Object Storage Service

Tool Guide (obsutil)

 Issue
 01

 Date
 2025-04-18





HUAWEI TECHNOLOGIES CO., LTD.

Copyright © Huawei Technologies Co., Ltd. 2025. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

Trademarks and Permissions

NUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd. All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:

https://www.huawei.com/en/psirt/vul-response-process

For vulnerability information, enterprise customers can visit the following web page: <u>https://securitybulletin.huawei.com/enterprise/en/security-advisory</u>

Contents

1 obsutil Introduction	1
2 Downloading and Installing obsutil	4
3 Getting Started	7
3.1 Preparing the Environment	7
3.2 Initializing the Configuration	9
3.3 Quick Start	12
4 Bucket Commands	15
4.1 Creating a Bucket	15
4.2 Listing Buckets	18
4.3 Querying Bucket Properties	20
4.4 Setting Bucket Properties	22
4.5 Deleting a Bucket	25
4.6 Configuring a Bucket Policy	
4.7 Obtaining a Bucket Policy	
4.8 Deleting a Bucket Policy	
5 Object Commands	
5.1 Creating a Folder	30
5.2 Uploading an Object	
5.3 Querying Object Properties	45
5.4 Setting Object Properties	47
5.5 Listing Objects	53
5.6 Copying an Object	58
5.7 Moving an Object	68
5.8 Downloading an Object	
5.9 Generating the Download Link of an Object	
5.10 Deleting an Object	92
5.11 Synchronously Uploading Incremental Objects	95
5.12 Synchronously Copying Incremental Objects	
5.13 Synchronously Downloading Incremental Objects	
5.14 Restoring Objects from the Archive Storage	
5.15 Resuming a Failed Upload Task	
5.16 Resuming a Failed Copy Task	134

5.17 Resuming a Failed Download Task	
5.18 Listing Multipart Upload Tasks	
5.19 Deleting a Multipart Upload Task	
5.20 Creating an Authorization Code for Directory Sharing	156
5.21 Listing Objects by Using an Authorization Code	158
5.22 Downloading Objects by Using an Authorization Code	161
6 Auxiliary Commands	172
6.1 Updating a Configuration File	172
6.2 Deleting Part Records	174
6.3 Viewing Command Help Information	175
6.4 Querying the Version Number	
6.5 Archiving Log Files	
6.6 Listing Failure Result Files	181
7 Common Examples	183
7.1 Upload	
7.2 Synchronous Upload	
7.3 Download	
7.4 Synchronous Download	
7.5 Copy	
7.6 Synchronous Copy	
7.7 Listing	
7.8 Listing Multipart Upload Tasks	
7.9 Deleting All Multipart Upload Tasks in a Bucket	
8 Fault Locating	193
8.1 Overview	
8.2 Log Files	
8.3 Result Lists	
8.4 Return Codes	
9 Best Practices	197
9.1 Using the obsutil help Command to Search for Functions	
9.2 Configuring Scheduled Tasks Using the Crontab Command	
9.3 Setting obsutil Commands as Built-in Commands	
9.4 Configuring Auto Obtaining of Access Keys for obsutil	
9.5 Fine-Tuning obsutil Performance	
9.6 Using obsutil for Resumable Data Transfer	
9.7 Using obsutil to Upload a Symbolic Link	
9.8 Configuring an HTTP Proxy for obsutil	
9.9 Using obsutil to Share Directories9.10 Using obsutil to Replicate Data Across Regions on the Client Side	
9.11 Limiting the Upload and Download Rate for obsutil	
10 FAQs	

10.1 After Some Files Are Deleted in My Local Directory, Can obsutil Synchronously Delete Them from the Bucket?	
10.2 Can I Use obsutil to Directly Save a Listing Result to a Local File?	. 210
10.3 Why Is the Size of Objects Queried by obsutil Inconsistent with That on OBS Console?	. 211
10.4 How Can I Find Out Why Some Tasks in a Batch Task Failed?	211
10.5 How Can I Locate and Rectify I/O Timeout and EOF Errors?	.211
10.6 Why Is a Question Mark Displayed in the Batch Task Progress Bar?	.212
10.7 Can Multiple config Files Be Placed in One Directory?	. 212
10.8 Why Is the Number of Objects I Downloaded from OBS to My Local Windows PC Different from That in OBS?	
10.9 Can I Rename an Object or a Folder?	.214
A Configuration Parameters	215

obsutil Introduction

obsutil is a command line tool for accessing and managing OBS on Huawei Cloud. You can use this tool to create buckets and upload, download, or delete files/ folders. If you are familiar with the command line interface (CLI), obsutil is a good choice in batch processing and automatic tasks.

obsutil is compatible with Windows, Linux, and macOS. **Table 1-1** lists the recommended operating system (OS) versions. To download the obsutil for your OS, refer to **Downloading and Installing obsutil**.

OS	Recommended Version	
Windows	Windows 7	
	Windows 8	
	Windows 10	
	Windows 11	
	Windows Server 2016	
	Windows Server 2019	
Linux	• SUSE 11	
	EulerOS 2	
	CentOS 7	
macOS	macOS 10.13.4	

Table 1-1 Recommended C	DS versions	for obsutil
-------------------------	-------------	-------------

Billing

obsutil is free of charge. However, when obsutil is used, requests will be sent to OBS, and you will be billed for using OBS. For details about OBS billing, see **Billing Overview**.

Advantages

obsutil has the following advantages:

- 1. Easy to use
- 2. Lightweight and installation-free
- 3. Compatible with Windows, Linux, and macOS
- 4. Excellent performance and diversified configurations

Application Scenarios

- Automatic backup and archiving, for example, periodically uploading local data to OBS
- Operations that you cannot perform using other tools (like OBS Browser+). Such operations include synchronously uploading, downloading, and copying objects.

Functions

Table 1-2 lists the functions of obsutil.

l
i

Function	Description
Basic bucket management	 Allows you to: Create buckets of different storage classes in specific regions. Delete buckets. Get the bucket list or configuration information.
Basic object management	 Allows you to upload, download, delete, and list objects. In detail, you can: Upload one or more files or folders. Upload large files in multipart uploads. Synchronously upload, download, and copy incremental objects. Copy a single object or a batch of objects sharing the same prefix. Move a single object or a batch of objects sharing the same prefix. Resume failed upload, download, or copy tasks.
Logging	Provides logging on your client to record operations on buckets and objects for analysis.

Advanced bucket and object management will be available in later versions of obsutil. You can now use them on OBS Console. For details, see **Object Storage Service User Guide**.

Command Line Structures

Below describes how to use commands in obsutil:

- In Windows
 obsutil command [parameters...] [options...]
- In Linux or macOS

./obsutil command [parameters...] [options...]

NOTE

- command indicates the command to execute, for example, **ls** or **cp**.
- *parameters* indicates mandatory parameters, for example, the bucket name during bucket creation.
- options indicates optional parameters. They must start with a hyphen (-).
- Square brackets ([]) are not part of a command. Remove them before executing a specific command.
- You must escape special characters (including & < > and spaces) in a command using quotation marks. Single quotation marks are for Linux or macOS, and double quotation marks for Windows.
- You can pass additional parameters in the *-key=value* or *-key value* format, for example, *-acl=private*, or *-acl private*. Both formats are the same. Choose one as you like.
- In Windows, you can execute **obsutil.exe** to enter interactive mode. In this mode, you can run *command [parameters...] [options...]* where **obsutil** is not needed. Below gives an example:
 Enter "exit" or "quit" to logout
 Enter "help" or "help command" to show help docs

Input your command: to snow help of Input your command: -->**ls -limit=3 -s** obs://bucket-001 obs://bucket-002 obs://bucket-003 Bucket number: 3

Input your command: -->

• If you log in to a remote macOS or Linux via SSH to use obsutil, set **TMOUT** to **0** to prevent obsutil from logging out due to SSH timeout.

2 Downloading and Installing obsutil

Download Links

 Table 2-1 lists the download links of obsutil for different OSs.

OS	Software Package	SHA-256 Checksum
Windows (64-bit)	obsutil_windows64	46440850e63600b31888f2c 7c8e79db99f01d4ddd4e2b8 a1e01af282ee4be93a
Linux AMD (64-bit) Linux x86 (64-bit)	obsutil_linux_amd64	5b1b04b8076ab6171abe9a8 b91ec55aea3ceb026f7edfbe 1fde831631a536c37
Linux Arm (64-bit)	obsutil_linux_arm64	89ac3d9a57b2d9633be61e0 50024d52483881975f058e2 0c5e981a45587f9fa4
macOS (64-bit)	obsutil_mac64	7bf5afa0960e1ab163dccf0a 53bcd6fe13b9d6f7d6ba541c 116103bfa8424118

Table 2-1 Download links of obsutil

Getting Started with obsutil

You can download obsutil and then use it without installation. Methods of downloading obsutil vary depending on the OS type.

NOTE

Make sure that your cloud server has been connected to the Internet, or the obsutil installation will fail.

Windows

Step 1 Download the obsutil package to your local PC.

Step 2 Decompress the software package.

Step 3 Use Command Prompt to go to the decompressed folder and run obsutil commands.

----End

Linux

Step 1 Open the CLI and run the wget command to download obsutil.

Linux AMD (64-bit) and Linux x86 (64-bit):

wget https://obs-community-intl.obs.ap-southeast-1.myhuaweicloud.com/obsutil/current/ obsutil_linux_amd64.tar.gz

Linux Arm (64-bit):

wget https://obs-community-intl.obs.ap-southeast-1.myhuaweicloud.com/obsutil/current/obsutil_linux_arm64.tar.gz

NOTE

- The URL after **wget** is the download link of the installation package, for which you do not need to make any adaptations. To download the package, just copy and run the whole **wget** command.
- You can also download the obsutil package from a PC running Windows and then use a cross-platform transfer tool (such as WinSCP) to transfer the package to your Linux host.
- **Step 2** Run the following command in the directory where the obsutil package resides: tar -xzvf obsutil_linux_amd64.tar.gz
- **Step 3** List the obsutil directory. **x.x.x** indicates the obsutil version.

ll

```
dr-x----- 2 root root 4096 Jan 5 2024 obsutil_linux_amd64_x.x.x
-rw------ 1 root root 3845484 Mar 27 17:05 obsutil_linux_amd64.tar.gz
```

- **Step 4** Go to the directory where obsutil resides. **x.x.x** indicates the obsutil version. cd obsutil_linux_amd64_x.x.x
- **Step 5** Run the following command to grant the execute permissions for obsutil: chmod 755 obsutil

NOTE

This step is required, or error "No such file or directory" will be reported when you are querying the obsutil version number.

Step 6 Continue to run the following command in the directory. If the version number of obsutil is returned, the installation is successful. ./obsutil version

Figure 2-1 Successful installation

obsutil version:L3, obssdk version:3.23.12	rsion
0030C11 VC131011.E0, 00330k VC131011.5.25.12	
L operating system:linux, arch:amd64	
[root@server-1	

----End

macOS

- **Step 1 Download the obsutil package** to your local PC.
- **Step 2** Decompress the software package.
- **Step 3** Open the CLI, go to the directory where obsutil belongs, and run the following command to add execute permissions to obsutil:

----End

NOTE

To improve the usability of obsutil, you can set obsutil commands as built-in commands by running the shell or batch script in the software package. This way, you can run obsutil commands in any directory. For details, see **Setting obsutil Commands as Built-in Commands**.

More Operations

If you need to use obsutil on a Huawei Cloud ECS, you can configure access over an intranet to reduce costs on traffic by referring to **Using obsutil on a Linux ECS to Access OBS over an Intranet**.

References

- Quick Start
- Using the obsutil help Command to Search for Functions

3 Getting Started

3.1 Preparing the Environment

To use obsutil, you need to register a cloud service account, enable OBS, and obtain the access keys (AK and SK) first.

Step 1 Register a cloud service account.

Before using OBS, ensure that you have a cloud service account.

- 1. Open a browser.
- 2. Log in to the Huawei Cloud official website.
- 3. In the upper right corner of the page, click **Register**.
- 4. Enter the registration information and click **Register**.

Step 2 Enable OBS.

Ensure that your account balance is sufficient before using OBS.

- 1. Log in to the management console.
- 2. Click **Billing & Costs** from the top menu bar. The **Billing Center** page is displayed.
- 3. Choose Funds Management > Top Up. The Top Up page is displayed.
- 4. Top up the account as prompted.
- 5. Go back to the management console page after the recharging is successful.
- 6. Choose Service List > Object Storage Service to access OBS Console.
- Step 3 (Optional) Create an IAM user.

For data security, it is recommended that you do not use the account directly to access OBS. Through the Identity and Access Management (IAM) service, you can create a user who has the permission to access OBS resources and manage buckets and objects on obsutil. If you do not need to use any IAM user, skip this step.

- 1. On the top navigation bar of the console, click = in the upper left corner and choose Management & Governance > Identity and Access Management.
- 2. On the displayed IAM console page, create a user group with OBS permissions configured.

For details, see **Creating a User Group**. After the user group is created, locate the row that displays **Global service** > **OBS** in the **User Group Permissions**, and click **Configure Policy** in the row to set OBS permissions for the user group.

NOTE

For details about OBS policies, see **Permissions Management**.

3. Create a user.

For details, see **Creating an IAM User**. When creating a user, set the **User Group** to the one created in **Step 3.2** with OBS permissions configured and set the access type to **Programmatic access**.

NOTE

- If the user group is not configured with OBS permissions, you can configure finegrain permissions on OBS Console through bucket policies or object policies. For details, see **OBS Permission Control**.
- If **Programming access** is not enabled during user creation, enable it by following the instructions provided in **Viewing or Modifying IAM User Information**.

Step 4 Obtain access keys.

OBS uses AKs and SKs in user accounts for signature verification to ensure that only authorized accounts can access specified OBS resources. Detailed explanations about AK and SK are as follows:

- Access key ID (AK): indicates the ID of the access key, which is a unique identifier used together with a secret access key to sign requests cryptographically.
- Secret access key (SK): indicates the private key used together with its associated AK to cryptographically sign requests. The AK and SK are used together to identify a request sender to prevent the request from being modified.

NOTE

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

Create access keys as follows:

- 1. In the upper right corner of the console page, select **My Credentials** under the username.
- 2. On the **My Credentials** page, select **Access Keys** in the navigation pane on the left.
- 3. On the Access Keys page, click Add Access Key.

NOTE

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

4. 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 select the verification by email or mobile phone.
- 5. Click **OK**.
- 6. In the **Download Access Key** dialog box that is displayed, click **OK** to save the access keys to your browser's default download path.

NOTE

Keep the access keys properly to prevent information leakage. If you click **Cancel** in the dialog box, the access keys will not be downloaded, and you cannot download them later. Re-create access keys if required.

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

NOTE

In the access key file, the value in the **Access Key ID** column is the AK, and the value in the **Secret Access Key** column is the SK.

----End

3.2 Initializing the Configuration

Before using obsutil, you need to configure the OBS endpoint and AK/SK for obsutil to interconnect with OBS. Then, you can use obsutil to operate OBS buckets and objects.

Prerequisites

- You have downloaded the obsutil software package. For details, see Downloading and Installing obsutil.
- You have obtained the enabled **regions and endpoints** of OBS. If you want to access OBS in the CN-Hong Kong region, the address you actually visit is **https://obs.ap-southeast-1.myhuaweicloud.com**.
- You have obtained the AK/SK from the access key management page. For details, see Preparing the Environment.

obsutil Initialization Methods

Method 1: Run the **config** command (for more information about **config**, see **Updating a Configuration File**):

In Windows

Using a permanent AK/SK pair: obsutil config -i=*ak* -k=*sk* -e=*endpoint* Using a temporary AK/SK pair and a security token: obsutil config -i=*ak* -k=*sk* -t=*token* -e=*endpoint* • In Linux or macOS

Using a permanent AK/SK pair:

./obsutil config -i=*ak* -k=*sk* -e=*endpoint*

Using a temporary AK/SK pair and a security token:

./obsutil config -i=*ak* -k=*sk* -t=*token* -e=*endpoint*

D NOTE

- After the command is executed, there will be the **.obsutilconfig** file created in the directory (~ in Linux or macOS or **C:\Users**<*username*> in Windows) where obsutil commands run. This file contains all the configuration information of obsutil.
- For details about the parameters in the **.obsutilconfig** file, see **Configuration Parameters**.
- You can obtain the temporary AK/SK pair and security token by referring to **Obtaining a User Token**.
- The **.obsutilconfig** file contains the AK/SK information, so it is hidden by default to prevent leakage. To query this file, run the following command in the directory where obsutil commands run.
 - In Windows dir
 - In Linux or macOS
 Is -a

or

ls -al

- obsutil encrypts the AK/SK information in the .obsutilconfig file.
- After you use -**i**, -**k**, and -**e** to configure user authentication, the **history** command in Linux can query the configured values.

Method 2: Initialize the configuration by **automatically obtaining access keys for obsutil**.

Method 3: Initialize obsutil in interactive mode.

Below gives an example of initializing obsutil in Linux.

- 1. Run the following command: ./obsutil config -interactive
- 2. Configure the **ak**, **sk**, **endpoint**, and **token** parameters.

The following table describes the parameters.

Table 3-1

Parameter	Optional or Mandatory	Description
ak	Mandatory	AK in permanent or temporary security credentials
sk	Mandatory	SK in permanent or temporary security credentials

Parameter	Optional or Mandatory	Description
endpoint	Mandatory	Endpoint for accessing OBS, which can contain the protocol type, domain name, and port number (optional), for example, https://your- <i>endpoint</i> : 443 . (For security purposes, you are advised to use HTTPS. The port number 443 can be omitted.)
		If you want to access OBS in the CN-Hong Kong region, the address you actually visit is https://obs.ap- southeast-1.myhuawei cloud.com.
		 NOTE To view the endpoints and regions available for OBS, see Regions and Endpoints.
token	Optional	Security token in the temporary security credentials. It is mandatory when temporary security credentials are used.

NOTE

For details about **config**, see **Updating a Configuration File**.

Checking the Connectivity

After the initial configuration is complete, run the following command to check the connectivity:

- In Windows
 obsutil ls -s
- In Linux or macOS ./obsutil ls -s

Check the command output:

• If it contains "Bucket number", the configuration is correct.

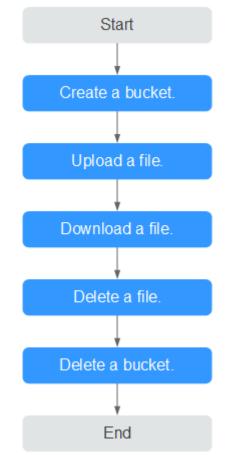
- If it contains "Http status [403]", the access keys are wrong.
- If it contains "A connection attempt failed", OBS cannot be connected. Then, check the network condition.
- If it contains "Error: cloud_url [url] is not in well format", the domain name to be accessed is incorrect. Check the domain name in the configuration file.

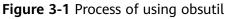
NOTE

If the command output contains "Http status [403]", you may not have the required permissions for obtaining the bucket list. A further analysis is required to identify the root cause.

3.3 Quick Start

This section uses the Linux OS as an example to describe how to use obsutil to perform basic operations in OBS. For details, see **Figure 3-1**.





Prerequisites

- You have obtained obsutil and completed initial configuration.
- The directory saving obsutil is accessed.

Procedure

Step 1 Run the ./obsutil mb obs://bucket-test -location=ap-southeast-1 command to create a bucket named bucket-test in the CN-Hong Kong region.

./obsutil mb obs://bucket-test -location=ap-southeast-1

Create bucket [bucket-test] successfully!

NOTE

In the preceding command, parameter **location** indicates the region where a bucket is created. It is mandatory only when the endpoint set during initial configuration belongs to any other regions than the default one CN North-Beijing1 (cn-north-1). To view the currently valid regions, see **Regions and Endpoints**.

Step 2 Run the ./obsutil cp /temp/test.txt obs://bucket-test/test.txt command to upload the test.txt file to bucket bucket-test.

./obsutil cp /temp/test.txt obs://bucket-test/test.txt

Parallel:5Jobs:5Threshold:52428800PartSize:5242880VerifyLength:falseVerifyMd5:falseCheckpointDir:/temp/.obsutil_checkpointFalse

test.txt:[=======] 100.00% 48.47 KB/s 0s Upload successfully, 4.44KB, /temp/test.txt --> obs://bucket-test1/test.txt

NOTE

To upload local folder **test** to the OBS bucket, run the following command:

./obsutil cp /test/ obs://bucket-test -r -f

- For details about the **cp** command, see **Uploading an Object**.
- For more upload scenarios, see Upload.
- Step 3 Run the ./obsutil cp obs://bucket-test/test.txt /temp/test1.txt command to download test.txt from bucket bucket-test to a local PC.

./obsutil cp obs://bucket-test/test.txt /temp/test1.txt

Parallel:5Jobs:5Threshold:52428800PartSize:52428800VerifyLength:falseVerifyMd5:falseCheckpointDir:/temp/.obsutil_checkpoint

NOTE

To download directory **test** from the OBS bucket to the local folder **temp**, run the following command:

./obsutil cp obs://bucket-test/test /temp -r -f

- For details about the **cp** command, see **Downloading an Object**.
- For more download scenarios, see **Download**.
- Step 4 Run the ./obsutil rm obs://bucket-test/test.txt -f command to delete object test.txt from bucket bucket-test.

./obsutil rm obs://bucket-test/test.txt -f

Delete object [test.txt] in the bucket [bucket-test] successfully!

Step 5 Run the ./obsutil rm obs://bucket-test -f command to delete bucket bucket-test.

./obsutil rm obs://bucket-test -f

Delete bucket [bucket-test] successfully!

----End

4 Bucket Commands

4.1 Creating a Bucket

Function

You can use this command to create a bucket. A bucket name must be unique in OBS. One account can create a maximum of 100 buckets.

NOTE

If you create a bucket and name it the same as an existing one in the same account and region, no error will be reported and status code 200 is returned. The bucket properties comply with those set in the first creation request. In other cases, creating a bucket with the same name as an existing one will receive the status code 409, indicating that the bucket already exists.

NOTICE

If the configured endpoint is a global domain name, you may need to wait several minutes before uploading objects to the created bucket. Therefore, set the endpoint to a regional domain name according to **Initializing the Configuration** if you want to upload objects instantly to the bucket.

Command Line Structure

- In Windows
 - obsutil mb obs://bucket [-fs] [-az=xxx] [-acl=xxx] [-sc=xxx] [-location=xxx] [-config=xxx] [-e=xxx] [i=xxx] [-k=xxx] [-t=xxx]
- In Linux or macOS
 ./obsutil mb obs://bucket [-fs] [-az=xxx] [-acl=xxx] [-sc=xxx] [-location=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Examples

• Take the Windows OS as an example. Run the **obsutil mb obs://bucket-test** command to create a bucket. The creation is successful.

obsutil mb obs://bucket-test Start at 2024-09-29 07:52:11.3769487 +0000 UTC

Create bucket [bucket-test] successfully, request id [000001923CC401864018BA75753D2D5F]

 Take the Windows OS as an example. Run the obsutil mb obs://bucket001 command to create a bucket with the same name as the bucket of another account. The creation fails.
 obsutil mb obs://bucket001 Start at 2024-09-30 07:03:50.1378331 +0000 UTC

Create bucket [bucket001] failed, status [409], error code [BucketAlreadyExists], error message [The requested bucket name is not available. The bucket namespace is shared by all users of the system. Please select a different name and try again.], request id [0000019241BE18DB4019EDD66E135C56]

Parameter Description

Parame ter	Optional or Mandatory	Description	
bucket	Mandatory	 Bucket name NOTE A bucket name must comply with the following rules: Contains 3 to 63 characters, including lowercase letters, digits, hyphens (-), and periods (.), and starts with a digit or letter. Cannot be an IP address. Cannot start or end with a hyphen (-) or period (.). Cannot contain two consecutive periods (.), for example, my.bucket. Cannot contain periods (.) and hyphens (-) adjacent to each other, for example, my.bucket or my.bucket. 	
fs	Optional (additional parameter)	Creates a parallel file system.	
az	Optional (additional parameter)	 Specifies a bucket's data redundancy policy. Possible values are: multi-az NOTE <pre> If multi-az is used, a bucket with the multi-AZ storage policy will be created. If this parameter is not specified, a bucket with the single-AZ storage policy will be created.</pre> 	
acl	Optional (additional parameter)	 Access control policies that can be specified when creating a bucket. Possible values are: private public-read public-read-write NOTE The preceding three values indicate private read and write, public read, and public read and write. 	

Parame ter	Optional or Mandatory	Description
sc	Optional (additional parameter)	 Default bucket storage class that can be specified when creating a bucket. Possible values are: standard: Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB. warm: Infrequent Access storage class. It is ideal for storing infrequently accessed (less than 12 times a year) data, but when needed, the access has to be fast. cold: Archive storage class. It provides secure, durable, and inexpensive storage for rarely-accessed (once a year) data.
location	Mandatory unless the region where the OBS service resides is not the default region (additional parameter)	Region where the bucket resides. NOTE This parameter indicates the region where a bucket will be created. It is mandatory only when the endpoint belongs to any other regions than the default one CN North-Beijing1 (cn-north-1). To view the currently valid regions, see Regions and Endpoints .
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

4.2 Listing Buckets

Function

You can use this command to obtain the bucket list. In the list, bucket names are displayed in lexicographical order.

Command Line Structure

- In Windows
 obsutil ls [-s] [-fs] [-sc] [-du] [-fs] [-j=1] [-limit=1] [-format=default] [-config=xxx] [-e=xxx]
 [-k=xxx] [-t=xxx]
- In Linux or macOS ./obsutil ls [-s] [-fs] [-sc] [-du] [-fs] [-j=1] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Examples

• Take the Windows OS as an example. Run the **obsutil ls -limit=5** command to obtain the bucket list.

obsutil ls -limit=5

Start at 2024-09-29 07:58:46.0506904 +0000 UTC

Bucket obs://bucket001	CreationDate 2018-09-03T01:53:0	Location 2Z examp	BucketType ole OBJECT
obs://bucket002	2018-11-01T01:40:0	1Z examp	ole OBJECT
obs://bucket003	2018-10-25T11:45:4	5Z examp	ole OBJECT
obs://bucket004	2018-10-26T02:33:0	9Z examp	ole OBJECT
obs://bucket005	2018-10-26T02:34:5	0Z examp	ole OBJECT
Bucket number : 5			

Parameter Description

Parameter	Optional or Mandatory	Description
S	Optional (additional parameter)	Displays simplified query result. NOTE In the simplified format, the returned result contains only the bucket name.
SC	Optional (additional parameter)	Queries the storage classes of the buckets when listing buckets.
j	Optional (additional parameter). It must be used together with sc .	Indicates the maximum number of concurrent tasks for querying the bucket storage class. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.

Parameter	Optional or Mandatory	Description
du	Optional (additional parameter)	Quickly returns the total size of listed objects, without displaying detailed object information. This parameter can be used together with other parameters. NOTE This parameter takes effect only on listing objects, but not on listing buckets.
fs	Optional (additional parameter)	 Specifies whether the method of listing parallel file systems is applied. If you are listing parallel file systems, you are recommended to add this parameter. CAUTION With this method, the listing time required varies largely depending on the directory structures. After this parameter is enabled, marker and limit will be ignored. Then, the size of the bucket, parallel file system, or directory will be calculated. This parameter is only supported by obsutil 5.5.12 and later.
limit	Optional (additional parameter)	Maximum number of buckets that can be queried. If the value is less than 0, all buckets are listed. If it is left blank, a maximum of 1000 buckets can be listed by default.
format	Optional (additional parameter)	Prints a listing result in the user- defined format. Currently, the value can only be default , indicating that the listing result is displayed in one row.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
е	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.

Parameter	Optional or Mandatory	Description
t	Optional (additional parameter)	Specifies the user's security token.

NOTE

In a bucket listing result, **BucketType** indicates the bucket type. **OBJECT** indicates an object storage bucket, while **POSIX** indicates a parallel file system.

4.3 Querying Bucket Properties

Function

You can use this command to query the basic properties of a bucket, including its default storage class, region, version ID, storage usage, bucket quota, AZ, whether it supports POSIX, and the number of objects in the bucket.

Command Line Structure

- In Windows obsutil stat obs://bucket [-acl] [-bf=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
- In Linux or macOS ./obsutil stat obs://bucket [-acl] [-bf=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

Examples

• Take the Windows OS as an example. Run the **obsutil stat obs://bucket-test** command to query the basic properties of bucket **bucket-test**. **obsutil stat obs://bucket-test**

Start at 2024-09-29 07:58:46.0506904 +0000 UTC

```
Bucket:
obs://bucket-test
StorageClass:
standard
ObsVersion:
3.0
AvailableZone:
multi-az
BucketType:
OBJECTObjectNumber:
8005
Size:
320076506
Quota:
0
```

Parameter Description

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name
acl	Optional	Queries the access control policies of the bucket while querying bucket properties.
bf	Optional (additional parameter)	 Display format of the used bucket capacity (in bytes). Possible values are: human-readable raw NOTE If this parameter is not configured, the display format of the used bucket capacity (in bytes) is determined by the humanReadableFormat parameter in the configuration file.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Response

Field	Description	
Bucket	Bucket name	
StorageClass	Default storage class of the bucket	
Location	Region where the bucket resides	
ObsVersion	Version of the bucket	
AvailableZone	AZ where the bucket is located	

Field	Description	
BucketType	Type of a bucket. OBJECT indicates a bucket for object storage. POSIX indicates a bucket used as a parallel file system.	
ObjectNumber	Number of objects in the bucket	
Size	Storage usage of the bucket, in bytes	
Quota	Bucket quota. Value 0 indicates that no upper limit is set for the bucket quota.	
Acl	Access control policy of the bucket	

4.4 Setting Bucket Properties

Function

You can use this command to set the properties of a bucket, such as storage classes and access policies.

Command Line Structure

- In Windows
 obsutil chattri obs://bucket [-sc=xxx] [-acl=xxx] [-aclXml=xxx] [-config=xxx] [-e=xxx] [-i=xxx]
 [-t=xxx]
- In Linux or macOS
 ./obsutil chattri obs://bucket [-sc=xxx] [-acl=xxx] [-acl=xxx] [-aclXml=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil chattri obs://buckettest -acl=private command to change the access control policy of the bucket to private read and write. obsutil chattri obs://bucket-test -acl=private

Start at 2024-09-29 07:58:46.0506904 +0000 UTC

Set the acl of bucket [bucket-test] to [private] successfully, request id [04050000016836C5DA6FB21F14A2A0C0]

Parameter Description

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name

Parameter	Optional or Mandatory	Description
SC	Optional (additional	Default storage class of the bucket. Possible values are:
	parameter)	• standard : Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB.
		• warm: Infrequent Access storage class. It is ideal for storing infrequently accessed (less than 12 times a year) data, but when needed, the access has to be fast.
		• cold : Archive storage class. It provides secure, durable, and inexpensive storage for rarely-accessed (once a year) data.
		NOTE If the multi-AZ mode is enabled for a bucket, the default storage class of the bucket cannot be set to cold .
acl	Optional (additional	Access control policies that can be specified for buckets. Possible values are:
	parameter)	• private
		• public-read
		• public-read-write
		NOTE The preceding three values indicate private read and write, public read, and public read and write.

Parameter	Optional or Mandatory	Description
aclXml	Optional (additional parameter)	Access control policy of the bucket, in XML format. <accesscontrolpolicy> <owner> <id>ownerid</id> </owner> <accesscontrollist> <grante> <grantee> <id>userid</id> </grantee> <permission>[WRITE/WRITE_ACP/READ/ READ_ACP/FULL_CONTROL]</permission> <grante> <grantee> <canned>Everyone</canned> </grantee> <canned>Everyone</canned> </grante> <accesscontrollist> </accesscontrollist> </grante></accesscontrollist> </accesscontrolpolicy> NOTE • Owner: Optional. Specify the bucket owner's ID. • In AccessControlList, the Grant field contains the authorized users. Grantee specifies the IDs of authorized users. Canned specifies the authorized user group (currently, only Everyone is supported). • The following permissions can be granted: WRITE (write), WRITE_ACP (write ACL), READ (read), READ_ACP (read ACL), and FULL_CONTROL (full control). NOTICE Because angle brackets (<) and (>) are unavoidably included in the parameter value, you must use quotation marks to enclose them for escaping when running the command. Use single quotation marks for Linux or macOS and quotation marks for Windows.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.

Parameter	Optional or Mandatory	Description
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

NOTE

Only one from **sc**, **acl**, or **aclXml** can be set for each command.

4.5 Deleting a Bucket

Function

You can use this command to delete a bucket. The bucket to be deleted must be empty (containing no objects, historical versions, or fragments).

NOTE

To delete a non-empty bucket, run the commands in **Deleting a Multipart Upload Task** and **Deleting an Object** to clear the bucket, and then run the following command to delete the bucket.

Command Line Structure

- In Windows obsutil rm obs://bucket [-f] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
- In Linux or macOS ./obsutil rm obs://bucket [-f] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil rm obs://bucket-test command to delete bucket bucket-test. obsutil rm obs://bucket-test

Start at 2024-09-30 07:58:33.736622 +0000 UTC

Do you want to delete bucket [bucket-test] ? Please input (y/n) to confirm:

Delete bucket [bucket-test] successfully!

Parameter Description

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name

Parameter	Optional or Mandatory	Description
f	Optional (additional parameter)	Runs in force mode.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

4.6 Configuring a Bucket Policy

Function

You can use this command to configure a bucket policy.

Command Line Structure

In Windows

obsutil bucketpolicy obs://bucket -method=put -localfile=xxx [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

In Linux or macOS
 ./obsutil bucketpolicy obs://bucket -method=put -localfile=xxx [-config=xxx] [-e=xxx] [-i=xxx]
 [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil bucketpolicy obs:// bucket -method=put -localfile=d:\temp\policy.json command to set a bucket policy based on file policy.json. obsutil bucketpolicy obs://bucket -method=put -localfile=d:\temp\policy.json

Start at 2024-09-30 07:59:28.8675681 +0000 UTC

Put bucketPolicy succeed, requestId is [04050000016836C5DA6FB21F14A2A0C0]

Parameter Description

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name
method	Mandatory	Specifies the method. Set this parameter to put when configuring a bucket policy.
localfile	Mandatory	Path of the local policy file to import
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

4.7 Obtaining a Bucket Policy

Function

You can use this command to obtain a bucket policy.

Command Line Structure

• In Windows

obsutil bucketpolicy obs://bucket -method=get [-localfile=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

• In Linux or macOS ./obsutil bucketpolicy obs://bucket -method=get [-localfile=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil bucketpolicy obs:// bucket -method=get -localfile=d:\temp\policy.json command to export the bucket policy to local file policy.json. obsutil bucketpolicy obs://bucket -method=get -localfile=d:\temp\policy.json Start at 2024-09-30 08:08:59.9412332 +0000 UTC

Export bucketPolicy to [d:\temp\policy.json] succeed, requestId is [04050000016836C5DA6FB21F14A2A0C0]

Parameter Description

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name
method	Mandatory	Specifies the method. Set this parameter to get when obtaining a bucket policy.
localfile	Optional (additional parameter)	If this parameter is set, the policy is exported to a local file. If not set, the policy is exported in a standard manner by default.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

4.8 Deleting a Bucket Policy

Function

You can use this command to delete a bucket policy.

Command Line Structure

- In Windows obsutil bucketpolicy obs://bucket -method=delete [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx] [-t=xxx]
- In Linux or macOS ./obsutil bucketpolicy obs://bucket -method=delete [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil bucketpolicy obs:// bucket -method=delete command to delete a bucket policy.
 obsutil bucketpolicy obs://bucket -method=delete

Start at 2024-09-30 08:09:29.6396262 +0000 UTC

Delete bucketPolicy succeed, requestId is [04050000016836C5DA6FB21F14A2A0C0]

Parameter Description

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name
method	Mandatory	Specifies the method. Set this parameter to delete when deleting a bucket policy.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

5 Object Commands

5.1 Creating a Folder

Function

You can use this command to create a folder in a specified bucket or local file system.

NOTICE

No error is returned if a folder with the same name as an existing one is created, and the content of the existing folder remains unchanged.

Command Line Structure

- In Windows
 - Creating a folder in a specified bucket
 obsutil mkdir obs://bucket/folder[/subfolder1/subfolder2] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Creating a folder in the local file system obsutil mkdir folder_url [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
- In Linux or macOS
 - Creating a folder in a specified bucket
 ./obsutil mkdir obs://bucket/folder[/subfolder1/subfolder2] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Creating a folder in the local file system
 ./obsutil mkdir folder_url [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil mkdir obs://buckettest/folder1/folder2 command to create a folder in a bucket. obsutil mkdir obs://bucket-test/folder1/folder2

The bucket [bucket-test] does not support POSIX, create folder(s) step by step Create folder [obs://bucket-test/folder1/] successfully, request id

[0000016979E1D23C860BB3D8E4577C5E] Create folder [obs://bucket-test/folder1/folder2] successfully, request id [0000016979E1D2B2860BB5181229C72C]

Parame ter	Optional or Mandatory	Description
bucket	Mandatory when creating a folder in a specified bucket	Bucket name
folder	Mandatory when creating a folder in a specified bucket	Folder path in the bucket. This value can contain multi-level folders. Separate each level with a slash (/).
folder_u rl	Mandatory when creating a folder in the local file system	Folder path in the local file system. The value can be an absolute path or a relative path.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

5.2 Uploading an Object

Function

You can use this command to upload one or more local files or folders to a specified path in OBS. These files can be texts, images, videos, or any other type of files.

NOTICE

Do not change the local file or folder when uploading it. Otherwise, the upload may fail or data may be inconsistent.

Restrictions

obsutil has restrictions on the size of files or folders to be uploaded. You can upload an empty file or folder of 0 bytes. You can also upload a single file or folder with a maximum size of 5 GB in normal mode or a single file with a maximum size of 48.8 TB in multipart mode.

Command Line Structure

- In Windows
 - Uploading a file

obsutil cp file_url obs://bucket[/key] [-arcDir=xxx] [-dryRun] [-link] [-u] [-vlength] [-vmd5] [p=1] [-threshold=5248800] [-acl=xxx] [-encoding-type=url] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-o=xxx] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

– Uploading a folder

obsutil cp folder_url obs://bucket[/key] -r [-arcDir=xxx] [-dryRun] [-link] [-f] [-flat] [-u] [vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-encoding-type=url] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

- Uploading multiple files/folders

obsutil cp file1_url,folder1_url|filelist_url obs://bucket[/prefix] -msm=1 [-r] [-arcDir=xxx] [dryRun] [-link] [-f] [-u] [-vlength] [-vmd5] [-flat] [-j=1] [-p=1] [-threshold=52428800] [acl=xxx] [-encoding-type=url] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [exclude=*.xxx][-timeRange=time1-time2] [-at] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

NOTE

In this command, **/prefix** is the name prefix for uploading folders. For the execution examples, see **Upload**.

- In Linux or macOS
 - Uploading a file

./obsutil cp file_url obs://bucket[/key] [-arcDir=xxx] [-dryRun] [-link] [-u] [-vlength] [-vmd5] [p=1] [-threshold=5248800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-o=xxx] [cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

– Uploading a folder

./obsutil cp folder_url obs://bucket[/key] -r [-arcDir=xxx] [-dryRun] [-link] [-f] [-flat] [-u] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-at] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] - Uploading multiple files/folders

/obsutil cp file1_url,folder1_url|filelist_url obs://bucket[/prefix] -msm=1 [-r] [-arcDir=xxx] [dryRun] [-link] [-f] [-u] [-vlength] [-vmd5] [-flat] [-j=1] [-p=1] [-threshold=52428800] [acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [t=xxx]

NOTE

In this command, **/prefix** is the name prefix for uploading folders. For the execution examples, see **Upload**.

Examples

Take the Windows OS as an example. Run the obsutil cp d:\temp\test.txt obs://bucket-test/key command to upload the test.txt file in the temp directory in the D: drive to bucket bucket-test and rename the file as key. obsutil cp d:\temp\test.txt obs://bucket-test/key
 Start at 2024-09-30 08:11:41.6724827 +0000 UTC

Parallel:5Jobs:5Threshold:50.00MBPartSize:autoVerifyLength:falseVerifyMd5:falseCheckpointDir:C:\Users\Administrator\.obsutil_checkpoint

[=====] 100.00% 1.68 MB/s 8.46MB/ 8.46MB 5s

Upload successfully, 8.46MB, n/a, d:\temp\test.txt --> obs://bucket-test/key, cost [5], status [200], request id [0000016979E1D2B2860BB5181229C72C]

 Take the Windows OS as an example. Run the obsutil cp d:\temp obs:// bucket-test -f -r command to recursively upload all files and subfolders in the temp directory in the D: drive to the temp folder in bucket bucket-test. obsutil cp d:\temp obs://bucket-test -f -r Start at 2024-09-30 08:14:12.1406275 +0000 UTC

Parallel:5Jobs:5Threshold:50.00MBPartSize:autoVerifyLength:falseVerifyMd5:falseCheckpointDir:C:\Users\Administrator\.obsutil_checkpoint

Task id: 104786c8-27c2-48fc-bc6a-5886596fb0ed OutputDir: C:\Users\Administrator\.obsutil_output

[======] 100.00% tps:35.71 2.02 KB/s 7.20MB/7.20MB 0s Succeed count: 5 Failed count: 0 Succeed bytes: xxx Metrics [max cost:90 ms, min cost:45 ms, average cost:63.80 ms, average tps:35.71, transferred size: 7.20MB]

Task id: 104786c8-27c2-48fc-bc6a-5886596fb0ed

• For more examples, see Upload.

Parameter	Optional or Mandatory	Description
file_url	Optional for uploading multiple files/ folders Mandatory for uploading a file	 Local file path NOTE Do not nest paths when uploading multiple files/folders. For example, you cannot specify /a/b/c and /a/b/ at the same time. If this parameter is configured when uploading multiple files/folders, msm must be set to 1. In this case, use commas (,) to separate multiple file paths, for example, file_url1,file_url2. Files and folders can both be included when uploading multiple files/folders. For example, file_url1,folder_url2.
folder_url	Optional for uploading multiple files/ folders Mandatory for uploading a folder	 Local folder path NOTE If flat is not configured when uploading a folder, the entire folder is uploaded. If flat is configured, all files in the folder are uploaded. Do not nest paths when uploading multiple files/folders. For example, you cannot specify /a/b/c and /a/b/ at the same time. If this parameter is configured when uploading multiple files/folders, msm must be set to 1. In this case, use commas (,) to separate multiple folder paths, for example, folder_url1,folder_url2. Files and folders can both be included when uploading multiple files/folders. For example, file_url1,folder_url1,file_url2,folder_url2.
filelist_url	Optional for uploading multiple files/ folders	 Indicates the path of the file that contains the list of files/folders to be uploaded. If this parameter is configured, msm must be set to 2. NOTE The list file is in common text file formats, such as TXT and CSV. Each line in the file indicates a file or folder to be uploaded. For example: file_url1 file_url2 folder_url1 folder_url2 Do not nest paths in the list file. For example, you cannot specify /a/b/c and /a/b/ at the same time.
bucket	Mandatory	Bucket name

Parameter	Optional or Mandatory	Description
key	Optional	Indicates the object name or object name prefix specified when uploading a file, or the object name prefix specified when uploading a folder. The rules are as follows:
		 If this parameter is left blank when uploading a file, the file is uploaded to the root directory of the bucket and the object name is the file name. If the value ends with a slash (/), the value is used as the object name prefix when the file is uploaded, and the object name is the value plus the file name. If the value does not end with a slash (/), the file is uploaded with the value as the object name.
		 If this parameter is left blank when uploading a folder, the folder is uploaded to the root directory of the bucket. If the value ends with a slash (/), the value is used as the object name prefix of the folder to be uploaded. If the value does not end with a slash (/), the folder to be uploaded is prefixed with the value plus a slash (/). NOTE For details about how to use this parameter, see Upload.
fr	Optional for uploading a file (additional parameter)	Generates an operation result list when uploading a file.
flat	Optional for uploading a folder or multiple files/ folders (additional parameter)	Uploads all files in a folder but not the folder itself.
arcDir	Optional (additional parameter)	Path to which the uploaded files are archived
dryRun	Optional (additional parameter)	Conducts a dry run.

Parameter	Optional or Mandatory	Description
link	Optional (additional parameter)	 Uploads the actual path of the symbolic-link file/folder NOTICE If this parameter is not specified and the file to be uploaded is a symbolic-link file whose target file does not exist, the exception message "The system cannot find the file specified" will be displayed in Windows OS, while the exception message "No such file or directory" will be displayed in macOS or Linux OS. Avoid the symbolic link loop of a folder, otherwise, the upload will exit due to panic. If you do not want the system to panic, set panicForSymbolicLinkCircle to false in the configuration file.
u	Optional (additional parameter)	Indicates incremental upload. If this parameter is set, each file can be uploaded only when it does not exist in the bucket, its size is different from the namesake one in the bucket, or it has the latest modification time. CAUTION When you compare each local file with data in the bucket, a billable HEAD request is generated. For details, see Requests .
vlength	Optional (additional parameter)	After the upload completes, check whether the sizes of the objects in the bucket are the same as those of the local files.
vmd5	Optional (additional parameter)	 After the upload completes, check whether the MD5 values of the objects in the bucket are the same as those of the local files. NOTE If the size of the file or folder to be uploaded is too large, using this parameter will degrade the overall performance due to MD5 calculation. After the MD5 verification is successful, this parameter value is used for metadata x-obs-meta-md5chksum, for later MD5 verification during download or copy. CAUTION If your object needs encryption, do not use this parameter.
р	Optional (additional parameter)	Indicates the maximum number of concurrent multipart upload tasks when uploading a file. The default value is the value of defaultParallels in the configuration file.

Parameter	Optional or Mandatory	Description
threshold	Optional (additional parameter)	Indicates the threshold for enabling multipart upload, in bytes. The default value is the value of defaultBigfileThreshold in the configuration file. NOTE
		 If the size of the file or folder to be uploaded is smaller than the threshold, upload it directly. Otherwise, a multipart upload is required.
		 If you upload a file or folder directly, no part record is generated, and resumable transmission is not supported.
		• This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).
acl	Optional (additional parameter)	Access control policies that can be specified when uploading files. Possible values are: • private • public-read
		 public-read-write bucket-owner-full-control
		NOTE The preceding four values indicate private read and write, public read, public read and write, and bucket owner full control.
sc	Optional (additional parameter)	Indicates the storage classes of objects that can be specified when uploading files. Possible values are:
		• standard : Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB.
		• warm: Infrequent Access storage class. It is ideal for storing infrequently accessed (less than 12 times a year) data, but when needed, the access has to be fast.
		• cold : Archive storage class. It provides secure, durable, and inexpensive storage for rarely-accessed (once a year) data.

Parameter	Optional or Mandatory	Description
meta	Optional (additional parameter)	 Indicates the standard and custom metadata that can be specified during file upload. This parameter should be configured in the following format: <i>key1:value1# key2:value2# key3:value3</i>. NOTE The format example above indicates that the objects copied to the bucket contain three groups of custom metadata: <i>key1:value1, key2:value2</i>, and <i>key3:value3</i>. Standard metadata headers include Content-Type, Content-Encoding, Cache-Control, Content-Disposition, Content-Language and Expires.
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart upload task, in bytes. The value ranges from 100KB to 5GB. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source file size.
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the subfolder in the home directory of the user who executes obsutil commands. NOTE A part record is generated during a multipart upload and saved to the upload subfolder. After the upload succeeds, its part record is deleted automatically. If the upload fails or is suspended, the system attempts to resume the task according to its part record when you perform the upload the next time.
r	Mandatory for uploading a folder (additional parameter) Optional for uploading multiple files/ folders	Indicates files and subfolders within the folder when uploading a folder recursively.

Parameter	Optional or Mandatory	Description
f	Optional for uploading a folder or multiple files/ folders (additional parameter)	Runs in force mode.
j	Optional for uploading a folder or multiple files/ folders (additional parameter)	Indicates the maximum number of concurrent tasks for uploading a folder. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.
encoding- type	Optional (additional parameter)	If the object name contains special characters, set this parameter to url . NOTE This parameter is only available for obsutil 5.5.12 and later.
msm	Mandatory for uploading multiple files/ folders (additional parameter)	 Enables the mode for uploading multiple files/folders. Possible values are 1 and 2. NOTE If msm is set to 1, the source URL indicates a list of file/folder names separated by commas. If msm is set to 2, the source URL indicates a file containing a list of file/folder names. If the file or folder name already contains commas (,), do not set msm to 1. If parameter r is not set, the folders in the list will not be uploaded.

Parameter	Optional or Mandatory	Description
exclude	de Optional for uploading a folder or multiple files/ folders (additional parameter)	 Indicates the file matching patterns that are excluded, for example: *.txt. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt. You can use * to represent * and \? to represent ?. If the name of the file to be uploaded matches the value of this parameter, the file is skipped.
		NOTICE
		 You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows.
		 The matching pattern applies to the absolute file path (including the file name and file directory).
		 The matching pattern takes effect only for files in the folder.
		 Multiple exclude parameters can be specified, for example, -exclude=*.xxx -exclude=*.xxx.

Parameter	Optional or Mandatory	Description
include	Optional for uploading a folder or multiple files/ folders (additional parameter)	 Indicates the file matching patterns that are included, for example: *.jpg. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. You can use * to represent * and \? to represent ?. Only after identifying that the name of the file to be uploaded does not match the value of exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is uploaded. If not, the file is skipped. NOTE Example of uploading files in a request with the include parameter contained: ./obsutil cp /localpath/ obs://test/ -include=/localpath/2022-12-09/* -f -r This command uploads files that are under localpath and start with /localpath/2022-12-09/ to bucket test. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute file path (including the file name and file directory). The matching pattern takes effect only for files in the folder. Multiple include parameters can be specified, for example, -include=*.xxxinclude=*.xxx.
at	Optional for uploading a folder or multiple files/ folders (additional parameter)	Indicates that only the files whose latest access time is within the value of timeRange are uploaded. NOTE • This parameter must be used together with timeRange .
disableDir Object	Optional for uploading multiple folders (additional parameter)	Indicates the folders themselves are not uploaded as an object. Configuring this parameter can avoid uploading empty folders to a bucket. If a folder contains files, the files will be uploaded and the original path format is retained.

Parameter	Optional or Mandatory	Description
timeRange	Optional for uploading a folder or multiple files/ folders (additional parameter)	Indicates the time range matching pattern when uploading files. Only files whose latest modification time is within the configured time range are uploaded.
		This pattern has a lower priority than the file matching patterns (exclude/include). That is, the time range matching pattern is executed after the configured file matching patterns.
		NOTE
		 The matching time range is represented in <i>time1-time2</i>, where <i>time1</i> must be earlier than or the same as <i>time2</i>. The time format is <i>yyyyMMddHHmmss</i>.
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		 If this parameter is set to *-<i>time2</i>, all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i>-*, all files whose latest modification time is later than <i>time1</i> are matched.
		NOTICE Time in the matching pattern is the UTC time.
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on folders.

Parameter	Optional or Mandatory	Description
0	Optional (additional parameter)	 Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output, the subfolder in the home directory of the user who executes obsutil commands. NOTE The naming rule for result lists is as follows: cp_{succeed failed warning}_report_time_TaskId.txt By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file. If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list cp_failed_report_time_TaskId.txt in the result list folder and the log files in the log path.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Response

Field	Description
Parallel	Parameter -p in the request

Field	Description
Jobs	Parameter -j in the request
Threshold	Parameter -threshold in the request
PartSize	Parameter -ps in the request
Exclude	Parameter - exclude in the request
Include	Parameter - include in the request
TimeRange	Parameter -timeRange in the request
VerifyLength	Parameter -vlength in the request
VerifyMd5	Parameter -vmd5 in the request
CheckpointDir	Parameter - cpd in the request
OutputDir	Parameter -o in the request
ArcDir	Parameter -arcDir in the request
Succeed count	Number of successful tasks
Failed count	Number of failed tasks
Skip count	Number of tasks that are skipped during incremental upload, download, or copy, and synchronous upload, download, or copy. NOTE Skipped tasks are recorded into successful tasks.
Warning count	 Number of tasks that are executed successfully but contain warnings. NOTE The task for which a warning is generated may be a failure or a success, which needs to be further determined according to the corresponding result list. The number of tasks that generate warnings is independent of the number of successful or failed tasks. The total number of tasks is the number of successful tasks plus the number of failed tasks.
Succeed bytes	Number of bytes that are successfully uploaded or downloaded.
max cost	Maximum duration of all tasks, in ms
min cost	Minimum duration of all tasks, in ms
average cost	Average duration of all tasks, in ms
average tps	The average number of tasks completed per second
Task id	Unique ID of an operation, which is used to search for the result list generated in a batch task

5.3 Querying Object Properties

Function

You can use this command to query the basic properties of an object.

Command Line Structure

- In Windows obsutil stat obs://bucket/key [-acl][-bf=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx] [-t=xxx]
- In Linux or macOS ./obsutil stat obs://bucket/key [-acl][-bf=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil stat obs://buckettest/key command to query the basic properties of an object. obsutil stat obs://bucket-test/key

Start at 2024-09-25 04:48:10.1147483 +0000 UTC

```
Key:
obs://bucket-test/key
LastModified:
2018-11-16T02:15:49Z
Size:
7
StorageClass:
standard
ETag:
43d93b553855b0e1fc67e31c28c07b65
ContentType:
text/plain
Type:
file
Metadata:
key=value
```

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name
key	Mandatory	Object name
acl	Optional	Queries the access control policies of the object at the same time.

Parameter	Optional or Mandatory	Description
bf	Optional (additional	Display format of the object size (in bytes). Possible values are:
	parameter)	human-readable
		• raw
		NOTE If this parameter is not configured, the display format of the object size (in bytes) is determined by the humanReadableFormat parameter in the configuration file.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Response

Field	Description
Кеу	Object name
LastModified	Latest modification time of the object
Size	Object size, in bytes
StorageClass	Storage class of the object
MD5	Real MD5 of the object NOTE You can query this value only after running the cp command and configuring the - vmd5 parameter.
ETag	ETag value of an object calculated on the server
ContentType	Content-Type of the object

Field	Description	
Туре	Object type	
Metadata	Customized metadata of the object	
	NOTE The value of this field can be queried only after the object has customized metadata.	

5.4 Setting Object Properties

Function

You can use this command to set properties of an object or set properties of objects in batches by a specified object name prefix.

D NOTE

You can set storage classes only for buckets whose version is 3.0.

Command Line Structure

- In Windows
 - Setting properties of a single object
 obsutil chattri obs://bucket/key [-meta=aaa:bbb#ccc:ddd] [-sc=xxx] [-acl=xxx] [-aclXml=xxx] [versionId=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
 - Setting properties of objects in batches
 obsutil chattri obs://bucket[/key] -r [-f] [-v] [-meta=aaa:bbb#ccc:ddd] [-sc=xxx] [-acl=xxx] [-acl=xxx]
- In Linux or macOS
 - Setting properties of a single object
 ./obsutil chattri obs://bucket/key [-meta=aaa:bbb#ccc:ddd] [-sc=xxx] [-acl=xxx] [-aclXml=xxx] [-versionId=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Setting properties of objects in batches
 ./obsutil chattri obs://bucket[/key] -r [-f] [-v] [-meta=aaa:bbb#ccc:ddd] [-sc=xxx] [-acl=xxx] [-acl=xxx] [-acl=xxx] [-acl=xxx] [-e=xxx] [-e=xxx] [-i=xxx] [-i=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example, run the obsutil chattri obs://buckettest/key -acl=public-read command to set the access permission to an object to public read.

obsutil chattri obs://bucket-test/key -acl=public-read Start at 2024-09-30 08:17:15.5572914 +0000 UTC

Set the acl of object [key] in the bucket [bucket-test] to [public-read] successfully, request id [04050000016836DDFA73B2B5320E2651]

 Take the Windows OS as an example, run the obsutil chattri obs://buckettest -r -f -acl=public-read command to set the access permission to all objects in the bucket to public read. obsutil chattri obs://bucket-test -r -f -acl=public-read Start at 2024-09-30 08:18:03.105373 +0000 UTC

[-----] 100.00% tps:155.15 5/5 233ms

Succeed count:5Failed count:0Metrics [max cost:177 ms, min cost:53 ms, average cost:102.40 ms, average tps:20.41]Task id:9d7f73ff-f747-4fdd-9b2a-815ba2dc3b07

Parameter	Optional or Mandatory	Description	
bucket	Mandatory	Bucket name	
key	Mandatory when setting properties of an object. Optional when setting properties of objects in batches.	Indicates the name of the object whose properties are to be set, or the name prefix of objects whose properties are to be set in batches. NOTE If this parameter is left blank during batch operation, properties of all objects in the bucket are set.	
meta	Optional (additional parameter)	 Standard or custom metadata that can be specified for destination objects in object replication. This parameter should be configured in the following format: <i>key1:value1# key2:value2# key3:value3.</i> NOTE This parameter takes effect only when it is used together with direct. This value indicates that the destination objects contain three groups of custom metadata: <i>key1:value1, key2:value2,</i> and <i>key3:value3.</i> Standard metadata headers include Content-Type, Content-Encoding, Cache-Control, Content-Disposition, Content-Language and Expires. 	
direct	Optional	Metadata operation indicator.	
	(additional parameter)	The value can be REPLACE_NEW or REPLACE .	
		REPLACE_NEW : The existing metadata value is replaced with a new one, the metadata lacking a value is assigned one, and the metadata not specified keeps unchanged.	
		REPLACE : The metadata is replaced with the header included in the current request and the metadata not specified is deleted.	
		NOTE This parameter takes effect only when it is used together with meta .	

Parameter	Optional or Mandatory	Description	
sc	Optional (additional parameter)	 Storage classes of objects. Possible values are: standard: Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB. warm: Infrequent Access storage class. It is ideal for storing infrequently accessed (less than 12 times a year) data, but when needed, the access has to be fast. 	
		 cold: Archive storage class. It provides secure, durable, and inexpensive storage for rarely-accessed (once a year) data. NOTE For an object whose storage class is cold, restore the object first and then specify its storage class. To restore an object, see Restoring Objects from the Archive Storage. 	
acl	Optional (additional parameter)	 Access control policies that can be specified for objects. Possible values are: private public-read public-read-write bucket-owner-full-control NOTE The preceding four values indicate private read and write, public read, public read and write, and bucket owner full control. 	

Parameter	Optional or Mandatory	Description
aclXml	Optional (additional parameter)	Access control policy of the bucket, in XML format. <accesscontrolpolicy> <owner> <id>ownerid</id> </owner> <accesscontrollist> <grant> <grante> <id>userid</id> </grante></grant> <grantee> <permission>[WRITE]WRITE_ACP/READ/READ_ACP/ FULL_CONTROL]</permission> <grantee> <canned>Everyone</canned> <grantee> <grantee> <canned>Everyone</canned> <grantee> <canned>Everyone</canned> <grantee> <permission>[WRITE]WRITE_ACP/READ/READ_ACP/ FULL_CONTROL]</permission> </grantee></grantee></grantee></grantee></grantee></grantee></accesscontrollist> NOTE • Owner: Optional. Specify the object owner's ID. • In AccessControlList, the Grant field contains the authorized users. Grantee specifies the IDs of authorized user group (currently, only Everyone is supported). • The following permissions can be granted: WRITE (write), WRITE_ACP (write ACL), READ (read), READ_ACP (read ACL), and FULL_CONTROL (full control). NOTICE Because angle brackets (<) and (>) are unavoidably included in the parameter value, you must use quotation marks to enclose them for escaping when running the command. Use single quotation marks for Linux or macOS and quotation marks for Windows.</accesscontrolpolicy>
fs	Optional (additional parameter)	 Specifies whether the method of listing parallel file systems is applied. If you are listing parallel file systems, you are recommended to add this parameter. CAUTION With this method, the listing time required varies largely depending on the directory structures. After this parameter is enabled, marker and limit will be ignored. Then, the size of the bucket, parallel file system, or directory will be calculated. This parameter is only supported by obsutil 5.5.12 and later.

Parameter	Optional or Mandatory	Description
versionId	Optional when setting properties of an object (additional parameter)	Version ID of the object whose properties are to be set
fr	Optional when setting properties of an object (additional parameter)	Generates an operation result list when setting properties of an object.
f	Optional when setting properties of objects in batches (additional parameter)	Runs in force mode.
r	Mandatory when setting properties of objects in batches (additional parameter)	Sets properties of objects in batches based on a specified object name prefix.
V	Optional when setting properties of objects in batches (additional parameter)	Sets properties of versions of objects in batches based on a specified object name prefix.

Parameter	Optional or Mandatory	Description
0	Optional (additional parameter)	Indicates the folder where operation result lists reside. After the command is executed, result lists (including success and failure files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands.
		 The naming rule for result lists is as follows: chattri_{succeed failed}_report_time_TaskId.txt By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file.
		 If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list chattri_failed_report_time_TaskId.txt in the result list folder and the log files in the log path.
j	Optional when setting properties of objects in batches (additional parameter)	Indicates the maximum number of concurrent tasks for setting object properties in batches. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

D NOTE

Only one from **acl**, **sc**, or **aclXml** can be set for each command.

Response

Refer to **Response** for uploading an object.

5.5 Listing Objects

Function

You can use this command to query objects or object versions in a bucket. All objects are listed in lexicographical order by object name and version ID.

Command Line Structure

In Windows

obsutil ls obs://bucket[/prefix] [-s] [-d][-fs] [-v] [-du] [-marker=xxx] [-versionIdMarker=xxx] [encoding-type=url][-bf=xxx] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [t=xxx]

In Linux or macOS
 ./obsutil ls obs://bucket[/prefix] [-s][-fs] [-d] [-v] [-du] [-marker=xxx] [-versionIdMarker=xxx] [encoding-type=url][-bf=xxx] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

Examples

 Example 1: Take the Windows OS as an example. Run the obsutil ls obs:// bucket-test -limit=10 command to list objects in the bucket. obsutil ls obs://bucket-test -limit=10 Start at 2024-09-30 08:18:50.0592788 +0000 UTC

Listing objects .

Folder list: obs://bucket-test/api/

Object list: key LastI obs://bucket-test/AUTHORS "796393c1eaf502ef56a85c2ceb640aea"	Modified Size St 2018-11-16T02:15:49Z	orageClass 33243	s ETag standard
obs://bucket-test/CONTRIBUTING.md standard "12d93325ba6131f852c	2018-11-16T02:15:492 Jaecd18dd65edc"	Z 130	56
obs://bucket-test/CONTRIBUTORS standard "b486b5003e6215c919	2018-11-16T02:15:49Z 9e86ab3ccec9fa"	4571	10
obs://bucket-test/LICENSE "5d4950ecb7b26d2c5e4e7b4e0dd74707'	2018-11-16T02:15:49Z '	1479	standard
obs://bucket-test/PATENTS "3a55d95595a6f9e37dee53826b4daff2"	2018-11-16T02:15:49Z	1303	standard
obs://bucket-test/README.md "97351fd7946b9ea021a31a86ba2a10ab'	2018-11-16T02:15:49Z	1399	standard
obs://bucket-test/VERSION "43d93b553855b0e1fc67e31c28c07b65"	2018-11-16T02:15:49Z	7 s	standard

obs://bucket-test/api/README "4e9e63a87075df60cdf65c8ce9e92117"	2018-11-16T02:15:49Z	521	standard
obs://bucket-test/api/except.txt "8eb96de3f60447e2f09a7531c99fb3ee"	2018-11-16T02:15:49Z	20194	standard
Next marker: api/except.txt Folder number: 1 File number: 9			
Example 2: Take the Windows bucket-test2 command to list obsutil ls obs://bucket-test2			util ls obs://
Start at 2024-09-30 08:21:06.6300221 +	0000 UTC		
Listing objects .			
Object list:			
key Lastl obs://bucket-test2/123 'd41d8cd98f00b204e9800998ecf8427e''	Modified Size 2022-03-29T09:17:51Z	StorageClass 0B st	s ETag andard
bbs://bucket-test2/1_2-3.txt 'd41d8cd98f00b204e9800998ecf8427e"	2022-03-29T09:17:51Z	OB s	tandard
obs://bucket-test2/1_2-3_33.txt "dcf204c11d791255adc63e61763c2426"	2022-03-29T09:17:51Z	200B	standard
	03-29T09:17:51Z 0B	standard	
"d41d8cd98f00b204e9800998ecf8427e"			
Total size of bucket: 200B Folder number: 0 File number: 4			
Example 3: Take the Windows bucket-test2/prefix command the bucket.			
obsutil ls obs://bucket-test2/prefix Start at 2024-09-30 08:24:36.7057148 +0	0000 UTC		
Listing objects .			
Object list: key Last1 obs://bucket-test2/prefix "d41d8cd98f00b204e9800998ecf8427e"	Modified Size 2022-03-29T09:17:51Z	StorageClass 0B st	s ETag andard
obs://bucket-test2/prefix2 "d41d8cd98f00b204e9800998ecf8427e"	2022-03-29T09:17:51Z	OB st	andard
Total size of prefix [prefix]:: 0B Folder number: 0 File number: 2			

• For more examples, see Listing.

Parameter Description

•

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name

Parameter	Optional or Mandatory	Description
prefix	Optional	Prefix of an object name for listing objects NOTE If this parameter is left blank, all objects in the bucket are listed.
S	Optional (additional parameter)	Displays simplified query result. NOTE In the simplified format, the returned result contains only the object name.
d	Optional (additional parameter)	Lists only objects and subdirectories in the current directory, instead of recursively listing all objects and subdirectories.
		In big data scenarios, parallel file systems usually have deep directory levels and each directory has a large number of files. In such case, you are advised to use this parameter to limit the scope to list. NOTE According to the naming conventions in OBS, a slash (/) is used as the directory separator.
v	Optional (additional parameter)	Lists versions of an object in a bucket. The result contains the latest version and historical versions (if any) of the object.
marker	Optional (additional parameter)	Object name to start with when listing objects in a bucket. All objects are listed in lexicographical order by object name. NOTE For details about how to use this parameter, see Listing.
versionldMa rker	Optional (additional parameter). It must be used together with the v and marker parameters.	Version ID to start with when listing versions of objects in a bucket. All versions and objects are listed in lexicographical order by object name and version ID. NOTE If the value of versionIdMarker is not a version ID specified by marker , versionIdMarker is invalid.
bf	Optional (additional parameter)	 Display formats of bytes in the listing result. Possible values are: human-readable raw NOTE If this parameter is not configured, the display format of bytes in the result is determined by the humanReadableFormat parameter in the configuration file.

Parameter	Optional or Mandatory	Description
du	Optional (additional parameter)	Quickly returns the total size of listed objects, without displaying detailed object information. This parameter can be used together with other parameters. NOTE If there are too many objects listed, wait for a while. CAUTION
		• This parameter is only supported by obsutil 5.4.6 and later.
fs	Optional (additional parameter)	Specifies whether the method of listing parallel file systems is applied. If you are listing parallel file systems, you are recommended to add this parameter. CAUTION
		 With this method, the listing time required varies largely depending on the directory structures.
		 After this parameter is enabled, marker and limit will be ignored. Then, the size of the bucket, parallel file system, or directory will be calculated.
		 This parameter is only supported by obsutil 5.5.12 and later.
limit	Optional (additional parameter)	Maximum number of objects that can be listed. If the value is less than or equal to 0, all objects are listed. If it is left blank, 1,000 objects are listed by default. NOTE If there are a large number of objects in a bucket, you are advised to set this parameter to limit the number of objects to be listed each time. If not all objects are listed, marker and versionIdMarker of the next request will be returned in the result, which you can use to list the remaining objects.
encoding- type	Optional (additional parameter)	If the object name contains special characters, set this parameter to url . NOTE This parameter is only available for obsutil 5.5.12 and later.
format	Optional (additional parameter)	Prints a listing result in the user-defined format. Currently, the value can only be default , indicating that the listing result is displayed in one row.

Parameter	Optional or Mandatory	Description
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
е	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Response

Field	Description	
Кеу	Object name	
LastModified	Time when the last modification was made to the object	
Size	Object size	
StorageClass	 Storage class of an object standard: Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB. warm: Infrequent Access storage class. It is ideal for storing infrequently accessed (less than 12 times a year) data, but when needed, the access has to be fast. cold: Archive storage class. It provides storage for rarely-accessed (once a year) data. 	

Field	Description
ETag	ETag of an object, which is a Base64-encoded 128-bit MD5 digest. ETag is the unique identifier of the object content. It can be used to determine whether the object content is changed. For example, if the ETag value is A when an object is uploaded, but this value has changed to B when the object is downloaded, it indicates that the object content has been changed. The ETag reflects changes to the object content, rather than the object metadata. An uploaded object or copied object has a unique ETag.

5.6 Copying an Object

Function

You can use this command to copy a single object or copy objects in batches by a specified object name prefix.

NOTICE

- Do not change the source objects in the OBS bucket when copying a single object or objects in batches. Otherwise, the operation may fail or data may be inconsistent.
- If the storage class of the object to be copied is **cold**, you must restore the object to be copied first. Otherwise, the copy fails.
- To copy objects, you must have the read permission on the objects to be copied and the write permission on the destination bucket.
- If the client-side cross-region replication function is not enabled, ensure that the source bucket and destination bucket are in the same region.
- If the source bucket is a parallel file system (supporting POSIX), the destination bucket must also be a parallel file system.

Command Line Structure

- In Windows
 - Copying a single object
 obsutil cp obs://srcbucket/key obs://dstbucket/[dest] [-dryRun][-u] [-crr] [-vlength] [-vmd5] [p=1] [-threshold=52428800] [-versionId=xxx] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [ps=auto] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
 - Copying objects in batches
 obsutil cp obs://srcbucket[/key] obs://dstbucket[/dest] -r [-dryRun][-f] [-flat] [-u] [-crr] [vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2]
 [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
- In Linux or macOS
 - Copying a single object

./obsutil cp obs://srcbucket/key obs://dstbucket/[dest] [-dryRun] [-u] [-crr] [-vlength] [-vmd5] [p=1] [-threshold=52428800] [-versionId=xxx] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [ps=auto] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Copying objects in batches

 /obsutil cp obs://srcbucket[/key] obs://dstbucket[/dest] -r [-dryRun] [-f] [-flat] [-u] [-crr] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

D NOTE

- The source path and destination path cannot be the same.
- The source path and destination path cannot be overlapped either. If the source path overlaps with the prefix of the destination path, recursive replication applies. If the destination path overlaps with the prefix of the source path, the replication may overwrite objects in the source path.

Examples

Take the Windows OS as an example. Run the obsutil cp obs://bucket-test/key obs://bucket-test2 command to copy a single object.
 obsutil cp obs://bucket-test/key obs://bucket-test2
 Start at 2024-09-30 08:30:09.0815415 +0000 UTC

Parallel: 3 Jobs: 3 Threshold: 50.00MB PartSize: auto CheckpointDir: xxxx

[=======] 100.00% 6/s 0s Waiting for the copied key to be completed on server side. Copy successfully, 19B, obs://bucket-test/key --> obs://bucket-test2/key ext.txt, cost [1708], status [200], request id [00000192420D227E4017336A12F1DC22]

 Take the Windows OS as an example. Run the obsutil cp obs://bucket-test/ temp/ obs://bucket-test2 -f -r command to copy objects in batches. obsutil cp obs://bucket-test/temp/ obs://bucket-test2 -r -f Start at 2024-09-30 08:34:02.7819703 +0000 UTC

Parallel: 5 Jobs: 5 Threshold: 50.00MB PartSize: auto CheckpointDir: xxxx

Task id: 0476929d-9d23-4dc5-b2f8-0a0493f027c5 OutputDir: xxxx

[======] 100.00% 10/s 0s Succeed count: 5 Failed count: 0

Metrics [max cost:298 ms, min cost:192 ms, average cost:238.00 ms, average tps:9.71, transferred size: 7.20MB] Task id: 0476929d-9d23-4dc5-b2f8-0a0493f027c5

- For more examples, see **Copy**.

Parameter	Optional or Mandatory	Description
srcbucket	Mandatory	Source bucket name
dstbucket	Mandatory	Destination bucket name

Parameter	Optional or Mandatory	Description
dest	Optional	Indicates the destination object name when copying an object, or the name prefix of destination objects when copying objects in batches.
key	Mandatory for copying an object. Optional for copying objects in batches.	 Indicates the source object name when copying an object, or the name prefix of source objects when copying objects in batches. The rules are as follows: This parameter cannot be left blank when copying an object. If dest is left blank, the source object is copied to the root directory of the destination bucket. If the value of dest ends with a slash (/), the destination object name is the value of dest plus the source object name. Otherwise, the destination object name is the value of dest. If this parameter is left blank when copying objects in batches, all objects in the source bucket are copied. If not, objects whose name prefix is the set value in the source bucket are copied. The rules for confirming the name of the destination object name. If the value of dest does not end with a slash (/), the destination object name. If the value of dest does not end with a slash (/), the destination object name. If the value of dest does not end with a slash (/), the destination object name. If the source object oname. If this parameter is configured but the flat parameter is not when copying objects in batches, the name of the source object otorains the name prefix of the parent object. If flat is configured, then the name of the source object. For details about how to use this parameter, see Copy.
fr	Optional for copying an object (additional parameter)	Generates an operation result list when copying an object.

Parameter	Optional or Mandatory	Description
flat	Optional for copying objects in batches (additional parameter)	The name prefix of the parent object is excluded when copying objects in batches.
dryRun	Optional (additional parameter)	Conducts a dry run.
crr	Optional (additional parameter)	Enables the client-side cross-region replication function. In this mode, data is directly copied to the destination bucket from the source bucket through data stream. The buckets can by any two OBS buckets. NOTE • If this parameter is configured, the configuration of
		client-side cross-region replication must be updated in the configuration file. For details, see Updating a Configuration File .
		 The configurations of the source bucket and destination bucket are respectively akCrr/skCrr/ tokenCrr/endpointCrr and ak/sk/token/endpoint in the configuration file.
		NOTICE When cross-region replication is enabled, the upload/ download bandwidth, CPU, and memory resources of the host where commands are executed will be occupied, which may deteriorate the host performance.
vlength	Optional (additional parameter)	Verifies whether the object size in the destination bucket is the same as that in the source bucket after the copy task completes.
		This parameter must be used together with crr .
vmd5	Optional (additional parameter)	Verifies whether the MD5 value of the destination bucket is the same as that of the source bucket after the copy task completes. NOTE
		 This parameter must be used together with crr. Objects in the source bucket must contain metadata x-obs-meta-md5chksum, or MD5 verification will be skipped. After the MD5 verification is successful, this parameter value is used for metadata x-obs-meta-md5chksum of the destination object, for later MD5 verification during download or copy.
		CAUTION If your object needs encryption, do not use this parameter.

Parameter	Optional or Mandatory	Description
u	Optional (additional parameter)	Indicates incremental copy. If this parameter is set, each object can be copied only when it does not exist in the destination bucket, its size is different from the namesake one in the destination bucket, or it has the latest modification time.
р	Optional (additional parameter)	Indicates the maximum number of concurrent multipart copy tasks when copying an object. The default value is the value of defaultParallels in the configuration file.
threshold	Optional (additional parameter)	 Indicates the threshold for enabling multipart copy, in bytes. The default value is the value of defaultBigfileThreshold in the configuration file. NOTE If the size of the object to be copied is smaller than the threshold, copy the object directly. If not, a multipart copy is required. If you copy an object directly, no part record is generated, and resumable transmission is not supported. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).
versionId	Optional for copying an object (additional parameter)	Source object version ID that can be specified when copying an object
acl	Optional (additional parameter)	 Access control policies for destination objects that can be specified when copying objects. Possible values are: private public-read public-read-write bucket-owner-full-control NOTE The preceding four values indicate private read and write, public read, public read and write, and bucket owner full control.

Parameter	Optional or Mandatory	Description
SC	Optional (additional parameter)	 Storage classes of the destination objects that can be specified when copying objects. Possible values are: standard: Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB. warm: Infrequent Access storage class. It is ideal for storing infrequently accessed (less than 12 times a year) data, but when needed, the access has to be fast. cold: Archive storage class. It provides secure, durable, and inexpensive storage for rarely-
meta	Optional (additional parameter)	 accessed (once a year) data. Standard or custom metadata that can be specified for destination objects in object replication. This parameter should be configured in the following format: <i>key1:value1#key2:value2#key3:value3</i>. NOTE The format example above indicates that the destination objects contain three groups of custom metadata: <i>key1:value1, key2:value2</i>, and <i>key3:value3</i>. Standard metadata headers include Content-Type, Content-Encoding, Cache-Control, Content-Disposition, Content-Language and Expires.
fs	Optional (additional parameter)	 Specifies whether the method of listing parallel file systems is applied. If you are listing parallel file systems, you are recommended to add this parameter. CAUTION With this method, the listing time required varies largely depending on the directory structures. After this parameter is enabled, marker and limit will be ignored. Then, the size of the bucket, parallel file system, or directory will be calculated. This parameter is only supported by obsutil 5.5.12 and later.

Parameter	Optional or Mandatory	Description
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart copy task, in bytes. The value ranges from 100KB to 5GB. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source object size.
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the subfolder in the home directory of the user who executes obsutil commands. NOTE A part record is generated during a multipart copy and saved to the copy subfolder. After the copy succeeds, its part record is deleted automatically. If the copy fails or is suspended, the system attempts to resume the task according to its part record when you perform the copy the next time.
r	Mandatory for copying objects in batches (additional parameter)	Copies objects in batches based on a specified name prefix of objects in the source bucket.
f	Optional for copying objects in batches (additional parameter)	Runs in force mode.
j	Optional for copying objects in batches (additional parameter)	Indicates the maximum number of concurrent tasks for copying objects in batches. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.

Parameter	Optional or Mandatory	Description
exclude	Optional for copying objects in batches (additional parameter)	 Indicates the matching patterns of source objects that are excluded, for example: *.txt. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt. You can use * to represent * and \? to represent ?. If the name of the object to be copied matches the value of this parameter, the object is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object in the bucket is obs://bucket/src1/src2/test.txt, then the absolute path of the object is src1/src2/test.txt. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple exclude parameters can be specified, for example, -exclude=*.xxx -exclude=*.xxx.

Parameter	Optional or Mandatory	Description
include	Optional for copying objects in batches (additional parameter)	 Indicates the matching patterns of source objects that are included, for example: *.jpg. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. You can use * to represent * and \? to represent ?. Only after identifying that the name of the file to be copied does not match the value of exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is copied. If not, the file is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs://bucket/src1/src2/test.txt, then the absolute path of the object is src1/src2/test.txt. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple include parameters can be specified, for example, -include=*.xxx -include=*.xxx.

Parameter	Optional or Mandatory	Description
timeRange	Optional for copying objects in batches	Indicates the time range matching pattern when copying objects. Only objects whose latest modification time is within the configured time range are copied.
	(additional parameter)	This pattern has a lower priority than the object matching patterns (exclude/include). That is, the time range matching pattern is executed after the configured object matching patterns.
		• The matching time range is represented in <i>time1-time2</i> , where <i>time1</i> must be earlier than or the same as <i>time2</i> . The time format is <i>yyyyMMddHHmmss</i> .
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		• If this parameter is set to *- <i>time2</i> , all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i> -*, all files whose latest modification time is later than <i>time1</i> are matched.
		NOTICE
		 Time in the matching pattern is the UTC time. This matching pattern applies only to objects whose names do not end with a slash (/).
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on objects whose names end with a slash (/).
0	Optional (additional parameter)	Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands. NOTE
		 The naming rule for result lists is as follows: cp_{succeed failed warning}_report_time_TaskId.txt
		 By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file.
		 If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list cp_failed_report_time_TaskId.txt in the result list folder and the log files in the log path.

Parameter	Optional or Mandatory	Description
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Refer to **Response** for uploading an object.

5.7 Moving an Object

Function

You can use this command to move a single object or move objects in batches by a specified object name prefix.

NOTICE

- Do not change the source objects in the OBS bucket when moving objects. Otherwise, the operation may fail or data may be inconsistent.
- If the storage class of the object to be moved is **cold**, you must restore the object first. Otherwise, the moving fails.
- The source objects are deleted after the move operation succeeds.
- If the source bucket is a parallel file system (supporting POSIX), the destination bucket cannot be an object storage bucket.

Command Line Structure

• In Windows

Moving a single object

obsutil mv obs://srcbucket/key obs://dstbucket/[dest] [-dryRun] [-u] [-p=1] [threshold=52428800] [-versionId=xxx] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

- Moving objects in batches
 obsutil mv obs://srcbucket[/key] obs://dstbucket[/dest] -r [-dryRun] [-f] [-flat] [-u] [-j=1] [p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [config=xxx] [-e=xxx] [-t=xxx] [-t=xxx]
- In Linux or macOS
 - Moving a single object

 /obsutil mv obs://srcbucket/key obs://dstbucket/[dest] [-dryRun] [-u] [-p=1] [threshold=52428800] [-versionId=xxx] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto]
 [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Moving objects in batches

 /obsutil mv obs://srcbucket[/key] obs://dstbucket[/dest] -r [-dryRun] [-f] [-flat] [-u] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-i=xxx] [-t=xxx] [-t=xxx]

NOTE

- The source path and destination path cannot be the same.
- The source and destination paths cannot be nested when moving objects in batches.
- Batch object move is not available for parallel file systems.

Examples

Take the Windows OS as an example. Run the obsutil mv obs://bucket-test/key obs://bucket-test2 command to move a single object.
 obsutil mv obs://bucket-test/key obs://bucket-test2
 Start at 2024-09-30 08:36:01.3934921 +0000 UTC

Parallel:5Jobs:5Threshold:50.00MBPartSize:autoCheckpointDir:xxxx

[======] 100.00% 6/s 0s

Waiting for the copied key to be completed on server side.

Move successfully, 19B, obs://bucket-test/key --> obs://bucket-test2/key, cost [1815], status [200], request id [00000192421282AC401423183A8B83A1]

 Take the Windows OS as an example. Run the obsutil mv obs://bucket-test/ temp/ obs://bucket-test2 -f -r command to move objects in batches. obsutil mv obs://bucket-test/temp/ obs://bucket-test2 -f -r Start at 2024-09-30 08:37:32.2454905 +0000 UTC

Parallel: 5 Jobs: 5 Threshold: 50.00MB PartSize: auto CheckpointDir: xxxx Task id: 0476929d-9d23-4dc5-b2f8-0a0493f027c5 OutputDir: xxxx

[======] 100.00% 10/s 0s Succeed count: 5 Failed count: 0 Metrics [max cost:298 ms, min cost:192 ms, average cost:238.00 ms, average tps:9.71, transferred size: 7.20MB] Task id: 0476929d-9d23-4dc5-b2f8-0a0493f027c5

Parameter	Optional or Mandatory	Description
srcbucket	Mandatory	Source bucket name
dstbucket	Mandatory	Destination bucket name
dest	Optional	Indicates the destination object name when moving a single object, or the name prefix of destination objects when moving objects in batches.
key	Mandatory for moving a single object Optional for moving objects in batches	 Indicates the source object name when moving a single object, or the name prefix of source objects when moving objects in batches. The rules are as follows: This parameter cannot be left blank when moving a single object. If dest is left blank, the source object is moved to the root directory of the destination bucket. If the value of dest ends with a slash (/), the destination object name is the value of dest plus the source object name. Otherwise, the destination object name is the value of dest. If this parameter is left blank when moving objects in batches, all objects in the source bucket are moved. If not, objects whose name prefix is the set value in the source bucket are moved. If not, objects whose name prefix is the set value in the source bucket are moved. If not, object name. If the value of dest ends with a slash (/), the destination object name. If the value of dest does not end with a slash (/), the destination object name. If the value of dest does not end with a slash (/), the destination object name. If this parameter is configured but parameter flat is not when moving objects in batches, the name of the source object contains the name prefix of the parent object. If flat is configured, then the name of the source object does not contain the name prefix of the parent object.
		 For details about how to use this parameter, see Command Line Structure.
fr	Optional for moving an object (additional parameter)	Generates an operation result list when moving an object.

Parameter	Optional or Mandatory	Description
flat	Optional for moving objects in batches (additional parameter)	The name prefix of the parent object is excluded when moving objects in batches.
dryRun	Optional (additional parameter)	Conducts a dry run.
u	Optional (additional parameter)	Indicates incremental move. If this parameter is set, each object can be moved only when it does not exist in the destination bucket, its size is different from the namesake one in the destination bucket, or it has the latest modification time. NOTE If the size and modification time of the destination object are the same as those of the source object, the source object is directly deleted instead of being moved.
р	Optional (additional parameter)	Indicates the maximum number of concurrent multipart move tasks when moving an object. The default value is the value of defaultParallels in the configuration file.
threshold	Optional (additional parameter)	 Indicates the threshold for enabling multipart move, in bytes. The default value is the value of defaultBigfileThreshold in the configuration file. NOTE If the size of the object to be moved is smaller than the threshold, move the object directly. If not, a multipart move is required. If you move an object directly, no part record is generated, and resumable transmission is not supported. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).
versionId	Optional for moving an object (additional parameter)	Source object version ID that can be specified when moving a single object NOTE Parameter versionId cannot be configured for buckets that support POSIX.

Parameter	Optional or Mandatory	Description
acl	Optional (additional parameter)	 Access control policies for destination objects that can be specified when moving objects. Possible values are: private public-read public-read-write bucket-owner-full-control NOTE The preceding four values indicate private read and write, public read, public read and write, and bucket owner full control.
SC	Optional (additional parameter)	 Storage classes of the destination objects that can be specified when moving objects. Possible values are: standard: Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB. warm: Infrequent Access storage class. It is ideal for storing infrequently accessed (less than 12 times a year) data, but when needed, the access has to be fast. cold: Archive storage class. It provides secure, durable, and inexpensive storage for rarely-accessed (once a year) data.
meta	Optional (additional parameter)	Metadata of destination objects that can be specified when moving objects. The format is <i>key1:value1#key2:value2#key3:value3</i> . NOTE The preceding value indicates that the destination objects in the bucket contain three groups of customized metadata after objects are moved: key1:value1 , key2:value2 , and key3:value3 .
fs	Optional (additional parameter)	 Specifies whether the method of listing parallel file systems is applied. If you are listing parallel file systems, you are recommended to add this parameter. CAUTION With this method, the listing time required varies largely depending on the directory structures. After this parameter is enabled, marker and limit will be ignored. Then, the size of the bucket, parallel file system, or directory will be calculated. This parameter is only supported by obsutil 5.5.12 and later.

Parameter	Optional or Mandatory	Description
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart move task, in bytes. The value ranges from 100KB to 5GB. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source object size.
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the subfolder in the home directory of the user who executes obsutil commands. NOTE A part record is generated during a multipart move and saved to the copy subfolder. After the move succeeds, its part record is deleted automatically. If the move fails or is suspended, the system attempts to resume the task according to its part record when you perform the move the next time.
r	Mandatory for moving objects in batches (additional parameter)	Moves objects in batches based on a specified name prefix of objects in the source bucket.
f	Optional for moving objects in batches (additional parameter)	Runs in force mode.
j	Optional for moving objects in batches (additional parameter)	Indicates the maximum number of concurrent tasks for moving objects in batches. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.

Parameter	Optional or Mandatory	Description
exclude	Optional for moving objects in batches (additional parameter)	 Indicates the matching patterns of source objects that are excluded, for example: *.txt. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt. You can use * to represent * and \? to represent ?. If the name of the object to be moved matches the value of this parameter, the object is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object in the bucket is obs://bucket/src1/src2/test.txt, then the absolute path of the object is src1/src2/test.txt. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple exclude parameters can be specified, for example, -exclude=*.xxx -exclude=*.xxx.

Parameter	Optional or Mandatory	Description
include	Optional for moving objects in batches (additional parameter)	 Indicates the matching patterns of source objects that are included, for example: *.jpg. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. You can use * to represent * and \? to represent ?. Only after identifying that the name of the file to be moved does not match the value of exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is moved. If not, the file is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs://bucket/src1/src2/test.txt, then the absolute path of the object is src1/src2/test.txt. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple include parameters can be specified, for example, -include=*.xxx -include=*.xxx.

Parameter	Optional or Mandatory	Description
timeRange	Optional for moving objects in batches	Indicates the time range matching pattern when moving objects. Only objects whose latest modification time is within the configured time range are moved.
	(additional parameter)	This pattern has a lower priority than the object matching patterns (exclude / include). That is, the time range matching pattern is executed after the configured object matching patterns.
		• The matching time range is represented in <i>time1-time2</i> , where <i>time1</i> must be earlier than or the same as <i>time2</i> . The time format is <i>yyyyMMddHHmmss</i> .
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		• If this parameter is set to *- <i>time2</i> , all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i> -*, all files whose latest modification time is later than <i>time1</i> are matched.
		• Time in the matching pattern is the UTC time.
	 This matching pattern applies only to objects whose names do not end with a slash (/). 	
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on objects whose names end with a slash (/).
0	Optional (additional parameter)	Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands. NOTE
		 The naming rule for result lists is as follows: mv_{succeed failed warning}_report_time_TaskId.txt
		 By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file.
		 If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list mv_failed_report_time_TaskId.txt in the result list folder and the log files in the log path.

Parameter	Optional or Mandatory	Description
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Refer to **Response** for uploading an object.

5.8 Downloading an Object

Function

You can use this command to download an object or download objects in batches by object name prefix to your local PC.

NOTICE

- Do not change the source objects in the OBS bucket when downloading a single object or objects in batches. Otherwise, the download may fail or data may be inconsistent.
- If the object to be downloaded is in the Infrequent Access storage class, you must restore it before downloading it. Otherwise, the download will fail.

Command Line Structure

- In Windows
 - Downloading a single object
 obsutil cp obs://bucket/key file_or_folder_url [-tempFileDir=xxx] [-dryRun] [-u] [-vlength] [vmd5] [-p=1] [-threshold=52428800] [-versionId=xxx] [-encoding-type=url] [-ps=auto] [cpd=xxx][-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

- Downloading objects in batches obsutil cp obs://bucket[/key] folder_url -r [-tempFileDir=xxx] [-dryRun] [-f] [-flat] [-u] [vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-encoding-type=url] [-ps=auto] [include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
- In Linux or macOS
 - Downloading a single object ./obsutil cp obs://bucket/key file or folder url [-tempFileDir=xxx] [-dryRun] [-u] [-vlength] [vmd5] [-p=1] [-threshold=52428800] [-versionId=xxx] [-encoding-type=url] [-ps=auto] [cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Downloading objects in batches ./obsutil cp obs://bucket[/key] folder_url -r [-tempFileDir=xxx] [-dryRun] [-f] [-flat] [-u] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-ps=auto] [-include=*.xxx] [exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [i=xxx] [-k=xxx] [-t=xxx]

Examples

Take the Windows OS as an example. Run the **obsutil cp obs://bucket**test/key d:\temp\test.txt command to download a single object. obsutil cp obs://bucket-test/key d:\temp\test.txt Start at 2024-09-30 08:39:34.180766 +0000 UTC

Parallel: 5	Jobs: 5	
Threshold: 50.00MB	PartSize:	auto
VerifyLength: false	VerifyMd5:	false
CheckpointDir: xxxx	-	
TempFileDir: xxxx		

======] 100.00% 4.86 KB/s 8.46MB/8.46MB 0s Download successfully, 8.46MB, obs://bucket-test/key --> d:\temp\test.txtt, cost [41], status [200], request id [000001924215BEC84019EDF4044A5451]

Take the Windows OS as an example. Run the **obsutil cp obs://bucket-test/ temp d:** -f -r command to download objects in batches. obsutil cp obs://bucket-test/temp d:\ -f -r

Start at 2024-09-30 08:41:56.0306522 +0000 UTC

Parallel: 5	Jobs: 5	
Threshold: 50.00MB	PartSize:	auto
VerifyLength: false	VerifyMd5:	false
CheckpointDir: xxxx		

Task id: 3066a4b0-4d21-4929-bb84-4829c32cbd0f OutputDir: xxxx TempFileDir: xxxx

[=======] 100.00% tps:17.86 155.59 KB/s 7.20MB/7.20MB 0s Succeed count: 6 Failed count: 0 70B Succeed bytes: Metrics [max cost:153 ms, min cost:129 ms, average cost:92.00 ms, average tps:70, transferred size: 7.20MB]

Task id: 3066a4b0-4d21-4929-bb84-4829c32cbd0f

For more examples, see **Download**.

Parameter	Optional or Mandatory	Description
file_or_fold er_url	Mandatory for downloading an object	Local file/folder path
folder_url	Mandatory for downloading objects in batches	Local folder path
bucket	Mandatory	Bucket name

Parameter	Optional or Mandatory	Description
key	Mandatory for downloading an object	Indicates the name of the object to be downloaded, or the name prefix of the objects to be downloaded in batches.
	Optional for downloading objects in a batch	This parameter cannot be left blank when downloading an object. The saving and naming rules are as follows:
		 If this parameter specifies a file or folder path that does not exist, the tool checks whether the value ends with a slash (/) or backslash (\). If yes, a folder is created based on the path, and the object is downloaded to this newly created directory.
		 If this parameter specifies a file or folder path that does not exist and the value does not end with a slash (/) or backslash (\), the object is downloaded to your local PC with the value of key as the file name.
		 If this parameter specifies an existing file, the object is downloaded to your local PC overwriting the existing file, with the value of key as the file name.
		 If this parameter specifies an existing folder, the object is downloaded to the directory specified by file_or_folder_url with the object name as the file name.
		During batch download, the saving rules are as follows:
		 If this parameter is left blank, all objects in the bucket are downloaded to the directory specified by folder_url.
		 If this parameter is configured, objects whose name prefix is the configured value in the bucket are downloaded to the directory specified by folder_url.
		 NOTE If this parameter is configured but the flat
		parameter is not configured when downloading objects in a batch, the name of the downloaded file contains the name prefix of the parent object. If flat is configured, then the name of the downloaded file does not contain the name prefix of the parent object.
		 For details about how to use this parameter, see Download.

Parameter	Optional or Mandatory	Description
fr	Optional for downloading an object (additional parameter)	Generates an operation result list when downloading an object.
flat	Optional for downloading objects in batches (additional parameter)	The name prefix of the parent object is excluded when downloading objects in batches.
tempFileDir	Optional (additional parameter)	 Indicates the directory for storing temporary files during multipart download. The default value is the value of defaultTempFileDir in the configuration file. NOTE If this parameter is left blank and the defaultTempFileDir parameter in the configuration file is also left blank, temporary files generated during multipart download are saved in the directory where to-be-downloaded files are located and end with the suffix of .obs.temp. Temporary files generated during multipart download are stored in this directory. Therefore, ensure that the user who executes obsutil has the write permission on the path. The available space of the partition where the path is located must be greater than the size of the objects to be downloaded.
dryRun	Optional (additional parameter)	Conducts a dry run.
u	Optional (additional parameter)Indicates incremental download. If parameter is set, each object can be downloaded only when it does not the local path, its size is different fr namesake one in the local path, or the latest modification time.	
vlength	Optional (additional parameter)	Checks whether the sizes of the local files are the same as those of the objects in the bucket after the download is complete.

Parameter	Optional or Mandatory	Description
vmd5	Optional (additional parameter)	Checks whether MD5 values of the local files are the same as those of the objects in the bucket after the download is complete. NOTE Objects in the bucket must contain metadata x - obs-meta-md5chksum , or MD5 verification will be skipped. CAUTION If your object needs encryption, do not use this parameter.
p	Optional (additional parameter)	Indicates the maximum number of concurrent multipart download tasks when downloading an object. The default value is the value of defaultParallels in the configuration file.
threshold	Optional (additional parameter)	 Indicates the threshold for enabling multipart download, in bytes. The default value is the value of defaultBigfileThres- hold in the configuration file. NOTE If the size of the object to be downloaded is smaller than the threshold, download the object directly. If not, a multipart download is required. If you download an object directly, no part record is generated, and resumable transmission is not supported. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).
versionId	Optional for downloading an object (additional parameter)	Source object version ID that can be specified when downloading an object
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart download task, in bytes. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source object size.

Parameter	Optional or Mandatory	Description
fs	Optional (additional parameter)	 Specifies whether the method of listing parallel file systems is applied. If you are listing parallel file systems, you are recommended to add this parameter. CAUTION With this method, the listing time required varies largely depending on the directory structures. After this parameter is enabled, marker and limit will be ignored. Then, the size of the bucket, parallel file system, or directory will be calculated. This parameter is only supported by obsutil 5.5.12 and later.
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the subfolder in the home directory of the user who executes obsutil commands. NOTE A part record is generated during a multipart download and saved to the down subfolder. After the download succeeds, its part record is deleted automatically. If the download fails or is suspended, the system attempts to resume the task according to its part record when you perform the download the next time.
r	Mandatory for downloading objects in batches (additional parameter)	Copies objects in batches based on a specified object name prefix.
f	Optional for downloading objects in batches (additional parameter)	Runs in force mode.
j	Optional for downloading objects in batches (additional parameter)	Indicates the maximum number of concurrent tasks for downloading objects in a batch. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.

Parameter	Optional or Mandatory	Description
encoding- type	Optional (additional parameter)	If the object name contains special characters, set this parameter to url . NOTE This parameter is only available for obsutil 5.5.12 and later.
exclude	Optional for downloading objects in batches (additional parameter)	 Indicates the matching patterns of source objects that are excluded, for example, *.txt. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt. You can use * to represent * and \? to represent ?. If the name of the object to be downloaded matches the value of this parameter, the object is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object is strip. for example, if the path of an object is strip. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple exclude parameters can be specified, for example, -exclude=*.xxx -exclude=*.xxx.

Parameter	Optional or Mandatory	Description
include	Optional for downloading objects in batches (additional	Specifies what formats of objects can be downloaded. If this parameter is set to *.jpg , only objects in the .jpg format can be downloaded.
	parameter)	NOTE
		 The asterisk (*) represents any group of characters, and the question mark (?) represents any single character.
		 You can use * to represent * and \? to represent ?.
		 Only after identifying that the name of the file to be downloaded does not match the value of exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is downloaded. If not, the file is skipped.
		NOTICE
		 You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows.
		 The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs://bucket/src1/src2/ test.txt, then the absolute path of the object is src1/src2/test.txt.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
		 Multiple include parameters can be specified, for example, -include=*.xxx -include=*.xxx.

Parameter	Optional or Mandatory	Description
timeRange	Optional for downloading objects in batches (additional parameter)	Indicates the time range matching pattern when downloading objects. Only objects whose latest modification time is within the configured time range are downloaded.
		This pattern has a lower priority than the object matching patterns (exclude/include). That is, the time range matching pattern is executed after the configured object matching patterns.
		NOTE
		• The matching time range is represented in <i>time1-time2</i> , where <i>time1</i> must be earlier than or the same as <i>time2</i> . The time format is <i>yyyyMMddHHmmss</i> .
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		 If this parameter is set to *-<i>time2</i>, all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1-*</i>, all files whose latest modification time is later than <i>time1</i> are matched.
		NOTICE
		• Time in the matching pattern is the UTC time.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on objects whose names end with a slash (/).

Parameter	Optional or Mandatory	Description
0	Optional (additional parameter)	Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands. NOTE
		 The naming rule for result lists is as follows: cp_{succeed failed warning}_report_time_TaskId.txt
		• By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file.
		 If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list cp_failed_report_time_TaskId.txt in the result list folder and the log files in the log path.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.
encoding- type	Optional (additional parameter)	If the object name contains special characters, set this parameter to url . NOTE This parameter is only supported by obsutil 5.5.12 and later.

Refer to **Response** for uploading an object.

5.9 Generating the Download Link of an Object

Function

You can use this command to generate the download link of a specified object in a bucket or generate the download links of objects in a bucket in batches by object name prefix.

Command Line Structure

- In Windows
 - Generating the download link of a single object obsutil sign obs://bucket/key [-e=300] [-config=xxx] [-endpoint=xxx] [-i=xxx] [-t=xxx] [-t=xxx]
 - Generating the download links of objects in batches by object name prefix obsutil sign obs://bucket[/key] -r [-e=300] [-timeRange=time1-time2] [-include=*.xxx] [-
- exclude=*.xxx] [-o=xxx] [-config=xxx] [-endpoint=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 In Linux or macOS
 - Generating the download link of a single object ./obsutil sign obs://bucket/key [-e=300] [-config=xxx] [-endpoint=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Generating the download links of objects in batches by object name prefix
 (absutil sign abs://buckat[/kgul.r[a=200] [timePaperetime1 time2] [include=* xxy] [

./obsutil sign obs://bucket[/key] -r [-e=300] [-timeRange=time1-time2] [-include=*.xxx] [-exclude=*.xxx] [-o=xxx] [-config=xxx] [-endpoint=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Examples

 In Windows, run obsutil sign obs://bucket-test/test.txt to generate a single object download link: obsutil sign obs://bucket-test/test.txt

Download url of [obs://bucket-test/test.txt] is: http://your-endpoint/bucket-test/test.txt?AccessKeyId=xxxx&Expires=1552548758&Signature=xxxx

 In Windows, run obsutil sign obs://bucket-test/test/ -r to generate object download links in batches: obsutil sign obs://bucket-test/test/ -r

Generate download urls for objects .

Generate the download url(s) for the objects in the bucket [**bucket-test**] finished Task id: af4dc692-6a88-4541-8156-ff1a889d2288

NOTE

- If there are a large number of objects, obsutil saves the object download links to a specific file. The file name is associated with the task ID. For instance, the task ID in the example above is af4dc692-6a88-4541-8156-ff1a889d2288, so the file name should be sign_succeed_report_{timestamp}_af4dc692-6a88-4541-8156-ff1a889d2288.txt.
- By default, the file is stored under folder **.obsutil_output** in your user directory. You can also use parameter **-o** to specify a new folder.

Parame ter	Optional or Mandatory	Description	
bucket	Mandatory	Bucket name	
key	Optional	Object name used for generating the download link of a single object, or object name prefix used for generating download links of objects in batches	
е	Optional (additional parameter)	Validity period of the generated download links of objects, in seconds. Minimum value: 60s. Default value: 300s	
r	Mandatory when generating download links of objects in batches (additional parameter)	Generates the download links of objects in batches by a specified object name prefix.	
exclude	Optional when generating download links of objects in batches (additional parameter)	 Indicates the matching patterns of objects that are excluded, for example: *.txt. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt. You can use * to represent * and \? to represent ?. If the name of the object to be downloaded matches the value of this parameter, the object is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs://bucket/src1/src2/test.txt. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple exclude parameters can be specified, for example, -exclude=*.xxx. 	

Parame ter	Optional or Mandatory	Description
include	Optional when generating download links of objects in batches (additional parameter)	 Indicates the matching patterns of objects that are included, for example: *.jpg. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. You can use * to represent * and \? to represent ?. Only after identifying that the name of the file to be downloaded does not match the value of exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is downloaded. If not, the file is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs://bucket/src1/src2/test.txt. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple include parameters can be specified, for example, -include=*.xxx.

Parame ter	Optional or Mandatory	Description
timeRan ge	Optional (additional parameter)	Indicates the time range matching pattern when generating download links of objects. Only the download links of objects whose latest modification time is within the configured time range are generated.
		This pattern has a lower priority than the object matching patterns (exclude / include). That is, the time range matching pattern is executed after the configured object matching patterns.
		NOTE
		• Time in the matching pattern is the UTC time.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
		• The matching time range is represented in <i>time1-time2</i> , where <i>time1</i> must be earlier than or the same as <i>time2</i> . The time format is <i>yyyyMMddHHmmss</i> .
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		• If this parameter is set to *- <i>time2</i> , all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i> -*, all files whose latest modification time is later than <i>time1</i> are matched.
0	Optional when generating download links of objects in batches	Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success and failure files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands. NOTE
	(additional parameter)	• The naming rule for result lists is as follows: sign_{succeed failed}_report_ <i>time</i> _TaskId.txt By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
endpoin t	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.

Parame ter	Optional or Mandatory	Description
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

5.10 Deleting an Object

Function

- You can use this command to delete a specified object.
- You can also use this command to delete objects in batches based on a specified object name prefix.

The delete operation cannot be undone.

Important Notes

In big data scenarios, parallel file systems usually have deep directory levels and each directory has a large number of files. In such case, deleting directories from parallel file systems may fail due to timeout. To address this problem, you are advised to delete directories in either of the following ways:

- 1. On the Hadoop client that has OBSA, an OBS client plugin, embedded, run the **hadoop fs rmr obs://{***Name of a parallel file system***}/{***Directory name***}** command.
- 2. Configure a lifecycle rule for directories so that they can be deleted in background based on the preset lifecycle rule.

Command Line Structure

- In Windows
 - Deleting a single object
 obsutil rm obs://bucket/key [-f] [-versionId=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Deleting objects in batches
 obsutil rm obs://bucket/[key] -r [-j=1] [-f] [-v] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx]
 [-t=xxx]
- In Linux or macOS
 - Deleting a single object
 ./obsutil rm obs://bucket/key [-f] [-versionId=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx]
 [-k=xxx] [-t=xxx]

Deleting objects in batches ./obsutil rm obs://bucket/[key] -r [-j=1] [-f] [-v] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Examples

• Take the Windows OS as an example. Run the **obsutil rm obs://buckettest/key -f** command to delete a single object named **key** in bucket **buckettest**.

obsutil rm obs://bucket-test/key -f

Start at 2024-09-25 04:48:10.1147483 +0000 UTC

Delete object [key] in the bucket [bucket-test] successfully, cost [152], request id [0000016979E1D2B2860BB5181229C72C]

 Take the Windows OS as an example. Run the obsutil rm obs://bucket-test r -f command to delete all objects in bucket bucket-test. obsutil rm obs://bucket-test -r -f Start at 2024-09-30 08:46:55.5335644 +0000 UTC

[======] 100.00% 21s Succeed count: 1313 Failed count: 0 Task id: 95936984-f81a-441a-bba0-1fd8254d9241

 Take the Windows OS as an example. Run the obsutil rm obs://buckettest/key -r -f command to delete all objects and folders prefixed with key in bucket bucket-test.
 obsutil rm obs://bucket-test/key -r -f

Start at 2024-09-30 08:49:09.5602115 +0000 UTC

[======] 100.00% 21s Succeed count: 10 Failed count: 0 Task id: 79ab59ec-7e00-4f22-8c88-465faa834125

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name
key	Mandatory for deleting a single object. Optional for deleting objects in batches.	Indicates the name of the object to be deleted, or the name prefix of the objects to be deleted in batches. NOTE If this parameter is left blank when deleting objects in batches, all objects in the bucket are deleted.
fr	Optional for deleting a single object (additional parameter)	Generates an operation result list when deleting an object.
f	Optional (additional parameter)	Runs in force mode.

Parameter	Optional or Mandatory	Description
versionId	Optional for deleting a single object (additional parameter)	Version ID of the object to be deleted.
r	Mandatory for deleting objects in batches (additional parameter)	Deletes objects in batches based on a specified object name prefix. CAUTION When you batch delete objects, all objects with the specified prefix will be deleted.
j	Optional for deleting objects in batches (additional parameter)	Indicates the maximum number of concurrent tasks for deleting objects in batches. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.
V	Optional for deleting objects in batches (additional parameter)	Deletes versions of an object and the delete markers in batches based on a specified object name prefix.
0	Optional (additional parameter)	Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success and failure files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands. NOTE
		 The naming rule for result lists is as follows: rm_{succeed failed}_report_time_TaskId.txt By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file. If there are multiple folders and files and you need to confirm the detailed error information about a failed
		task, refer to the failure list rm_failed_report _ <i>time</i> _ TaskId.txt in the result list folder and the log files in the log path.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .

Parameter	Optional or Mandatory	Description
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Refer to **Response** for uploading an object.

5.11 Synchronously Uploading Incremental Objects

Function

You can use this command to synchronize all content in a local source path to the specified target OBS bucket to ensure data consistency. Incremental synchronization has the following meanings:

- 1. "Incremental" means that the local source files are compared with their counterparts in the destination bucket and only those with content changes are uploaded.
- 2. "Synchronization" means that after the command is executed, all source files in the local path have their counterparts in the target OBS bucket.

NOTICE

- Do not change the local file or folder during synchronization. Otherwise, the synchronization may fail or data may be inconsistent.
- Each file can be synchronously uploaded only when it does not exist in the bucket, its size is different from the namesake one in the bucket, or it has the latest modification time.

When you compare each local file with data in the bucket, a billable HEAD request is generated. For details, see **Requests**.

Command Line Structure

- In Windows
 - Uploading a file synchronously
 obsutil sync file_url obs://bucket[/key] [-arcDir=xxx] [-dryRun] [-link] [-vlength] [-vmd5] [-p=1]
 [-threshold=5248800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-o=xxx] [cpd=xxx] [-fr] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Uploading a folder synchronously
 obsutil sync folder_url obs://bucket[/key] [-arcDir=xxx] [-dryRun] [-link] [-vlength] [-vmd5] [j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-at] [-mf] [-o=xxx] [-cpd=xxx] [config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
- In Linux or macOS
 - Uploading a file synchronously
 ./obsutil sync file_url obs://bucket[/key] [-arcDir=xxx] [-dryRun] [-link] [-vlength] [-vmd5] [p=1] [-threshold=5248800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-o=xxx] [cpd=xxx] [-fr] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Uploading a folder synchronously
 ./obsutil sync folder_url obs://bucket[/key] [-arcDir=xxx] [-dryRun] [-link] [-vlength] [-vmd5] [j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-at] [-mf] [-o=xxx] [-cpd=xxx] [config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil sync d:\temp\test.txt obs://bucket-test/key command to synchronously upload a file. obsutil sync d:\temp\test.txt obs://bucket-test/key

Start at 2024-09-25 04:48:10.1147483 +0000 UTC

Parallel: 5 Jobs: 5 Threshold: 50.00MB PartSize: auto VerifyLength: false VerifyMd5: false CheckpointDir: C:\Users\Administrator\.obsutil_checkpoint

[=====] 100.00% 1.68 MB/s 8.46MB/ 8.46MB 5s

Upload successfully, 8.46MB, d:\temp\test.txt --> obs://bucket-test/key, cost [55], status [200], request id [00000192421F4E224012B8470C0CCCDC]

 Take the Windows OS as an example. Run the obsutil sync d:\temp obs:// bucket-test/temp command to synchronously upload a folder. obsutil sync d:\temp obs://bucket-test/temp

Start at 2024-09-25 04:48:10.1147483 +0000 UTC

Parallel:5Jobs:5Threshold:50.00MBPartSize:autoVerifyLength:falseVerifyMd5:falseCheckpointDir:C:\Users\Administrator\.obsutil_checkpointTask id:104786c8-27c2-48fc-bc6a-5886596fb0ed

OutputDir: C:\Users\Administrator\.obsutil_output

[======] 100.00% tps:35.71 2.02 KB/s

7.20MB/7.20MB 0s Succeed count: 5 Failed count: 0 Succeed bytes: xxxx Metrics [max cost:90 ms, min cost:45 ms, average cost:63.80 ms, average tps:35.71, transferred size: 7.20MB] Task id: 104786c8-27c2-48fc-bc6a-5886596fb0ed

• For more examples, see **Synchronous Upload**.

Paramete r	Optional or Mandatory	Description
file_url	Mandatory for uploading a file synchronousl y	Local file path
folder_url	Mandatory for uploading a folder synchronousl y	Local folder path
bucket	Mandatory	Bucket name

Optional or Mandatory	Description
Optional	Indicates the object name or object name prefix specified when uploading a file synchronously, or the object name prefix specified when uploading a folder synchronously.
	The rules are as follows:
	 If this parameter is left blank when synchronously uploading a file, the file is uploaded to the root directory of the bucket and the object name is the file name. If the value ends with a slash (/), the value is used as the object name prefix when the file is uploaded, and the object name is the value plus the file name. If the value does not end with a slash (/), the file is uploaded with the value as the object name.
	• If this parameter is left blank when synchronously uploading a folder, all objects in the root directory of the bucket are the same as the files in the local folder. If this parameter is configured, objects whose name prefix is the configured value are the same as the files in the local folder.
	 NOTE If the value of this parameter does not end with a slash (/) when synchronously uploading a folder, the obsutil tool automatically adds a slash (/) at the end of the configured value as the object name prefix. For details about how to use this parameter, see
	Synchronous Upload.
Optional for synchronousl y uploading a file (additional parameter)	Generates an operation result list when synchronously uploading a file.
Optional (additional parameter)	Path to which the synchronously uploaded files are archived
Optional (additional parameter)	Conducts a dry run.
-	Mandatory Optional Optional for synchronousl y uploading a file (additional parameter) Optional (additional parameter)

Paramete r	Optional or Mandatory	Description
link	Optional (additional parameter)	 Uploads the actual path of the symbolic-link file/ folder NOTICE If this parameter is not specified and the file to be uploaded is a symbolic-link file whose target file does not exist, the exception message "The system cannot find the file specified" will be displayed in Windows OS, while the exception message "No such file or directory" will be displayed in macOS or Linux OS. Avoid the symbolic link loop of a folder, otherwise, the upload will exit due to panic. If you do not want the system to panic, set panicForSymbolicLinkCircle to false in the configuration file.
vlength	Optional (additional parameter)	After the synchronous upload is complete, check whether the sizes of the objects in the bucket are the same as those of the local files.
vmd5	Optional (additional parameter)	 After the synchronous upload is complete, check whether the MD5 values of the objects in the bucket are the same as those of the local files. NOTE If the size of the file or folder to be uploaded is too large, using this parameter will degrade the overall performance due to MD5 calculation. After the MD5 verification is successful, this parameter value is used for metadata x-obs-meta-md5chksum, for later MD5 verification during download or copy. CAUTION If your object needs encryption, do not use this parameter.
р	Optional (additional parameter)	Indicates the maximum number of concurrent multipart upload tasks when uploading a file. The default value is the value of defaultParallels in the configuration file.
threshold	Optional (additional parameter)	 Indicates the threshold for enabling multipart upload, in bytes. The default value is the value of defaultBigfileThreshold in the configuration file. NOTE If the size of the file or folder to be uploaded is smaller than the threshold, upload it directly. Otherwise, a multipart upload is required. If you upload a file or folder directly, no part record is generated, and resumable transmission is not supported. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).

Paramete r	Optional or Mandatory	Description
acl	Optional (additional parameter)	 Access control policies that can be specified when synchronously uploading files. Possible values are: private public-read public-read-write bucket-owner-full-control NOTE The preceding four values indicate private read and write, public read, public read and write, and bucket owner full control.
SC	Optional (additional parameter)	 Indicates the storage classes of objects that can be specified when synchronously uploading files. Possible values are: standard: Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB. warm: Infrequent Access storage class. It is ideal for storing infrequently accessed (less than 12 times a year) data, but when needed, the access has to be fast. cold: Archive storage class. It provides secure, durable, and inexpensive storage for rarely-accessed (once a year) data.
meta	Optional (additional parameter)	Indicates the customized metadata that can be specified when uploading files. The format is key1:value1#key2:value2#key3:value3 . NOTE The preceding value indicates that the object in the bucket contains three groups of customized metadata after the file is uploaded: key1:value1 , key2:value2 , and key3:value3 .
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart upload task, in bytes. The value ranges from 100KB to 5GB. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source file size.

Paramete r	Optional or Mandatory	Description
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the subfolder in the home directory of the user who executes obsutil commands. NOTE A part record is generated during a multipart upload and saved to the upload subfolder. After the upload succeeds, its part record is deleted automatically. If the upload fails or is suspended, the system attempts to resume the task according to its part record when you perform the upload the next time.
j	Optional for synchronousl y uploading a folder (additional parameter)	Indicates the maximum number of concurrent tasks for uploading a folder synchronously. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.
exclude	Optional for synchronousl y uploading a folder (additional parameter)	 Indicates the file matching patterns that are excluded, for example: *.txt. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt. You can use * to represent * and \? to represent ?. If the name of the file to be uploaded matches the value of this parameter, the file is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute file path (including the file name and file directory). The matching pattern takes effect only for files in the folder. Multiple exclude parameters can be specified, for example, -exclude=*.xxx -exclude=*.xxx.

Paramete r	Optional or Mandatory	Description
include	Optional for synchronousl y uploading a folder (additional parameter)	 Indicates the file matching patterns that are included, for example: *.jpg. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. You can use * to represent * and \? to represent ?. Only after identifying that the name of the file to be uploaded does not match the value of exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is uploaded. If not, the file is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute file path (including the file name and file directory). The matching pattern takes effect only for files in the folder. Multiple include parameters can be specified, for example, -include=*.xxx -include=*.xxx.
at	Optional for synchronousl y uploading a folder (additional parameter)	 Indicates that when synchronously uploading a folder, only the files whose latest access time is within the value of timeRange are uploaded. NOTE This parameter must be used together with timeRange.
disableDir Object	Optional for synchronousl y uploading folders (additional parameter)	Indicates the folders themselves are not uploaded as an object. Configuring this parameter can avoid uploading empty folders to a bucket. If a folder contains files, the files will be uploaded and the original path format is retained.

Paramete r	Optional or Mandatory	Description	
timeRang e	Optional for synchronousl y uploading a folder	Indicates the time range matching pattern when synchronously uploading files. Only files whose latest modification time is within the configured time range are uploaded.	
	(additional parameter)	This pattern has a lower priority than the file matching patterns (exclude / include). That is, the time range matching pattern is executed after the configured file matching patterns.	
		NOTE	
		• The matching time range is represented in <i>time1-time2</i> , where <i>time1</i> must be earlier than or the same as <i>time2</i> . The time format is <i>yyyyMMddHHmmss</i> .	
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000. 	
		 If this parameter is set to *-<i>time2</i>, all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i>-*, all files whose latest modification time is later than <i>time1</i> are matched. 	
		NOTICE Time in the matching pattern is the UTC time.	
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on folders.	
0	Optional (additional parameter)	Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands.	
		NOTE	
		 The naming rule for result lists is as follows: sync_{succeed failed warning}_report_time_TaskId.txt 	
		• By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file.	
		 If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list sync_failed_report_time_TaskId.txt in the result list folder and the log files in the log path. 	

Paramete r	Optional or Mandatory	Description
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Refer to **Response** for uploading an object.

5.12 Synchronously Copying Incremental Objects

Function

This function synchronizes all objects in the specified path in the source bucket with objects in the specified path in the destination bucket to keep data consistency. Incremental synchronization has the following meanings: (1) Increment: Compare the source object with the target object and copy only the source object that has changes. (2) Synchronization: After the command is executed, ensure that the specified path of the source bucket is a subset of the destination bucket. That is, any object in the specified path of the source bucket has its corresponding object in the destination bucket.

NOTICE

- Do not change the source objects in the OBS bucket when synchronously copying objects. Otherwise, the synchronization may fail or data may be inconsistent.
- If the storage class of the object to be copied is **cold**, you must restore the object to be copied first. Otherwise, the copy fails.
- To copy objects, you must have the read permission on the objects to be copied and the write permission on the destination bucket.
- If the client-side cross-region replication function is not enabled, ensure that the source bucket and destination bucket are in the same region.
- Each object can be synchronously copied only when it does not exist in the destination bucket, its size is different from the namesake one in the destination bucket, or it has the latest modification time.
- If the source bucket is a parallel file system (supporting POSIX), the destination bucket cannot be an object storage bucket.

When you compare each local file with data in the bucket, a billable HEAD request is generated. For details, see **Requests**.

Command Line Structure

In Windows

obsutil sync obs://srcbucket[/key] obs://dstbucket[/dest] [-dryRun] [-crr] [-vlength] [-vmd5] [-j=1] [p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

• In Linux or macOS

./obsutil sync obs://srcbucket[/key] obs://dstbucket[/dest] [-dryRun] [-crr] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [e=xxx] [-i=xxx] [-t=xxx] [-t=xxx]

D NOTE

The source path and destination path cannot be the same or nested when synchronously copying objects.

Examples

 Take the Windows OS as an example. Run the obsutil sync obs://buckettest/temp/ obs://bucket-test2/temp/ command to synchronously copy objects.

obsutil sync obs://bucket-test/temp/ obs://bucket-test2/temp

Start at 2024-09-25 04:48:10.1147483 +0000 UTC

Parallel: 5 Jobs: 5 Threshold: 50.00MB PartSize: auto CheckpointDir: C:\Users\Administrator\.obsutil_checkpoint Task id: 104786c8-27c2-48fc-bc6a-5886596fb0e OutputDir: C:\Users\Administrator\.obsutil_output

[======] 100.00% 10/s 0s Succeed count: 5 Failed count: 0 Metrics [max cost:298 ms, min cost:192 ms, average cost:238.00 ms, average tps:9.71, transferred size: 7.20MB] Task id: 0476929d-9d23-4dc5-b2f8-0a0493f027c5

• For more examples, see **Synchronous Copy**.

Parameter	Optional or Mandatory	Description
srcbucket	Mandatory	Source bucket name
dstbucket	Mandatory	Destination bucket name
dest	Optional	Name prefix of destination objects. NOTE If the value of this parameter does not end with a slash (/), the obsutil tool automatically adds a slash (/) at the end of the configured value as the name prefix of destination objects.
key	Optional	Name prefix of source objects The rules are as follows:
		 If this parameter is left blank, objects whose name prefix is the value of dest in the destination bucket are the same as all objects in the source bucket.
		• If this parameter is configured, objects whose name prefix is the value of dest in the destination bucket are the same as objects whose name prefix is this configured value in the source bucket.
		NOTE
		 If the value of this parameter does not end with a slash (/), the obsutil tool automatically adds a slash (/) at the end of the configured value as the name prefix of source objects.
		• For details about how to use this parameter, see Synchronous Copy .
dryRun	Optional (additional parameter)	Conducts a dry run.

Parameter	Optional or Mandatory	Description
crr	Optional (additional parameter)	 Enables the client-side cross-region replication function. In this mode, data is directly copied to the destination bucket from the source bucket through data stream. The buckets can by any two OBS buckets. NOTE If this parameter is configured, ensure that the configuration of client-side cross-region replication is updated in the configuration file. For details, see Updating a Configuration File. The configurations of the source bucket and destination bucket are respectively akCrr/ skCrr/tokenCrr/endpointCrr and ak/sk/ token/endpoint in the configuration file. NOTICE When cross-region replication is enabled, the upload/download bandwidth, CPU, and memory resources of the host where commands are executed will be occupied, which may deteriorate the host performance.
vlength	Optional (additional parameter)	Verifies whether the object size in the destination bucket is the same as that in the source bucket after the copy task completes. NOTE This parameter must be used together with crr .
vmd5	Optional (additional parameter)	 Verifies whether the MD5 value of the destination bucket is the same as that of the source bucket after the copy task completes. NOTE This parameter must be used together with crr. Objects in the source bucket must contain metadata x-obs-meta-md5chksum, or MD5 verification will be skipped. After the MD5 verification is successful, this parameter value is used for metadata x-obs-meta-md5chksum of the destination object, for later MD5 verification during download or copy. CAUTION If your object needs encryption, do not use this parameter.
р	Optional (additional parameter)	Indicates the maximum number of concurrent multipart copy tasks when copying an object. The default value is the value of defaultParallels in the configuration file.

Parameter	Optional or Mandatory	Description
threshold	Optional (additional parameter)	 Indicates the threshold for enabling multipart copy, in bytes. The default value is the value of defaultBigfileThreshold in the configuration file. NOTE If the size of the object to be copied is smaller than the threshold, copy the object directly. If not, a multipart copy is required. If you copy an object directly, no part record is generated, and resumable transmission is not supported. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).
acl	Optional (additional parameter)	 Access control policies for destination objects that can be specified when copying objects. Possible values are: private public-read public-read-write bucket-owner-full-control NOTE The preceding four values indicate private read and write, public read, public read and write, and bucket owner full control.
SC	Optional (additional parameter)	 Storage classes of the destination objects that can be specified when copying objects. Possible values are: standard: Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB. warm: Infrequent Access storage class. It is ideal for storing infrequently access has to be fast. cold: Archive storage class. It provides secure, durable, and inexpensive storage for rarely-accessed (once a year) data.

Parameter	Optional or Mandatory	Description
meta	Optional (additional parameter)	Metadata of destination objects that can be specified when copying objects. This parameter should be configured in the following format: <i>key1:value1#key2:value2#key3:value3</i> . NOTE The format example above indicates that the destination objects contain three groups of custom metadata: <i>key1:value1, key2:value2</i> , and <i>key3:value3</i> .
fs	Optional (additional parameter)	 Specifies whether the method of listing parallel file systems is applied. If you are listing parallel file systems, you are recommended to add this parameter. CAUTION With this method, the listing time required varies largely depending on the directory structures. After this parameter is enabled, marker and limit will be ignored. Then, the size of the bucket, parallel file system, or directory will be calculated. This parameter is only supported by obsutil 5.5.12 and later.
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart copy task, in bytes. The value ranges from 100KB to 5GB. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source object size.
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the subfolder in the home directory of the user who executes obsutil commands. NOTE A part record is generated during a multipart copy and saved to the copy subfolder. After the copy succeeds, its part record is deleted automatically. If the copy fails or is suspended, the system attempts to resume the task according to its part record when you perform the copy the next time.

Parameter	Optional or Mandatory	Description
j	Optional for copying objects in batches (additional parameter)	Indicates the maximum number of concurrent tasks for copying objects synchronously. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.
exclude	Optional for copying objects in batches (additional parameter)	 Indicates the matching patterns of source objects that are excluded, for example: *.txt. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt. You can use * to represent * and \? to represent ?. If the name of the object to be copied matches the value of this parameter, the object is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object is show the absolute path of the object taxt. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple exclude parameters can be specified, for example, -exclude=*.xxx -exclude=*.xxx.

Parameter	Optional or Mandatory	Description
include	Optional for copying objects in batches (additional parameter)	 Indicates the matching patterns of source objects that are included, for example: *.jpg. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. You can use * to represent * and \? to represent ?. Only after identifying that the name of the file to be copied does not match the value of exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is copied. If not, the file is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special
		the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows.
		 The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs://bucket/src1/src2/ test.txt, then the absolute path of the object is src1/src2/test.txt.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
		 Multiple include parameters can be specified, for example, -include=*.xxx -include=*.xxx.

Parameter	Optional or Mandatory	Description
timeRange	Optional (additional parameter)	Indicates the time range matching pattern when synchronously copying objects. Only objects whose latest modification time is within the configured time range are copied.
		This pattern has a lower priority than the object matching patterns (exclude/include). That is, the time range matching pattern is executed after the configured object matching patterns.
		NOTE
		• The matching time range is represented in <i>time1-time2</i> , where <i>time1</i> must be earlier than or the same as <i>time2</i> . The time format is <i>yyyyMMddHHmmss</i> .
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		 If this parameter is set to *-<i>time2</i>, all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i>-*, all files whose latest modification time is later than <i>time1</i> are matched.
		NOTICE
		• Time in the matching pattern is the UTC time.
		• This matching pattern applies only to objects whose names do not end with a slash (/).
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on objects whose names end with a slash (/).

Parameter	Optional or Mandatory	Description
0	Optional (additional parameter)	 Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output, the subfolder in the home directory of the user who executes obsutil commands. NOTE The naming rule for result lists is as follows: sync_{succeed failed warning}_report_time_TaskId.txt By default, the maximum size of a single result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file. If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list sync_failed_report_time_TaskId.txt in the result list folder and the log files in the log path.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

5.13 Synchronously Downloading Incremental Objects

Function

This function synchronizes all content in the specified path of the source bucket to the destination bucket on OBS, ensuring that the content is consistent between the specified path of the source bucket and the destination bucket. Incremental synchronization has the following meanings: (1) Increment: Compare the source object with the target file and download only the source object that has changes. (2) Synchronization: After the command is executed, ensure that the specified path of the source bucket is a subset of the local target path. That is, any object in the specified path of the source bucket has its corresponding file in the local target path.

NOTICE

- Do not change the source objects in the OBS bucket when synchronously downloading objects. Otherwise, the synchronization may fail or data may be inconsistent.
- If the storage class of the object to be downloaded is **cold**, you must restore the object first. Otherwise, the download fails.
- Each object can be synchronously downloaded only when it does not exist in the local path, its size is different from the namesake one in the local path, or it has the latest modification time.
- A single file cannot be downloaded synchronously. Only folders can be downloaded synchronously.

When you compare each local file with data in the bucket, a billable HEAD request is generated. For details, see **Requests**.

Command Line Structure

In Windows

obsutil sync obs://bucket[/key] folder_url [-tempFileDir=xxx] [-dryRun] [-vlength] [-vmd5] [-j=1] [p=1] [-threshold=52428800] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

In Linux or macOS

 ./obsutil sync obs://bucket[/key] folder_url [-tempFileDir=xxx] [-dryRun] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil sync obs://buckettest/temp d:\ temp command to download objects synchronously. obsutil sync obs://bucket-test/temp d:\temp Start at 2024-09-30 08:53:22.327072 +0000 UTC

Parallel: 5	Jobs: 5	
Threshold: 50.00MB	PartSize:	auto
VerifyLength: false CheckpointDir: xxxx	VerifyMd5:	false
Task id: 3066a4b0-4d21 OutputDir: xxxx TempFileDir: xxxx	-4929-bb84-4829c	32cbd0f
-		======] 100.00% tps:17.86 155.59 KB/s
7.20MB/7.20MB 0s		
Succeed count: 6	Failed count: 0	
Succeed bytes: xxxx		

Task id: 3066a4b0-4d21-4929-bb84-4829c32cbd0f

• For more examples, see **Synchronous Download**.

Parameter	Optional or Mandatory	Description
folder_url	Mandatory	Local folder path
bucket	Mandatory	Bucket name
key	Optional	Indicates the name prefix of objects to be synchronously downloaded.
		The rules are as follows:
		 If this parameter is left blank, all files in the folder specified by folder_url are the same as all objects in the bucket.
		 If this parameter is configured, all files in the folder specified by folder_url are the same as the objects whose name prefix is the configured value in the bucket.
		NOTE
		 If the value of this parameter does not end with a slash (/), the obsutil tool automatically adds a slash (/) at the end of the configured value as the object name prefix.
		 For details about how to use this parameter, see Synchronous Download.

Parameter	Optional or Mandatory	Description
tempFileDir	Optional (additional parameter)	 Indicates the directory for storing temporary files during synchronous download. The default value is the value of defaultTempFileDir in the configuration file. NOTE Temporary files generated during multipart download are stored in this directory. Therefore, ensure that the user who executes obsutil has the write permission on the path. The available space of the partition where the path is located must be greater than the size of the objects to be downloaded.
dryRun	Optional (additional parameter)	Conducts a dry run.
vlength	Optional (additional parameter)	Checks whether the sizes of the local files are the same as those of the objects in the bucket after the download is complete.
vmd5	Optional (additional parameter)	Checks whether MD5 values of the local files are the same as those of the objects in the bucket after the download is complete. NOTE Objects in the bucket must contain metadata x- obs-meta-md5chksum, or MD5 verification will be skipped. CAUTION If your object needs encryption, do not use this parameter.
р	Optional (additional parameter)	Indicates the maximum number of concurrent multipart download tasks when downloading an object. The default value is the value of defaultParallels in the configuration file.

Parameter	Optional or Mandatory	Description
threshold	Optional (additional parameter)	 Indicates the threshold for enabling multipart download, in bytes. The default value is the value of defaultBigfileThreshold in the configuration file. NOTE If the size of the object to be downloaded is smaller than the threshold, download the object directly. If not, a multipart download is required. If you download an object directly, no part record is generated, and resumable transmission is not supported.
		• This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart download task, in bytes. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source object size.
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint, the subfolder in the home directory of the user who executes obsutil commands. NOTE A part record is generated during a multipart download and saved to the down subfolder. After the download succeeds, its part record is deleted automatically. If the download fails or is suspended, the system attempts to resume the task according to its part record when you perform the download the next time.
j	Optional for downloading objects in batches (additional parameter)	Indicates the maximum number of concurrent tasks for downloading objects synchronously. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.

Parameter	Optional or Mandatory	Description
exclude	Optional for downloading objects in batches (additional parameter)	 Indicates the matching patterns of source objects that are excluded, for example: *.txt. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt. You can use * to represent * and \? to represent ?. If the name of the object to be downloaded matches the value of this parameter, the object is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object is src1/src2/test.txt. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple exclude parameters can be specified, for example, -exclude=*.xxx - exclude=*.xxx.

Parameter	Optional or Mandatory	Description
include	Optional for downloading objects in batches (additional parameter)	 Indicates the matching patterns of source objects that are included, for example: *.jpg. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. You can use * to represent * and \? to represent ?. Only after identifying that the name of the file to be downloaded does not match the value of exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is downloaded. If not, the file is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object is src1/src2/test.txt. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple include parameters can be specified, for example, -include=*.xxx - include=*.xxx.

Parameter	Optional or Mandatory	Description
timeRange	Optional (additional parameter)	Indicates the time range matching pattern when synchronously downloading objects. Only objects whose latest modification time is within the configured time range are downloaded.
		This pattern has a lower priority than the object matching patterns (exclude / include). That is, the time range matching pattern is executed after the configured object matching patterns.
		NOTE
		• The matching time range is represented in <i>time1-time2</i> , where <i>time1</i> must be earlier than or the same as <i>time2</i> . The time format is <i>yyyyMMddHHmmss</i> .
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		 If this parameter is set to *-<i>time2</i>, all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i>- *, all files whose latest modification time is later than <i>time1</i> are matched.
		NOTICE
		 Time in the matching pattern is the UTC time.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on objects whose names end with a slash (/).

Parameter	Optional or Mandatory	Description
0	Optional (additional parameter)	Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands. NOTE
		 The naming rule for result lists is as follows: sync_{succeed failed warning}_report_time_TaskId.txt
		 By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file.
		 If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list sync_failed_report_time_TaskId.txt in the result list folder and the log files in the log path.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

5.14 Restoring Objects from the Archive Storage

Function

You can use this command to restore a specified **cold** object or restore **cold** objects with a specific prefix in batches.

NOTE

- Object content cannot be read during restoration.
- After an object is restored, the time it requires before the object can be downloaded depends on the OBS server.
- obsutil currently does not support the restoration status query. You can use OBS Browser
 + or call the API for obtaining object metadata to query the restoration status.

To prolong the validity period of the Archive data restored, you can repeatedly restore the Archive data, but you will be billed for each restore. After a second restore, the validity period of Standard object copies will be prolonged, and you need to pay for storing these copies during the prolonged period.

Command Line Structure

- In Windows
 - Restoring an object
 obsutil restore obs://bucket/key [-d=1] [-t=xxx] [-versionId=xxx] [-fr] [-o=xxx] [-config=xxx] [e=xxx] [-i=xxx] [-teken=xxx]
 - Restoring objects in batches
 obsutil restore obs://bucket[/key] -r [-f] [-v] [-d=1] [-t=xxx] [-o=xxx] [-j=1] [-config=xxx] [e=xxx] [-i=xxx] [-k=xxx] [-token=xxx]
 - Restoring all objects in a specific directory at a time obsutil restore obs://bucket/folder/ -r [-f] [-v] [-d=1] [-t=xxx] [-o=xxx] [-j=1] [-config=xxx] [e=xxx] [-i=xxx] [-k=xxx] [-token=xxx]
- In Linux or macOS
 - Restoring an object
 ./obsutil restore obs://bucket/key [-d=1] [-t=xxx] [-versionId=xxx] [-fr] [-o=xxx] [-config=xxx] [e=xxx] [-i=xxx] [-k=xxx] [-token=xxx]
 - Restoring objects in batches
 ./obsutil restore obs://bucket[/key] -r [-f] [-v] [-d=1] [-t=xxx] [-o=xxx] [-j=1] [-config=xxx] [-e=xxx] [-i=xxx] [-token=xxx]
 - Restoring all objects in a specific directory at a time
 ./obsutil restore obs://bucket/folder/ -r [-f] [-v] [-d=1] [-t=xxx] [-o=xxx] [-j=1] [-config=xxx] [-e=xxx] [-i=xxx] [-texen=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil restore obs://buckettest/key command to restore a single object whose storage class is cold. obsutil restore obs://bucket-test/key Start at 2024-09-30 08:56:17.9537365 +0000 UTC Start to restore object [key] in the bucket [bucket-test] successfully, cost [252] ms, request id [0000019242250F754015F23EE0B7876E]

 Take the Windows OS as an example. Run the obsutil restore obs://buckettest -r -f command to restore objects whose storage class is cold in the bucket in batches. obsutil restore obs://bucket-test -r -f

Start at 2024-09-30 08:57:11.3565648 +0000 UTC

[======] 100.00% 3s Succeed count: 12 Failed count: 0 Metrics [max cost:264 ms, min cost:54 ms, average cost:119.33 ms, average tps:19.70] Task id: 96f104ee-d0bf-40ff-95dd-31dec0d8f4f4

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name
key	Mandatory for restoring a single object whose storage class is cold	Indicates the name of the object to be restored or the name prefix of the objects to be restored in batches. NOTE If this parameter is left blank when batch restoring objects all objects where storage class is cold in
	Optional for batch restoring objects whose storage class is cold	objects, all objects whose storage class is cold in the bucket are restored.
d	Optional (additional parameter)	Storage duration after objects whose storage class is cold are restored, in days. The value ranges from 1 to 30. The default value is 1 .
t	Optional (additional parameter)	 Options for restoring objects. Possible values are: standard expedited NOTE expedited indicates that objects can be quickly restored from Archive storage within 1 to 5 minutes. standard indicates that objects can be restored from Archive storage within 3 to 5 hours. If this parameter is not configured, an expedited restore is used by default.
versionId	Optional for restoring a single object whose storage class is cold (additional parameter)	Version ID of the to-be-restored object whose storage class is cold

Parameter	Optional or Mandatory	Description
fs	Optional (additional parameter)	Specifies whether the method of listing parallel file systems is applied. If you are listing parallel file systems, you are recommended to add this parameter. CAUTION
		 With this method, the listing time required varies largely depending on the directory structures.
		 After this parameter is enabled, marker and limit will be ignored. Then, the size of the bucket, parallel file system, or directory will be calculated.
		 This parameter is only supported by obsutil 5.5.12 and later.
encoding- type	Optional (additional parameter)	If the object name contains special characters, set this parameter to url . NOTE This parameter is only available for obsutil 5.5.12 and later.
fr	Optional for restoring a single object whose storage class is cold (additional parameter)	Generates an operation result list when restoring a single object whose storage class is cold .
f	Optional for batch restoring objects whose storage class is cold (additional parameter)	Runs in force mode.
r	Mandatory for batch restoring objects whose storage class is cold (additional parameter)	Restores objects whose storage class is cold in batches by object name prefix.
V	Optional for batch restoring objects whose storage class is cold (additional parameter)	Restores versions of objects whose storage class is cold in batches by object name prefix.

Parameter	Optional or Mandatory	Description
0	Optional (additional parameter)	Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success and failure files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands.
		 NOTE The naming rule for result lists is as follows: restore_{succeed failed}_report_time_TaskId.txt By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file.
j	Optional for batch restoring objects whose storage class is cold (additional parameter)	Maximum number of concurrent tasks for batch restoring objects whose storage class is cold . The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
token	Optional (additional parameter)	Specifies the user's security token.

5.15 Resuming a Failed Upload Task

Function

You can use this command to resume a failed upload task based on the task ID.

Command Line Structure

• In Windows

obsutil cp -recover=xxx [-arcDir=xxx] [-dryRun] [-f] [-u] [-vlength] [-vmd5] [-j=1] [-p=1] [threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-clear] [-config=xxx] [-e=xxx] [i=xxx] [-k=xxx] [-t=xxx]

• In Linux or macOS

./obsutil cp -recover=xxx [-arcDir=xxx] [-dryRun] [-f] [-u] [-vlength] [-vmd5] [-j=1] [-p=1] [threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-clear] [-config=xxx] [-e=xxx] [i=xxx] [-k=xxx] [-t=xxx]

Examples

Take the Windows OS as an example. Run the obsutil cp -recover
 104786c8-27c2-48fc-bc6a-5886596fb0ed -f command to resume the failed upload task.

obsutil cp -recover 104786c8-27c2-48fc-bc6a-5886596fb0ed -f Start at 2024-10-08 01:10:07.3809685 +0000 UTC

Parallel: 5	Jobs: 5	
Threshold: 50.00MB	PartSize:	auto
VerifyLength: false CheckpointDir: xxxx	VerifyMd5:	false

Task id: a628d6da-c562-4a1f-b687-4fa125de0dc3 OutputDir: xxxx TempFileDir: xxxx

[======] 100.00% tps:35.71 2.02 KB/s 7.20MB/7.20MB 0s Succeed count: 1 Failed count: 0 Succeed bytes: xxx Metrics [max cost:90 ms, min cost:45 ms, average cost:63.80 ms, average tps:35.71, transferred size:70B]

Task id: a628d6da-c562-4a1f-b687-4fa125de0dc3

Paramete r	Optional or Mandatory	Description
recover	Mandatory (additional parameter)	 ID of the upload task to be resumed NOTE You can obtain the task ID after an upload task is complete, or query it based on the name of the result list. A task ID is the 36 characters, excluding the suffix .txt. You can locate the upload task to be resumed in the directory where the result lists reside. For details about the directory of the result lists, see additional parameter o.
arcDir	Optional (additional parameter)	Path to which the uploaded files are archived
dryRun	Optional (additional parameter)	Conducts a dry run.
u	Optional (additional parameter)	Indicates incremental upload. If this parameter is set, each file can be uploaded only when it does not exist in the bucket, its size is different from the namesake one in the bucket, or it has the latest modification time. CAUTION When you compare each local file with data in the bucket, a billable HEAD request is generated. For details, see Requests .
vlength	Optional (additional parameter)	After the upload completes, check whether the sizes of the objects in the bucket are the same as those of the local files.
vmd5	Optional (additional parameter)	 After the upload completes, check whether the MD5 values of the objects in the bucket are the same as those of the local files. NOTE If the size of the file or folder to be uploaded is too large, using this parameter will degrade the overall performance due to MD5 calculation. After the MD5 verification is successful, this parameter value is used for metadata x-obs-meta-md5chksum, for later MD5 verification during download or copy. CAUTION If your object needs encryption, do not use this parameter.

Paramete r	Optional or Mandatory	Description
р	Optional (additional parameter)	Indicates the maximum number of concurrent multipart upload tasks when uploading a file. The default value is the value of defaultParallels in the configuration file.
threshold	Optional (additional parameter)	 Indicates the threshold for enabling multipart upload, in bytes. The default value is the value of defaultBigfileThreshold in the configuration file. NOTE If the size of the file or folder to be uploaded is smaller than the threshold, upload it directly. Otherwise, a multipart upload is required. If you upload a file or folder directly, no part record is generated, and resumable transmission is not supported. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).
acl	Optional (additional parameter)	 Access control policies that can be specified when uploading files. Possible values are: private public-read public-read-write NOTE The preceding three values indicate private read and write, public read, and public read and write.
SC	Optional (additional parameter)	 Indicates the storage classes of objects that can be specified when uploading files. Possible values are: standard: Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB. warm: Infrequent Access storage class. It is ideal for storing infrequently accessed (less than 12 times a year) data, but when needed, the access has to be fast. cold: Archive storage class. It provides secure, durable, and inexpensive storage for rarely-accessed (once a year) data.

Paramete r	Optional or Mandatory	Description
meta	Optional (additional parameter)	Indicates the customized metadata that can be specified when uploading files. The format is key1:value1#key2:value2#key3:value3 . NOTE The preceding value indicates that the object in the bucket contains three groups of customized metadata after the file is uploaded: key1:value1 , key2:value2 , and key3:value3 .
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart upload task, in bytes. The value ranges from 100KB to 5GB. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source file size.
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the subfolder in the home directory of the user who executes obsutil commands. NOTE A part record is generated during a multipart upload and saved to the upload subfolder. After the upload succeeds, its part record is deleted automatically. If the upload fails or is suspended, the system attempts to resume the task according to its part record when you perform the upload the next time.
f	Optional (additional parameter)	Runs in force mode.
j	Optional (additional parameter)	Maximum number of concurrent tasks for uploading a folder. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.

Paramete r	Optional or Mandatory	Description	
exclude	Optional (additional parameter)	Indicates the file matching patterns that are excluded, for example: *.txt .	
		 The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt. 	
		 You can use * to represent * and \? to represent ?. 	
		• If the name of the file to be uploaded matches the value of this parameter, the file is skipped.	
		NOTICE	
		 You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. 	
		 The matching pattern applies to the absolute file path (including the file name and file directory). 	
		 The matching pattern applies only to files in a folder. 	
		 Multiple exclude parameters can be specified, for example, -exclude=*.xxx -exclude=*.xxx. 	

Paramete r	Optional or Mandatory	Description
include	Optional (additional parameter)	 Indicates the file matching patterns that are included, for example: *.jpg. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. You can use ** to represent * and \? to represent ?. Only after identifying that the name of the file to be uploaded does not match the value of
		exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is uploaded. If not, the file is skipped. NOTICE
		• You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows.
		 The matching pattern applies to the absolute file path (including the file name and file directory).
		 The matching pattern applies only to files in a folder.
		 Multiple include parameters can be specified, for example, -include=*.xxx -include=*.xxx.
at	Optional (additional parameter)	Indicates that when resuming a failed upload task, only the files whose latest access time is within the value of timeRange are uploaded. NOTE This parameter must be used together with
		timeRange.

Paramete r	Optional or Mandatory	Description
timeRang e	Optional (additional parameter)	Indicates the time range matching pattern when uploading files. Only files whose latest modification time is within the configured time range are uploaded.
		This pattern has a lower priority than the file matching patterns (exclude/include). That is, the time range matching pattern is executed after the configured file matching patterns.
		NOTE
		 The matching time range is represented in <i>time1-time2</i>, where <i>time1</i> must be earlier than or the same as <i>time2</i>. The time format is <i>yyyyMMddHHmmss</i>.
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		 If this parameter is set to *-<i>time2</i>, all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i>-*, all files whose latest modification time is later than <i>time1</i> are matched.
		NOTICE Time in the matching pattern is the UTC time.
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on objects whose names end with a slash (/).

Paramete r	Optional or Mandatory	Description
0	Optional (additional parameter)	Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands.
		 NOTE The naming rule for result lists is as follows: cp_{succeed failed warning}_report_time_TaskId.txt. By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file. If there are multiple folders and files and you
		need to confirm the detailed error information about a failed task, refer to the failure list cp_failed_report_ <i>time_TaskId</i> . txt in the result list folder and the log files in the log path.
clear	Optional (additional parameter)	Deletes the failure result files after the upload task is resumed.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
е	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

5.16 Resuming a Failed Copy Task

Function

You can use this command to resume a failed copy task based on the task ID.

Command Line Structure

• In Windows

obsutil cp -recover=xxx [-dryRun] [-f] [-u] [-crr] [-vlength] [-vmd5] [-j=1] [-p=1] [threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-clear] [-config=xxx] [-e=xxx] [i=xxx] [-k=xxx] [-t=xxx]

In Linux or macOS

./obsutil cp -recover=xxx [-dryRun] [-f] [-u] [-crr] [-vlength] [-vmd5] [-j=1] [-p=1] [threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-clear] [-config=xxx] [-e=xxx] [i=xxx] [-k=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil cp recover=0476929d-9d23-4dc5-b2f8-0a0493f027c5 -f command to copy objects in batches.

```
obsutil cp -recover=0476929d-9d23-4dc5-b2f8-0a0493f027c5 -f
Start at 2024-10-08 01:10:07.3809685 +0000 UTC
```

Parallel:	5	Jobs:	5	
Threshold:	50.00MB	Pai	rtSize:	auto
VerifyLength: false		Verif	yMd5:	false
Checkpoint	Dir: xxxx			

Task id: a628d6da-c562-4a1f-b687-4fa125de0dc3 OutputDir: xxxx TempFileDir: xxxx

[======] 100.00% tps:35.71 2.02 KB/s 7.20MB/7.20MB 0s Succeed count: 1 Failed count: 0 Succeed bytes: xxx Metrics [max cost:90 ms, min cost:45 ms, average cost:63.80 ms, average tps:35.71, transferred size:70B]

Task id: a628d6da-c562-4a1f-b687-4fa125de0dc3

Parameter	Optional or Mandatory	Description
recover	Mandatory (additional parameter)	 ID of the copy task to be resumed. NOTE You can obtain the task ID after a copy task is complete, or query it based on the name of the result list. A task ID is the 36 characters, excluding the suffix .txt. You can locate the copy task to be resumed in the directory where the result lists reside. For details about the directory of the result lists, see additional parameter o.
dryRun	Optional (additional parameter)	Conducts a dry run.
crr	Optional (additional parameter)	 Enables the client-side cross-region replication function. In this mode, data is directly copied to the destination bucket from the source bucket through data stream. The buckets can by any two OBS buckets. NOTE If this parameter is configured, ensure that the configuration of client-side cross-region replication is updated in the configuration file. For details, see Updating a Configuration File. The configurations of the source bucket and destination bucket are respectively akCrr/ skCrr/tokenCrr/endpointCrr and ak/sk/ token/endpoint in the configuration file. NOTICE When cross-region replication is enabled, the upload/download bandwidth, CPU, and memory resources of the host where commands are executed will be occupied, which may deteriorate the host performance.
vlength	Optional (additional parameter)	Verifies whether the object size in the destination bucket is the same as that in the source bucket after the copy task completes. NOTE This parameter must be used together with crr .

Parameter	Optional or Mandatory	Description	
vmd5	Optional (additional parameter)	 Verifies whether the MD5 value of the destination bucket is the same as that of the source bucket after the copy task completes. NOTE This parameter must be used together with crr. Objects in the source bucket must contain metadata x-obs-meta-md5chksum, or MD5 verification will be skipped. After the MD5 verification is successful, this parameter value is used for metadata x-obs-meta-md5chksum of the destination object, for later MD5 verification during download or copy. CAUTION If your object needs encryption, do not use this parameter. 	
u	Optional (additional parameter)	Indicates incremental copy. If this parameter is set, each object can be copied only when it does not exist in the destination bucket, its size is different from the namesake one in the destination bucket, or it has the latest modification time. CAUTION When you compare each local file with data in the bucket, a billable HEAD request is generated. For details, see Requests .	
р	Optional (additional parameter)	Indicates the maximum number of concurrent multipart copy tasks when copying an object. The default value is the value of defaultParallels in the configuration file.	
threshold	Optional (additional parameter)	 Indicates the threshold for enabling multipart copy, in bytes. The default value is the value of defaultBigfileThreshold in the configuration file. NOTE If the size of the object to be copied is smaller than the threshold, copy the object directly. If not, a multipart copy is required. If you copy an object directly, no part record is generated, and resumable transmission is not supported. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). 	

Parameter	Optional or Mandatory	Description
acl	Optional (additional parameter)	Access control policies for destination objects that can be specified when copying objects. Possible values are:
		• private
		• public-read
		 public-read-write
		NOTE The preceding three values indicate private read and write, public read, and public read and write.
sc	Optional (additional parameter)	Storage classes of the destination objects that can be specified when copying objects. Possible values are:
		• standard : Standard storage class. It features low access latency and high throughput, and is applicable to storing frequently accessed data (multiple accesses per month) or data that is smaller than 1 MB.
		• warm: Infrequent Access storage class. It is ideal for storing infrequently accessed (less than 12 times a year) data, but when needed, the access has to be fast.
		• cold : Archive storage class. It provides secure, durable, and inexpensive storage for rarely-accessed (once a year) data.
meta	Optional (additional parameter)	Metadata of destination objects that can be specified when copying objects. This parameter should be configured in the following format: <i>key1:value1# key2:value2# key3:value3.</i> NOTE The format example above indicates that the destination objects contain three groups of custom metadata: <i>key1:value1, key2:value2</i> , and <i>key3:value3.</i>

Parameter	Optional or Mandatory	Description
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart copy task, in bytes. The value ranges from 100KB to 5GB. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source object size.
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the subfolder in the home directory of the user who executes obsutil commands. NOTE A part record is generated during a multipart copy and saved to the copy subfolder. After the copy succeeds, its part record is deleted automatically. If the copy fails or is suspended, the system attempts to resume the task according to its part record when you perform the copy the next time.
f	Optional (additional parameter)	Runs in force mode.
j	Optional (additional parameter)	Indicates the maximum number of concurrent tasks for copying objects in batches. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.

Parameter	Optional or Mandatory	Description
exclude	Optional (additional parameter)	Indicates the matching patterns of source objects that are excluded, for example: *.txt .
		NOTE
		 The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt.
		 You can use * to represent * and \? to represent ?.
		 If the name of the object to be copied matches the value of this parameter, the object is skipped.
		NOTICE
		 You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows.
		 The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs:// bucket/src1/src2/test.txt, then the absolute path of the object is src1/src2/test.txt.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
		 Multiple exclude parameters can be specified, for example, -exclude=*.xxx - exclude=*.xxx.

Parameter	Optional or Mandatory	Description
include	Optional (additional parameter)	Indicates the matching patterns of source objects that are included, for example: *.jpg . NOTE
		 The asterisk (*) represents any group of characters, and the question mark (?) represents any single character.
		 You can use * to represent * and \? to represent ?.
		 Only after identifying that the name of the file to be copied does not match the value of exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is copied. If not, the file is skipped.
		NOTICE
		 You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows.
		 The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs:// bucket/src1/src2/test.txt, then the absolute path of the object is src1/src2/test.txt.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
		 Multiple include parameters can be specified, for example, -include=*.xxx - include=*.xxx.

Parameter	Optional or Mandatory	Description
timeRange	Optional (additional parameter)	Indicates the time range matching pattern when copying objects. Only objects whose latest modification time is within the configured time range are copied.
		This pattern has a lower priority than the object matching patterns (exclude / include). That is, the time range matching pattern is executed after the configured object matching patterns.
		NOTE
		• The matching time range is represented in <i>time1-time2</i> , where <i>time1</i> must be earlier than or the same as <i>time2</i> . The time format is <i>yyyyMMddHHmmss</i> .
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		 If this parameter is set to *-<i>time2</i>, all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i>- *, all files whose latest modification time is later than <i>time1</i> are matched.
		NOTICE
		 Time in the matching pattern is the UTC time.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on objects whose names end with a slash (/).

Parameter	Optional or Mandatory	Description
0	Optional (additional parameter)	 Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output, the subfolder in the home directory of the user who executes obsutil commands. NOTE The naming rule for result lists is as follows: cp_{succeed failed warning}_report_time_TaskId.txt. By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file. If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list cp_failed_report_time_TaskId.txt in the result list folder and the log files in the log path.
clear	Optional (additional parameter)	Deletes the failure result files after the copy task is resumed.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Refer to **Response** for uploading an object.

5.17 Resuming a Failed Download Task

Function

You can use this command to resume a failed download task based on the task ID.

Command Line Structure

• In Windows

In Linux or macOS

obsutil cp -recover=xxx [-dryRun] [-tempFileDir=xxx] [-f] [-u] [-vlength] [-vmd5] [-j=1] [-p=1] [threshold=52428800] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [o=xxx] [-cpd=xxx] [-clear] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

./obsutil cp -recover=xxx [-dryRun] [-tempFileDir=xxx] [-f] [-u] [-vlength] [-vmd5] [-j=1] [-p=1] [threshold=52428800] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-

Examples

 Take the Windows OS as an example. Run the obsutil cp recover=3066a4b0-4d21-4929-bb84-4829c32cbd0f d:\ -f -r command to download objects in batches. obsutil cp -recover=3066a4b0-4d21-4929-bb84-4829c32cbd0f -f -r Start at 2024-10-08 01:10:07.3809685 +0000 UTC

o=xxx] [-cpd=xxx] [-clear] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Parallel: 5 Jobs: 5 Threshold: 50.00MB PartSize: auto VerifyLength: false VerifyMd5: false CheckpointDir: xxxx

Task id: a628d6da-c562-4a1f-b687-4fa125de0dc3 OutputDir: xxxx TempFileDir: xxxx

[======] 100.00% tps:35.71 2.02 KB/s 7.20MB/7.20MB 0s Succeed count: 1 Failed count: 0 Succeed bytes: xxx Metrics [max cost:90 ms, min cost:45 ms, average cost:63.80 ms, average tps:35.71, transferred size:70B]

Task id: a628d6da-c562-4a1f-b687-4fa125de0dc3

Parameter	Optional or Mandatory	Description
recover	Mandatory (additional parameter)	 ID of the download task to be resumed NOTE You can obtain the task ID after a download task is complete, or query it based on the name of the result list. A task ID is the 36 characters, excluding the suffix .txt. You can locate the download task to be resumed in the directory where the result lists reside. For details about the directory of the result lists, see additional parameter o.
tempFileDir	Optional (additional parameter)	 Indicates the directory for storing temporary files during download. The default value is the value of defaultTempFileDir in the configuration file. NOTE Temporary files generated during multipart download are stored in this directory. Therefore, ensure that the user who executes obsutil has the write permission on the path. The available space of the partition where the path is located must be greater than the size of the objects to be downloaded.
dryRun	Optional (additional parameter)	Conducts a dry run.
u	Optional (additional parameter)	Indicates incremental download. If this parameter is set, each object can be downloaded only when it does not exist in the local path, its size is different from the namesake one in the local path, or it has the latest modification time. CAUTION When you compare each local file with data in the bucket, a billable HEAD request is generated. For details, see Requests .
vlength	Optional (additional parameter)	Checks whether the sizes of the local files are the same as those of the objects in the bucket after the download is complete.

Parameter	Optional or Mandatory	Description
vmd5	Optional (additional parameter)	Checks whether MD5 values of the local files are the same as those of the objects in the bucket after the download is complete. NOTE Objects in the bucket must contain metadata x- obs-meta-md5chksum, or MD5 verification will be skipped. CAUTION If your object needs encryption, do not use this parameter.
р	Optional (additional parameter)	Indicates the maximum number of concurrent multipart download tasks when downloading an object. The default value is the value of defaultParallels in the configuration file.
threshold	Optional (additional parameter)	 Indicates the threshold for enabling multipart download, in bytes. The default value is the value of defaultBigfileThres- hold in the configuration file. NOTE If the size of the object to be downloaded is smaller than the threshold, download the object directly. If not, a multipart download is required. If you download an object directly, no part record is generated, and resumable transmission is not supported. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart download task, in bytes. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source object size.

Parameter	Optional or Mandatory	Description
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the subfolder in the home directory of the user who executes obsutil commands.
		NOTE A part record is generated during a multipart download and saved to the down subfolder. After the download succeeds, its part record is deleted automatically. If the download fails or is suspended, the system attempts to resume the task according to its part record when you perform the download the next time.
f	Optional (additional parameter)	Runs in force mode.
j	Optional (additional parameter)	Indicates the maximum number of concurrent tasks for downloading objects in batches. The default value is the value of defaultJobs in the configuration file.
		NOTE The value is ensured to be greater than or equal to 1.

Parameter	Optional or Mandatory	Description
exclude	Optional (additional parameter)	Indicates the matching patterns of source objects that are excluded, for example: *.txt .
		NOTE
		 The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt.
		 You can use * to represent * and \? to represent ?.
		 If the name of the object to be downloaded matches the value of this parameter, the object is skipped.
		NOTICE
		 You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows.
		 The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs:// bucket/src1/src2/test.txt, then the absolute path of the object is src1/src2/test.txt.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
		 Multiple exclude parameters can be specified, for example, -exclude=*.xxx - exclude=*.xxx.

Parameter	Optional or Mandatory	Description
include	Optional (additional parameter)	Indicates the matching patterns of source objects that are included, for example: *.jpg. NOTE
		 The asterisk (*) represents any group of characters, and the question mark (?) represents any single character.
		 You can use * to represent * and \? to represent ?.
		• Only after identifying that the name of the file to be downloaded does not match the value of exclude , the system checks whether the file name matches the value of this parameter. If yes, the file is downloaded. If not, the file is skipped.
		NOTICE
		 You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows.
		 The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs:// bucket/src1/src2/test.txt, then the absolute path of the object is src1/src2/test.txt.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
		 Multiple include parameters can be specified, for example, -include=*.xxx - include=*.xxx.

Parameter	Optional or Mandatory	Description
timeRange	Optional (additional parameter)	Indicates the time range matching pattern when downloading objects. Only objects whose latest modification time is within the configured time range are downloaded.
		This pattern has a lower priority than the object matching patterns (exclude / include). That is, the time range matching pattern is executed after the configured object matching patterns.
		NOTE
		• The matching time range is represented in <i>time1-time2</i> , where <i>time1</i> must be earlier than or the same as <i>time2</i> . The time format is <i>yyyyMMddHHmmss</i> .
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		 If this parameter is set to *-<i>time2</i>, all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i>- *, all files whose latest modification time is later than <i>time1</i> are matched.
		NOTICE
		• Time in the matching pattern is the UTC time.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on objects whose names end with a slash (/).

Parameter	Optional or Mandatory	Description
0	Optional (additional parameter)	 Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output, the subfolder in the home directory of the user who executes obsutil commands. NOTE The naming rule for result lists is as follows: cp_{succeed failed warning}_report_time_TaskId.txt. By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file. If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list cp_failed_report_time_TaskId.txt in the result list folder and the log files in the log path.
clear	Optional (additional parameter)	Deletes the failure result files after the download task is resumed.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Refer to **Response** for uploading an object.

5.18 Listing Multipart Upload Tasks

Function

You can use this command to query multipart upload tasks in a bucket.

Command Line Structure

- In Windows
 obsutil ls obs://bucket[/prefix] [-s] [-d] -m [-a] [-uploadIdMarker=xxx] [-marker=xxx] [-limit=1] [format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
- In Linux or macOS

 /obsutil ls obs://bucket[/prefix] [-s] [-d] -m [-a] [-uploadIdMarker=xxx] [-marker=xxx] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx] [-t=xxx]

Examples

 Take the Windows OS as an example. Run the obsutil ls obs://bucket-test m -limit=10 command to query the multipart upload tasks in the bucket. obsutil ls obs://bucket-test -m -limit=10

Listing multipart uploads.

Upload list: Key Initiated StorageClass UploadId obs://bucket-test/aaa 2018-11-27T03:49:07Z standard 000001675348ED21860C3F61EF955BD3

obs://bucket-test/dir1/10GB.txt 2018-11-07T06:58:09Z standard 00000166ECF6CF7C860D1DBAF3F76013

obs://bucket-test/dir1/1GB.txt 2018-11-07T06:58:09Z standard 00000166ECF6CF6F860B7FBE95D01B03

obs://bucket-test/dir1/50GB.txt 2018-11-07T06:58:09Z standard 00000166ECF6CF86860D1DC2C8E8F66B

obs://bucket-test/dir1/5GB.txt 2018-11-07T06:58:09Z standard 00000166ECF6CF75860CDA7780CB52C3

obs://bucket-test/test11/20GB.txt 2018-11-27T08:21:26Z standard 0000016754423D24860CA8A4D06C2054

Folder number: 0 Upload number: 6

• For more examples, see Listing Multipart Upload Tasks.

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name

Parameter	Optional or Mandatory	Description
prefix	Optional	Object name prefix for listing multipart uploads NOTE If this parameter is left blank, all multipart upload tasks in the bucket are listed.
S	Optional (additional parameter)	Displays simplified query result. NOTE In the simplified format, the returned result contains only the object name and upload ID of the multipart upload.
d	Optional (additional parameter)	Lists only the multipart upload tasks and sub-directories in the current directory are listed, instead of recursively listing all the multipart upload tasks and sub-directories.
m	Mandatory (additional parameter)	Lists multipart upload tasks in the bucket.
а	Optional (additional parameter)	Lists the objects and the multipart upload tasks in the bucket.
marker	Optional (additional parameter)	Indicates the upload ID after which the multipart upload listing begins. All returned multipart upload tasks are listed in lexicographical order by object name involved in the tasks. NOTE For details about how to use this parameter, see Listing Multipart Upload Tasks .
uploadIdMa rker	Optional (additional parameter)	Indicates the upload ID after which the multipart upload listing begins. This parameter must be used together with marker . All returned multipart upload tasks are listed in lexicographical order by object name and upload ID involved in the tasks.
limit	Optional (additional parameter)	Maximum number of objects that can be listed. If the value is less than or equal to 0 , all objects are listed. NOTE If there are a large number of multipart upload tasks in a bucket, you are advised to set this parameter to limit the number of multipart upload tasks each time. If not all tasks are listed, marker and uploadIdMarker of the next request will be returned in the result, which you can use to list the remaining tasks.

Parameter	Optional or Mandatory	Description
format	Optional (additional parameter)	Prints a listing result in the user-defined format. Currently, the value can only be default , indicating that the listing result is displayed in one row.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

5.19 Deleting a Multipart Upload Task

Function

- You can use this command to delete a multipart upload task in a specified bucket by using the multipart upload ID.
- You can also use this command to delete multipart upload tasks in batches based on a specified object name prefix.

Command Line Structure

- In Windows
 - Deleting a single multipart upload task obsutil abort obs://bucket/key -u=xxx [-f] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
 - Deleting multipart upload tasks in batches
 obsutil abort obs://bucket[/key] -r [-f] [-o=xxx] [-j=1] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
 t=xxx]
- In Linux or macOS
 - Deleting a single multipart upload task
 ./obsutil abort obs://bucket/key -u=xxx [-f] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Deleting multipart upload tasks in batches
 ./obsutil abort obs://bucket[/key] -r [-f] [-o=xxx] [-j=1] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

Examples

• Take the Windows OS as an example. Run the **obsutil abort obs://bucket-test/key -u=xxx -f** command to delete a single multipart upload task. obsutil abort obs://bucket-test/key -u=xxx -f

Start at 2024-10-08 01:25:55.6771288 +0000 UTC

[------] 100.00% tps:0.00 1/1 106ms Succeed count: 1 Failed count: 0 Metrics [max cost:54 ms, min cost:54 ms, average cost:54.00 ms, average tps:8.77]

Task id: 4972589c-c775-41be-a288-bbee3edaaee9

 Take the Windows OS as an example. Run the obsutil abort obs://buckettest -r -f command to delete all multipart upload tasks in the bucket in batches.

obsutil abort obs://bucket-test -r -f Start at 2024-10-08 01:28:29.1980739 +0000 UTC

[------] 100.00% tps:2924.55 3/3 202ms Succeed count: 3 Failed count: 0 Metrics [max cost:148 ms, min cost:61 ms, average cost:113.33 ms, average tps:14.63]

Task id: cd2fd08e-fc31-47d9-b4b0-9f9a3376435f

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Bucket name
key	Mandatory for deleting a multipart upload task. Optional for deleting multipart upload tasks in batches.	Indicates the object name involved in a multipart upload task to be deleted, or the name prefix of the objects involved in multipart upload tasks to be deleted in batches. NOTE If this parameter is left blank when deleting multipart upload tasks in batches, all multipart upload tasks in the bucket are deleted.
u	Mandatory for deleting a single multipart upload task (additional parameter)	ID of the multipart upload task to be deleted NOTE You can obtain the value of this parameter from Listing Multipart Upload Tasks.
fr	Optional for deleting a single multipart upload task (additional parameter)	Generates an operation result list when deleting a multipart upload task.

Parameter	Optional or Mandatory	Description	
f	Optional (additional parameter)	Runs in force mode.	
r	Mandatory for deleting multipart upload tasks (additional parameter)	Deletes multipart upload tasks in batches based on a specified object name prefix.	
j	Optional for deleting multipart upload tasks (additional parameter)	Indicates the maximum number of concurrent tasks for deleting multipart uploads in batches. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.	
0	Optional (additional parameter)	 Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success and failure files) are generated in the folder. The default value is .obsutil_output, the subfolder in the home directory of the user who executes obsutil commands. NOTE The naming rule for result lists is as follows: abort_{succed failed}_report_time_TaskId.txt By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file. 	
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .	
е	Optional (additional parameter)	Specifies the endpoint.	
i	Optional (additional parameter)	Specifies the user's AK.	
k	Optional (additional parameter)	Specifies the user's SK.	
t	Optional (additional parameter)	Specifies the user's security token.	

Refer to **Response** for uploading an object.

5.20 Creating an Authorization Code for Directory Sharing

Function

You can use this command to specify the bucket name, object name prefix, and access code to create an authorization code for directory sharing.

Command Line Structure

- In Windows
 - obsutil create-share obs://bucket[/prefix] [-ac=xxx] [-vp=xxx] [-dst=xxx] [-config=xxx] [-e=xxx] [i=xxx] [-k=xxx] [-t=xxx]
- In Linux or macOS
 ./obsutil create-share obs://bucket[/prefix] [-ac=xxx] [-vp=xxx] [-dst=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Examples

 In Windows, you can run the obsutil create-share obs://bucket/test/ ac=123456 -vp=1m command to create an authorization code that is valid within one month. obsutil create-share obs://bucket/test/ -ac=123456 -vp=1m

Access Code: 123456

Valid Until: Sat, 26 Oct 2019 11:28:10 GMT +8000

Parameter	Optional or Mandatory	Description
bucket	Mandatory	Name of a bucket

Parameter	Optional or Mandatory	Description	
prefix	Optional	Prefix of an object name. If this parameter is specified, objects starting with this prefix are shared. If this parameter is left blank, all objects in the bucket are shared. NOTE It is recommended that the value end with a slash (/).	
ас	Optional (additional parameter)	 Access code. NOTE If no access code is passed using this parameter, obsutil tool prompts you to enter the access code in interactive mode. An access code is a six-digit string. 	
νр	Optional (additional parameter)	 Validity period of an authorization code. The default value is one day, indicating that the generated authorization code is valid for only one day. NOTE This parameter supports different time units, including: m (month), w (week), d (day), h (hour), min (minute), and s (second). For example, 1d indicates that the authorization code is valid within one day, 2w indicates that the code is valid within two weeks, and 3h indicates that the code is valid within three hours. The default time unit is s (second), for example, 3600 indicates that the authorization code is valid within 3600 seconds. 	
dst	Optional (additional parameter)	Path for storing the generated authorization code.	
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .	
е	Optional (additional parameter)	Specifies the endpoint.	
i	Optional (additional parameter)	Specifies the user's AK.	

Parameter	Optional or Mandatory	Description
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Field	Description
Authorization Code	Code for authorization
Access Code	Access code
Valid Until	Expiration time

5.21 Listing Objects by Using an Authorization Code

Function

You can use this command to query objects in a bucket with an authorization code. The returned objects are sorted in lexicographical order.

Command Line Structure

- In Windows
 - Enter an authorization code directly.
 obsutil share-ls authorization_code [-ac=xxx] [-prefix=xxx] [-s] [-d] [-marker=xxx] [-bf=xxx] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Use the file path to pass an authorization code. obsutil share-ls file://authorization_code_file_url [-ac=xxx] [-prefix=xxx] [-s] [-d] [-marker=xxx] [-bf=xxx] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
- In Linux or macOS
 - Enter an authorization code directly.
 ./obsutil share-ls authorization_code [-ac=xxx] [-prefix=xxx] [-s] [-d] [-marker=xxx] [-bf=xxx] [-limit=1] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
 - Use the file path to pass an authorization code.
 ./obsutil share-ls file://authorization_code_file_url [-ac=xxx] [-prefix=xxx] [-s] [-d] [-marker=xxx]
 [-bf=xxx] [-limit=1] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Examples

• In Windows, you can run the **obsutil share-ls xxx -ac=123456 -limit=1** command to query objects in a bucket using an authorization code.

obsutil share-ls xxx -ac=123456 -limit=1 The authorized prefix is [test/test.tar.gz] Listing objects . Object list: LastModified key Size StorageClass ETag obs://bucket-test/test/test.tar.gz 2019-07-11T14:50:59Z 48.92KB standard "1dd27294ad2f152b43cd111e9fe3990f"

Total size of prefix [test/]: 48.92KB Folder number: 0 File number: 1 The authorized prefix is [test/]

In Windows, you can run the obsutil share-ls xxx -ac=123456 -limit=1 • command to query directories in a bucket using an authorization code. obsutil share-ls xxx -ac=123456 -limit=1

The authorized prefix is [test]

Listing objects .

Folder list: obs://bucket-test/test/

Object list: key

LastModified Size StorageClass ETag obs://bucket-test/test/test.tar.gz "1dd27294ad2f152b43cd111e9fe3990f" 2019-07-11T14:50:59Z 48.92KB standard

Total size of prefix [test/]: 48.92KB Folder number: 1 File number: 1 The authorized prefix is [test/]

Parameter	Optional or Mandatory	Description
authorizatio n_code or file:// authorizatio n_code_file_ url	Mandatory	Code for authorization NOTE If the authorization code starts with file:// , the authorization code is obtained from a local file.
ac	Optional (additional parameter)	 Access code NOTE If no access code is specified using this parameter, obsutil tool prompts you to enter the access code in interactive mode. An access code is a six-digit string.

Parameter	Optional or Mandatory	Description
prefix	Optional (additional parameter)	 Object name prefix specified for listing objects using an authorization code. NOTE If this parameter is specified, objects starting with this prefix are listed. If this parameter is left blank, all objects in the authorized path are shared.
S	Optional (additional parameter)	Displays simplified query result. NOTE In the simplified format, the returned result contains only the object name.
d	Optional (additional parameter)	Lists only objects and subdirectories in the current directory, instead of recursively listing all objects and subdirectories. NOTE According to the naming conventions in OBS, a slash (/) is used as the directory separator.
marker	Optional (additional parameter)	Object name to start with when listing objects in a bucket using an authorization code. All objects are listed in lexicographical order by object name.
bf	Optional (additional parameter)	 Display formats of bytes in the listing result. Possible values are: human-readable raw NOTE If this parameter is not configured, the display format of bytes in the result is determined by the humanReadableFormat parameter in the configuration file.
limit	Optional (additional parameter)	Maximum number of objects that can be listed. If the value is less than or equal to 0, all objects are listed. If it is left blank, 1000 objects are listed by default. NOTE If there are a large number of objects in a bucket, you are advised to set this parameter to limit the number of objects to be listed each time. If not all objects are listed, marker of the next request will be returned in the result, which you can use to list the remaining objects.
format	Optional (additional parameter)	Prints a listing result in the user-defined format. Currently, the value can only be default , indicating that the listing result is displayed in one row.

Parameter	Optional or Mandatory	Description
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

5.22 Downloading Objects by Using an Authorization Code

Function

You can use this command to download an object or download objects in a batch by object name prefix to your local PC.

NOTICE

Do not change the source objects in the OBS bucket when downloading a single object or objects in batches. Otherwise, the download may fail or data may be inconsistent.

Command Line Structure

- In Windows
 - Enter the authorization code to download a single object. obsutil share-cp authorization_code file_or_folder_url -key=xxx [-ac=xxx] [-dryRun] [tempFileDir=xxx] [-u] [-vlength] [-vmd5] [-p=1] [-threshold=52428800] [-ps=auto] [-cpd=xxx][fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
 - Use the file path to transfer the authorization code and download a single object.

obsutil share-cp file://authorization_code_file_url file_or_folder_url -key=xxx [-ac=xxx] [-dryRun] [-tempFileDir=xxx] [-u] [-vlength] [-vmd5] [-p=1] [-threshold=52428800] [-ps=auto] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

- Enter the authorization code to download objects in a batch. obsutil share-cp authorization_code folder_url -r [-key=xxx] [-ac=xxx] [-dryRun] [tempFileDir=xxx] [-f] [-u] [-vlength] [-vmd5] [-flat] [-j=1] [-p=1] [-threshold=52428800] [ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
- Use the file path to transfer the authorization code and download objects in a batch.
 obsutil share-cp file://authorization_code_file_url folder_url -r [-key=xxx] [-ac=xxx] [-dryRun] [-

tempFileDir=xxx] [-f] [-u] [-vlength] [-vmd5] [-flat] [-j=1] [-p=1] [-threshold=52428800] [ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]

• In Linux or macOS

- Enter the authorization code to download a single object.
 ./obsutil share-cp authorization_code file_or_folder_url -key=xxx [-ac=xxx] [-dryRun] [-tempFileDir=xxx] [-u] [-vlength] [-vmd5] [-p=1] [-threshold=52428800] [-ps=auto] [-cpd=xxx][-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx]
- Use the file path to transfer the authorization code and download a single object.
 ./obsutil share-cp file://authorization_code_file_url file_or_folder_url -key=xxx [-ac=xxx] [-dryRun] [-tempFileDir=xxx] [-u] [-vlength] [-vmd5] [-p=1] [-threshold=52428800] [-ps=auto] [-cpd=xxx][-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
- Enter the authorization code to download objects in a batch.
 ./obsutil share-cp authorization_code folder_url -r [-key=xxx] [-dc=xxx] [-dryRun] [-tempFileDir=xxx] [-f] [-u] [-vlength] [-vmd5] [-flat] [-j=1] [-p=1] [-threshold=52428800] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-t=xxx]
- Use the file path to transfer the authorization code and download objects in a batch.

./obsutil share-cp file://authorization_code_file_url folder_url -r [-key=xxx] [-ac=xxx] [-dryRun] [tempFileDir=xxx] [-f] [-u] [-vlength] [-vmd5] [-flat] [-j=1] [-p=1] [-threshold=52428800] [ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Examples

 In Windows, you can run the obsutil share-cp xxx d:\temp\test.txt ac=123456 -key=src/test.txt command to download a single object. obsutil share-cp xxx d:\temp\test.txt -ac=123456 -key=test/test.txt The authorized prefix is [test/test.txt]

Start at 2024-09-30 07:55:36.7698045 +0000 UTC

Parallel: 3 Jobs: 3 Threshold: 50.00MB PartSize: auto VerifyLength: false VerifyMd5: false CheckpointDir: xxxx TempFileDir: xxxx Waiting to prepare the temp file [xxx]. Download successfully, 19B, n/a, https://endpoint:443/test/test.txt --> d:\temp\test.txt The authorized prefix is [test/test.txt]

In Windows, you can run the obsutil share-cp xxx d:\temp -ac=123456 -f -r command to download objects in a batch.
 obsutil share-cp xxx d:\temp -ac=123456 -f -r The authorized prefix is [test]

Start at 2024-09-30 08:05:07.0097508 +0000 UTC

Parallel: 3	Jobs:	3	
Threshold: 50.0	OMB	PartSize:	auto
VerifyLength: fals	se V	erifyMd5:	false
CheckpointDir: xx	xx	-	

Task id: 1a50b1dd-3f92-42ae-a974-ff8fe514c2c2 OutputDir: xxxx TempFileDir: xxxx [=======] 100.00% 155.59 KB/s 0s Succeed count: 6 Failed count: 0 Succeed bytes: xxx Metrics [max cost:153 ms, min cost:129 ms, average cost:92.00 ms, average tps:17.86, transferred size:70B] Task id: 1a50b1dd-3f92-42ae-a974-ff8fe514c2c2 The authorized prefix is [test/]

Parameter	Optional or Mandatory	Description
authorizatio n_code or file:// authorizatio n_code_file_ url	Mandatory	Code for authorization NOTE If the authorization code starts with file:// , the authorization code is obtained from a local file.
file_or_folde r_url	Mandatory for downloading an object	Local file/folder path
folder_url	Mandatory for downloading objects in a batch	Local folder path

Parameter	Optional or Mandatory	Description
key	Mandatory for downloading an object	Indicates the name of the object to be downloaded, or the name prefix of the objects to be downloaded in batches.
	(additional parameter) Optional for	This parameter cannot be left blank when downloading an object. The saving and naming rules are as follows:
	downloading objects in a batch	 If this parameter specifies a file or folder path that does not exist, the tool checks whether the value ends with a slash (/) or backslash (\). If yes, a folder is created based on the path, and the object is downloaded to this newly created directory.
		 If this parameter specifies a file or folder path that does not exist and the value does not end with a slash (/) or backslash (\), the object is downloaded to your local PC with the value of the parameter as the file name.
		• If this parameter specifies an existing file, the object is downloaded to your local PC overwriting the existing file, with the value of the parameter as the file name.
		 If this parameter specifies an existing folder, the object is downloaded to the directory specified by file_or_folder_url with the object name as the file name.
		During batch download, the saving rules are as follows:
		 If this parameter is left blank, all objects in the authorized path specified in the authorization code are downloaded to the folder specified by folder_url.
		 If this parameter is configured, objects whose name prefix is the configured value in the bucket are downloaded to the directory specified by folder_url.
		NOTEIf this parameter is specified, objects starting with
		 this prefix are listed. If this parameter is configured but the flat parameter is not configured when downloading objects in a batch, the name of the downloaded file contains the name prefix of the parent object. If flat is configured, then the name of the downloaded file does not contain the name prefix of the parent object.

Parameter	Optional or Mandatory	Description
		NOTICE During batch download, if the flat option is selected and the object name prefix is empty or does not end with a slash (/) during authorization code creation, the downloaded object list may be empty.
ac	Optional (additional parameter)	 Access code NOTE If no access code is specified using this parameter, obsutil tool prompts you to enter the access code in interactive mode. An access code is a six-digit string.
r	Mandatory for downloading objects in batches (additional parameter)	Copies objects in batches based on a specified object name prefix.
fr	Optional for downloading an object (additional parameter)	Generates an operation result list when downloading an object.
flat	Optional for downloading objects in batches (additional parameter)	The name prefix of the parent object is excluded when downloading objects in a batch.
tempFileDir	Optional (additional parameter)	 Indicates the directory for storing temporary files during multipart download. The default value is the value of defaultTempFileDir in the configuration file. NOTE If this parameter is left blank and the defaultTempFileDir parameter in the configuration file is also left blank, temporary files generated during multipart download are saved in the directory where to-be-downloaded files are located and end with the suffix of .obs.temp. Temporary files generated during multipart download are stored in this directory. Therefore, ensure that the user who executes obsutil has the write permission on the path. The available space of the partition where the path is located must be greater than the size of the objects to be downloaded.

Parameter	Optional or Mandatory	Description
dryRun	Optional (additional parameter)	Conducts a dry run.
u	Optional (additional parameter)	Indicates incremental download. If this parameter is set, each object can be downloaded only when it does not exist in the local path, its size is different from the namesake one in the local path, or it has the latest modification time. CAUTION When you compare each local file with data in the bucket, a billable HEAD request is generated. For details, see Requests .
vlength	Optional (additional parameter)	Checks whether the sizes of the local files are the same as those of the objects in the bucket after the download is complete.
vmd5	Optional (additional parameter)	Checks whether MD5 values of the local files are the same as those of the objects in the bucket after the download is complete. NOTE Objects in the bucket must contain metadata x-obs- meta-md5chksum , or MD5 verification will be skipped. CAUTION If your object needs encryption, do not use this parameter.
р	Optional (additional parameter)	Indicates the maximum number of concurrent multipart download tasks when downloading an object. The default value is the value of defaultParallels in the configuration file.
threshold	Optional (additional parameter)	 Indicates the threshold for enabling multipart download, in bytes. The default value is the value of defaultBigfileThreshold in the configuration file. NOTE If the size of the object to be downloaded is smaller than the threshold, download the object directly. If not, a multipart download is required. If you download an object directly, no part record is generated, and resumable transmission is not supported. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).

Parameter	Optional or Mandatory	Description
ps	Optional (additional parameter)	 Indicates the size of each part in a multipart download task, in bytes. The default value is the value of defaultPartSize in the configuration file. NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). The parameter can be set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source object size.
cpd	Optional (additional parameter)	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the subfolder in the home directory of the user who executes obsutil commands. NOTE A part record is generated during a multipart download and saved to the down subfolder. After the download succeeds, its part record is deleted automatically. If the download fails or is suspended, the system attempts to resume the task according to its part record when you perform the download the next time.
f	Optional for downloading objects in a batch (additional parameter)	Runs in force mode.
j	Optional for downloading objects in a batch (additional parameter)	Indicates the maximum number of concurrent tasks for downloading objects in a batch. The default value is the value of defaultJobs in the configuration file. NOTE The value is ensured to be greater than or equal to 1.

Parameter	Optional or Mandatory	Description
exclude	exclude Optional for downloading objects in a batch (additional parameter)	 Indicates the matching patterns of source objects that are excluded, for example: *.txt. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. For instance, abc*.txt indicates any file whose name starts with abc and ends with .txt.
		 You can use * to represent * and \? to represent ?. If the name of the object to be downloaded matches the value of this parameter, the object is skipped.
		NOTICE
		 You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows.
	 The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs://bucket/src1/src2/test.txt, then the absolute path of the object is src1/src2/test.txt. 	
		 This matching pattern applies only to objects whose names do not end with a slash (/).
		 Multiple exclude parameters can be specified, for example, -exclude=*.xxx -exclude=*.xxx.

Parameter	Optional or Mandatory	Description
include	Optional for downloading objects in a batch (additional parameter)	 Indicates the matching patterns of source objects that are included, for example: *.jpg. NOTE The asterisk (*) represents any group of characters, and the question mark (?) represents any single character. You can use * to represent * and \? to represent ?. Only after identifying that the name of the file to be downloaded does not match the value of exclude, the system checks whether the file name matches the value of this parameter. If yes, the file is downloaded. If not, the file is skipped. NOTICE You are advised to use quotation marks for the matching pattern to prevent special characters from being escaped by the OS and leading to unexpected results. Use single quotation marks for Linux or macOS and quotation marks for Windows. The matching pattern applies to the absolute path of an object, including the object name prefix and object name starting from the root directory. For example, if the path of an object in the bucket is obs://bucket/src1/src2/test.txt. This matching pattern applies only to objects whose names do not end with a slash (/). Multiple include parameters can be specified, for example, -include=*.xxx.

Parameter	Optional or Mandatory	Description
timeRange	Optional for downloading objects in a batch (additional parameter)	Indicates the time range matching pattern when downloading objects. Only objects whose latest modification time is within the configured time range are downloaded.
		This pattern has a lower priority than the object matching patterns (exclude/include). That is, the time range matching pattern is executed after the configured object matching patterns. NOTE
		 The matching time range is represented in <i>time1-time2</i>, where <i>time1</i> must be earlier than or the same as <i>time2</i>. The time format is <i>yyyyMMddHHmmss</i>.
		 Automatic formatting is supported. For example, yyyyMMdd is equivalent to yyyyMMdd000000, and yyyyMM is equivalent to yyyyMM01000000.
		• If this parameter is set to *- <i>time2</i> , all files whose latest modification time is earlier than <i>time2</i> are matched. If it is set to <i>time1</i> -*, all files whose latest modification time is later than <i>time1</i> are matched.
		NOTICE
		• Time in the matching pattern is the UTC time.
		 This matching pattern applies only to objects whose names do not end with a slash (/).
mf	Optional (additional parameter)	Indicates that the name matching pattern (include or exclude) and the time matching pattern (timeRange) also take effect on objects whose names end with a slash (/).

Parameter	Optional or Mandatory	Description
0	Optional (additional parameter)	Indicates the folder where operation result lists reside. After the command is executed, result lists (possibly including success, failure, and warning files) are generated in the folder. The default value is .obsutil_output , the subfolder in the home directory of the user who executes obsutil commands. NOTE • The naming rule for result lists is as follows: share- cp_{succeed failed
		 warning}_report_time_TaskId.txt By default, the maximum size of a single result list is 30 MB and the maximum number of result lists that can be retained is 1024. You can set the maximum size and number by configuring recordMaxLogSize and recordBackups in the configuration file.
		• If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list in the result list folder and the log files in the log path.
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

Refer to **Response** for uploading an object.

6 Auxiliary Commands

6.1 Updating a Configuration File

Function

You can update items in the **.obsutilconfig** file, including the endpoint, AK, SK, and token.

Configuration Parameters describes parameters in the **.obsutilconfig** file.

Command Line Structure

- In Windows
 - Updating settings in interactive mode obsutil config [-interactive] [-crr] [-config=xxx]
 - Directly updating settings
 obsutil config [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] [-crr] [-config=xxx]
- In Linux or macOS
 - Updating settings in interactive mode ./obsutil config [-interactive] [-crr] [-config=xxx]
 - Directly updating settings
 ./obsutil config [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] [-crr] [-config=xxx]

NOTE

• After you use -i, -k, and -e to configure user authentication, the history command in Linux or other OSs can query the configured values.

Examples

• Take Windows as an example. Run the **obsutil config -interactive** command to update the access keys and OBS endpoint in the default configuration file. **obsutil config -interactive**

 xxxxxxxxxxxxxxxxxxxxxxxxxxx Please input your token: xxxxxxxxxxxxxxxxxxxxxxxxxxxxx Config file url: C:\Users\tools\.obsutilconfig

Update config file successfully!

• Take Linux as an example. Run the ./obsutil config -interactive command to update the access keys and OBS endpoint in the default configuration file. ./obsutil config -interactive

Update config file successfully!

Parameter	Optional or Mandatory	Description
interactive	Optional (additional parameter)	Updates settings in interactive mode. NOTICE For security purposes, you are advised to use the interactive mode to configure the AK and SK.
e	Optional (additional parameter)	Changes the value of endpoint or endpointCrr . For details, see Configuration Parameters .
i	Optional (additional parameter)	Changes the value of ak or akCrr . For details, see Configuration Parameters .
k	Optional (additional parameter)	Changes the value of sk or skCrr . For details, see Configuration Parameters .
t	Optional (additional parameter)	Changes the value of token or tokenCrr . For details, see Configuration Parameters .
crr	Optional (additional parameter)	Updates the settings of client-side cross- region replication.
		NOTE If this parameter is not specified, endpoint , ak , sk , and token will be updated by the e , i , k , and t parameters.
		If this parameter is specified, endpointCrr , akCrr , skCrr , and tokenCrr will be updated by the e , i , k , and t parameters.

Parameter	Optional or Mandatory	Description
config	Optional (additional parameter)	User-defined configuration file for executing the current command. To learn the parameters that can be configured in this file, see Configuration Parameters .
		By specifying this parameter and a path, you can update parameters in the user- defined configuration file. Otherwise, parameters in the default configuration file will be updated.

6.2 Deleting Part Records

Function

You can use this command to delete part records from a specified directory.

Command Line Structure

- In Windows
 obsutil clear [checkpoint_dir] [-u] [-d] [-c] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
- In Linux or macOS ./obsutil clear [checkpoint_dir] [-u] [-d] [-c] [-config=xxx] [-e=xxx] [-i=xxx] [-t=xxx] [-t=xxx]

Examples

• Take the Windows OS as an example. Run the **obsutil clear** -**u** command to delete the part records of multipart upload tasks in the default directory. **obsutil clear** -**u**

Parameter	Optional or Mandatory	Description
checkpoint_ dir	Optional	Indicates the folder where the part records reside. The default value is .obsutil_checkpoint , the same subfolder where obsutil commands reside.
u	Optional (additional parameter)	Deletes the part records of all multipart upload tasks. NOTE At the same time, the system attempts to delete the multipart upload tasks in the part records.

Parameter	Optional or Mandatory	Description
d	Optional (additional parameter)	Deletes the part records of all multipart download tasks. NOTE At the same time, the system attempts to delete the fragments in the part records.
с	Optional (additional parameter)	Deletes the part records of all multipart copy tasks. NOTE At the same time, the system attempts to delete the multipart copy tasks in the part records.
config	Optional (additional parameter)	User-defined configuration file for executing a command. For details about parameters that can be configured, see Configuration Parameters .
е	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

NOTE

You must configure at least one among the **u**, **d** and **c** parameters.

6.3 Viewing Command Help Information

Function

You can use this command to view the commands supported by obsutil or view the help information of a specific command.

Command Line Structure

• In Windows obsutil help [command] In Linux or macOS
 ./obsutil help [command]

Examples

 Take the Windows OS as an example. Run the **obsutil help mb** command to view the help information about the command for creating a bucket.
 obsutil help mb

Summary:

create a bucket with the specified parameters

Syntax:

obsutil mb obs://bucket [-acl=xxx] [-location=xxx] [-fs] [-az=xxx] [-sc=xxx] [-config=xxx] [-i=xxx] [-k=xxx] [-t=xxx] [-e=xxx]

Options:

-fs create a bucket that supports POSIX

-az=xxx

the AZ of the bucket, possible values are [multi-az]

-sc=xxx

the default storage class of the bucket, possible values are [standard|warm|cold|deep-archive]

-acl=xxx

the ACL of the bucket, possible values are [private|public-read|public-read-write]

- -location=xxx the region where the bucket is located
- -epid=xxx the enterprise project id of the bucket
- -kms=xxx the encryption id of the bucket
- -config=xxx the path to the custom config file when running this command
- -e=xxx endpoint
- -i=xxx access key ID
- -k=xxx security key ID
- -t=xxx
- security token
- Summary: create a bucket with the specified parameters
- Syntax:
- obsutil mb obs://bucket [-fs] [-az=xxx] [-acl=xxx] [-sc=xxx] [-location=xxx] [-config=xxx]
- Options: -fs
 - create a bucket that supports POSIX
- -az=xxx
- the AZ of the bucket, possible values are [multi-az]
- -acl=xxx
- the ACL of the bucket, possible values are [private|public-read|public-read-write]

-sc=xxx

the default storage class of the bucket, possible values are: [standard|warm|cold]

-location=xxx

the region where the bucket is located

-config=xxx

the path to the custom config file when running this command

Parameter Description

Optional or Mandatory	Description
-	DescriptionCurrently, the help documents of the following commands are available:• For abort, see Deleting a Multipart Upload Task.• For chattri, see Setting Object Properties.• For cp, see Uploading an Object,
	Mandatory

6.4 Querying the Version Number

Function

You can use this command to query the current version of obsutil.

Command Line Structure

- In Windows
 obsutil version
- In Linux or macOS ./obsutil version

Examples

Take the Windows OS as an example.

obsutil version

obsutil version:5.1.9, obssdk version:2.2.12 operating system:windows, arch:amd64

6.5 Archiving Log Files

Function

You can use this command to archive log files to a local PC or to a specified bucket.

Command Line Structure

- In Windows
 - Archiving to a local PC obsutil archive [file_or_folder_url] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Archiving to a specified bucket obsutil archive obs://bucket[/key] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
- In Linux or macOS
 - Archiving to a local PC obsutil archive [file_or_folder_url] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
 - Archiving to a specified bucket obsutil archive obs://bucket[/key] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Examples

• Take the Windows OS as an example. Run the **obsutil archive** command to archive log files to the same directory where the tool is executed. **obsutil archive**

[------] 100.00% 15/15 35ms Succeed to archive log files to [D:\obsutil\obsutil_log.zip]

Parameter Description

Parameter	Optional or Mandatory	Description
file_or_fol der_url	Optional	 Indicates the path to which log files are archived. The rules are as follows: If this parameter is left blank, log files are archived to the same directory where obsutil commands reside with obsutil_log.zip as the archive file name. If this parameter specifies a file or folder path that does not exist, the tool checks whether the value ends with a slash (/) or backslash (\). If yes, a folder is created based on the path, and log files are archived to the newly created directory with obsutil_log.zip as the archive file name. If this parameter specifies a file or folder path that does not exist and the value does not end with a slash (/) or backslash (\), log files are archived to a local PC with the value as the archive file name. If this parameter specifies an existing .zip file, then log files are archived to a local PC overwriting the existing file, with the value as the archive file name. If this parameter specifies an existing .zip file, then log files are archived to a local PC overwriting the existing file, with the value as the archive file name. If this parameter specifies an existing .zip file, then log files are archived to a local PC overwriting the existing file, with the value as the archive file name. If this parameter specifies an existing folder, then log files are archived to the specified directory with obsutil_log.zip as the archive file name.
bucket	Mandatory for archiving log files to a specified bucket	Bucket name

Parameter	Optional or Mandatory	Description
key	Optional for archiving log files to a specified bucket	Indicates the object name or object name prefix when archiving log files to a specified bucket.The rules are as follows:If this parameter is left blank, log files are
		archived to the root directory of the bucket with obsutil_log.zip as the object name.
		 If the value ends with a slash (/), the value is used as the object name prefix when archiving log files, and the object name is the value plus obsutil_log.zip. Otherwise, log files are archived with the value as the object name.
config	Optional (additional parameter)	User-defined configuration file for executing a command. For details about parameters that can be configured, see Configuration Parameters .
e	Optional (additional parameter)	Specifies the endpoint.
i	Optional (additional parameter)	Specifies the user's AK.
k	Optional (additional parameter)	Specifies the user's SK.
t	Optional (additional parameter)	Specifies the user's security token.

6.6 Listing Failure Result Files

Function

This command is used to list the last-modified failure result files generated by the **cp** or **sync** commands in a specified folder.

Command Line Structure

- In Windows
 obsutil ls -failed [-limit=1000] [-o=xxx]
- In Linux or macOS ./obsutil ls -failed [-limit=1000] [-o=xxx]

Examples

The following example is based on a Windows OS:

obsutil ls -failed		
FileName	ModTime	
TaskId cp_failed_report_xxx_ccb9f044-7768- CST ccb9f044-7768-4948-ae6d-462		2019-05-26 19:33:13.516813 +0800
sync_failed_report_xxx_59104460-d10 CST 59104460-d10e-42a1-9f76-03		2019-05-27 14:30:19.91126 +0800

Get 2 last modified failed report files.

Parameter Description

Parameter	Optional or Mandatory	Description
failed	Mandatory	 This command is used to list the last-modified failure result files generated by the cp or sync commands in a specified folder. NOTE This command does not support recursive scanning of all files in a specified folder. It can only list failure result files in the root directory of a specified path.
limit	Optional (additional parameter)	Maximum number of objects that can be queried. If the value is less than or equal to 0, all objects are listed. If it is left blank, 1000 objects are listed by default.
0	Optional (additional parameter)	Specifies the folder where the result files are located. If this parameter is not specified, the .obsutil_output subfolder in the user directory for running obsutil commands is used by default.

7 Common Examples

7.1 Upload

All commands in this section use the Linux operating system as an example to describe how to upload files.

Assume that a local folder is in the following structure:

src1 src2 test1.txt test2.txt src3 test3.txt

Based on the preceding folder structure, different upload scenarios require different commands.

 To upload the test3.txt file in the local src1 folder to the root directory of bucket bucket-test, the command is as follows: ./obsutil cp /src1/test3.txt obs://bucket-test

After the upload completes, the following object is generated in the bucket: ./obs://bucket-test/test3.txt

• To upload the **test3.txt** file in the local **src1** folder to the root directory of bucket **bucket-test** and rename it to **aaa.txt**, the command is as follows: ./obsutil cp /src1/test3.txt obs://bucket-test/aaa.txt

After the upload completes, the following object is generated in the bucket: ./obs://bucket-test/aaa.txt

 To upload the test3.txt file in the local src1 folder to the src folder in bucket bucket-test, the command is as follows: ./obsutil cp /src1/test3.txt obs://bucket-test/src/

After the upload completes, the following object is generated in the bucket: ./obs://bucket-test/src/test3.txt

• To recursively upload the entire local **src2** folder to the root directory of bucket **bucket-test** in force mode, the command is as follows: ./obsutil cp /src1/src2 obs://bucket-test -r -f

After the upload completes, the following objects are generated in the bucket:

obs://bucket-test/src2/ obs://bucket-test/src2/test1.txt obs://bucket-test/src2/test2.txt

 To recursively upload the entire local src1 folder to the src folder in bucket bucket-test in force mode, the command is as follows: ./obsutil cp /src1 obs://bucket-test/src -r -f

After the upload completes, the following objects are generated in the bucket:

obs://bucket-test/src/src1/ obs://bucket-test/src/src1/src2/ obs://bucket-test/src/src1/src2/test1.txt obs://bucket-test/src/src1/src2/test2.txt obs://bucket-test/src/src1/src3/ obs://bucket-test/src/src1/test3.txt

• To recursively upload the all files and subfolders in the local **src1** folder to the **src** folder in bucket **bucket-test** in force mode, the command is as follows: ./obsutil cp /src1 obs://bucket-test/src -r -f -flat

After the upload completes, the following objects are generated in the bucket:

obs://bucket-test/src/ obs://bucket-test/src/src2/ obs://bucket-test/src/src2/test1.txt obs://bucket-test/src/src2/test2.txt obs://bucket-test/src/src3/ obs://bucket-test/src/test3.txt

• To upload the **file1** file to the **bucket-test** bucket, and resume the upload if the upload fails, run the following commands: ./obsutil cp /file1 obs://bucket-test/file -f

The upload fails. The command output is as follows:

./obsutil cp /file1 obs://bucket-test/file -f

Parallel: 3	Jobs: 3	
Threshold: 524288000	PartSize:	5242880
VerifyLength: false	VerifyMd5:	false
CheckpointDir: xxxx	-	

] 66.08% ?/s 3.35GB/4.88GB ?

Run the preceding command again to resume the upload. The command output is as follows:

obsutil cp /file1 obs://bucket-test/file -f Parallel: 3 Jobs: 3 Threshold: 524288000 PartSize: 5242880 VerifyLength: false VerifyMd5: false CheckpointDir: xxxx

[======] 100% 307.42MB/s 4.88GB/4.88GB 5.308s

Upload successfully, 4.88GB, n/a, /file1 --> obs://bucket-test/file, cost [6325], status [200], request id [xxxxx]

To incrementally upload all files from the local src1 folder to the src directory of bucket bucket-test, the command is as follows:
 ./obsutil cp /src1 obs://bucket-test/src -f -r -u

Four objects are successfully uploaded, one of which is a new object. The command output contains **Skip count**.

./obsutil cp /src1 obs://bucket-test/src -f -r -u Start at 2024-10-08 02:00:18.8906532 +0000 UTC

Parallel: 5	Jobs: 5	
Threshold: 50.00MB	PartSize:	auto
VerifyLength: false	VerifyMd5:	false
CheckpointDir: xxxx	-	

Task id: 6a97974a-7929-4188-9736-fcd637d16584 OutputDir: xxxx [=======] 100% tps:0.00 ?/s 2.09KB/2.09KB 5ms Succeed count: 4 Failed count: 0 Skip count: 3 Succeed bytes: 2.09KB Metrics [max cost:6 ms, min cost:6 ms, average cost:1.50 ms, average tps:52.63, transferred size :2.09KB

Run the following command to exclude the src2 folder (including all files and folders contained) when uploading the src1 folder:
 ./obsutil cp /src1 obs://bucket-test/src -exclude "*src1/src2*" -f -r -mf

Five objects are successfully uploaded, and the upload information contains **exclude** and the specific content.

./obsutil cp /src1 obs://bucket-test/src -exclude "*src1/src2*" -f -r -mf Start at 2024-10-08 02:04:27.7752009 +0000 UTC 5 Parallel: 5 Jobs: Threshold: 50.00MB PartSize: auto VerifyLength: false VerifyMd5: false Exclude: *src1/src2* Include: CheckpointDir: xxxx OutputDir: xxxx

[======] 100.00% tps:35.82 ?/s 5/5 2.39KB/ 2.39KB 340ms Succeed count: 3 Failed count: 0 Succeed bytes: 2.39KB

Metrics [max cost:338 ms, min cost:91 ms, average cost:240.40 ms, average tps:14.62, transferred size:2.39KB]

After the upload completes, the following objects are generated in the bucket:

obs://bucket-test/src/src1/ obs://bucket-test/src/src1/src3/ obs://bucket-test/src/src1/test3.txt

D NOTE

Resumable upload is available only for large files. Specifically, the file size is greater than 5 GB or the file size is greater than the threshold (50 MB by default).

7.2 Synchronous Upload

All commands in this section use the Linux operating system as an example to describe how to perform synchronous upload operations.

Assume that a local folder is in the following structure:

— src1
src2
test1.txt
test2.txt
src3
└── test3.txt

Assume that bucket **bucket-test** contains the following objects:

```
obs://bucket-test/src1/
obs://bucket-test/src1/src2/
obs://bucket-test/src1/src2/test1.txt
obs://bucket-test/src1/src3/
```

Based on the structure of the preceding local folder and objects in the bucket, different synchronous upload scenarios require different commands.

 To synchronize the test3.txt file in the local src1 folder to the root directory of bucket bucket-test, the command is as follows: ./obsutil sync /src1/test3.txt obs://bucket-test

After the synchronization is successful, the **test3.txt** file is directly uploaded to the bucket because there is no **test3.txt** in bucket **bucket-test**. Then, objects in the bucket are as follows:

obs://bucket-test/test3.txt obs://bucket-test/src1/ obs://bucket-test/src1/src2/ obs://bucket-test/src1/src2/test1.txt obs://bucket-test/src1/src3/

 To synchronize all files and subfolders in the local src1 folder to the src1 folder in bucket bucket-test, the command is as follows: ./obsutil sync /src1 obs://bucket-test/src1

After the synchronization, the objects in the bucket are as follows:

obs://bucket-test/src1/ obs://bucket-test/src1/test3.txt obs://bucket-test/src1/src2/ obs://bucket-test/src1/src2/test1.txt obs://bucket-test/src1/src2/test2.txt obs://bucket-test/src1/src3/

7.3 Download

All commands in this section use the Linux operating system as an example to describe how to download files.

Assume that bucket **bucket-test** contains the following objects:

obs://bucket-test/test1.txt obs://bucket-test/test2.txt obs://bucket-test/test3.txt obs://bucket-test/test4.txt obs://bucket-test/test6.txt obs://bucket-test/src1/ obs://bucket-test/src1/test7.txt obs://bucket-test/src2/ obs://bucket-test/src2/test8.txt obs://bucket-test/src2/src3/ obs://bucket-test/src2/src3/test9.txt

Based on the structure of objects in the bucket, different download scenarios require different commands.

 To download the test1.txt file from bucket bucket-test to the local src1 folder, the command is as follows: ./obsutil cp obs://bucket-test/test1.txt /src1

After the download is complete, the following file is generated on the local PC:

```
└── src1
└── test1.txt
```

• Run the following command to download the **test1.txt** file to your local PC. If there is no **test.txt** on the local PC, the **test1.txt** file is directly downloaded and you can rename it to **test.txt**. If **test.txt** already exists, **test1.txt** is downloaded and overwrites the original local **test.txt** file after renaming. ./obsutil cp obs://bucket-test/test1.txt /test.txt

After the download is complete, the following file is generated on the local PC:

└── test.txt

 To recursively download the entire src2 folder from bucket bucket-test to the local src1 folder in force mode, the command is as follows: ./obsutil cp obs://bucket-test/src2 /src1 -r -f

After the download is complete, the following files are generated on the local PC:

```
─_ src1
└─_ src2
└─_ src3
└─_ test9.txt
└─_ test8.txt
```

• To recursively download all files and subfolders in the **src2** folder from bucket **bucket-test** to the local **src1** folder in force mode, the command is as follows: ./obsutil cp obs://bucket-test/src2 /src1 -r -f -flat

After the download is complete, the following files are generated on the local PC:

```
└── src1
│── src3
└── test9.txt
└── test8.txt
```

• To recursively download the all objects in bucket **bucket-test** to the local **src0** folder in force mode, the command is as follows:

./obsutil cp obs://bucket-test /src0 -r -f

After the download is complete, the following files are generated on the local PC:

 Run the following command to exclude the src2 folder (including all files and folders contained) when downloading the src1 folder from the bucket-test bucket:

./obsutil cp obs://bucket-test/src1/ src1 -exclude "*src1/src2*" -r -f -mf

Four objects are successfully downloaded, and the download information contains **exclude** and the specific content.

./obsutil cp obs://bucket-test/src1/ src1 -exclude "*src1/src2*" -r -f -mf

Parallel: 5	Jobs:	5	
Threshold: 50.00MB	PartS	ze: auto	
VerifyLength: false	VerifyM	d5: false	
Exclude: *src1/src2*			
Include:			
CheckpointDir: xxxx			
OutputDir: xxxx			
TempFileDir: xxxx			
[======================================			=======] 100.00% tps:87.78 ?/s 4/4 2.39KB/
2.39KB 223ms			

```
Succeed count: 4 Failed count: 0
Succeed bytes: 2.39KB
Metrics [max cost:147 ms, min cost:77 ms, average cost:56.00 ms, average tps:8.85, transferred
size:2.39KB]
```

After the download is complete, the following files are generated on the local PC:

```
src1

src3

test9.txt

test7.txt
```

7.4 Synchronous Download

All commands in this section use the Linux operating system as an example to describe how to perform synchronous download operations.

Assume that bucket bucket-test contains the following objects:

```
obs://bucket-test/src1/
obs://bucket-test/src1/test3.txt
obs://bucket-test/src1/src2/
obs://bucket-test/src1/src2/test1.txt
obs://bucket-test/src1/src2/test2.txt
obs://bucket-test/src1/src3/
```

Assume that a local folder is in the following structure:

└── src1 └── test3.txt

Based on the structure of the preceding local folder and objects in the bucket, different synchronous download scenarios require different commands.

 To synchronize all files and subfolders in the src1 folder in bucket bucket-test to the local src1 folder, the command is as follows: ./obsutil sync obs://bucket-test/src1 /src1

After the synchronization is successful, the following files are generated in the local **src1** folder:

```
src1

src2

test1.txt

test2.txt

src3

test3.txt
```

7.5 Copy

All commands in this section use the Linux operating system as an example to describe how to copy files.

Assume that bucket **bucket-src** contains the following objects:

obs://bucket-src/test1.txt obs://bucket-src/test2.txt obs://bucket-src/test3.txt obs://bucket-src/test4.txt obs://bucket-src/test5.txt obs://bucket-src/src1/ obs://bucket-src/src1/test7.txt obs://bucket-src/src2/ obs://bucket-src/src2/test8.txt obs://bucket-src/src2/src3/ obs://bucket-src/src2/src3/test9.txt

Based on the structure of objects in the bucket, different copy scenarios require different commands.

 To copy the test1.txt file from bucket bucket-src to bucket bucket-dest, the command is as follows:

./obsutil cp obs://bucket-src/test1.txt obs://bucket-dest

After the copy is complete, the following object is generated in bucket **bucket-dest**:

obs://bucket-dest/test1.txt

• To copy the content of the **test1.txt** file in bucket **bucket-src** to the **text.txt** file in bucket **bucket-dest**, the command is as follows: ./obsutil cp obs://bucket-src/test1.txt obs://bucket-dest/test.txt

After the copy is complete, the following object is generated in bucket **bucket-dest**:

obs://bucket-dest/test.txt

 To copy the test1.txt file in bucket bucket-src to the text folder in bucket bucket-dest, the command is as follows: ./obsutil cp obs://bucket-src/test1.txt obs://bucket-dest/test/

After the copy is complete, the following object is generated in bucket **bucket-dest**:

obs://bucket-dest/test/test1.txt

 Run the following command to recursively copy the entire src2 folder in bucket bucket-src to bucket bucket-dest in force mode: ./obsutil cp obs://bucket-src/src2 obs://bucket-dest -r -f

After the copy is complete, the following objects are generated in bucket **bucket-dest**:

obs://bucket-dest/src2/ obs://bucket-dest/src2/test8.txt obs://bucket-dest/src2/src3/ obs://bucket-dest/src2/src3/test9.txt

 To recursively copy all files and subfolders in the src2 folder in bucket bucketsrc to bucket bucket-dest in force mode, the command is as follows: ./obsutil cp obs://bucket-src/src2 obs://bucket-dest -r -f -flat

After the copy is complete, the following objects are generated in bucket **bucket-dest**:

obs://bucket-dest/test8.txt obs://bucket-dest/src3/ obs://bucket-dest/src3/test9.txt

7.6 Synchronous Copy

All commands in this section use the Linux operating system as an example to describe how to perform synchronous copy operations.

Assume that the source bucket **bucket-src** contains the following objects:

obs://bucket-src/src1/ obs://bucket-src/src1/test3.txt obs://bucket-src/src1/src2/ obs://bucket-src/src1/src2/test1.txt obs://bucket-src/src1/src2/test2.txt obs://bucket-src/src1/src3/

Assume that the destination bucket **bucket-dest** contains the following objects:

obs://bucket-dest/src1/ obs://bucket-dest/src1/test3.txt

Based on the structure of objects in the bucket, different synchronous copy scenarios require different commands.

• To synchronize all files and subfolders in the **src1** folder in bucket **bucket-src** to the **src1** folder in bucket **bucket-dest**, the command is as follows: ./obsutil sync obs://bucket-src/src1 obs://bucket-dest/src1

After the synchronous copy is complete, the objects in the destination bucket **bucket-dest** are as follows:

```
obs://bucket-dest/src1/
obs://bucket-dest/src1/test3.txt
obs://bucket-dest/src1/src2/
obs://bucket-dest/src1/src2/test1.txt
obs://bucket-dest/src1/src2/test2.txt
obs://bucket-dest/src1/src3/
```

7.7 Listing

All commands in this section use the Linux operating system as an example to describe how to list files.

Assume that bucket **bucket-test** contains the following objects:

```
obs://bucket-test/test1.txt
obs://bucket-test/test2.txt
obs://bucket-test/test3.txt
obs://bucket-test/test4.txt
obs://bucket-test/test5.txt
obs://bucket-test/src1/
obs://bucket-test/src1/test7.txt
obs://bucket-test/src2/
obs://bucket-test/src2/test8.txt
```

Based on the structure of objects in the bucket, different object listing scenarios require different commands.

• To list three objects in bucket **bucket-test**, the command is as follows: ./obsutil ls obs://bucket-test -limit=3

The returned result is listed in lexicographical order by object name and version ID as follows:

obs://bucket-test/test1.txt obs://bucket-test/test2.txt obs://bucket-test/test3.txt

• To list three objects following **test3.txt** in bucket **bucket-test**, the command is as follows:

./obsutil ls obs://bucket-test -limit=3 -marker=test3.txt

The returned result is listed in lexicographical order by object name and version ID as follows:

obs://bucket-test/test4.txt obs://bucket-test/test5.txt obs://bucket-test/test6.txt To list the files and subdirectories in the root directory of bucket bucket-test in non-recursive mode, that is, files in the subdirectories are not listed, the command is as follows: ./obsutil ls obs://bucket-test -d

The returned result is listed in lexicographical order by object name and version ID as follows:

obs://bucket-test/test1.txt obs://bucket-test/test2.txt obs://bucket-test/test3.txt obs://bucket-test/test4.txt obs://bucket-test/test5.txt obs://bucket-test/test6.txt obs://bucket-test/src1/ obs://bucket-test/src2/

7.8 Listing Multipart Upload Tasks

All commands in this section use the Linux operating system as an example to describe how to list multipart upload tasks.

Assume that bucket **bucket-test** contains the following multipart upload tasks:

obs://bucket-test/task1.txt uploadid1 obs://bucket-test/task1.txt uploadid2 obs://bucket-test/task2.txt uploadid3 obs://bucket-test/task3.txt uploadid4 obs://bucket-test/src1/ obs://bucket-test/src1/task4.txt uploadid5 obs://bucket-test/src2/ obs://bucket-test/src2/task5.txt uploadid6

• Run the following command to list three multipart upload tasks in bucket **bucket-test**:

./obsutil ls obs://bucket-test -m -limit=3

The returned result is listed in lexicographical order by object name as follows:

obs://bucket-test/task1.txt uploadid1 obs://bucket-test/task1.txt uploadid2 obs://bucket-test/task2.txt uploadid3

 To list the rest multipart upload tasks following uploadid1, the command is as follows:

./obsutil ls obs://bucket-test -m -limit=3 -marker=task1.txt -uploadIdMarker=uploadid1

The returned result is listed in lexicographical order by object name and upload ID as follows:

obs://bucket-test/task1.txt uploadid2 obs://bucket-test/task2.txt uploadid3 obs://bucket-test/task3.txt uploadid4

7.9 Deleting All Multipart Upload Tasks in a Bucket

All commands in this section use the Linux operating system as an example to describe how to delete all multipart upload tasks in a bucket.

Assume that bucket **bucket-test** contains the following multipart upload tasks:

obs://bucket-test/task1.txt uploadid1 obs://bucket-test/task1.txt uploadid2 obs://bucket-test/task2.txt uploadid3 obs://bucket-test/task3.txt uploadid4 obs://bucket-test/src1/ obs://bucket-test/src1/task4.txt uploadid5 obs://bucket-test/src2/ obs://bucket-test/src2/task5.txt uploadid6

You can run the following command to delete all fragments of multipart upload tasks in the bucket at a time:

./obsutil abort obs://bucket-test -r -f

8 Fault Locating

8.1 Overview

obsutil provides multiple methods for users to locate and analyze faults. **Table 8-1** details the methods. Generally, you need to combine these methods for a precise fault locating.

Method	Description
Log Files	obsutil log files include tool logs and SDK logs. The tool logs record the success information and exceptions generated during obsutil running. The SDK logs record the success information and exceptions generated during requesting for OBS.
Result Lists	Result lists are generated after batch tasks complete and may include success, failure, and warning files.
Return Codes	obsutil yields different return codes based on different execution results. You can analyze and troubleshoot faults according to these return codes.

8.2 Log Files

Configuring Log Files

obsutil log files include tool logs and SDK logs. You can add the following parameters to the **.obsutilconfig** file to enable the two logging functions.

• Tool logging (records the log information generated during obsutil running): configure **utilLogPath**, **utilLogBackups**, **utilLogLevel**, and **utilMaxLogSize**.

 SDK logging (records the log information generated when using obsutil to call OBS server-side APIs): configure sdkLogPath, sdkLogBackups, sdkLogLevel, and sdkMaxLogSize.

NOTE

- For details about the parameter description, see Configuration Parameters.
- **utilLogPath** and **sdkLogPath** indicate the absolute paths of the log files, not the folders that store the log files.
- If **utilLogPath** and **sdkLogPath** are not specified, tool logging and SDK logging are not enabled, and therefore no log file is generated during obsutil running.
- Log files that are rolled over are named as follows: *filename.log.number*

NOTICE

If multiple obsutil processes are running at the same time, log files may fail to be written concurrently or may be lost. In this case, add parameter **-config** when running commands to configure an independent configuration file for each process. Make sure that **utilLogPath** and **sdkLogPath** are set to different paths for each process.

Collecting Log Files

You can collect logs in either of the following methods:

Method 1: Use auxiliary commands by referring to Archiving Log Files.

Method 2: Locate the paths specified by **utilLogPath** and **sdkLogPath** in the configuration file, and then search for the log files in the corresponding paths in the local file system.

8.3 Result Lists

Configuring Result Lists

Result lists are generated when batch tasks complete. By default, they are saved to the subfolder **.obsutil_output** in the home directory of the user who executes obsutil commands. You can specify another folder to save them by setting the additional parameter **-o** when executing a command.

Viewing Result Lists

Result lists are classified into success, failure, and warning lists. The naming rule is as follows: *Operation* **_{succeed | failed | warning}_report**_*Time*_*TaskId*.txt. For example, the name of the result list file for successfully uploading a folder is cp_succeed_report_20190417021908_fbbc83e3-98ac-4d19b23a-64023b1e0c34.txt, among which, fbbc83e3-98ac-4d19b23a-64023b1e0c34 indicates the task ID.

- If the number of successes, failures, or warnings is zero, the corresponding result list is not generated.
- The task ID of a result list is unique for each operation.
- The maximum size of a result list is 30 MB and the maximum number of lists that can be retained is 1024.
- If there are multiple folders and files and you need to confirm the detailed error information about a failed task, refer to the failure list in the result list folder and the **log files** in the log path.
- To prevent the batch operation efficiency from being reduced by too many result lists, you are advised to periodically archive and back up the result lists in .obsutil_output to another folder.

8.4 Return Codes

If obsutil is invoked by processes, the command output cannot be viewed in real time. obsutil generates different return codes based on different execution results. **Table 8-2** describes the return codes. You can use either the following methods to obtain the return code of the latest execution result and then analyze and rectify the fault based on it:

- In the macOS or Linux OS, run the following command to obtain the return code of the latest execution result: echo \$?
- In the Windows OS, run the following command to obtain the return code of the latest execution result: echo %errorlevel%

Return Code	Meaning	Example Scenario
0	Execution succeeded.	An object is successfully uploaded.
1	The file does not exist.	The entered file path does not exist for uploading a file by running the cp command.
2	The task does not exist.	The specified task ID does not exist for resuming a failed upload task by running the cp command.

Table 8-2 Return codes

Return Code	Meaning	Example Scenario
3	Parameter error	 At least one entered additional parameters is not supported for uploading a file by running the cp command. The entered value of cloud_url is invalid for downloading a file by running the cp command. NOTE cloud_url indicates the bucket path or object path. Set cloud_url in the format of obs://bucketname when downloading all objects in a bucket. Set cloud_url in the format of obs://bucketname/key when downloading a specified object in a bucket.
4	Bucket status error	The specified destination bucket does not exist for uploading a folder by running the cp command.
5	Initialization error during command execution	 An error occurs when loading the configuration file. Parameter -o is configured when running the cp command to upload a folder, but the folder specified by -o for saving the result lists fails to be created.
6	Execution error.	When you run the ls command to query the bucket list, the query fails because the network times out.
7	The operation is not supported.	Running the chattri command to change object properties is not supported because the bucket version is not 3.0.
8	A batch task succeeded partially.	Some objects fail to be downloaded during a batch download by running the cp command.
9	Interruption error	Users press Ctrl + C to interrupt the command execution.
-1	Unknown error	-

9 Best Practices

9.1 Using the obsutil help Command to Search for Functions

obsutil provides **help** commands for viewing the help documents of each command. To query the help document of the bucket creation command, perform the following steps:

- **Step 1** Run the **obsutil help** command to query the list of all supported commands.
- **Step 2** Find the abbreviation of the command to be viewed based on the document description in the command list. For example, the abbreviation of the command for creating a bucket is **mb**.
- **Step 3** Run the **obsutil help mb** command to view the usage and detailed functions of the **mb** command, illustrated as follows:

```
Summary:
create a bucket with the specified parameters
Svntax:
 obsutil mb obs://bucket [-fs] [-az=xxx] [-acl=xxx] [-sc=xxx] [-location=xxx] [-config=xxx]
Options:
 -fs
  create a bucket that supports POSIX
 -az=xxx
  the AZ of the bucket, possible values are [multi-az]
 -acl=xxx
  the ACL of the bucket, possible values are [private|public-read|public-read-write]
 -sc=xxx
  the default storage class of the bucket, possible values are: [standard|warm|cold]
 -location=xxx
  the region where the bucket is located
 -config=xxx
 the path to the custom config file when running this command
```

Step 4 Run the **obsutil mb obs://bucket-test -location xxx** command to create a bucket named **bucket-test** in the *xxx* region.

----End

- For more information about the **help** command, see **Viewing Command Help Information**.
- You can set the **helpLanguage** parameter in the configuration file to configure the language type of the **help** command. For example, **helpLanguage=Chinese** indicates that the language type of the help command is Chinese.
- The supported languages are Chinese and English. The default language is English.

9.2 Configuring Scheduled Tasks Using the Crontab Command

Scenario

Go to the **/root** directory at 21:30 every day and upload the **/src/src1** folder to bucket **obs://bucket-test** in the incremental mode.

Prerequisites

You have properly enabled the scheduled crond service in the Linux OS.

NOTE

Run the **service crond status** command to check whether the service is enabled.

Procedure

- **Step 1** Run the **crontab** -**e** command to open the configuration file for setting a scheduled task.
- **Step 2** Enter the Insert mode to edit the configuration file.

30 21 * * * cd /root && nohup ./obsutil cp /src/src1 obs://bucket-test -r -f -u &>obsutil_crond.log &

Assume that the obsutil tool is in the **/root** directory. The preceding configuration is described as follows: Go to the **/root** directory at 21:30 every day, upload the **/src/src1** folder to bucket **obs://bucket-test** in incremental mode, and redirect the command output to the **obsutil_crond.log** file in the **/root** directory.

- **Step 3** Press **Esc** to exit the Insert mode. Then input **:wq** and press **Enter** to save the configuration and exit.
- **Step 4** Run the **crontab -l** command to check whether the scheduled task is configured successfully.

----End

FAQs

- 1. How do I determine whether a scheduled task is being executed?
 - Run the **tail /var/log/cron** command to view the latest scheduled task execution records.
 - Run the **ps** -**ef** | **grep obsutil** command to check whether obsutil is being executed.
- 2. How do I forcibly stop an ongoing scheduled task?
 - a. Run the **ps -ef | grep obsutil** command to check the process of obsutil.
 - b. Run the **kill -9** *PID* command to forcibly stop the process, where *PID* indicates the queried process ID.

9.3 Setting obsutil Commands as Built-in Commands

Scenario

Because obsutil is external software, you need to access the directory where obsutil resides before running obsutil commands, which is not convenient.

An OS provides built-in commands so that the command-dependent directories are loaded to the memory when the system is started. In this way, you can run the commands in any directory, which improves the tool's usability.

This section introduces how to set obsutil commands as built-in commands in different OSs.

Setting obsutil Commands as Built-in Commands in Windows

Option 1: Automatic configuration by running a script (recommended)

- **Step 1** Go to the directory where the obsutil tool is located.
- **Step 2** Ensure that **setup.bat** and **obsutil.exe** are in the same directory. Double-click the **setup.bat** file.
- **Step 3** Check whether the configuration is successful: Run the **obsutil help** command in the CLI. If the help information is displayed, the configuration is successful.

NOTE

If the message **'obsutil' is not recognized as an internal or external command, either an operable program or batch file** is displayed after you run the **obsutil help** command when the automatic configuration completes, the configuration fails. In this case, manual configuration is required.

----End

Option 2: Manual configuration

- **Step 1** In the CLI, run the **echo %PATH%** command to query all the paths configured in the current system. Then select one as the operation path.
- **Step 2** Run the **mklink** *PATH***/obsutil.exe** *OBSUTIL_PATH* command to set obsutil commands as built-in commands in the system.

PATH indicates the operation path selected in step 1. *OBSUTIL_PATH* indicates the absolute path of **obsutil.exe**.

Step 3 Check whether the configuration is successful: Run the **obsutil help** command in the CLI. If the help information is displayed, the configuration is successful.

----End

Setting obsutil Commands as Built-in Commands in Linux or macOS

Option 1: Automatic configuration by running a script (recommended)

Step 1 Run the script with the path of obsutil as the parameter. Assume that obsutil is stored in the **/home/user1** directory. Run the following command: source setup.sh /home/user1/obsutil

NOTE

- You must execute the script as user **root**.
- The script parameter is the relative or absolute path of obsutil.
- You must run the source setup.sh or . setup.sh command to execute the script.
- **Step 2** Check whether the configuration is successful: Run the **obsutil help** command in any directory. If the help information is displayed, the configuration is successful.

NOTE

If the message **-bash: obsutil: command not found** is displayed after you run the **obsutil help** command when the automatic configuration completes, log in to the Linux or macOS CLI again. If the problem persists, manual configuration is required.

----End

Option 2: Manual configuration

Step 1 Run the following command to create a directory for the obsutil tool: mkdir /obsutil

NOTE

- Skip this step if the directory already exists.
- You must run the command as user **root**.
- **Step 2** Run the following command to grant the **755** permission for the tool's directory: chmod 755 /obsutil

NOTE

- Skip this step if the permission for the directory is **drwxr-xr-x**.
- You must run the command as user **root**.
- Step 3 Copy the obsutil tool to the directory created in step 1 and change its permission to 711. Assume that the original path of the tool is /home/test/obsutil. Run the following command:

cp /home/test/obsutil /obsutil chmod 711 /obsutil/obsutil **Step 4** Run the **vi /etc/profile** command, type **i** to enter the Insert mode to edit the file. Add **export PATH=\$PATH:/obsutil** at the end of the file. Then press **ESC** to exit the editing mode, and then type **:wq!** and press **Enter** to save the file and exit.

NOTE

Skip this step if the new line already exists in the **/etc/profile** file.

- **Step 5** Run the **echo \$PATH** command to query the current environment variables. If **:/ obsutil** in included in the query result, indicating that the **/obsutil** environment variable already exists, go to the next step. Otherwise, run the **source /etc/profile** command.
- **Step 6** Check whether the configuration is successful: Run the **obsutil help** command in any directory. If the help information is displayed, the configuration is successful.

----End

FAQs

1. How do I locate the obsutil configuration file after setting obsutil commands to built-in commands?

The **.obsutilconfig** file in the same directory where obsutil commands reside is the configuration file of the obsutil tool. You can also run the **obsutil config** command to obtain the configuration file path. An example is provided as follows:

```
obsutil config
Config file url:
D:\tools\.obsutilconfig
```

- 2. How do I delete obsutil commands after setting them as built-in commands?
 - In Windows:
 - i. Run the **where obsutil** command to locate the path of obsutil commands. where obsutil
 - E:\tools\bin\obsutil.exe
 - ii. Run the **del** *PATH* command to delete obsutil commands. del E:\tools\bin\obsutil.exe

NOTE

Replace *PATH* with the path of obsutil commands. **E:\tools\bin\obsutil.exe** is used in the preceding example.

- In Linux or macOS:
 - Run the which obsutil command to locate the path of obsutil commands. which obsutil /obsutil/obsutil
 - ii. Run the **rm -rf** *PATH* command to delete obsutil commands. rm -rf /obsutil/obsutil

NOTE

Replace *PATH* with the path of obsutil commands. **/obsutil/obsutil** is used in the preceding example.

iii. Restore the system environment variable: Delete the path of obsutil that is set in the **/etc/profile** file.

NOTE

If the **/etc/profile** file contains line **export PATH=\$PATH:/obsutil**, delete the line. Or if the file contains line **export PATH=\$PATH:/test/bin:/obsutil:/ test1**, delete **:/obsutil** from the line.

- 3. What should I do if the execution of built-in obsutil commands fails in Linux or macOS?
 - If the message Permission denied is displayed after executing obsutil help, run the chmod 755 OBSUTIL_PATH command (replace OBSUTIL_PATH with the path of obsutil) to add an execute permission for the obsutil tool.
 - If the message **command not found** is displayed, log in again.
 - If the message Cannot create parent folder for xx/.obsutilconfig, xx
 Permission denied is displayed, check whether the home directory of the user exists.

NOTICE

In the Ubuntu OS, if you run the **useradd** command to add a user, the home directory of the user is not created by default. You need to create it manually. Therefore, you are advised to run the **adduser** command to add a user.

4. What can I do if no log file is generated after running built-in obsutil commands in Linux or macOS?

If you have properly configured **sdkLogPath** and **utilLogPath** in the configuration file, but still no log file is generated after command execution, then check whether the user who runs the command has the read and write permissions on **sdkLogPath** and **utilLogPath**.

9.4 Configuring Auto Obtaining of Access Keys for obsutil

Feature Description

In addition to obtaining access keys (AK/SK) from the **.obsutilconfig** file, obsutil has the feature of automatically obtaining access keys in chain mode, to obtain access keys from system environment variables or to obtain temporary access keys through ECS APIs.

With this feature enabled, obsutil attempts to obtain available access keys from the **.obsutilconfig** file, system environment variables, and then ECS in sequence.

Configuration Method

Step 1 Enable the feature of automatically obtaining access keys in chain mode.

Set autoChooseSecurityProvider in the .obsutilconfig file to true.

Step 2 Configure the methods of obtaining access keys for obsutil.

- .obsutilconfig file: Configure parameters ak and sk. The AK and SK have been automatically written into the .obsutilconfig file during initialization (see Initializing the Configuration). You can modify them if needed.
- System environment variables: Set OBS_ACCESS_KEY_ID to ak, OBS_SECRET_ACCESS_KEY to sk, and OBS_SECURITY_TOKEN to token (can be left blank).
- ECS APIs: Ensure that the ECS has been bound to an IAM agency that has OBS permissions configured. For details, see Obtaining a Security Key from an ECS.

----End

Process of Searching for Access Keys

- 1. Search for access keys in the **.obsutilconfig** file (**ak**, **sk**, and **token**). If access keys are not left blank, the search ends. If they are not blank, continue to the next step.
- Search for access keys from system environment variables (OBS_ACCESS_KEY_ID, OBS_SECRET_ACCESS_KEY, and OBS_SECURITY_TOKEN). If access keys are not left blank, the search ends. If they are left blank, continue to the next step.
- 3. Search for temporary access keys from the ECS where obsutil runs. If the temporary access keys are not left blank, the search ends. Otherwise, access OBS as an anonymous user.

NOTE

- The "access keys are not empty" means both AK and SK are not empty, but the token can be empty.
- In the environment variables, the value of OBS_ACCESS_KEY_ID is the AK, that of OBS_SECRET_ACCESS_KEY is the SK, and that of OBS_SECURITY_TOKEN corresponds to the token.
- If the cross-region replication mode is enabled, OBS_ACCESS_KEY_ID_CRR corresponds to akCrr, OBS_SECRET_ACCESS_KEY_CRR corresponds to skCrr, and OBS_SECURITY_TOKEN_CRR corresponds to tokenCrr.
- The prerequisite for querying access keys from an ECS is that obsutil is running on this ECS and the ECS has been bound to an agency.
- If the function of automatically obtaining access keys is enabled, you still need to specify the endpoint.

9.5 Fine-Tuning obsutil Performance

By default, obsutil uploads, downloads, and copies files or objects whose size is greater than 50 MB in multiple parts. **Table 9-1** details related parameters in the **.obsutilconfig** file.

Parameter	Description	
defaultBigfileThres- hold	Indicates the threshold for triggering multipart tasks, in bytes. If the size of a file to be uploaded, downloaded, or copied is greater than the threshold, the file is uploaded, downloaded, or copied in multiple parts. The default value is 50MB .	
defaultPartSize	Size of each part, in bytes. The default value is auto . NOTE	
	• For multipart upload and copy, the value ranges from 100KB to 5GB .	
	• For multipart download, the value is unrestricted.	
defaultParallels	Maximum number of concurrent tasks in the multipart mode. The default value is 5.	

Table 9-1 Multipart-related parameters
--

Generally, multipart tasks not only speed up transmission but also allow you to resume failed tasks. By default, the part size of a multipart task can be automatically adjusted by the obsutil in the **auto** mode. In practice, however, to further improve the upload and download performance, you can adjust the part size according to the file size and the network conditions, to obtain the maximum transmission efficiency and ensure the successful completion of a transmission task.

Adjust the number of concurrent tasks in the multipart mode according to the following formula:

defaultParallels = Min(Number of CPUs x 2, Object size/defaultPartSize x 1.5)

In the upload, download, and copy commands, parameters **-p** and **-ps** are used to modify the number of concurrent tasks in the multipart mode and part size respectively, and then deliver the multipart task based on the parameter values configured in the command. The default values in the configuration file are used if you do not set them in a command.

Adjust the number of concurrent tasks in the multipart mode according to the following formula:

p = Min(Number of CPUs x 2, Object size/ps x 1.5)

For batch upload and download tasks, adjust the maximum number of concurrent tasks in a multipart upload, indicated by the parameter **defaultJobs (-j)**, for better performance.

If you have a large number of **small files** (each is usually several MB) to be uploaded or downloaded, set **defaultJobs** (-j) to a larger value for better performance. In this case, adjusting **defaultParallels** (-p) and **defaultPartSize** (**ps)** may be ineffective.

If you want to upload or download **large files**, set **defaultParallels (-p)** and **defaultPartSize (-ps)** to a larger value for better performance. However, if there

are too many concurrent tasks (calculated from **defaultJobs** x **defaultParallels**), the upload and download performance may deteriorate because of resource switchover and preemption between threads, and some tasks may fail due to network fluctuations.

NOTE

- Resources of a running host are limited. Therefore, if the number of concurrent tasks in the multipart mode is set too large, the performance of obsutil upload, download, or copy may deteriorate due to resource switchover and preemption between threads. In this case, you need to adjust the values of **defaultParallels** (-**p**) and **defaultPartSize** (-**ps**) based on the actual file size and network status. To perform a pressure test, lower the two values at first, and then gradually increase them to determine the optimal values.
- If the values of **defaultParallels** (-**p**) and **defaultPartSize** (-**ps**) are too large, an EOF error may occur due to network instability. In this case, set the two parameters to smaller values.
- If a batch operation is performed, the destination object size can be set to the average size of the objects to be operated.
- If a batch task fails due to timeout, EOF, and other common network issues, retry the task through an incremental operation (specifically, configuring parameter -u in the cp command). You can also restore the failed task (by configuring parameter -recover in the cp command) based on the generated task ID.

9.6 Using obsutil for Resumable Data Transfer

obsutil supports resumable data transfer (upload, download, and copy) for large files by using the multipart algorithms for upload, download, and copy. You can set the threshold size for starting a multipart upload, download, or copy task based on your actual requirements to resume the upload, download, or copy task if the task fails or is interrupted. You can specify the threshold size for starting a multipart task in either of the following ways:

Method 1

Set **defaultBigfileThreshold** in the configuration file. For details, see **Configuration Parameters**.

Method 2

Set threshold, a command-level parameter, when you run commands for object uploads, object downloads, object copy, synchronous uploads of incremental objects, synchronous downloads of incremental objects, and synchronous copy of incremental objects.

Example: **obsutil cp d:\temp\test.txt obs://bucket-test/key threshold=52428800**

In this command:

- obsutil cp d:\temp\test.txt obs://bucket-test/key uploads file test.txt in the temp directory under D drive to bucket bucket-test and renames the file key.
- **-threshold=52428800** starts a multipart upload when the threshold 50 MB is reached.

The following example is based on a Windows OS: obsutil cp d:\temp\test.txt obs://bucket-test/key -threshold=52428800

Parallel: 3	Jobs: 3	
Threshold: 50.00MB	PartSize:	auto
VerifyLength: false	VerifyMd5:	false
CheckpointDir: xxxx		

[======] 100.00% 1.68 MB/s 5s Upload successfully, 8.46MB, d:\temp\test.txt --> obs://bucket-test/key

D NOTE

- Priority: Command level parameter **threshold** has higher priority than the **defaultBigfileThreshold** in the configuration file.
- The threshold size of a multipart task applies to single files or objects. When the size of a file or object is greater than the threshold value, the multipart algorithm is applied to the file or object.
- The multipart algorithm and resumable data transfer are forcibly bound together. That is, once the multipart algorithm is used, the resumable data transfer is enabled for the task.

9.7 Using obsutil to Upload a Symbolic Link

obsutil supports the upload of the real path to which the symbolic link points when a file or folder is uploaded. You can specify the command-level parameter **link** to implement this function when running commands for upload or incremental synchronization upload.

NOTE

- obsutil can identify symbolic links pointing to folders. If a symbolic link points to a folder, obsutil recursively scans the contents in the folder.
- Avoid the symbolic link loop of a folder, otherwise, the upload will exit due to panic. If you do not want the system to panic, set **panicForSymbolicLinkCircle** to **false** in the configuration file.
- The symbolic link and the shortcut on the Windows OS are two different types. obsutil cannot identify the shortcut on the Windows OS.

9.8 Configuring an HTTP Proxy for obsutil

You can configure an HTTP proxy in either of the following ways:

Method 1: Set the **proxyUrl** parameter in the **.obsutilconfig** file, for example, **proxyUrl=http://username:password@your-proxy:8080**.

Method 2: Use the system environment variable HTTPS_PROXY or HTTP_PROXY, for example, HTTPS_PROXY=http://username:password@your-proxy:8080.

- HTTP proxy format: http://[Username:Password@]Proxy server address.Port number. The Username and Password are optional.
- The **proxyUrl** parameter and system environment variables are in the following priority order: **proxyUrl** > **HTTPS_PROXY** > **HTTP_PROXY**.
- The user name and password cannot contain colons (:) and at signs (@), which will result in parsing errors.

9.9 Using obsutil to Share Directories

The directory sharing function allows the owner of a bucket to share directories in a bucket or the entire bucket with other users by using the authorization code and access code. If you have the valid authorization code and access code of a shared folder, you can use OBS tools (OBS Browser and obsutil) to access the folder, list objects, and download objects. Alternatively, you can directly enter the authorization code in the address box of a web browser to list and download objects.

obsutil provides three commands to implement directory sharing. The procedure is as follows:

Step 1 Run the obsutil create-share command to create an authorization code for sharing a directory. For example, you can run the following command to share the test directory in the bucket named bucket with the access code set to 123456 and the validity period set to 10 days:

obsutil create-share obs://bucket/test/ -ac=123456 -vp=10d

NOTE

- When creating an authorization code, you are advised to end the name of the directory to be shared always with a slash (/). If no directory name is specified in the command (for example, only **obs://bucket** is specified in the command), the entire bucket is shared.
- If you do not use the **ac** option to set the access code, obsutil will prompt you to enter the access code. The access code must be a six-digit string.
- For details about this command, see **Creating an Authorization Code for Directory Sharing**.
- **Step 2** Run the **obsutil share-ls** command to list objects in the bucket. For example, to list the first 100 objects in the **test** directory in the bucket using the authorization code, run the following command:

obsutil share-ls file://d:/authorizationCode.txt -ac=123456 -prefix=test/ -limit=100

NOTE

- If the value of **prefix** is not specified, all objects in the authorized path are listed by default. If you do not want to list all objects, set **prefix** to a subset of the authorized path in the authorization code.
- For details about this command, see Listing Objects by Using an Authorization Code.
- Step 3 Run the obsutil share-cp command to download objects from the bucket. For example, if you want to download all objects in the sub subdirectory of the test directory, run the following command:

obsutil share-cp file://d:/authorizationCode.txt ./ -ac=123456 -key=test/sub/ -r -f

For details about this command, see **Downloading Objects by Using an Authorization Code**.

----End

D NOTE

- You can also create an authorization code on OBS Console or OBS Browser, and then use obsutil to list and download objects.
- You can use obsutil to create an authorization code and enter the authorization code in the address box of a web browser to list and download objects, or use the authorization code to log in to OBS Browser to list and download objects.

9.10 Using obsutil to Replicate Data Across Regions on the Client Side

obsutil client supports cross-region replication. You can directly replicate data from a source bucket to the destination bucket through data streams. The source bucket and destination bucket can be any two OBS buckets. Objects can be replicated between buckets in different regions under the same account or across accounts. The following procedure describes how to replicate data between buckets across accounts and regions:

- **Step 1** Run the **obsutil config** command to configure the AK, SK, and endpoint of the source bucket account.
 - In Windows
 obsutil config -i=src_ak -k=src_sk -e=src_endpoint -crr
 - In Linux ./obsutil config -i=*src_ak* -k=*src_sk* -e=*src_endpoint* -crr
- **Step 2** Run the **obsutil config** command to configure the AK, SK, and endpoint of the destination bucket account.
 - In Windows obsutil config -i=*dst_ak* -k=*dst_sk* -e=*dst_endpoint*
 - In Linux ./obsutil config -i=*dst_ak* -k=*dst_sk* -e=*dst_endpoint*
- **Step 3** Check the connectivity to ensure that the destination bucket is correctly configured.
 - In Windows
 obsutil ls -s
 - In Linux or macOS ./obsutil ls -s

Check the command output:

- If it contains "Bucket number", the configuration is correct.
- If it contains "Http status [403]", the access keys are wrong.
- If it contains "A connection attempt failed", OBS cannot be connected. Then, check the network condition.
- If it contains "Error: cloud_url [url] is not in well format", the domain name to be accessed is incorrect. Check the domain name in the configuration file.

If the command output contains "Http status [403]", you may not have the required permissions for obtaining the bucket list. A further analysis is required to identify the root cause.

- **Step 4** Run the **cp** command to specify that cross-region replication method is used to copy objects from the source bucket to the destination bucket.
 - In Windows obsutil cp obs://src-bucket obs://dst-bucket -f -r -crr
 - In Linux ./obsutil cp obs://src-bucket obs://dst-bucket -f -r -crr

----End

NOTE

- To use the cross-region replication function, you need to specify the **-crr** parameter. If this parameter is specified, update the configuration of the client-side cross-region replication in the configuration file. For details, see **Updating a Configuration File**.
- The configurations of the source bucket and destination bucket are respectively **akCrr/ skCrr/tokenCrr/endpointCrr** and **ak/sk/token/endpoint** in the configuration file.
- The preceding procedure is also applicable to the situation when the source and destination buckets belong to the same account.

When the **-crr** parameter is used, the source object's standard metadata, including **Cache-Control**, **Expires**, **Content-Encoding**, **Content-Disposition**, **Content-Type**, and **Content-Language**, will not be copied.

When the **-crr** parameter is used for cross-region replication, the ACL of the source object will not be copied. You can use **[-acl=***xxx***]** to specify the ACL for the target object. If the ACL is not specified, the object inherits the ACL of the bucket by default.

9.11 Limiting the Upload and Download Rate for obsutil

obsutil allows you to configure the **rateLimitThreshold** parameter in the **.obsutilconfig** file to limit the upload and download rate.

For detailed parameter description, see **Configuration Parameters**. If you do not configure this parameter, the upload and download rate will not be limited, but depend on the user's network bandwidth and the number of concurrent tasks. For details about the optimization, see **Fine-Tuning obsutil Performance**.

Parameter **rateLimitThreshold** limits the global rate of obsutil tasks. This means that if you upload and download files in batches using the **cp** and **sync** commands, the actual upper rate is the one specified by **rateLimitThreshold**, not by the value obtained as follows: Number of concurrent tasks x Value of **rateLimitThreshold**.

10 FAQs

10.1 After Some Files Are Deleted in My Local Directory, Can obsutil Synchronously Delete Them from the Bucket?

No.

obsutil allows you to upload your local directory to an OBS bucket. After the synchronization, if you delete some files in the local directory and then perform an incremental upload, obsutil only checks whether there are incremental files that need to be uploaded. obsutil cannot detect the deleted files, so these files will not be deleted in the bucket.

If new files are added to your local directory during an upload, the number of objects uploaded by obsutil may be inconsistent with that in the local directory. To keep the files same in both places, run the incremental upload command after the upload is complete.

10.2 Can I Use obsutil to Directly Save a Listing Result to a Local File?

obsutil does not allow you to directly save a listing result to a local file, but you can redirect a listing result displayed on the screen in standard output to a specified local file by relying on the redirection supported by OS. The following uses the listing of objects in a bucket as an example:

- In Windows, run the following redirection command in the CLI: obsutil ls obs://bucketName -format=default -limit=0 > D:/result.txt
- In Linux or macOS, run the following command: ./obsutil ls obs://bucketName -format=default -limit=0 > /root/result.txt

NOTE

In Windows, you need to perform the redirection operation in the CLI to redirect the output to a local file. Do not run the redirection command in the obsutil executable file because obsutil does not support redirection.

10.3 Why Is the Size of Objects Queried by obsutil Inconsistent with That on OBS Console?

When you use obsutil to list all objects in a bucket, the listing result contains the total size of the objects. If the total size is different from that on OBS Console or OBS Browser+, go through the following list to locate the fault:

- To query the number of objects in a bucket and the space occupied by the objects, both OBS Browser+ and OBS Console call the API for Querying Information About Used Space in a Bucket. The results obtained by them correspond to the output by calling the stat command in obsutil. On OBS Browser+ and OBS Console, the bucket storage statistics are measured in the backend and are not real time. Therefore, you are advised to use obsutil to query the storage usage.
- 2. Check whether there are object fragments in the bucket on OBS Browser+ and OBS Console by referring to Listing Multipart Upload Tasks. OBS bucket storage statistics cover the size of both objects and object fragments in a bucket, but obsutil lists only the objects in a bucket.

10.4 How Can I Find Out Why Some Tasks in a Batch Task Failed?

After a batch task is completed, there will be result telling you how many tasks succeeded or failed. To find out why those tasks failed, you can check their result list or obsutil log file.

After a batch task is executed, its task ID will be generated. You can query the list of failed tasks in the **.obsutil_output** directory based on the task ID. The list is named in the **cp_{failed}_report_***Time_TaskId*.**txt** format and contains detailed error information about each failed task.

In addition, you can view the obsutil log file to query the error records during obsutil running. You are advised to set the log level to **DEBUG** for fault locating. For details about the log configuration and log path, see **Log Files**.

10.5 How Can I Locate and Rectify I/O Timeout and EOF Errors?

I/O timeout and EOF errors usually occur when requests fail due to network fluctuations. Go through the following list to locate the cause.

- 1. Ping the bucket domain name (*bucketName.endpoint*) to check the network connection between your local PC and the bucket. If the network connection is abnormal, resolve the network issue first.
- 2. If these errors are prone to occur and the bucket domain name can be pinged, use HTTP for the endpoint and capture network packets. Based on the packets captured, check whether packet loss occurs on the actual network link and resolve the errors accordingly.

Solutions:

- 1. If the network connection is abnormal, resolve the local network problem first. If you need to configure a proxy, see **Configuring an HTTP Proxy for obsutil**.
- If these errors occur occasionally, retry the command of your operation. During upload, download, or replication operations, specify the -u parameter in the cp command to perform an incremental upload. In this way, you do not need to retry the tasks that have been successfully completed in a batch task.
- 3. If the network condition is poor, you can decrease the values of **defaultParallels (-p)** and **defaultJobs (-j)** to reduce the number of concurrent upload, download, or replication tasks, to make errors less likely occur.

10.6 Why Is a Question Mark Displayed in the Batch Task Progress Bar?

Possible causes:

When a batch upload or download task is being executed, if there are a large number of objects involved, obsutil needs to traverse all objects to collect the total number and size of objects in the task. During this collection process, a question mark (?) is displayed in the progress bar.

Solution:

After the information is collected, corresponding task details will be displayed in the progress bar.

10.7 Can Multiple config Files Be Placed in One Directory?

No.

obsutil allows you to configure multiple **config** files, but they cannot be stored in the same directory due to encryption requirements.

If multiple **config** files are required, initialize them to put them in different directories.

10.8 Why Is the Number of Objects I Downloaded from OBS to My Local Windows PC Different from That in OBS?

Windows does not distinguish the cases of files or folders while OBS does. For example, in Windows, **FOO.txt** and **foo.txt** are regarded as the same file, but in OBS, they are regarded as different files.

In the following example, a bucket has objects with the same name but different cases.

obsutil ls obs://test Start at 2024-01-17 01:33:37.6378193 +0000 UTC

Listing objects .

Folder list: obs://test/TEST/ obs://test/test/

Object list: key LastModified Size StorageClass ETag obs://test/TEST/FOO.txt 2024-01-17T01:33:21Z 493B standard "735b0ca5680bfa12d8895702662192ec"

obs://test/foo.txt 2024-01-17T01:32:59Z 5B standard "5d41402abc4b2a76b9719d911017c592"

Total size of bucket: 498B Folder number: 2 File number: 2

Run the **cp** command to download all objects from this bucket to the **test** directory on your local Windows PC.

obsutil cp obs://test ./test -r -f

Start at 2024-01-17 01:39:40.4700311 +0000 UTC

Parallel:5Jobs:5Threshold:50.00MBPartSize:autoVerifyLength:falseVerifyMd5:falseCheckpointDir:C:\Users\Administrator\.obsutil_checkpoint

Task id: 72c5ac1c-68c0-4206-95d6-18a193bfa77e OutputDir: C:\Users\Administrator\.obsutil_output TempFileDir: C:\Users\Administrator\temp

[------] 100.00% tps:484.98 162.11KB/s 4/4 498B/498B 205ms Succeed count: 4 Failed count: 0 Succeed bytes: 498B Metrics [max cost:86 ms, min cost:85 ms, average cost:43.25 ms, average tps:9.13, transferred size:498B]

Task id: 72c5ac1c-68c0-4206-95d6-18a193bfa77e

Run the **dir /b /s** command to view all files, folders, and files in subdirectories in the **test** directory. It is found that there are only the **TEST** folder and the **TEST foo.txt** file kept.



In this case, the local files downloaded may not be completely the same as the original OBS objects. The local file **TEST\foo.txt** is actually **TEST/FOO.txt** (but not **test/foo.txt**) in OBS. This is determined by the download success sequence of files. How files and folders are kept on a local Windows PC is based on the following rules:

- Folders: The folder that is first successfully downloaded is kept. For example, if TEST/ is downloaded successfully before test/, only the TEST/ folder will be kept on your local Windows PC.
- Files: The name of the file that is first downloaded successfully and the content of the file that is last downloaded successfully will be kept. For example, if test/foo.txt is downloaded before TEST/FOO.txt, the file name will be foo.txt, but the file content will be that of FOO.txt.

10.9 Can I Rename an Object or a Folder?

Yes. You can run **the mv command** to rename an object or a folder. The following give two examples in Windows.

 Running obsutil mv obs://bucket-test/key obs://bucket-test/key2 to rename object key to key2 obsutil mv obs://bucket-test/key obs://bucket-test/key2

Parallel: 5 Jobs: 5 Threshold: 50.00MB PartSize: auto CheckpointDir: xxxx

Move successfully, 19B, obs://bucket-test/key --> obs://bucket-test/key2, cost [96], status [200], request id [xxxxxxxx]

Running obsutil mv obs://bucket-test/temp/ obs://bucket-test/temp2 -flat
 -f -r to rename folder temp to temp2
 -hutil mu obs//bucket test/temp2 (flat f r

obsutil mv obs://bucket-test/temp/ obs://bucket-test/temp2/ -flat -f -r

Parallel:5Jobs:5Threshold:50.00MBPartSize:autoCheckpointDir:xxxxVutputDir:vxxx

[=====] 100.00% tps:0.00 2/2 174ms Succeed count: 5 Failed count: 0

Metrics [max cost:298 ms, min cost:192 ms, average cost:238.00 ms, average tps:9.71, transferred size:70B]

Task id: 0476929d-9d23-4dc5-b2f8-0a0493f027c5

NOTICE

When renaming a folder, you must add the -flat parameter.

A Configuration Parameters

You can set obsutil parameters in the .obsutilconfig file.

Configuration file format:

endpoint=<value> ak=<value> sk=<value> token=<value> endpointCrr=<value> akCrr=<value> skCrr=<value> tokenCrr=<value> connectTimeout=<value> socketTimeout=<value> maxRetryCount=<value> maxConnections=<value> defaultBigfileThreshold=<value> defaultPartSize=<value> defaultParallels=<value> defaultJobs=<value> defaultJobsCacheCount=<value> rateLimitThreshold=<value> sdkLogBackups=<value> sdkLogLevel=<value> sdkLogPath=<value> sdkMaxLogSize=<value> utilLogBackups=<value> utilLogLevel=<value> utilLogPath=<value> utilMaxLogSize=<value> writeBufferIoSize=<value> readBufferIoSize=<value> recordMaxLogSize=<value> recordBackups=<value> humanReadableFormat=<value> showProgressBar=<value> showStartTime=<value> helpLanguage=<value> defaultTempFileDir=<value> checkSourceChange=<value> skipCheckEmptyFolder=<value> fsyncForDownload=<value> memoryEconomicalScanForUpload=<value> forceOverwriteForDownload=<value> panicForSymbolicLinkCircle=<value> autoChooseSecurityProvider=<value> fastFailThreshold=<value> abortHttpStatusForResumableTasks=<value> showBytesForCopy=<value>

proxyUrl=<value> faultTolerantMode=<value>

 Table A-1 describes the parameters.

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
endpoint	Man dato ry	Endpoint for accessing OBS, which can contain the protocol type, domain name, and port number (optional), for example, https://your- endpoint:443. (For security purposes, you are advised to use HTTPS. The port number 443 can be omitted.) If you want to access OBS in the CN-Hong Kong region, the address you actually visit is https://obs.ap- southeast-1.myhuaweicloud.c om. NOTE • To view the endpoints and regions available for OBS, see Regions and Endpoints. • If the configured endpoint does not contain any protocol, the HTTPS protocol is used by default.	N/A
ak	Opti onal	 Access key ID. This parameter is optional when autoChooseSecurityProvider is set to true. NOTE For more information, visit the access key management console. After you run obsutil for the first time, the tool encrypts the AK to ensure the key security. 	N/A

Table A-1	obsutil	configuration	parameters
		5	1

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
sk	Opti onal	 Secret access key. This parameter is optional when autoChooseSecurityProvider is set to true. NOTE For more information, visit the access key management console. After you run obsutil for the first time, the tool encrypts the SK to ensure the key security. 	N/A
token	Opti onal	Security token. If this parameter is empty, the security token is not set. When temporary AK and SK are used to access OBS, the token must be carried.	N/A
endpointCrr	Opti onal	 Endpoint for accessing OBS in the region where the source bucket resides when the client-side cross-region replication function is enabled, which can contain the protocol type, domain name, and port number, for example, https:// your-endpoint:443. (For security purposes, you are advised to use HTTPS. The port number 443 can be omitted.) NOTE To view the endpoints and regions available for OBS, see Regions and Endpoints. If the configured endpoint does not contain any protocol, the HTTPS protocol is used by default. 	N/A
akCrr	Opti onal	AK for the source bucket when the client-side cross-region replication function is enabled	N/A
skCrr	Opti onal	SK for the source bucket when the client-side cross-region replication function is enabled	N/A

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
tokenCrr	Opti onal	Security token used by the source bucket when the client- side cross-region replication is used. If this parameter is left blank, no security token is set. This parameter is mandatory when temporary access keys are used for access.	N/A
connectTimeout	Opti onal	Timeout interval for establishing an HTTP/HTTPS connection, in seconds. The default value is 30 .	[5, 120]
socketTimeout	Opti onal	Timeout interval for reading and writing data, in seconds. The default value is 310 .	[5, 600]
maxRetryCount	Opti onal	Maximum number of retry attempts. The default value is 3 . NOTE When an OBS request completes but HTTP status code 408 or 5 <i>XX</i> is returned, or when a timeout error occurs in an OBS request, the request is retried.	[0, 5]
maxConnections	Opti onal	Maximum number of HTTP connections that can be accessed. The default value is 1000 .	N/A
defaultBigfileTh- reshold	Opti onal	Threshold for triggering a multipart upload, in bytes. If the size of a file to be uploaded, downloaded, or copied is greater than the specified threshold, the file will be uploaded, downloaded, or copied using a multipart upload. The default value is 50MB . NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).	Greater than 5MB

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
defaultPartSize	Opti onal	 Size of each part, in bytes. The default value is auto. NOTE For a multipart upload and copy, the value ranges from 100KB to 5GB. For multipart download, the value is unrestricted. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). If this parameter is set to auto. In this case, obsutil automatically sets the part size for each multipart task based on the source object size. 	[9MB, 100MB]
defaultParallels	Opti onal	Maximum number of concurrent tasks in the multipart mode. The default value is 5 .	Set this parameter according to Fine- Tuning obsutil Performance.
defaultJobs	Opti onal	Maximum number of concurrent tasks in batches. The default value is 5 . NOTE Batch tasks include uploading, downloading, and copying folders, as well as restoring and deleting objects in batches.	[1, 50]
defaultJobsCache- Count	Opti onal	Cache size of a batch task queue, indicating the maximum number of tasks that can be cached. The default value is 1000000 . NOTE More cached tasks consume more memory resources. Therefore, you are advised to adjust the value of this parameter based on site requirements.	Default value

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
rateLimitThreshol d	Opti onal	Traffic control threshold of an upload or a download request, in bytes per second. The default value is 0 , indicating that traffic is not limited. The minimum value is 10KB .	Greater than 100KB
		NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).	
sdkLogBackups	Opti onal	Maximum number of SDK log files that can be retained. The default value is 10 .	N/A
sdkLogLevel	Opti onal	 SDK log level. Possible values are: DEBUG INFO WARN ERROR The default value is WARN. 	N/A

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
sdkLogPath	Opti onal	 Absolute path of SDK logs. The value must be a file path. The default value is the path of the obssdk.log file in the subfolder obsutil_log of the user's home directory (HOME in Linux or macOS and C:\Users \<username> in Windows).</username> NOTE If this parameter is left blank, no SDK log is generated. The path must be a file path and cannot be a folder path. After the SDK log function is enabled, all logs of requests to OBS are saved in the SDK log file for problem analysis and location. Ensure that the user who runs the command has the read and write permissions on the path. NOTICE If multiple obsutil processes are running at the same time, log files may fail to be written concurrently or may be lost. In this case, add parameter -config when running commands to configure an independent configuration file for each process. 	N/A
sdkMaxLogSize	Opti onal	Size of an SDK log file, in bytes. The default value is 30MB . NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).	[10MB, 100MB]
utilLogBackups	Opti onal	Maximum number of obsutil log files that can be retained. The default value is 10 .	N/A

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
utilLogLevel	Opti onal	obsutil log level. Possible values are: • DEBUG • INFO • WARN • ERROR The default value is INFO .	N/A
utilLogPath	Opti onal	 Absolute path of obsutil logs. The value must be a file path. The default value is the path of the obsutil.log file in the subfolder .obsutil_log of the user's home directory (HOME in Linux or macOS and C:\Users \<username> in Windows).</username> NOTE If this parameter is left blank, no obsutil log is generated. The path must be a file path and cannot be a folder path. After the obsutil log function is enabled, all logs generated during commands executing are saved in the obsutil log file for problem analysis and location. Ensure that the user who runs the command has the read and write permissions on the path. NOTICE If multiple obsutil processes are running at the same time, log files may fail to be written concurrently. In this case, add parameter -config when running commands to configure an independent configuration file for each process. 	N/A
utilMaxLogSize	Opti onal	Size of an obsutil log file, in bytes. The default value is 30MB . NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).	[10MB, 100MB]

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
writeBufferIoSize	Opti onal	 Size of the cache for downloading data, in bytes. The default value is 65536. NOTE Set this parameter based on site requirements. If the size of the file to be downloaded is large, set this parameter to a large value. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). 	N/A
readBufferIoSize	Opti onal	 Size of the cache for uploading data, in bytes. The default value is 8192. NOTE Set this parameter based on site requirements. If a large number of small files are uploaded, set this parameter to a small value. If large files are uploaded, set this parameter to a large value. This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes). 	[4096, 65536]
recordMaxLogSize	Opti onal	Size of a result list containing success, failure, or warning lists in a batch task, in bytes. The default value is 30MB . NOTE This parameter value can contain a unit, for example, 1MB (indicating 1048576 bytes).	[5MB, 100MB]
recordBackups	Opti onal	Maximum number of result lists of successful or failed batch tasks that can be retained. The default value is 1024 .	N/A

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
humanReadableF ormat	Opti onal	Indicates whether to convert the number of bytes in the object listing result and result list content to the human- readable format. The default value is true .	N/A
showProgressBar	Opti onal	Indicates whether to display the progress bar on the console. The value true indicates that the progress bar is displayed. The default value is true .	N/A
showStartTime	Opti onal	Indicates whether to display the start time on the console. The value true indicates that start time is displayed. The default value is true .	N/A
helpLanguage	Opti onal	Language of the help documents. Options are as follows: • Chinese • English The default value is English .	N/A
defaultTempFile- Dir	Opti onal	 Indicates the directory for storing temporary files during download. The default value is the .obsutil_tempfile subfolder in the user directory (HOME in Linux or macOS and C:\Users \<usersamelerates< td=""><td>N/A</td></usersamelerates<>	N/A

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
checkSourceChan ge	Opti onal	Indicates whether to check the change of source files or objects during upload/download/copy. The value true indicates that the function is enabled.	N/A
skipCheckEmptyF older	Opti onal	Indicates whether to skip checking empty folders on the OBS server during download. true indicates to skip the check. The default value is false . NOTICE If this parameter is set to true , the directory structure downloaded to your local PC may be different from that in OBS.	N/A
fsyncForDownload	Opti onal	 Indicates whether to forcibly synchronize memory data to disks during download. The value true indicates to enable forcible synchronization. The default value is false. NOTE Set this parameter to true for scenarios that require high data reliability. If this parameter is set to true, the download performance will be deteriorated. Therefore, exercise caution when using this parameter. 	N/A
memoryEconomic alScanForUpload	Opti onal	Indicates whether to use the scanning mode that occupies less memory space when uploading a folder. The value true indicates using this method. The default value is true .	N/A

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
forceOverwriteFor Download	Opti onal	Indicates to forcibly overwrite the local executable file (even if the local executable file is running) when downloading objects to the Linux OS or macOS. The value true means to overwrite, and the default value is true .	N/A
panicForSymboli- cLinkCircle	Opti onal	Indicates the processing method after a symbolic link loop is detected during upload. The value false indicates that errors are only recorded. The value true indicates that panic is triggered. The default value is false .	N/A
autoChooseSecur- ityProvider	Opti onal	Indicates whether to automatically obtain the access keys from the configuration file, environment variables, and ECS APIs in sequence. The default value is false . NOTE For details about this parameter, see Configuring Auto Obtaining of Access Keys for obsutil .	N/A
fastFailThreshold	Opti onal	Threshold for fast failure upon 4XX errors of batch tasks. When the number of 4XX errors exceeds the threshold, the fast failure process is triggered. All tasks that are not executed or being scanned are suspended. The default value is 5 . NOTE The fast failure mechanism is to avoid excessive traffic generated during batch task execution. To start a fast failure as soon as possible, set this parameter to 0 or -1 , indicating that the fast failure process starts immediately whenever a 4XX error occurs.	N/A

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
abortHttpStatus- ForResumableTas ks	Opti onal	 HTTP status codes for fast interruption of multipart upload, download, and copy tasks. If a sub-task of a multipart task receives an HTTP code that falls into this range, the multipart task is immediately interrupted. The default value is 401,403,404,405,409. NOTE Multiple HTTP status codes can be carried and separated by commas (,), for example, 401,403,404. The status code must be a 4XX HTTP status code. Other status codes are ignored. 	Default value
showBytesForCop y	Opti onal	Indicates whether the progress bar displays the rate in bytes when objects are copied between buckets. The default value is false .	N/A
proxyUrl	Opti onal	 HTTP proxy example: http:// username:password@your- proxy:8080 NOTE HTTP proxy format: http:// [Username:Password@]Proxy server address:Port number. The Username and Password are optional. The proxyUrl parameter and system environment variables are in the following priority order: proxyUrl > HTTPS_PROXY > HTTP_PROXY. The user name and password cannot contain colons (:) and at signs (@), which will result in parsing errors. 	N/A

Parameter	Opt iona l or Ma nda tory	Description	Recommended Value
faultTolerantMod e	Opti onal	Indicates whether to ignore the panic caused by read operations. Default value false indicates that the panic is not ignored.	N/A

D NOTE

- Set parameters with N/A as the recommended value based on your needs.
- You are advised to specify **sdkLogPath** and **utilLogPath** to enable SDK logging and obsutil logging.
- The values of defaultBigfileThreshold, defaultPartSize, rateLimitThreshold, sdkMaxLogSize, utilMaxLogSize, recordMaxLogSize, readBufferIoSize, and writeBufferIoSize can contain a unit, for example, 1MB (indicating 1048576 bytes).