>
> - Changing the storage class of a bucket does not change the storage class of existing objects in the bucket.
>
> - An object inherits the bucket storage class by default, if no other storage class is specified for the object upon its upload. When the bucket storage class is changed, newly uploaded objects inherit the new bucket storage class by default.

Step 4 In the displayed dialog box, click **OK** to close the dialog box.

----End

4.3 Searching for a Bucket

This section describes how to search for a bucket by entering characters contained in its name on OBS Browser.

Procedure

Step 1 Log in to OBS Browser.

Step 2 In the search box in the upper right corner on the main page, enter characters contained in the name of the desired bucket.

Step 3 Click .

The found buckets are displayed in the bucket list. If you want to search for another bucket, click to delete the keyword you entered and enter a bucket name again.

If you want to leave the object list page and quickly switch to another bucket, you can click on the left side to expand the hidden bucket list and search for the desired bucket in the search box above the bucket list.

For example, if you want to search for buckets whose names contain **test**, you only need to enter **test** in the search box in the upper right corner on the main page and click . Then, all buckets that contain **test** in their names are displayed in the bucket list.

----End

4.4 Viewing Basic Information of a Bucket

This section describes how to view basic information about a bucket, including the owner, capacity, location, and object quantity using OBS Browser.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the blank area in the row of the bucket about which you want to query the basic information and click **Basic Information**.

Step 3 The **Basic Information** window is popped up displaying related information. For details, **Figure 4-5**.

Figure 4-5 Basic information about the bucket
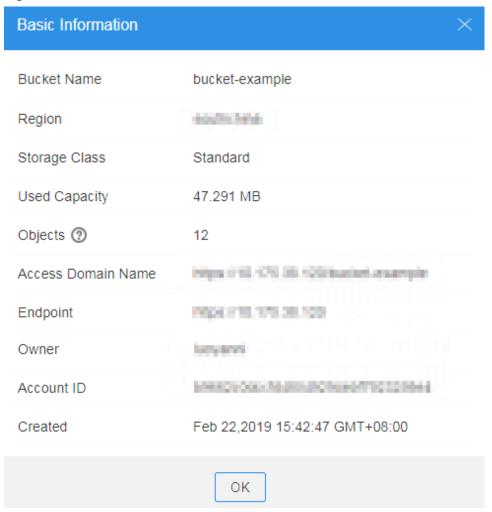
Table 4-2 Parameter description

Parameter	Description
Bucket Name	Name of the bucket.
Region	Region where the bucket is stored.
Storage Class	Indicates the storage class of the bucket, which can be **Standard**, **Infrequent Access**, or **Archive**.
Used Capacity	Total capacity used by objects in the bucket. By default, the bucket storage capacity is unlimited. You can set a quota to put an upper limit to the bucket storage capacity. For details, see **PutBucketQuota**.
Objects	The number of objects is the sum of folders and files (including the latest versions and historical versions).
Access Domain Name	Indicates the access domain name of the bucket, which is in the format of *Bucket name.Domain name*.

Parameter	Description
Endpoint	Indicates the domain name of the region where the bucket resides. OBS provides an endpoint for each region, facilitating users to access resources in each region.
Multi-AZ Mode	Status of the multi-AZ mode. If the multi-AZ mode is enabled, data is stored in multiple AZs. The multi-AZ mode does not apply to buckets in the Archive storage class.
Owner	Bucket owner is the account that creates the bucket
Account ID	Unique identity of the bucket owner. It is the same as **Account ID** on the **My Credentials** page.
Created	Time when the creation of a bucket is completed.

☐ **NOTE**

The statistics of **Used Capacity** and **Objects** are not real-time data, which are usually updated 15 minutes in delay.

----End

4.5 Managing Fragments

Background Information

Data can be uploaded to OBS using multipart uploads. Fragments are generated, if a multipart upload fails because of the following reasons (included but not limited to):

- The network is in poor condition, and connection to the OBS server is frequently interrupted.
- The upload task is manually suspended.
- The device is faulty.
- The device is powered off suddenly.

If a file fails to be uploaded or the upload task is suspended, fragments are generated and stored in OBS. You can resume the upload through task management. After the resumable upload completes, the fragments will be cleared automatically.

You can also use the fragment management function to clear fragments. If you resume an upload task after clearing the fragments, the upload progress will be lost and the task needs to be re-uploaded.

NOTICE

The fragment storage in OBS is billed.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the blank area in the row of the bucket and choose **More** > **Manage Fragment**.

Step 3 In the **Manage Fragment** dialog box, click **Check** to refresh the fragment list.

Select a fragment and click   🗑   on the right to delete it.

You can click **Delete All** above the fragment list to delete all the fragments.

Step 4 In the dialog box that is displayed, confirm the information and click **Yes**.

Step 5 In the displayed dialog box, click **OK** to close the dialog box.

Step 6 In the **Manage Fragment** dialog box, click **Close** to close the dialog box and return to the OBS Browser home page.

----End

4.6 Deleting a Bucket

You can delete unwanted buckets on OBS Console to free up the quota of buckets.

Prerequisites

- All objects in the bucket have been deleted.
- This operation can be performed only by the bucket owner.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the blank area in the row of the bucket and choose **More** > **Delete**.

📖 **NOTE**

The name of a deleted bucket can be reused for a bucket or a parallel file system at least 30 minutes after the deletion.

Step 3 In the displayed dialog box, click **OK**.

Step 4 In the displayed dialog box, click **OK** to close the dialog box.

----End

5 Managing Objects

5.1 Object Management Methods

OBS Browser supports three methods of object management.

1. You can use the operation buttons above the object list to manage objects.

   Click the blank area in the row where the object to be managed resides. You can manage the object using the buttons displayed in the upper part. For details, see **Figure 5-1**.
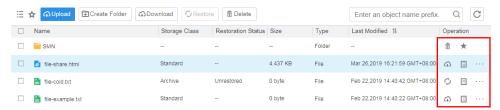
   Figure 5-1 First method of managing objects

2. Use the operation button on the right of the row where the object is located to manage the object.

   Click the operation buttons in the row of the object to be managed. For details, see **Figure 5-2**.

   Figure 5-2 Second method of managing objects

3. Use the shortcut menu to manage objects.

   Right-click the blank area in the row where the object to be managed resides. A shortcut menu is displayed for managing the object. For details, see **Figure 5-3**.
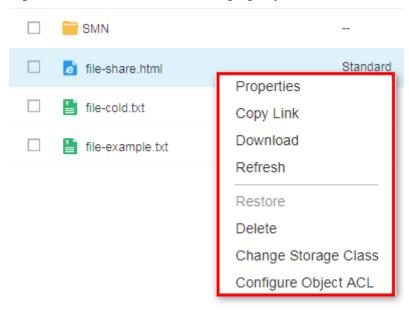
Figure 5-3 Third method of managing objects

5.2 Creating a Folder

This section describes how to create a folder using OBS Browser.

Background Information

Unlike the file system, OBS does not involve the concepts of file and folder. For easy data management, OBS provides a method to simulate folders. On OBS, an object is simulated as a folder by adding a slash (/) to the object name on OBS Browser.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket in which you want to create a folder. Click **Create Folder**.

Step 3 In the dialog box that is displayed, enter a folder name and click **OK**.

- You can create single-level or multi-level folders.
- The name cannot contain the following special characters: \:*?"<>|+
- The name cannot start or end with a period (.) or slash (/).
- The absolute path of the folder cannot exceed 1023 characters.
- Any single slash (/) separates and creates multiple levels of folders at once.
- The name cannot contain two or more consecutive slashes (/).

Step 4 In the dialog box that is displayed, click **OK**.

----End

5.3 Uploading a File or Folder

Background Information

Files are uploaded in multiparts on OBS Browser. OBS Browser supports the upload of a single file with the maximum size of 48.8 TB.

OBS Browser supports resumable transfer. If the upload task is suspended or fails, restart the task. According to the part information recorded in the task, the successfully uploaded parts will not be uploaded again, and other parts will be requested for uploading.

If batch upload is required for tens of thousands of objects, it is recommended that you use obsutil. For details, see **Uploading an Object**

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket to which the file or folder will be uploaded.

Step 3 Click **Upload**. The **Upload Object** dialog box is displayed. For details, see **Figure 5-4**.

You can select either files or folders to upload. For details, see **Step 4** and **Step 5**.
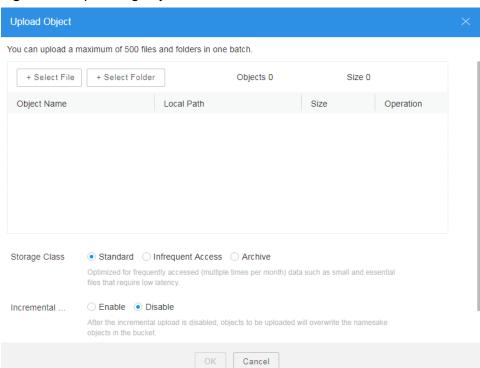
Figure 5-4 Uploading objects

Step 4 Click **Select File**. The local file browser dialog box is displayed. Select the file that you want to upload and click **Open**.

You can upload a maximum of 500 files or folders at a time.

📖 **NOTE**

If the files that you want to upload to OBS are stored in Microsoft OneDrive, it is recommended that the names of these files contain a maximum of 32 characters to ensure compatibility.

Step 5 Click **Select Folder**, select a folder, and click **OK**.

Step 6 Select a storage class. If no storage class is selected, the object inherits the bucket storage class by default.

Step 7 Incremental Upload: If this function is enabled, an object is compared with its namesake object (if any) in the bucket before upload. The comparison includes object size, content, modification time, and other properties. The object is uploaded to overwrite the namesake object in the bucket only when any of these properties changes. Otherwise, the upload is skipped. If this function is disabled, no comparison is conducted before the upload, namesake objects are overwritten by the newly uploaded ones.

Step 8 Click **OK** to upload the file or folder.

----End

Related Operations

You can modify the storage class of the object after uploading the file or folder.

Step 1 Log in to OBS Browser.

Step 2 In the bucket list, click the bucket name.

Step 3 Select the target object and choose **More** > **Change Storage Class** on the right.

Step 4 Select the desired storage class and click **OK**.

Step 5 In the displayed dialog box, click **OK** to close the dialog box.

----End

📖 **NOTE**

- Objects can be changed from Standard to the Infrequent Access or Archive storage class, or from Infrequent Access to the Standard or Archive storage class. Archive objects must be restored before they can be changed to the Standard or Infrequent Access storage class. Changing from Infrequent Access or Archive to other storage classes incurs restore fees. Select an appropriate change option based on your actual needs.
- When the storage class is changed to Archive, the object restore status changes to **Unrestored**.
- You can also configure a lifecycle rule to change the storage class of an object. For details, see **Lifecycle Management Overview**.

Follow-up Procedure

You can click **More** > **Copy Link** on the right of an object to copy the URL of the object.

5.4 Downloading a File or Folder

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket from which you want to download a file or folder. Select a file or folder. Then right-click and select **Download**.

OBS Browser supports batch download. You can select multiple files or folders and click **Download** above the object list to download objects in batches.

📖 **NOTE**

- Before downloading **Archive** files, ensure that they are in the **Restored** state.

Step 3 In the dialog box that is displayed, select a path to save the file or folder and click **OK**.

If name-sake files exist in the target path, a message will be displayed indicating "Files with the same name exist in the target path. Select the saving method." You can choose to **Keep All** or **Overwrite**.

----End

5.5 MD5 Verification

Background Information

During file upload and download, the file may be damaged due to network issues. Through MD5 verification, you can verify the correctness of uploaded or downloaded documents.

Enabling MD5 verification on OBS Browser will affect upload and download speeds. This section describes the MD5 verification function using file upload as an example.

📖 **NOTE**

MD5 verification is performed only when the file to be downloaded contains MD5 information.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click in the upper right corner of the page and choose **System Configuration**.

Step 3 Choose **System Configuration** > **Basic Configuration**. In the dialog box that is displayed, select **Enable MD5 verification**, as shown in **Figure 5-5**.
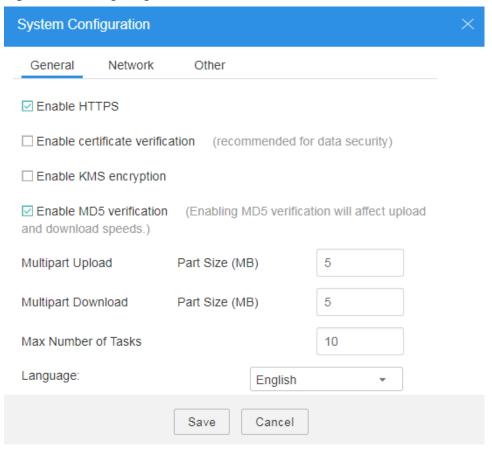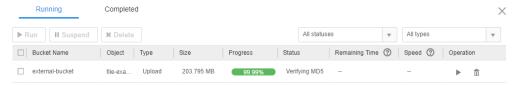
Figure 5-5 Configuring MD5 verification

Step 4 Click **Save**.

Step 5 Select the bucket to which the file is to be uploaded and upload the file.
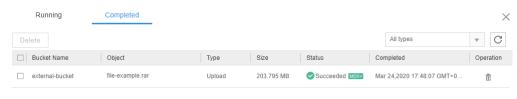
In the upper right corner of the page, click [icon]. Then in the running task list, you can view the file uploading status and MD5 verification status. For details, see **Figure 5-6**.

Figure 5-6 MD5 verification statuses

Step 6 After the file is uploaded, you can view the MD5 verification result in the completed task list, as shown in **Figure 5-7**.

Figure 5-7 MD5 verification results

----End

5.6 Renaming a File

Renaming a file is implemented by copying the file and saving it in another name, and then deleting the original file.

Prerequisites

You must have the permission to read the ACL of the object to be renamed.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket from which you want to copy a file or folder.

Step 3 Right-click the object to be renamed and choose **Rename** from the shortcut menu.

Step 4 In the **Rename** dialog box that is displayed, enter a new name for the object and click **Save**.

The status of the renaming task is displayed on the task management page.

----End

📖 **NOTE**

Renaming an object restored from the Archive storage class will change the storage class of the object to Standard.

If an object of the Standard storage class is stored in an Archive bucket, renaming the object changes its storage class to Archive.

5.7 Copying a File or Folder

This section describes how to copy a file or folder stored on OBS from the source path to the target path.

If batch copying is required for tens of thousands of objects, it is recommended that you use obsutil. For details, see **Copying an Object**.

Prerequisites

You must have the permission to read the ACL of the object to be renamed.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket from which you want to copy a file or folder.

Step 3 Then, right-click the file or folder and choose **Copy** from the shortcut menu.

Step 4 Select a path to save the file or folder. Right-click the path and choose **Paste** from the shortcut menu.

📖 **NOTE**

- You can only copy a file or folder in the current region.
- The source path and target path must be different.

----End

5.8 Moving a File or Folder

Moving a file or folder is implemented by copying the file or folder to a specified location, and then deleting the original file or folder.

If batch moving is required for tens of thousands of objects, it is recommended that you use obsutil. For details, see **Copying an Object** and **Deleting an Object**.

Prerequisites

You must have the permission to read and write the ACL of the object to be renamed.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket from which you want to copy a file or folder.

Step 3 Then, right-click the file or folder and choose **Copy** from the shortcut menu.

Step 4 Select a path to save the file or folder. Right-click the path and choose **Paste** from the shortcut menu.

📖 **NOTE**

- You can only move a file or folder in the current region.
- The source path and target path must be different.

----End

5.9 Sharing a File

Scenarios

You can allow other users to access a file stored in OBS by sharing the temporary URL of the file.
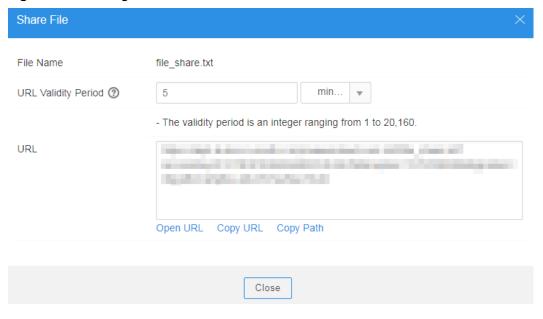
Limitations and Constraints

- An object shared from OBS Browser can be valid for one minute to 30 days. Or, you can configure a **bucket policy or object policy** to grant the access permissions of a folder to other users permanently.

- Only buckets of version 3.0 or later support the folder sharing function. You can view the bucket version in the **Basic Information** area on the **Overview** page of a bucket.

- Encrypted objects cannot be shared.

- To share an archived object, restore it first.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket to be operated.

Step 3 Select the file to be shared and click ⮐ . The **Share File** dialog box is displayed.

Figure 5-8 Sharing File

Step 4 URL related operations

- Click **Open URL** to preview the file on a new page or directly download it to your default download path.

- Click **Copy URL** to share the link to other users, so that they can enter the link to a web browser to access the file.

- You can also click **Copy Path** to share the file path to users who have access permissions to the bucket. Then the users can search for the file by pasting the path to the search box of the bucket.

📖 **NOTE**

Within the validity period of a URL, any user who has the URL can access the file.

Step 5 Click **Close** to close the dialog box.

----End

5.10 Sharing a Folder

Scenarios

With the folder sharing function, you can share your folder in OBS to other users.

Background Information

Folder sharing is temporary and has a validity period. You need to prepare a six-digit extraction code before sharing a folder. After the sharing link of the folder is created, OBS automatically aggregates the download links of all objects in the folder to a static website that is hosted by a public OBS bucket. Then users who have the temporary URL and extraction code can access the static website and download files.
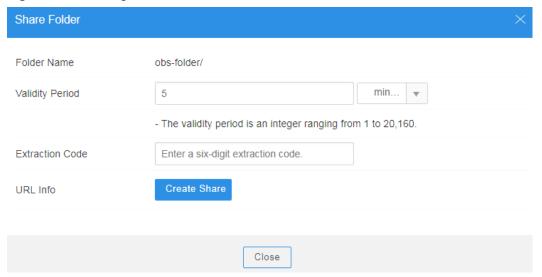
Limitations and Constraints

- A folder shared from OBS Browser can be valid for one minute to 30 days. Or, you can configure a **bucket policy or object policy** to grant the access permissions of a folder to other users permanently.
- The folder sharing function is restricted to a few regions only.
- Only buckets of version 3.0 or later support the folder sharing function. You can view the bucket version in the **Basic Information** area on the **Overview** page of a bucket.
- Archived objects in the folder need to be restored in the bucket before they can be downloaded.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket to be operated.

Step 3 Select the folder to be shared and click   . The **Share Folder** dialog box is displayed.

Figure 5-9 Sharing a folder

Step 4 Configure parameters.

Table 5-1 Parameters for sharing a folder

Parameter	Description
URL Validity Period	The validity period is measured by minutes or hours, and ranges from one minute to 30 days. The default value is five minutes. Within the URL validity period, any user who has the URL can access the folder.
Extraction Code	A six-digit code. An extraction code is required to access a shared folder.

Step 5 Click **Create Share** to generate sharing URL of the folder.

Step 6 You can send the URL and extraction code to other users. Then they can access the folder using the URL and the extraction code.

----End

Verification

Verify that other users can access the shared folder through the URL.

Step 1 Open a web browser, enter the shared URL, and open it.

Step 2 In the dialog box that is displayed, enter the extraction code and confirm the access to the shared folder.

----End

Verify that other users can access the shared folder through OBS Browser.

Step 1 Log in to OBS Browser.

Step 2 Click **Obtain Share** on the upper right corner the page. The **Add Account** dialog box is displayed.

Step 3 On the **Obtain Share** page, enter the copied link and extraction code.

Step 4 Click **OK** to access the shared folder.

----End

5.11 Accessing an Object Using Its Object URL

The object uniform resource locator (URL) (object sharing) function allows anonymous users to access object data using object URLs.

Prerequisites
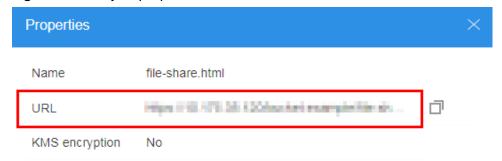
An anonymous user has been assigned with the permission to read the specified object.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket to be operated.

Step 3 Click the bucket for which you want to configure the object URL function, and click ▤ next to the object to be shared to view the object URL. For details, see **Figure 5-10**.

If you select **OBS** when logging in to OBS Browser, the object URL is in the format of https://*bucket name.domain name*/*directory level*/*object name*. If the object is in the root directory of the bucket, the URL does not contain a *directory level*.

Figure 5-10 Object properties

- If you select **Other object storage services** when logging in to OBS Browser, the object URL is in the format of https://*storage server IP address* or *domain name*/*bucket name*/*directory level*/*object name*. If the object is in the root directory of the bucket, the URL does not contain a *directory level*.

- If you select **OBS** when logging in to OBS Browser, the object URL is in the format of https://*bucket name.domain name*/*directory level*/*object name.* If the object is in the root directory of the bucket, the URL does not contain a *directory level*.

□ NOTE

To allow anonymous users to access objects stored in a bucket of Archive storage class using the URL, ensure that the objects are in the **Restored** state.

The method of using a browser to access objects varies depending on the object type. You can directly open **.txt** and **.html** files using a browser. However, when you open **.exe** and **.dat** files using a browser, the files are automatically downloaded to your local computer.

Step 4 Click ⬚ to copy the URL of the object

Step 5 In the displayed dialog box, click **OK** to close the dialog box.

Step 6 Paste the object URL in the address box of a web browser. Press **Enter**, and then you can access the object.

----End

5.12 Searching for a File or Folder

This section describes how to search for a file or folder by name prefix on OBS Browser.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket where the target file or folder resides.

Step 3 In the search box above the object list, enter the name prefix of the file or folder that you want to search for.

In the root directory of the bucket, files and folders whose name starts with the specified prefix are displayed.

□ NOTE

If you want to search for objects within a folder, you can use either of the following methods:

- In the search box of the root directory, enter *folder path*/*object name prefix*. For example, if you enter **abc/123/example**, all files and folders whose name is prefixed with **example** in the **abc/123** folder are displayed.

- Alternatively, you can open the specific folder, and enter the object name prefix in the search box of that folder. For example, you can open the **abc/123** folder and enter **example** in the search box. Then all files and folders whose name is prefixed with **example** in the **abc/123** folder are displayed.

Step 4 Click 🔍.

----End

5.13 Deleting a File or Folder

This section describes how to delete a file or folder.

Background Information

Deleting a folder will delete all files in the folder. Ensure that all files in a folder can be deleted before deleting the folder.

Deleting unnecessary files or folders saves space and costs.

You can use the lifecycle management function to implement periodic deletion of unneeded files or batch delete all files or folders in OBS. For details, see **Lifecycle Management Overview**.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Select the object that you want to delete and choose **More** > **Delete** on the right.

OBS Browser supports batch deletion. You can select multiple files or folders and click **Delete** above the list to delete them in batches.

Step 3 In the displayed dialog box, click **OK**.

----End

5.14 Restoring an Archive File in OBS

You need to restore an Archive object before downloading it.

Background Information

The Archive storage class is applicable to archiving rarely-accessed (such as once a year) data. The application scenarios include data archiving and long-term data retention for backup, allowing users to safely store data at a low price. However, it can take up to hours to restore data from OBS **Archive**.

If an **Archive** object is being restored, you cannot suspend or delete the restoration task.

You cannot restore an object that is in the **Restoring** state.

For details about data restore duration and pricing, see **Product Pricing Details**.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket in which the file that you want to restore resides. The object list is displayed.

Step 3 Click the restore icon next to the object that you want to restore. Alternatively, you can select an object and click the restore icon on the top of the object list.

You can also select multiple files or folders and click **Restore** above the list to restore them.

☐ NOTE

Objects that are being restored cannot be added for batch restoration.
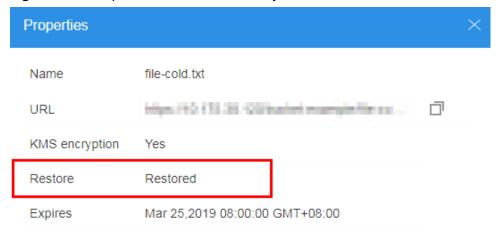
Table 5-2 Parameters for restoring objects

Parameter	Description
Validity Period	How long the object will remain in the **Restored** state. It starts once the object is restored. The value is an integer ranging from 1 to 30 (days). The default value is **30**. For example, if you set **Validity Period** to **20** when restoring an object, 20 days after the object is successfully restored, its status will change from **Restored** to **Unrestored**.
Speed	How fast an object will be restored. ● **Expedited**: Archive objects can be restored within 1 to 5 minutes, and Deep Archive objects can be restored within 3 to 5 hours. ● **Standard**: Archive objects can be restored within 3 to 5 hours, and Deep Archive objects can be restored within 5 to 12 hours.

Step 4 Click **OK**.

Click ⊟ and the **Properties** dialog box is displayed. For details, see **Figure 5-11**. You can view the restoration status.

Figure 5-11 Properties of the restored object

You can download the file only after its status changes to **Restored**. You can click the **Refresh** button in the upper right corner to refresh the restoration tasks and

to view the restoration progress. The system also automatically refreshes the restoration tasks every 5 minutes.

📖 **NOTE**

> The system checks the file restore status at UTC 00:00 every day. The system starts counting down the expiration time from the time when the latest check is complete.

----End

Related Operations

Within the validity period of a restored object, you can restore the object again. The validity period is then extended because it will start again when the latest restore is complete.

📖 **NOTE**

> If a restored object is restored again, its expiration time should be later than the time set for the previous restore. Assume that an object is restored on January 1 and will expire 30 days later (on January 30). If the object is restored again on January 10 and is made to be expired earlier than January 30 (less than 20 days later), this restore action is considered invalid.

6 Server-Side Encryption

6.1 Server-Side Encryption Overview

After server-side encryption is enabled, objects to be uploaded will be encrypted and stored on the server. When downloading the encrypted objects, the encrypted data will be decrypted on the server and displayed in plaintext to users.

Key Management Service (KMS) uses Hardware Secure Modules (HSMs) to ensure key security, enabling users to easily create and manage encryption keys. Keys are not displayed in plaintext outside HSMs, which prevents key disclosure. All operations performed on keys are controlled and logged, and usage of all keys is recorded, meeting regulatory compliance requirements.

OBS Browser supports server-side encryption with KMS-managed keys (SSE-KMS). In the SSE-KMS mode, OBS uses the keys provided by KMS for server-side encryption.

With OBS Browser, you can make API calls to use keys provided by KMS for server-side encryption.

6.2 Uploading a File with Server-Side Encryption

OBS allows users to encrypt objects on the server side so that the objects can be securely stored in OBS.

Prerequisites

In the region where the OBS is deployed, add the **KMS Administrator** permission to the user group. For details about how to add permissions, see the *IAM User Guide*.

Procedure

Step 1 Log in to OBS Browser.
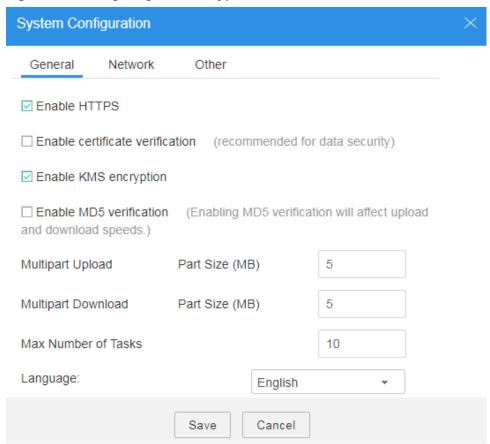
Step 2 In the upper right corner on the page, click .

Step 3   Choose **System Configuration** > **General**. For details, see **Figure 6-1**.

Figure 6-1 Configuring KMS encryption

Step 4   Select **Enable HTTPS** and **Enable KMS encryption**.

Step 5   Click **Save**.

Step 6   Verify the encryption status.
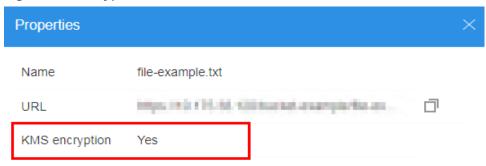
After HTTPS and KMS encryption are enabled, objects uploaded to OBS are encrypted with keys provided by KMS. By default, the key **obs/default** is selected for encryption.

After objects are uploaded, click 🗒 on the right of the object list. In the **Properties** dialog box that is displayed, you can view the object encryption status. **Yes** indicates that server-side encryption has been implemented for the object. **No** indicates that server-side encryption has not been implemented for the object. The object encryption status cannot be changed.

📖 **NOTE**

HTTPS must be enabled when you enable KMS encryption to upload objects. Therefore, if you deselect **Enable HTTPS**, **Enable KMS encryption** is deselected automatically.

Figure 6-2 Encryption status

☐ **NOTE**

- Server-side encryption does not support HTTP. To use server-side encryption, enable HTTPS.
- A key in use cannot be deleted. Otherwise, the object encrypted with this key cannot be downloaded.

----End

7 Permission Control

7.1 Overview

OBS Browser supports permission control based on bucket policies, object policies, bucket ACLs, and object ACLs.

- Bucket policy: A bucket policy applies to the configured OBS bucket and objects in the bucket. An OBS bucket owner can use a bucket policy to grant permissions of buckets and objects in the buckets to IAM users or other accounts.

- Object policy: An account can use an object policy to assign IAM users under the account the read and write permissions to a specific object.

- Access Control List (ACL): OBS provides ACL settings at bucket and object levels. Bucket and object ACLs are attached to accounts.

Bucket Policy

A bucket owner can edit a bucket policy to implement fine-grained bucket access control.
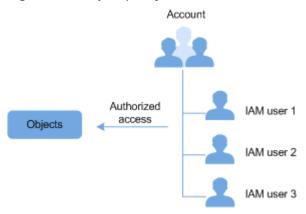
A bucket policy can be used to control access to the bucket and objects in the bucket. Specifically, you can define the effect, authorized users, resources, actions, and conditions of a bucket policy. Permissions attached to a bucket apply to all the objects in the bucket. After a bucket policy is created, access requests to the bucket are controlled by the bucket policy. The bucket policy controls access requests by allowing or denying the requests.

For details, see the **Policy Format**.

Object Policy

By configuring object policies, you can control the access permissions of other IAM users under the current account to the authorized objects.

Figure 7-1 Object policy

ACLs

A bucket or object ACL can assign the following users the read and write permissions to OBS resources:

Table 7-1 Users supported by OBS

Principal	Description
Owner	The owner of a bucket is the account that created the bucket. The bucket owner has all bucket access permissions by default. The read and write permissions to the bucket ACL are permanently available to the bucket owner, and cannot be modified.
	The owner of an object is the account that uploads the object, who may not be the owner of the bucket to which the object belongs. The object owner has the read access to the object, as well as the read and write permission to the object ACL, and such access permissions cannot be modified.
	NOTICE Do not modify the bucket owner's read and write access permissions for the bucket.
Anonymous User	Unregistered common users of cloud services. If the permissions to access a bucket or an object are granted to anonymous users, everyone can access the object or bucket without identity authentication.
	NOTICE If the permissions to access a bucket or an object are granted to anonymous users, everyone can access the object or bucket without identity authentication.

Principal	Description
Registered User	A registered user refers to any account registered with the cloud services, excluding IAM users or user groups created by any account. To obtain access permissions, a registered user must be authenticated (AK and SK are used for the identity authentication). If the registered user group is granted with the write permission for a bucket, any registered and authenticated cloud service account can upload objects to the bucket, overwrite objects in the bucket, and delete objects from the bucket.
Log Delivery User **NOTE** Only the bucket ACL supports authorizing permissions to the log delivery user.	A log delivery user only delivers access logs of buckets and objects to the specified target bucket. OBS does not create or upload any file to a bucket automatically. Therefore, if you want to record bucket access logs, you need to grant the permission to the log delivery user who will deliver the access logs to your specified target bucket. The user only delivers logs within the service scope of OBS. **NOTICE** After logging is enabled, the bucket write permission, as well as the ACL read permission for the target bucket will be enabled automatically for the log delivery user. If you manually disable such permissions, bucket logging fails.

Table 7-2 lists the access permissions controlled by a bucket ACL.

Table 7-2 Access permissions controlled by a bucket ACL

Permission	Option	Description
Access to Bucket	Read	A grantee with the read access to a bucket can obtain the list of objects in the bucket and the metadata of the bucket.
	Write	A grantee with the write access to a bucket can upload, overwrite, and delete any object in the bucket.
Access to ACL	Read	A grantee with the read access to a bucket ACL can obtain the ACL of the bucket. The bucket owner has this permission permanently by default.
	Write	A grantee with the write access to a bucket ACL can update the ACL of the bucket. The bucket owner has this permission permanently by default.

Table 7-3 lists the access permissions of an object ACL.

Table 7-3 Access permissions controlled by an object ACL

Permission	Option	Description
Access to Object	Read	A grantee with the read access to an object can obtain the content of the object and the metadata of the object.
Access to ACL	Read	A grantee with the read access to an object ACL can obtain the ACL of the object. The object owner has this permission permanently by default.
	Write	A grantee with the write access to an object ACL can update the ACL of the object. The object owner has this permission permanently by default.

 NOTE

Every time you change the bucket or object access permission setting in an ACL, it overwrites the existing setting instead of adding a new access permission to the bucket or object.

7.2 Configuring a Bucket Policy

A bucket policy defines the access control policy of resources (buckets and objects) in OBS.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the blank area in the row of the bucket for which you want to configure a bucket policy and choose **More** > **Configure Bucket Policy**.

Step 3 In the **Configure Bucket Policy** dialog box, input required parameters.

The size of a bucket policy cannot exceed 20 KB.

Table 7-4 describes the parameters of bucket policies. All fields except the **Effect** field are optional.

Table 7-4 Parameters in bucket policies

Parameter	Description	Mandatory or Not
Version	The value can be **2008-10-17**.	Optional
Id	The ID of the bucket policy. The value must be unique.	Optional
Statement	The description of the bucket policy. The statement defines complete permission control. Each bucket policy can have multiple statements, and each statement contains the following parameters: ● Sid ● Effect ● Principal ● NotPrincipal ● Action ● NotAction ● Resource ● NotResource ● Condition	Mandatory
Effect	Effect of the bucket policy. The statement can be set to accept or reject requests. Possible values are **Allow** and **Deny**	Mandatory
Sid	The statement ID.	Optional
Principal/ NotPrincipal	Users on whom the bucket policy statement takes effect Either **Principal** or **NotPrincipal** must be selected to specify the user on whom the bucket policy statement takes effect or does not take effect.	Mandatory
Action/ NotAction	OBS operations on which the bucket policy statement takes effect Either **Action** or **NotAction** must be selected to specify whether the bucket policy statement takes effect on the OBS operation.	Mandatory

Parameter	Description	Ma nda tor y or Not
Resource/ NotResour ce	Objects on which the bucket policy statement takes effect Either **Resource** or **NotResource** must be selected to specify whether the bucket policy statement takes effect on the OBS resources.	Ma nda tory
Condition	The conditions under which the bucket policy takes effect	Opt ion al

Step 4   Click **Save**.

Step 5   In the displayed dialog box, click **OK** to close the dialog box.

----End

7.3 Configuring an Object Policy
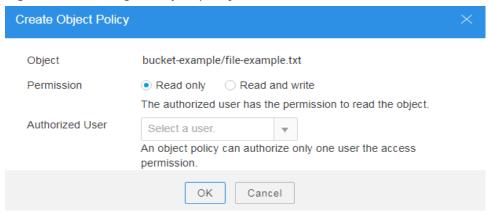
Procedure

Step 1   Log in to OBS Browser.

Step 2   Click the bucket where the desired object resides. The object list is displayed.

Step 3   Select the object, and click **Configure Object Policy** on top of the object list.

Step 4   If it is the first time that a policy is configured for the object, enter the account ID in the **Configure Object Policy** dialog box, and click **OK**. If this is not the first time, skip this step.

You can view the account ID on the My Credential page on the management console.

Step 5   Click Create Backup Policy.

Step 6   In the Create Object Policy dialog box, set the permission and select the IAM user to be authorized.

Figure 7-2 Creating an object policy

Step 7 Click **OK**.

Step 8 In the **Configure Object Policy** dialog box, click **Save**.

Step 9 In the displayed dialog box, click **OK** to close the dialog box.

----End
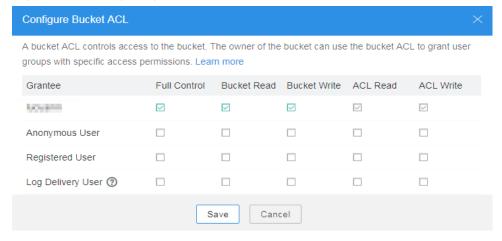
7.4 Configuring a Bucket ACL

By configuring ACL, you can authorize bucket and ACL permissions to the **Owner**, **Anonymous User**, **Registered User**, or **Log Delivery User** groups.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the blank area in the row of the bucket for which you want to set ACL permissions and click **Configure Bucket ACL**.

Step 3 Configure the ACL of the bucket and click **Save**. For details, see **Figure 7-3**.

Figure 7-3 Configuring a bucket ACL

Step 4 In the displayed dialog box, click **OK** to close the dialog box.

----End
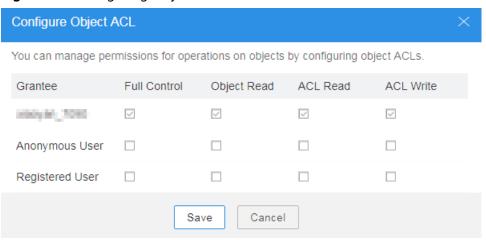
7.5 Configuring Object ACL

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the bucket where the desired object resides. The object list is displayed.

Step 3 In the row of the target object, click ··· on the right, and choose **Configure Object ACL**.

Step 4 Configure the ACL of the bucket and click **Save**. For details, see **Figure 7-4**.

Figure 7-4 Configuring Object ACL

Step 5 In the displayed dialog box, click **OK** to close the dialog box.

----End
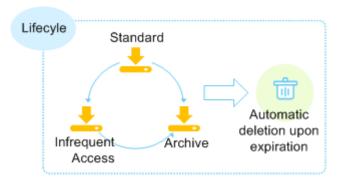
8 Lifecycle Management

8.1 Lifecycle Management Overview

Lifecycle management means periodically deleting objects in a bucket or transitioning between object storage classes by configuring rules.

Figure 8-1 Lifecycle management

Lifecycle management applies to the following scenarios:

● Some periodically uploaded files need only to be retained for one week or one month, and can be deleted once they have expired.

● Some documents are seldom accessed after a certain period of time. These documents need to be transitioned to **Infrequent Access** or **Archive** storage or be deleted.

You can define lifecycle rules for identifying objects and manage lifecycles of the objects based on the rules.

Objects that are no longer frequently accessed can be transitioned to **Infrequent Access** or **Archive**, saving your costs. In short, transition basically means that the object storage class is altered without copying the object. You can also manually change the storage class of an object on the Objects page. For details, see **Uploading a File or Folder**.

Lifecycle rules have two key elements:

- Policy: You can specify the prefix of object names so that objects whose names have this prefix are restricted by the rules. You can configure lifecycle rules for a bucket so that all objects in the bucket can be restricted by the lifecycle rules.

- Time You can specify the number of days after which objects that have been last updated and meet specified conditions are automatically transitioned to **Infrequent Access**, **Archive**, or expire and are then automatically deleted.

    - **Transition to Infrequent Access**: You can specify the number of days after which objects that have been last updated and meet specified conditions are automatically transitioned to **Infrequent Access**.

    - **Transition to Archive**: You can specify the number of days after which objects that have been last updated and meet specified conditions are automatically transitioned to **Archive**.

    - Expiration time: You can specify the number of days after which objects are automatically deleted or the day after which an object that matches with a rule is deleted.

The previous number of days for objects to be transitioned to **Infrequent Access** is at least 30. If objects are configured able to be transitioned to **Infrequent Access** and **Archive** both, the number of days for transition to **Archive** must be at least 30 greater than that for transition to **Infrequent Access**. For example, if the number of days for transition to **Infrequent Access** is 33, that for transition to **Archive** must be 63 at least. If only transition to **Archive** is enabled and transition to **Infrequent Access** is disabled, there is no limit on the number of days for transition. The expiration time must be greater than the two transition times.

8.2 Configuring a Lifecycle Rule

You can configure a lifecycle management rule for a bucket, and applicable objects in the bucket will be managed by the rule.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the blank area in the row of the bucket for which you want to configure a lifecycle rule and choose **More** > **Configure Lifecycle Rule**.

Step 3 In the **Configure Lifecycle Rule** dialog box, click **Create**. The dialog box in **Figure 8-2** is displayed.
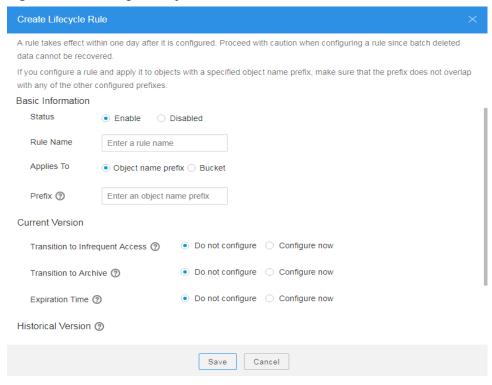
Figure 8-2 Creating a lifecycle rule on OBS Browser

Step 4 Configure a lifecycle rule.

Basic Information:

- **Status**:

  Select **Enable** to enable the lifecycle rule.

- **Rule Name**:

  Identify lifecycle rules. The **Rule Name** contains a maximum of 255 characters.

- **Prefix**: It is optional.
  - If this field is configured, objects with the specified prefix will be managed by the lifecycle rule. The prefix cannot start with a slash (/) or contain two consecutive slashes (//), and cannot contain the following special characters: **\:*?"<>|**
  - If this field is not configured, all objects in the bucket will be managed by the lifecycle rule.

  📖 **NOTE**

  - When **Object name prefix** is selected and the specified prefix and the prefix of an existing lifecycle rule overlap, OBS regards the two rules as one and disables the one to be configured. For example, if a rule with prefix **abc** exists in the system, another rule whose prefix starts with **abc** cannot be configured.
  - If a lifecycle rule whose **Applies To** is set to **Object name prefix** has been configured, you cannot configure a lifecycle rule whose **Applies To** is set to **Bucket**.

Current Version or **Historical Version**:

📖 **NOTE**

- **Current Version** and **Historical Version** are two concepts for **Versioning**. If **Versioning** is enabled, uploading objects with the same name to the same path generates different versions. The object uploaded lastly is called **Current Version**, and the object uploaded earlier is called **Historical Version**. For more information, see **Versioning**.

- You can configure either the **Current Version** or **Historical Version**, or both of them.

- **Transition to Infrequent Access After (Days)**: After this number of days since the last update, objects meeting certain conditions will be transitioned to Infrequent Access. This number must be at least 30.

- **Transition to Archive After (Days)**: After this number of days since the last update, objects meeting certain conditions will be transitioned to Archive. If you configure to transition objects first to Infrequent Access and then Archive, the objects must stay Infrequent Access at least 30 days before they can be transitioned to Archive. If only transition to Archive is used, but transition to Infrequent Access is not, there is no limit on the number of days for transition.

- Object deletion upon expiration: You can specify the number of days after which objects that have been last updated and meet the specified conditions are automatically deleted. The expiration time must be greater than the two transition times.

For example, on January 7, 2015, you saved the following files in OBS:

- log/test1.log
- log/test2.log
- doc/example.doc
- doc/good.txt

On January 10, 2015, you saved the following files:

- log/clientlog.log
- log/serverlog.log
- doc/work.doc
- doc/travel.txt

On January 10, 2015, you set the expiration time of objects prefixed with **log** to one day later, you may encounter the following situations:

- Objects **log/test1.log** and **log/test2.log** uploaded on January 7, 2015 may be deleted after the last system scan. The deletion may happen on January 10, 2015 or January 11, 2015, depending on the time of the last system scan.

- Objects **log/clientlog.log** and **log/serverlog.log** uploaded on January 10, 2015 are usually deleted on January 11, 2015 or January 12, 2015, depending on the time of the last system scan. If the objects have been stored for more than one day at the time of the last system scan, the objects are deleted upon the scan. Or, they are deleted at the next system scan or later whenever their storage duration meets the specified expiration time requirement.

On the day of operation, you can set the objects with the name prefix **log** to be transitioned to **Infrequent Access** 30 days later, transitioned to **Archive** 60 days later, and deleted 100 days later, then OBS will transition **log/clientlog.log**, **log/serverlog.log**, **log/test1.log**, and **log/test2.log** to **Infrequent Access** when their storage duration exceeds 30 days, transition them to **Archive** when their storage

duration exceeds 60 days, and delete them when their storage duration exceeds 100 days, respectively.

> **📖 NOTE**
>
> In theory, it takes 24 hours at most to execute a lifecycle rule. Because OBS calculates the lifecycle of an object from the next 00:00 (UTC time) after the object is uploaded, there may be a delay in transitioning objects between storage classes and deleting expired objects. Generally, the delay does not exceed 48 hours. If you make changes to an existing lifecycle rule, the rule will take effect again.

Step 5 Click **Save**.

Step 6 In the **Create Lifecycle Rule** dialog box, click **Save**.

Step 7 In the displayed dialog box, click **OK** to close the dialog box.

----End

9 CORS

9.1 CORS Overview

CORS is a browser-standard mechanism provided by the World Wide Web Consortium (W3C). It defines the interaction methods between client-side web applications in one origin and resources in another origin. For general web page requests, website scripts and contents in one origin cannot interact with those in another origin because of Same Origin Policies (SOPs).

The CORS specification is supported to allow cross-origin requests to access OBS resources.

Typical application scenarios of CORS are as follows:

- Enables JavaScript and HTML5 to be used for establishing web applications that can directly access resources in OBS. No proxy servers are required for transfer.

- Enables the dragging function of HTML5 to be used to upload files to OBS (with the upload progress displayed) or update OBS contents using web applications.

- Hosts external web pages, style sheets, and HTML5 applications in different origins. Web fonts or pictures in OBS can be shared by multiple websites.

The configuration of CORS takes effect within two minutes.

9.2 Configuring CORS

This section describes how to use CORS in HTML5 to implement cross-origin access.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Select the bucket to be configured and click **More** > **Configure CORS Rule**.

Step 3 Click **Add**.

📖 **NOTE**

> You can set a maximum of 100 CORS rules for one bucket.

Step 4 In the **Add CORS Rule** dialog box that is displayed, enter CORS rules.

Figure 9-1 Adding a CORS rule
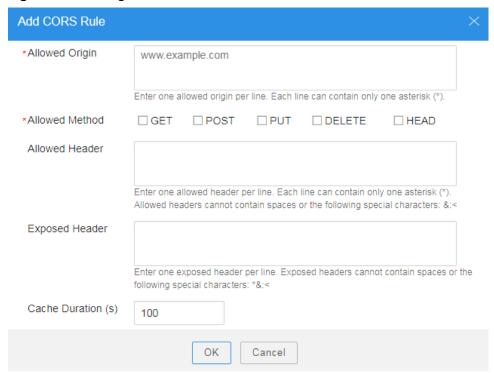
Table 9-1 describes parameters in CORS rules.

📖 **NOTE**

> If CDN acceleration is enabled for the bucket, HTTP header must be configured on CDN. For details, see **HTTP Header Settings**.

Table 9-1 Parameters in CORS rules

Parameter	Description
Allowed Origin	Specifies the origin of cross-origin requests. That is, requests from the origin can access the bucket. This parameter is mandatory. Multiple matching rules are allowed. One rule occupies one line, and allows one wildcard character (*****) at most. Example: http://rds.example.com https://*.vbs.example.com
Allowed Method	Specifies the method of cross-origin requests, that is, the operation type of buckets and objects. This parameter is mandatory. The following methods are included: Get, Post, Put, Delete, and Head.

Parameter	Description
Allowed Header	Specifies the allowed header of cross-origin requests. This parameter is optional. Only CORS requests matching the allowed header are valid. You can enter multiple allowed headers (one per line) and each line can contain one wildcard character (*) at most. Spaces and special characters including **&:<** are not allowed.
Exposed Header	Specifies the supplemented header in CORS responses, providing additional information for clients. This parameter is optional. You can enter multiple exposed headers (one per line). Spaces and special characters including ***&:<** are not allowed.
Cache Duration (s)	Mandatory. Specifies the duration that your browser can cache CORS responses, expressed in seconds. The default value is 100.

Step 5 Click **OK**.

Step 6 Click **OK** to save the rules.

After CORS is successfully configured, only the addresses specified in **Allowed Origin** can access a bucket in OBS using the method specified in **Allowed Method**. For example, you configure CORS parameters for bucket **testbucket** as follows: **Allowed Origin: www.example.com**; **Allowed Method: GET**; **Allowed Header**: left blank; **Exposed Header**: left blank; **Cache Duration (s): 100**. Then OBS only allows GET requests from **www.example.com** to access bucket **testbucket**, without restrictions on request headers. The client can cache the CORS response for 100 seconds.

Step 7 In the displayed dialog box, click **OK** to close the dialog box.

----End

10 Logging

10.1 Logging Overview

You can enable logging to facilitate analysis or audit as required. Access logs enable a bucket owner to analyze the property, type, or trend of requests to the bucket in depth. When the logging function of a bucket is enabled, OBS will log access requests for the bucket automatically, and write the generated log files to the specified bucket (target bucket).

NOTICE

Uploading bucket logs to the target bucket incurs billable PUT requests. For details about the pricing, see **Requests**.

OBS can record bucket access requests in logs for request analysis and log audit.

Logs occupy some OBS storage space rented by users, causing extra fees. For this reason, OBS does not collect bucket access logs by default.

The log files are generated and uploaded by OBS to the bucket where the logs are stored. Therefore, OBS requires the authorization to upload the generated log files. Therefore, before configuring logging for a bucket, you need to create an IAM agency for OBS and add this agency when configuring logging for the bucket. By default, when configuring permissions for an agency, you only need to grant the agency the permission to upload log files (PutObject) to the bucket for storing log files. In the following example, **mybucketlogs** is the bucket. If the default encryption function is enabled for the log storing bucket, the IAM agency also requires the KMS Administrator permissions in the region where the log storing bucket resides.

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
                "obs:object:PutObject"
            ],
            "Resource": [
```

```
          "OBS:*:*:object:mybucketlogs/*"
       ],
       "Effect": "Allow"
     }
  ]
}
```

After logging is configured, you can view operation logs in the bucket that stores the logs in approximately fifteen minutes.

The following shows an example access log of the target bucket:

```
787f2f92b20943998a4fe2ab75eb09b8 bucket [13/Aug/2015:01:43:42 +0000] xx.xx.xx.xx
787f2f92b20943998a4fe2ab75eb09b8 281599BACAD9376ECE141B842B94535B
REST.GET.BUCKET.LOCATION
- "GET /bucket?location HTTP/1.1" 200 - 211 - 6 6 "-"  "HttpClient" - -
```

The access log of each bucket contains the following information.

Table 10-1 Bucket log format

Parameter	Value Example	Description
BucketOwner	787f2f92b20943998a4fe2ab75eb09b8	Account ID of the bucket owner
Bucket	bucket	Name of the bucket
Time	[13/Aug/2015:01:43:42 +0000]	Timestamp of the request (UTC)
Remote IP	xx.xx.xx.xx	IP address from where the request is initiated
Requester	787f2f92b20943998a4fe2ab75eb09b8	Requester ID • When an account initiates a request, this is the account ID. When an IAM user initiates a request, this is the ID of the account to which the IAM user belongs. • When a request is initiated by an anonymous user, the value of this parameter is **Anonymous**.

Parameter	Value Example	Description
RequestID	281599BACAD9376ECE141B842B94535B	Request ID
Operation	REST.GET.BUCKET.LOCATION	Name of the operation
Key	-	Object name
Request-URI	GET /bucket?location HTTP/1.1	URI of the request
HTTPStatus	200	Return code
ErrorCode	-	Error code
BytesSent	211	Size of the HTTP response, expressed in bytes
ObjectSize	-	Object size (bytes)
TotalTime	6	Processing time on the server (ms)
Turn-AroundTime	6	Total time for processing the request (ms)
Referer	-	Header field **Referer** of the request
User-Agent	HttpClient	User-Agent header of the request
VersionID	-	Version ID carried in the request
STSLogUrn	-	Federated authentication and agency information
StorageClass	STANDARD_IA	Current storage class of the object
TargetStorageClass	GLACIER	Storage class that the object will be transited to

10.2 Configuring Logging

After logging is enabled for a bucket, OBS automatically converts bucket logs into objects following the naming rules and writes the objects into a target bucket.
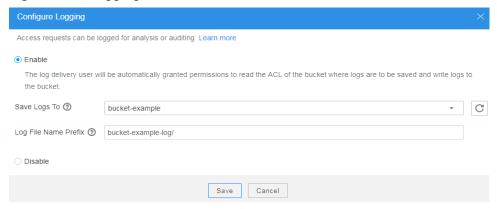
Prerequisites

The log delivery user group has been assigned the **Write** and **ACL View** permissions for the target bucket. For details, see **Configuring a Bucket ACL**.

Procedure

Step 1 Log in to OBS Browser.

Step 2 Click the blank area in the row of the bucket for which you want to configure logging and choose **More** > **Configure Logging**.

Step 3 Select **Enable**. See **Figure 10-1**.

Figure 10-1 Logging

Step 4 Select an existing bucket where you want to store log files.

Step 5 Enter a prefix for the **Log File Name Prefix**.

After logging is enabled, generated logs are named in the following format:

*<Log File Name Prefix>*YYYY-mm-DD-HH-MM-SS-*<UniqueString>*

- *<Log File Name Prefix>* is the shared prefix of log file names.
- **YYYY-mm-DD-HH-MM-SS** indicates when the log is generated.
- *<UniqueString>* indicates a character string generated by OBS.

On OBS Console, if the configured *<Log File Name Prefix>* ends with a slash (/), logs generated in the bucket are stored in the *<Log File Name Prefix>* folder in the bucket, facilitating the management of log files.

Example:

- If the bucket named **bucket** is used to save log files, and the log file name prefix is set to **bucket-log/**, all log files delivered to this bucket are saved in the **bucket-log** folder. A log file is named as follows: **2015-06-29-12-22-07-N7MXLAF1BDG7MPDV**.
- If the bucket named **bucket** is used to save log files, and the log file name prefix is set to **bucket-log**, all log files are saved in the root directory of the bucket. A log file is named as follows: **bucket-log2015-06-29-12-22-07-N7MXLAF1BDG7MPDV**.

Step 6 Click **Save**.

Step 7 In the displayed dialog box, click **OK** to close the dialog box.

----End

11 CDN Facilitated Download

11.1 CDN Download Overview

On OBS Console, you can bind domain names to a bucket, and enable CDN for access acceleration. After domain names are bound successfully, you can also see the configuration on OBS Browser. You can enable the CDN download function for bound domain names on OBS Browser. When objects are downloaded through OBS Browser, the system retrieves objects from the optimal CDN node, improving the download speed.

Before enabling CDN download, ensure that the bucket policy is Public Read, or object policies are Public Read for objects that are accessed.

11.2 Configuring CDN Download

Procedure

Step 1 Log in to OBS Browser.

Step 2 Select the bucket to be operated and click **More** > **Configure CDN Download**. The **Configure CDN Download** dialog box is displayed.

Step 3 Select **Enable CDN Download** next to the domain name that you want to configure.

Step 4 Click **OK**.

Step 5 In the displayed dialog box, click **OK** to close the dialog box.

----End

12 External Buckets

12.1 External Bucket Overview

The bucket owner can authorize other accounts the read and write access to the bucket. If you are authorized with such permissions, you can add the bucket on OBS Browser as an external bucket. After the external bucket is added successfully, you can operate the bucket on OBS Browser. For details about what actions you can perform on the external bucket, check the permission settings.

Authorizing Permissions Required for Adding a Bucket as an External Bucket on OBS Browser

The read and write access to a bucket can be granted through the bucket ACL or bucket policy.

Permissions controlled by a bucket ACL are as follows:

Table 12-1 Permissions controlled by a bucket ACL

Bucket ACL	Option	Mapped Action in a Custom Bucket Policy
Access to Bucket	Read	• HeadBucket • ListBucket • ListBucketVersions • ListBucketMultipartUploads
	Write	• PutObject • DeleteObject • DeleteObjectVersion
Access to ACL	Read	GetBucketAcl
	Write	PutBucketAcl

Permissions controlled by a standard bucket policy are as follows:

Table 12-2 Permissions controlled by a standard bucket policy

Parameter	Public Read	Public Read and Write
Effect	Allow	Allow
Principal	* (Any user)	* (Any user)
Resources	* (All objects in a bucket)	* (All objects in a bucket)
Actions	- GetObject - GetObjectVersion - HeadBucket - ListBucket	- GetObject - GetObjectVersion - PutObject - DeleteObject - DeleteObjectVersion - HeadBucket - ListBucket
Conditions	N/A	N/A

If a custom bucket policy is used to authorize such permissions, the HeadBucket, ListBucket, GetObject, and GetObjectVersion actions must be allowed. More actions can be allowed according to your actual needs.

Operations That Can Be Performed on the External Bucket

Operations that can be performed by the authorized user on the external bucket:

- You can add an external bucket but cannot restore objects from the Archive storage class in the bucket. You can view the object restore status only when the owner of those objects authorizes you the permission to read the objects.

- You (the user who adds the bucket) can perform only authorized actions on original objects in the bucket. If you want to have additional operation permissions for objects in the bucket, you need to have the permissions authorized by the object owner.

- If you upload an object to the added external bucket, the read access to the object and the object ACL will be automatically authorized to the bucket owner and configured in the object ACL settings.

- If you upload an encrypted object to the added external bucket, the bucket owner cannot access the object because the bucket owner does not have the key.

- To download an object from the external bucket, the user who adds the bucket must be authorized the permission to read the object, and encrypted objects cannot be downloaded.

12.2 Adding External Buckets

OBS Browser supports the external bucket adding function.

Prerequisites

You have obtained the permissions to read and write the external bucket that you want to add.

For example: Account **A** wants to add bucket **testbucket** of account **B**. Account **B** must authorize account **A** the permission to read bucket **testbucket**. If account **A** needs to upload object to bucket **testbucket**, account **B** needs to authorize account **A** the permission to write the bucket. Account **A** is the owner of the uploaded objects, and needs to authorize account **B** the permission to read and write the objects.

> 📖 **NOTE**
>
> An external bucket (version 3.0 or later) can be added only when its standard bucket policy is set to **Public Read and Write**.
>
> If the standard bucket policy of an external bucket is set to **Public Read and Write** upon its creation, you need to manually change the standard bucket policy to **Private** and then change it back to **Public Read and Write** again, so that it can be added.
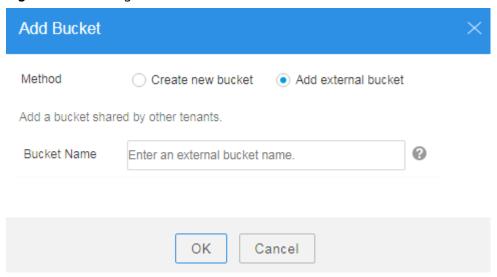
Procedure

Step 1 Log in to OBS Browser.

Step 2 Click **Add Bucket** on the upper left corner of the page. The **Add Bucket** dialog box is displayed.

Step 3 Select **Add external bucket** and enter the bucket name.

Figure 12-1 Adding an external bucket

After successfully adding an external bucket, you can see the external bucket in the bucket list and have the ACL access permissions for the bucket.

Step 4 Click **OK**.

Step 5 In the displayed dialog box, click **OK** to close the dialog box.

----End

12.3 Application Example 1: Authorizing Access Permissions Required for Adding an External Bucket Through the Bucket ACL

A bucket ACL can be used to grant the read and write access to a bucket. If only the read access to the bucket is granted, the authorized user can only add the bucket and list objects in the bucket, but cannot upload objects to the bucket. If the read and write access to the bucket is granted, the authorized user can upload objects to the bucket. Permissions controlled by a bucket ACL are as follows:

Table 12-3 Permissions controlled by a bucket ACL

Bucket ACL	Option	Mapped Action in a Custom Bucket Policy
Access to Bucket	Read	- HeadBucket - ListBucket - ListBucketVersions - ListBucketMultipartUploads
	Write	- PutObject - DeleteObject - DeleteObjectVersion
Access to ACL	Read	GetBucketAcl
	Write	PutBucketAcl

Procedure

Step 1 Log in to OBS Console.

Step 2 In the bucket list, click the bucket name you want. The **Objects** page is displayed.

Step 3 In the navigation pane on the left, click **Permissions** to go to the permission management page.

Step 4 Click **Bucket ACL**. The **Bucket ACL** page is displayed.

Step 5 Click **Add**, enter the account ID of the user that will add the bucket to OBS Browser, and select the read and write access to the bucket.

> 📖 NOTE
>
> If you want to authorize such access to all users, in the **Public Permissions** area, authorize the **Anonymous User** the read and write access to the bucket.

Step 6 Click **Save**.

----End

Verification

Step 1 Log in to OBS Browser.

Step 2 Click **Add Bucket** on the upper left corner of the page. The **Add Bucket** dialog box is displayed.

Step 3 Select **Add external bucket** and enter the bucket name.

Step 4 Click **OK**. The external bucket is added successfully.

Step 5 Click the newly added external bucket to open the bucket.

Step 6 Click **Upload Object**, and objects can be successfully uploaded to the bucket.

Step 7 Select an object in the bucket and click **Delete**. The object can be deleted successfully.

----End

12.4 Application Example 2: Authorizing Access Permissions Required for Adding an External Bucket Through the Standard Bucket Policy

A standard bucket policy can be used to grant the read and write access to a bucket. The standard bucket policy grants the public read and write access to the bucket, that is, all users can access the bucket. Permissions controlled by a standard bucket policy are as follows:

Table 12-4 Permissions controlled by a standard bucket policy

Parameter	Public Read	Public Read and Write
Effect	Allow	Allow
Principal	* (Any user)	* (Any user)
Resources	* (All objects in a bucket)	* (All objects in a bucket)
Actions	- GetObject - GetObjectVersion - HeadBucket - ListBucket	- GetObject - GetObjectVersion - PutObject - DeleteObject - DeleteObjectVersion - HeadBucket - ListBucket
Conditions	N/A	N/A

Procedure

Step 1 Log in to OBS Console.

Step 2 In the bucket list, click the bucket name you want. The **Objects** page is displayed.

Step 3 In the navigation pane on the left, click **Permissions** to go to the permission management page.

Step 4 Select the **Public Read and Write** policy from the standard bucket policies that are pre-defined.

Step 5 In the dialog box that is displayed, click **Yes**.

----End

Verification

Step 1 Log in to OBS Browser.

Step 2 Click **Add Bucket** on the upper left corner of the page. The **Add Bucket** dialog box is displayed.

Step 3 Select **Add external bucket** and enter the bucket name.

Step 4 Click **OK**. The external bucket is added successfully.

Step 5 Click the newly added external bucket to open the bucket.

Step 6 Click **Upload Object**, and objects can be successfully uploaded to the bucket.

Step 7 Select an object in the bucket and click **Delete**. The object can be deleted successfully.

----End

12.5 Application Example 3: Authorizing Access Permissions Required for Adding an External Bucket Through the Custom Bucket Policy

A custom bucket policy can be used to grant the read and write access permissions to the bucket to be added.

If a custom bucket policy is used to authorize such permissions, the HeadBucket, ListBucket, GetObject, and GetObjectVersion actions must be allowed. More actions can be allowed according to your actual needs.

Procedure

Step 1 Log in to OBS Console.

Step 2 In the bucket list, click the bucket name you want. The **Objects** page is displayed.

Step 3 In the navigation pane on the left, click **Permissions** to go to the permission management page.

Step 4 In the **Custom Bucket Policies** area, click **Create Bucket Policy**. The **Create Bucket Policy** dialog box is displayed.

Step 5 Set the following parameters to authorize another account with the permission to access the bucket:

Table 12-5 Parameters for authorizing the permission to access a specified bucket

Parameter	Value
Policy Mode	**Customized**
Effect	**Allow**
Principal	**Include** **Other account**: Enter the account ID. If you want to authorize the permissions all users, enter *****. **NOTE** The account ID and user ID can be obtained on the **My Credentials** page of the account or user to be authorized. If you authorize the permission to only an account, you do not need to enter user IDs. If you want to authorize the permission to an IAM user, you need to enter the account ID and user ID. You can authorize the permission to multiple IAM users. Use commas (,) to separate the user IDs.
Resources	**Include** - Leave it blank.
Actions	**Include** - HeadBucket - ListBucket

Step 6 Click **OK**.

Step 7 Create another bucket policy and set the parameters according to the following table to grant the authorized account with access permissions to resources in the bucket.

Table 12-6 Parameters for authorizing the permission to access a specified bucket

Parameter	Value
Policy Mode	**Customized**
Effect	**Allow**
Principal	Keep the value consistent with the preceding policy.

Parameter	Value
Resources	● **Include** ● Resource name: *****
Actions	● **Include** ● GetObject ● GetObjectVersion ● PutObject ● DeleteObject ● DeleteObjectVersion

Step 8 Click **OK**.

----End

Verification

Step 1 Log in to OBS Browser.

Step 2 Click **Add Bucket** on the upper left corner of the page. The **Add Bucket** dialog box is displayed.

Step 3 Select **Add external bucket** and enter the bucket name.

Step 4 Click **OK**. The external bucket is added successfully.

Step 5 Click the newly added external bucket to open the bucket.

Step 6 Click **Upload Object**, and objects can be successfully uploaded to the bucket.

Step 7 Select an object in the bucket and click **Delete**. The object can be deleted successfully.

----End

13 Task Management

OBS Browser supports the management of upload, download, renaming, moving, copying, deletion, restoration, restoration and download tasks. You can suspend, cancel, or delete tasks through task management.

📖 **NOTE**

If the number of displayed items exceeds 200,000, the system will save the first 100,000 items of tasks that are created earlier to the **history** directory in the decompression path of OBS Browser. These items are saved in the **historyDBData**[*time stamp*]**.csv** files, for example, **historyDBData20170426T063744.csv**.

Procedure

Step 1 Log in to OBS Browser.

Step 2 In the upper right corner on the page, click .

Step 3 You can select a task type from the drop-down list box in the upper right corner to query the running tasks.

Step 4 **Optional**: Select a running task and click ❙❙ to suspend the task. For a paused task, you can click ▶ to continue the task.

Step 5 **Optional**: Select a running task and click 🗑 to delete the task.

Step 6 **Optional**: If the task fails, select the failed task, move the pointer over ⋯ to view the failure cause, and click ▶ to run the task again.

Step 7 **Optional**: You can select multiple tasks and click **Run**, **Suspend**, or **Cancel** to manage the tasks.

📖 **NOTE**

If you cancel a running task, the task will be deleted.

A restoring task cannot be cancelled, but a failed restoration task will be deleted if you cancel it.

Step 8 **Optional**: Click the **Completed** button on the top of the page to view completed

tasks. Click next to a finished task to delete the task record.

----End

14 Related Operations

14.1 Configuring the System

This section describes how to modify system configurations.

Procedure

Step 1 Log in to OBS Browser.

Step 2 In the upper right corner of OBS Browser, click and choose **System Configuration**. The **System Configuration** dialog box is displayed, see **Figure 14-1**.

Figure 14-1 Configuring the system

Step 3 Click **General** and modify basic parameters as required.

Table 14-1 describes the parameters that can be modified.

Table 14-1 General configuration parameters

Parameter	Description
Enable HTTPS	If this option is selected, all communication information is encrypted and transferred to OBS over HTTPS.
Enable certificate verification	When this option is selected, the client will verify the server certificate.
Enable KMS encryption	When **Enable HTTPS** and **Enable KMS encryption** are selected, KMS encryption will be implemented for all objects uploaded to OBS.

Parameter	Description
Multipart Upload, Part Size (MB)	Objects whose size is larger than the specified part size (5 MB by default) are segmented and uploaded at the OBS background. The size of each part can be set in this dialog box. The value of **Part Size (MB)** can range from 5 MB to 5 GB. **NOTE** Multipart upload is used by default. Recommended settings of **Part Size (MB)** are as follows: To maximize client performance, set **Part Size (MB)** based on the upload speed. You are advised to set the **Part Size (MB)** value larger than the maximum upload speed. For example, if the maximum upload speed is 10 MB/s, set **Part Size (MB)** to an integer greater than 10 MB. It is recommended that the part size be set to a value two to three times the maximum upload speed.
Multipart Download, Part Size (MB)	OBS uses the multipart download function by default. Objects whose size is larger than the specified part size (5 MB by default) are segmented into multiple parts at the background and then downloaded. The part size can be set in this dialog box. The value of **Part Size (MB)** can range from 5 MB to 5 GB.
Maximum number of tasks	Specifies the maximum number of upload and download tasks. Enter an integer ranging from 2 to 20.
Language	Selects the language of the display. Changes take effect after the software is restarted.

Step 4 Click **Network** and set proxy server information as required. See **Figure 14-2**.

Figure 14-2 Network configurations
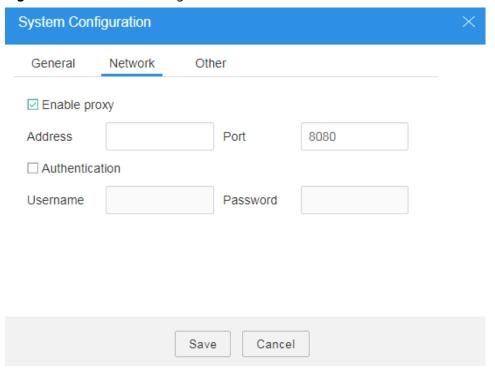
Table 14-2 Network configuration parameters

Parameter	Description
Enable proxy	If this option is selected, the **Authentication** option is displayed. Set the following parameters to access OBS through the proxy server: ● Address: domain name or IP address of the proxy server ● Port: port of the proxy server (default port is **8080**)

Step 5 Click **Other** and set the parameters as required. For details, see **Figure 14-3**.

Figure 14-3 Other configurations
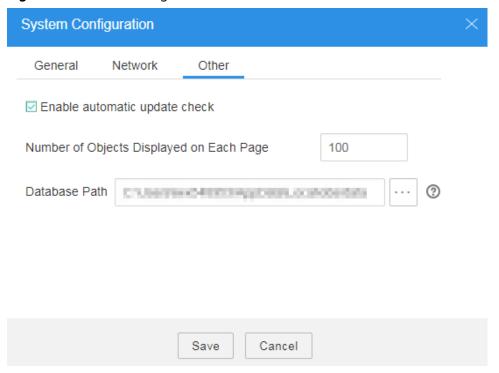
Table 14-3 Other parameters

Parameter	Description
Enable automatic update check	If this option is selected, each time when you log in to OBS Browser, a check will be automatically performed to determine whether the current software version is the latest.
Number of Objects Displayed on Each Page	Set the number of objects that are displayed on each page. The default value is 100. The value ranges from 50 to 300. After setting the value, click the ⟳ button in the upper right corner of the page so that the setting takes effect.
Database Path	Path for storing OBS Browser data. You can click ⋯ to modify the path. **CAUTION** The database path cannot contain any Chinese character.

Step 6 Click **Save** to save the system configuration.

----End

14.2 Managing Accounts

Account names on OBS Browser are used to distinguish one from another, which are irrelevant to the registered cloud service account. An OBS Browser account has one pair of AK and SK, but a pair of AK and SK can be used by multiple OBS Browser accounts. A maximum of 10 accounts can be added to OBS Browser.

OBS Browser uses the AK and SK for identity authentication. AKs and SKs are the access keys created by on the management console (on **My Credential** page) by cloud service accounts and IAM users. For details, see **Creating Access Keys (AK and SK)**.

1. AKs and SKs are required for logging in to OBS Browser to access OBS resources.

2. Once an AK and SK are entered, IAM receives the AK and SK, finds the cloud service account or IAM user that own the pair of AK and SK, and checks which OBS permissions the account or IAM user has.

3. Then IAM authorizes the user who tries to log in to OBS Browser the OBS permissions accordingly.

4. The user can access OBS resources through OBS Browser.

Add an Account

Step 1 Log in to OBS Browser.

Step 2 In the upper right corner of OBS Browser, click the account name, and select **Manage Account**.

Step 3 In the **Manage Account** dialog box that is displayed, click **Add Account**.

Step 4 In the **Add Account** dialog box, enter the account information.

The account information includes the following parameters:

- Account name: The account name is used only to uniquely identify an account and can be different from the OBS account registered with the cloud services. The account name cannot exceed 50 characters.

- Service: OBS Browser can connect to **OBS** or **Other object storage services**.
    - When accessing OBS, select **OBS**. For details, see **Figure 14-4**.

Figure 14-4 Add a new account - OBS

– When connecting to other object storage services, select **Other object storage services**. For details, see **Figure 14-5**.

Specify **Server Address**. You can enter the IP address or domain name in the following format: *server IP address or domain name:server port* (the protocol port of HTTPS is **443** and that of HTTP is **80**). The HTTPS server is used by default. If you want to use the HTTP server, click ⚙ in the upper right corner and click **System Configuration**. In the **System Configuration** dialog box that is displayed, deselect **Enable HTTPS**.

Add Account ✕

If you have created access keys, obtain the access key ID and its secret access key from the credentials.csv file downloaded from OBS Console. You can also click here to create a pair of access keys on the Access Keys tab.

Account Name	⦾
Service	Other object storage services ▾ ⦾
Server Address	⦾
Access Key ID	
Secret Access Key	●●●●●●●●●●●●
Access Path	Enter the access path. ⦾

☑ Remember my secret access key

OK   Cancel
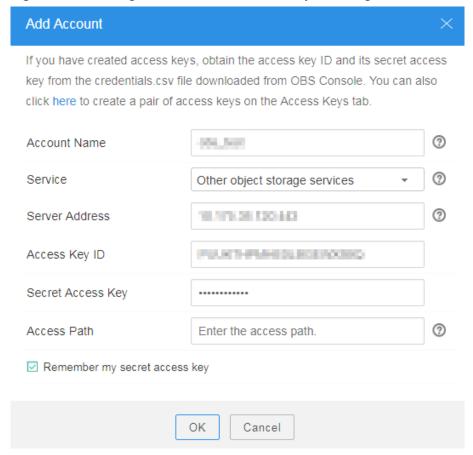
- AK and SK: Enter the AK and SK created on the **My Credential** page after you register with a cloud service. For details about how to obtain AKs and SKs, see **Creating Access Keys (AK and SK)**.

- Access path

  You can enter the bucket name or object path. After the login is successful, the bucket or object is accessed.

  – Bucket access path format: *Bucket name*

  – Object access path format: *Bucket name/Object name*

    The object name refers to a folder or a file.

- **Remember my secret access key** is selected by default. If you do not select **Remember my secret access key**, you must input the secret access key each time you log in to OBS Browser.

Step 5 Click **OK**.

After saving the account information, you can click the account name in the upper corner of the page and the newly added account is displayed in the account name drop-down list. You can click the desired account to switch to that account from the current login account.

----End

Edit an Account

Step 1 Log in to OBS Browser.

Step 2 In the upper right corner of OBS Browser, click the account name, and select **Manage Account**.

Step 3 Click **Edit** in the row where the desired account resides.

Step 4 Modify account information as required.

Step 5 Click **OK** to save the modification.

----End

Deleting an Account

Step 1 Log in to OBS Browser.

Step 2 In the upper right corner of OBS Browser, click the account name, and select **Manage Account**.

Step 3 Click **Delete** in the row where the desired account resides.

Step 4 Click **OK** and the account is deleted.

----End

14.3 Access Path
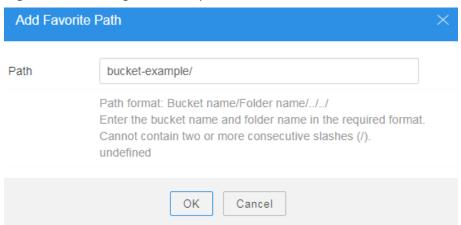
OBS Browser can directly access buckets, folders, or objects through access paths.

The format of an access path is as follows: *Bucket name*/*Folder name*/.../, which cannot contain two or more adjacent slashes (/).

Procedure

Step 1 Log in to OBS Browser.

Step 2 Manually add an access path to the folder for favorite paths.

1. Click ☆ . The folder of **Favorite Paths** is displayed in the navigation pane on the left. A maximum of 100 paths can be added to the folder of favorite paths.
2. Click **Add Path**. The **Add Favorite Path** dialog box is displayed.
3. Enter the access path. See **Figure 14-6**.

4. Click **OK**. The access path is added to the **Favorite Paths** folder.

5. In the displayed dialog box, click **OK** to close the dialog box.

Step 3 Add an access path to the **Favorites**.

1. Select a bucket or a folder in the bucket.

2. Click ⭐ to add the access path to the **Favorite Paths** folder. See **Figure 14-7**.

Figure 14-7 Adding an access path - method 2

Step 4 In the **Favorite Paths** folder, click the access path. OBS Browser opens the content in the access path.

Step 5 Add an access path when adding an account. For details, see **Managing Accounts**. If you log in to OBS Browser using the account, the content in the access path is opened.

----End

14.4 Updating OBS Browser

This section describes how to update OBS Browser.

Procedure

Every time you log in to OBS Browser, the system automatically checks for update by default. If the current version is not the latest, upgrade the software as instructed.

You can log in to OBS Browser, click in the upper corner of the page, select **System Configuration**, click **Other** in the **System Configuration** dialog box that is displayed, and deselect **Automatic Update Check**. Then you can manually upgrade OBS Browser as follows:

Step 1 Log in to OBS Browser.

Step 2 In the upper right corner, click and select **Check for Update**.

Step 3 Update the OBS Browser version as instructed.

Click on OBS Browser. Then click **Check for Update**. If **It is already the latest version** is displayed, the upgrade is successful.

----End

15 Troubleshooting

15.1 Login Page of OBS Browser Does Not Respond upon User Login

Question

When a user attempts to log in to OBS Browser, the login page does not respond.

Answer

Delete the **obs** folder in the **AppData\Local** directory on the C drive to clear OBS Browser related data. Then log in to OBS Browser again and reconfigure required information. If you want to retain the account information, save the **accounts**, **setting**, and **temp files** in the **obs** folder before deleting the folder. Start OBS Browser again and overwrite the three files in the **obs** folder.

15.2 No Upload Task Is Created After a Large Number of Files Are Selected for Upload On OBS Browser

Question

Why is no upload task created and nothing displayed on the page after a large number of files are selected for upload using OBS Browser? For example, after a user logs in to OBS Browser and chooses **Upload** > **Upload File** to select a large number of files from drive C for upload, no upload task is created and nothing is displayed on the page.

Answer

Confirm the number of files to be uploaded. The **Upload File** function of OBS Browser allows 500 files to be uploaded at the same time. If more files need to be uploaded, place the files in a folder and use the **Upload Folder** function to upload the folder.

Confirm the total name length of all files to be uploaded cannot exceed approximately 25,500 characters. If the name length exceeds this threshold, the system stops responding to the upload request.

15.3 Blue or Black Screen of Death Occurs During File Upload on OBS Browser

Question

Why does blue or black screen of death occur when files are uploaded using OBS Browser?

Answer

When OBS Browser is used to upload a large number of files or a few of big files, most of the memory is used. Check whether the available memory space of the PC is greater than or equal to 512 MB. If the available memory space of the PC is smaller than 512 MB, close some application programs to release the memory, or add new memory or virtual memory to the PC.

If the problem persists after memory is added, collect dump files by performing the following procedure, and contact customer service personnel for locating and solving this problem.

Step 1 Right-click **Computer** and choose **Properties** from the shortcut menu. In the dialog box that is displayed, click the **Advanced** tab. On the **Advanced** tab page, click **Settings**. In the dialog box that is displayed, view the path under **Dump file**.

Figure 15-1 shows a screenshot of the Windows 7 Pro SP1 64-bit operating system, which is used as an example.
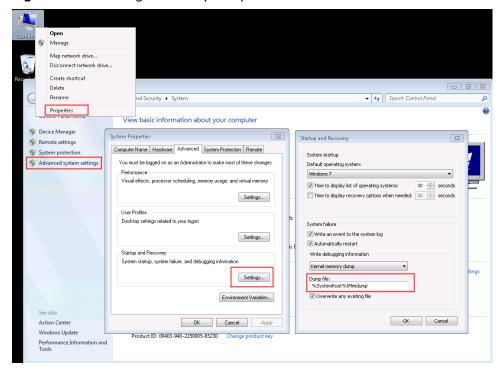
Figure 15-1 Viewing the Dump file path

Step 2 Go to the path, copy and compress all files of which the extension is **.dump**, and send the compressed package to customer service personnel for locating and solving this problem.

----End

15.4 Tasks Are Not Displayed in the Task List When Objects Are Managed Using OBS Browser

Question

Why are the tasks not displayed in the task list when objects are being uploaded, downloaded, or deleted using OBS Browser?

Answer

When OBS Browser is used to upload, download, or delete objects, the internal database of OBS Browser is invoked. By default, binary data generated while the database is running is saved in the personal folder of the Windows operating system user.

If the username of the Windows operating system contains non-letter characters, the internal database of OBS Browser cannot identify the save path of data. As a result, upload, download, and deletion tasks cannot be added to the task list. In such a scenario, perform the following operations to specify the save path of the binary data generated while the internal database is running.

Step 1   In the upper right corner of OBS Browser homepage, click ⚙ and choose **System Configuration**.

Step 2   Click **Other**.

Step 3   Click [...] next to the text box of **Database Path**. The local file browser is displayed.

Step 4   Select a path and click **OK**.

Step 5   In the **System Configuration** dialog box, click **Save** to save the selected path.

Then, perform the upload, download, or deletion tasks again. Task information will be displayed in the task list.

Step 6   If still no task information is displayed, close OBS Browser, delete the **obs\data** folder from the database path, and restart OBS Browser to try again.

----End

15.5 Time Difference Is Longer Than 15 Minutes Between the Client and the Server

Question

OBS is unavailable when an error is reported stating "Time difference is longer than 15 minutes between the client and server."

Answer

For security purposes, OBS checks the time difference between OBS Browser and the server. If the time difference is greater than 15 minutes, this error is reported. Adjust the local time based on the UTC time to avoid this problem.

15.6 An Error Occurs During the Start-up of OBS Browser, Indicating That the Task Management Function Cannot Work Properly Due to Unavailability of the Database

Question

When starting OBS Browser on a Windows OS, an error occurs with the message displays as "The task management function cannot work properly because the database is unavailable. Shut down OBS Browser, delete all files in the obs folder in the database path (check the path at System Configuration > Other), and restart OBS Browser." But the problem cannot be resolved by deleting the obs folder.

Answer

> Log in to Windows OS using a user whose name contains only digits and letters (either uppercase or lowercase, or both), and then start OBS Browser.

15.7 File Upload or Download Through OBS Browser Is Stuck

Question

- When uploading or downloading a large file through OBS Browser, you are prompted by the message **max call stack size exceeded** in the middle of the upload or download process.
- When a folder is being uploaded through OBS Browser, the operating system returns an I/O status exception and the upload progress stays at 99.99%.
- When a large file is being uploaded through OBS Browser, the progress bar does not respond and the upload keeps waiting.

Answer

The preceding problems are caused by earlier OBS Browser versions. You can upgrade your OBS Browser to the latest version and the problems will be resolved.

It is recommended that you enable the automatic update check function of OBS Browser to obtain the latest version in time.

To enable the function, perform the following steps: **Settings** > **System Configuration** > **Other** > **Auto Check for Latest Version**.

16 List of OBS Error Codes

If a request fails to be processed due to errors, an error response is returned. An error response contains an error code and error details. **Table 16-1** lists some common error codes in OBS error responses.

Table 16-1 List of OBS error codes

Error Code	Description
Obs.0000	Unauthorized gateway or proxy server. Enter the correct authentication information.
Obs.0001	Internal server error. Contact the customer service.
Obs.0002	Bad gateway or unable to connect to the proxy server. Contact the customer service.
Obs.0003	The server is busy. Try again later.
Obs.0004	The requested bucket name already exists. Bucket namespace is shared by all users in the system. Enter a different name and retry.
Obs.0005	Deletion failed. Check whether objects exist in the bucket or log in to OBS Console to check whether objects of historical versions exist in the bucket.
Obs.0006	Access to the local file denied. Please retry.
Obs.0007	Failed to upload the target file because it has been occupied. Check whether the file is accessible.
Obs.0008	Failed to read the file. Please retry.
Obs.0009	Insufficient local disk space. Delete unnecessary files to ensure that the disk space is sufficient.
Obs.0010	You do not have permission to read or write a local file. Apply for the required permissions.

Error Code	Description
Obs.0011	The account balance is in arrears or insufficient. Please top up and retry.
Obs.0012	Invalid access key ID or secret access key.
Obs.0013	Failed to upload the file in multiparts. Please cancel this task and retry.
Obs.0014	All access requests to this object are invalid.
Obs.0015	This operation cannot be performed because: The target bucket does not exist. The target bucket does not belong to you. You do not have the required permission. For this reason, ensure that the log delivery user has the **Bucket Write** and **ACL View** permissions for the target bucket. Configure the permissions in the Configure Permission dialog box.
Obs.0016	Your operation is conflicting with another user's operation. Try again later.
Obs.0017	The bucket that you attempt to access is not found in this region. Select another bucket.
Obs.0018	Incorrect region. Enter a correct region.
Obs.0019	Time difference is longer than 15 minutes between the client and server.
Obs.0020	The server load is too heavy. Try again later.
Obs.0021	Failed to delete some objects. Try again later.
Obs.0022	Failed to download some objects. Try again later.
Obs.0023	The number of buckets has reached the upper limit.
Obs.0024	Failed to process the request. Please retry.
Obs.0025	Invalid IP address or domain name. Try another one.
Obs.0026	The absolute path of a file cannot exceed 1023 characters. Please retry.
Obs.0027	Unable to connect to the proxy server. Check whether the network is normal and the authentication information about the proxy server is correct.
Obs.0028	The target bucket does not exist or is not in the same region with the source bucket.
Obs.0029	Incorrect signature. Contact the customer service.
Obs.0030	Failed to find or create the path. Check whether the path exists or is too long.

Error Code	Description
Obs.0031	The account has not been registered with the system. Only a registered account can be used.
Obs.0032	The bucket policy is invalid. Configure it again.
Obs.0033	The requested bucket name already exists. Bucket namespace is shared by all users in the system. Enter a different name and retry.
Obs.0034	The end time of the new validity period must be later than that of the old validity period.
Obs.0035	Cannot determine whether the bucket has objects or fragments. Check whether you have the read permission for this bucket.

17 Change History

Release Date	What's New
2019-05-22	This issue is the tenth official release. This issue incorporates the following changes: ● Updated the description of permission control.
2019-05-07	This issue is the ninth official release. This issue incorporates the following changes: ● Added the function for adding access paths.
2019-02-27	This issue is the eighth official release. This issue incorporates the following changes: ● Added the CDN download function.
2019-02-22	This issue is the seventh official release. This issue incorporates the following changes: ● Updated the section "External Buckets Overview".
2018-11-30	This issue is the sixth official release. This issue incorporates the following changes: ● Updated the section "Object Policy".

Release Date	What's New
2018-10-31	This issue is the fifth official release. This issue incorporates the following changes: ● Added access to the specific bucket or object on the login page. ● Added the description about configuring the database path in the system configuration of OBS Browser. ● Added the file renaming function. ● Added the object copying function. ● Added the object moving function. ● Added the object ACL function. ● Added the object policy function.
2018-05-31	This issue is the fourth official release. This issue incorporates the following changes: ● Added the section "Getting Started". ● Added the description of storage class. ● Added certificate verification. ● Modified the restrictions on mounting external buckets through OBS Browser. ● Updated the screenshots based on changes in the GUI.
2018-3-20	This issue is the third official release. This issue incorporates the following changes: ● Added object storage classes. ● Added the functions of changing bucket and object storage classes. ● Changed the software framework and updated screenshots.
2018-01-19	This issue is the second official release. This issue incorporates the following changes: ● Added the section of "Accessing OBS with Domain Names". ● Added the section of "Time Difference Is Longer Than 15 Minutes Between the Client and the Server". ● Added error codes. ● Updated the method to obtain AK/SK, Account ID, and User ID. ● Updated the description about the ACL.
2017-12-31	This issue is the first official release.