

Object Storage Service

Guía del usuario

Edición 01
Fecha 2020-08-31



Copyright © Huawei Technologies Co., Ltd. 2023. Todos los derechos reservados.

Quedan terminantemente prohibidas la reproducción y la divulgación del presente documento en todo o en parte, de cualquier forma y por cualquier medio, sin la autorización previa de Huawei Technologies Co., Ltd. otorgada por escrito.

Marcas y permisos



HUAWEI y otras marcas registradas de Huawei pertenecen a Huawei Technologies Co., Ltd.

Todas las demás marcas registradas y los otros nombres comerciales mencionados en este documento son propiedad de sus respectivos titulares.

Aviso

Las funciones, los productos y los servicios adquiridos están estipulados en el contrato celebrado entre Huawei y el cliente. Es posible que la totalidad o parte de los productos, las funciones y los servicios descritos en el presente documento no se encuentren dentro del alcance de compra o de uso. A menos que el contrato especifique lo contrario, ninguna de las afirmaciones, informaciones ni recomendaciones contenidas en este documento constituye garantía alguna, ni expresa ni implícita.

La información contenida en este documento se encuentra sujeta a cambios sin previo aviso. En la preparación de este documento se realizaron todos los esfuerzos para garantizar la precisión de sus contenidos. Sin embargo, ninguna declaración, información ni recomendación contenida en el presente constituye garantía alguna, ni expresa ni implícita.

Índice

| | |
|---|-----------|
| 1 Antes de comenzar..... | 1 |
| 2 Uso de OBS..... | 3 |
| 3 Acceso a OBS..... | 6 |
| 4 Descripción general de la función..... | 12 |
| 5 Gestión de permisos..... | 13 |
| 6 Clases de almacenamiento..... | 14 |
| 7 Gestión de buckets..... | 19 |
| 7.1 Información general..... | 19 |
| 7.2 Creación de un bucket..... | 20 |
| 7.3 Enumeración de buckets..... | 22 |
| 7.4 Consulta de información de buckets..... | 22 |
| 7.5 Gestión de cuotas de bucket..... | 23 |
| 7.6 Eliminación de buckets..... | 24 |
| 8 Gestión de objetos..... | 25 |
| 8.1 Descripción general del objeto..... | 25 |
| 8.2 Creación de la carpeta..... | 28 |
| 8.3 Carga de un objeto..... | 29 |
| 8.4 Descarga de un objeto..... | 32 |
| 8.5 Enumeración de objetos..... | 33 |
| 8.6 Copia de objetos..... | 34 |
| 8.7 Carga de varias partes..... | 35 |
| 8.8 Consulta de información de objeto..... | 40 |
| 8.9 Compartir un archivo..... | 41 |
| 8.10 Compartir una carpeta..... | 43 |
| 8.11 Gestión de metadatos de objetos..... | 45 |
| 8.12 Restaurar objetos desde el almacenamiento de archivos..... | 56 |
| 8.13 Lectura directa..... | 57 |
| 8.14 Eliminación de un objeto..... | 57 |
| 8.15 Recuperación de un objeto..... | 59 |
| 8.16 Gestión de fragmentos..... | 60 |

| | |
|---|------------|
| 9 Sistema de archivos paralelo..... | 61 |
| 10 Procesamiento de imágenes..... | 62 |
| 11 Gestión del ciclo de vida..... | 63 |
| 12 Replicación entre regiones..... | 74 |
| 13 Encriptación del lado del servidor..... | 78 |
| 14 Alojamiento de sitio web estático..... | 80 |
| 15 CORS..... | 89 |
| 16 Vinculación de nombre de dominio definido por el usuario..... | 92 |
| 17 Vuelta a la fuente..... | 94 |
| 18 Validación de URL..... | 96 |
| 19 Etiquetas..... | 99 |
| 20 Inventario de buckets..... | 100 |
| 21 Notificación de evento..... | 106 |
| 22 Registro..... | 114 |
| 23 Control de versiones..... | 118 |
| 24 Monitoreo..... | 122 |
| 25 Auditoría..... | 124 |
| 26 Descompresión en línea (OBT)..... | 125 |
| 27 Historial de revisiones..... | 127 |

1 Antes de comenzar

Object Storage Service (OBS) proporciona una serie de documentos para ayudarle a comprender y utilizar mejor el servicio.

Antes de usar OBS, lea este documento y otros relacionados de ayuda (consulte [Tabla 1-1](#)) para comprender los conceptos básicos, escenarios de aplicación y operaciones de OBS, para que pueda comenzar rápidamente con OBS.

Tabla 1-1 Documentos de OBS

| Documento | Descripción |
|--|--|
| Conocimiento progresivo | Proporciona los enlaces a <i>Inicio rápido</i> , <i>Guía del usuario</i> , <i>Prácticas recomendadas</i> , <i>FAQs</i> y <i>Soluciones</i> , para que pueda localizar fácilmente y con precisión el contenido que desea ver. |
| ¿Qué hay de nuevo? | Describe las últimas actualizaciones de OBS, incluidas las nuevas funciones y documentos. |
| Descripción general de la función | Una introducción general sobre las funciones de OBS y regiones donde estas funciones están disponibles. |
| Descripción general del servicio | Conceptos básicos, ventajas, escenarios de la aplicación, facturación y gestión de permisos para OBS. |
| Primeros pasos | Una guía breve que le ayudará a comenzar con las operaciones básicas, como crear bucket, cargar y descargar objetos. |
| Guía de operación de la consola | Le guía a través de las operaciones de la consola OBS con ejemplos de uso elaborados. |
| Guía de herramientas | Le presenta las herramientas de OBS, incluyendo OBS Browser+, obsutil y obsfs, con ejemplos de uso proporcionados. |
| Guía de características de procesamiento de imágenes | Una guía de operación para el procesamiento de imágenes de OBS. Cubre conceptos básicos, funciones y preguntas frecuentes, y demuestra cómo usar esta función a través de la consola de OBS o por la llamada a OBS API. |

| Documento | Descripción |
|---|---|
| Guía de características del sistema de archivos paralelo | Presenta los conceptos básicos, escenarios de aplicación, limitaciones, operaciones de usuario y facturación para el sistema de archivos paralelos (PFS) de OBS. |
| Prácticas recomendadas | Una colección de las prácticas recomendadas de OBS para escenarios típicos, que le ayudan a alcanzar sus objetivos de negocio con un mejor rendimiento, costos más bajos y operaciones más convenientes. |
| Referencia de API | Enumera las REST API proporcionadas por OBS, incluidas muestras de solicitud, muestras de respuesta y descripciones de parámetros. Le ayuda a usar APIs para llevar a cabo un desarrollo secundario. |
| Referencia de los SDK | Describe los kits de desarrollo de software (SDK) de OBS en lenguajes de programación convencionales como Java, Python, C, GO, Android e iOS. Los SDK proporcionan detalles de parámetros, ejemplos y preguntas frecuentes para ayudarle a realizar sin problemas la instalación y el desarrollo. |
| FAQs | Le ofrece las respuestas a las preguntas más frecuentes sobre OBS. |
| Videos | Videos tutoriales sobre las funciones básicas de OBS se proporcionan para su referencia. |

2 Uso de OBS

Puede gestionar los recursos de OBS de las siguientes maneras:

| Herramienta | Descripción | Cómo utilizarla | Referencia |
|----------------|---|---|---|
| Consola de OBS | La consola de OBS es una interfaz gráfica de usuario basada en web. Puede administrar todos sus recursos de OBS a través de esta consola. | Necesita una cuenta (incluida la contraseña) o un usuario de IAM para iniciar sesión en la consola de OBS. Para obtener más información, consulte Iniciar sesión en la consola de OBS . | Guía de operación de la consola |
| SDK | Los SDK de OBS encapsulan las API proporcionadas por OBS para simplificar el desarrollo. Los usuarios pueden usar las funciones de API proporcionadas por los SDK de OBS para acceder a las capacidades de servicio de OBS. | Configure el punto final durante la preparación del entorno y establezca las claves de acceso (AK/SK) durante la inicialización. Para obtener más información, consulte Referencia del SDK . | Referencia de los SDK |

| Herramienta | Descripción | Cómo utilizarla | Referencia |
|--------------|---|---|---|
| API | OBS proporciona las REST API que admiten solicitudes de HTTP/HTTPS. Puedes llamar a estas API para crear, modificar y eliminar buckets, así como para cargar, descargar y eliminar objetos. | <p>Configure el endpoint durante la preparación del entorno y haga que las claves de acceso (AK y SK) formen parte de su firma en cualquier solicitud que envíe. Si utiliza las REST API para desarrollar un programa, debe calcular la firma basándose en el algoritmo de firma definido por OBS y agregar la firma a las solicitudes.</p> <p>Para obtener más información, consulte Autenticación de firma de usuario.</p> | Referencia de API |
| OBS Browser+ | OBS Browser+, una herramienta GUI, es un cliente de Windows para administrar los recursos de OBS. Con OBS Browser+, puede administrar fácilmente los recursos de OBS desde un extremo local. | <p>Descargue OBS Browser+ y utilice las claves de acceso (AK y SK) para la autenticación de identidad. Puede usar OBS Browser+ para acceder directamente a los recursos OBS o puede configurar una dirección de servidor para acceder a los recursos de OBS.</p> <p>Para obtener detalles acerca de cómo configurar el AK, el SK y el extremo del navegador OBS+, consulte ¿Dónde puedo obtener claves de acceso (AK y SK).</p> | Introducción a OBS Browser+ |
| obsutil | obsutil es una herramienta de línea de la orden para acceder a OBS. Puede usarlo para realizar operaciones comunes de configuración y gestión en OBS. Si está familiarizado con la interfaz de línea de comandos (CLI), obsutil es la herramienta recomendada para el procesamiento por lotes y tareas automatizadas. | <p>Descargue obsutil, configure la dirección del servidor y use las claves de acceso (AK y SK) para la autenticación de identidad.</p> <p>Para obtener detalles acerca de cómo configurar el AK, el SK y el punto final de obsutil, consulte Realización de la configuración inicial.</p> | Introducción a obsutil |

| Herramienta | Descripción | Cómo utilizarla | Referencia |
|--------------------|--|---|--|
| obsfs | obsfs, construido sobre la base de Filesystem in Userspace (FUSE), es una herramienta de sistema de archivos proporcionada por OBS para montar sistemas de archivos paralelos a sistemas operativos Linux. Le permite acceder fácilmente al espacio de almacenamiento prácticamente ilimitado en OBS de la misma manera que usaría un sistema de archivos local. | Descargue la herramienta obsfs y use las claves de acceso (AK y SK) para la autenticación de identidad. Para obtener detalles acerca de cómo configurar el AK y el SK de obsfs, consulte Inicialización de obsfs . | Introducción a los obsfs |

3 Acceso a OBS

Nombre de dominio OBS

Los dos conceptos siguientes están relacionados con los nombres de dominio de OBS:

1. **Endpoint:** OBS proporciona un endpoint para cada región. Un endpoint es un nombre de dominio para acceder a OBS en una región determinada y se utiliza para recibir solicitudes de acceso enviadas desde esa región. Para obtener una lista completa de los endpoints de OBS y las regiones correspondientes, consulte [Regiones y endpoints](#).
2. **Bucket domain name:** Cada bucket en OBS tiene un nombre de dominio. Un nombre de dominio es la dirección de Internet de un bucket y se puede usar para acceder al bucket a través de Internet. Se utiliza típicamente en el desarrollo de aplicaciones en la nube y escenarios de intercambio de datos.

Un nombre de dominio de bucket de OBS está en el formato *BucketName.Endpoint*.

BucketName indica el nombre de un bucket y *Endpoint* indica el nombre de dominio OBS de la región donde se encuentra el bucket.

Tabla 3-1 enumera el nombre de dominio del bucket y otros nombres de dominio de OBS, incluidos sus formatos y protocolos.

Tabla 3-1 Domain name de OBS

| Tipo | Estructura | Descripción | Tipo de protocolo |
|--------------------------------|--|---|-----------------------|
| Nombre de dominio de la región | [Estructura] Endpoint [Ejemplo] obs.ap-southeast-1.myhuaweicloud.com | Cada región tiene un endpoint, que es el nombre de dominio de la región. Para obtener una asignación completa entre regiones y endpoints de OBS, consulte Regiones y endpoints . | HTT PS HTT P |

| Tipo | Estructura | Descripción | Tipo de protocolo |
|--|--|---|-----------------------|
| Nombre de dominio de Bucket | [Estructura] BucketName.Endpoint [Ejemplo] bucketname.obs.ap-southeast-1.myhuaweicloud.com | Después de crear un bucket, puede usar el nombre de dominio para acceder al bucket. Puede ensamblar el nombre de dominio del bucket poniéndolo y el endpoint del servicio OBS juntos, o puede obtenerlo desde Consulta de información de buckets en OBS Console y OBS Browser+. | HTT PS HTT P |
| Nombre de dominio de objeto | [Estructura] BucketName.Endpoint/ObjectName [Ejemplo] bucketname.obs.ap-southeast-1.myhuaweicloud.com/object.txt | Después de cargar un objeto en un bucket, puede usar el nombre de dominio del objeto para tener acceso al objeto. Puede ensamblar el nombre de dominio del objeto poniéndolo, el endpoints del servicio OBS y el nombre del objeto juntos, o puede obtenerlo por Consulta de información de objeto en OBS Console o OBS Browser+. Alternativamente, puede llamar a la API GetObjectUrl a través de un SDK para obtener el nombre de dominio del objeto. | HTT PS HTT P |
| Alojamiento de sitio web estático | [Estructura] BucketName.obs-website.Endpoint [Ejemplo] bucketname.obs-website.ap-southeast-1.myhuaweicloud.com | Un static website domain name es un nombre de dominio de bucket cuando el bucket está configurado para alojar un sitio web estático. | HTT PS HTT P |
| Vinculación de nombre de dominio definido por el usuario | Nombres de dominio que han sido licenciados por el Ministerio de Industria y Tecnología de la Información (MIIT) de China. | Puede vincular un nombre de dominio de usuario a un bucket para que pueda acceder al bucket a través del nombre de dominio de usuario. | HTT P |

Puntos de conexión

OBS tiene un endpoint en cada región.

Generalmente, el endpoint transportado en una solicitud de acceso a OBS debe ser el endpoint de la región donde reside el recurso solicitado. Sin embargo, en algunos casos especiales, puede utilizar cualquier endpoint.

1. Escenarios en los que el endpoint de la solicitud debe ser coherente con el endpoint de la región donde residen los recursos solicitados

Cuando se accede a un bucket o a un objeto, el endpoint de la solicitud debe ser el punto final de la región donde reside el bucket o el objeto.

Por ejemplo, si el bucket **mybucket** está en la región **ap-southeast-1**, puede enumerar objetos en el bucket enviando una solicitud que se muestra en el siguiente ejemplo:

Un ejemplo correcto de solicitud y respuesta para listar objetos:

[Solicitud]

```
GET / HTTP/1.1
Host: mybucket.obs.ap-southeast-1.myhuaweicloud.com
Accept: */*
Date: Thu, 10 Mar 2016 08:51:25 GMT
Authorization: authorization
```

[Respuesta]

```
HTTP/1.1 200 OK
x-obs-request-id: 0001EF710C000001536176DA465E4E6G
x-obs-id-2: Rdj0zZvRkihRcjCQUqjkDGt8JuAgi2CGuLiP7Pv/cYYplsS0xTFJQHP5vSg5yOYC
Content-Type: application/xml
Date: Thu, 10 Mar 2016 16:58:12 GMT
x-obs-bucket-location: ap-southeast-1
Content-Length: 259
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ListBucketResult xmlns="http://obs.myhuaweicloud.com/doc/2015-06-30/">
  <Name>mybucket</Name>
  <Prefix/>
  <Marker/>
  <MaxKeys>1000</MaxKeys>
  <IsTruncated>>false</IsTruncated>
  <Contents>
    <Key>object001</Key>
    <LastModified>2015-07-01T00:32:16.482Z</LastModified>
    <ETag>"2fa3bcaaec668adc5da177e67a122d7c"</ETag>
    <Size>12041</Size>
    <Owner>
      <ID>b4bf1b36d9ca43d984fbc9491b6fce9</ID>
    </Owner>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
</ListBucketResult>
```

Si el endpoint de la solicitud no es coherente con el de la región solicitada, se devuelve un mensaje de error que indica que el bucket no existe.

En el ejemplo anterior, el bucket **mybucket** está en la región **ap-southeast-1**. Si utiliza el endpoint **cn-south-1** (**mybucket.obs.cn-south-1.myhuaweicloud.com**) para acceder al bucket, se devuelve HTTP 404, indicando que el bucket no existe. En este caso, puedes llamar a la API para [obtener la ubicación del bucket](#) para obtener el ID de región del bucket y, a continuación, volver a enviar la solicitud.

Un ejemplo incorrecto de solicitud y respuesta para listar objetos:

[Solicitud]

```
GET / HTTP/1.1
Host: mybucket.obs.cn-south-1.myhuaweicloud.com
Accept: */*
Date: Thu, 10 Mar 2016 08:51:25 GMT
Authorization: authorization
```

[Respuesta]

```
HTTP/1.1 404 NoSuchBucket
x-obs-request-id: 0001EF710C000001536176DA465E4E6H
x-obs-id-2: Rdj0zZvRkihRcjCQUqjkDGt8JuAgi2CGuLiP7Pv/cYYplsS0xTFJQHP5vSg5yOYL
Date: Thu, 10 Mar 2016 08:51:30 GMT
Content-Length: 0
```

2. Escenarios en los que se puede utilizar cualquier endpoint en una solicitud

Para obtener la lista de bucket o la información de región de un bucket, puedes usar cualquier punto de enlace de región en las solicitudes de API que envíes, ya que estas API buscan los bucket solicitados en todas las regiones.

Un ejemplo de solicitud y respuesta para obtener información de región de un bucket:

Por ejemplo, si el bucket **mybucket** está en la región **ap-southeast-1** y el endpoint de la región **cn-south-1** se usa en la solicitud, la información de ubicación del bucket todavía se puede obtener.

[Solicitud]

```
GET /?location HTTP/1.1
Host: mybucket.obs.cn-south-1.myhuaweicloud.com
Accept: */*
Date: Thu, 10 Mar 2016 08:51:25 GMT
Authorization: authorization
```

[Respuesta]

```
HTTP/1.1 200 OK
x-obs-request-id: 0001EF710C000001536176DA465E4E6G
x-obs-id-2: Rdj0zZvRkihRcjCQUqjkDGt8JuAgi2CGuLiP7Pv/cYYplsS0xTFJQHP5vSg5yOYC
Content-Type: application/xml
Date: Thu, 10 Mar 2016 16:58:12 GMT
Content-Length: length

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Location xmlns="http://obs.myhuaweicloud.com/doc/2015-06-30/">ap-
southeast-1</Location>
```

Acceso a OBS a través de Internet

El acceso a OBS a través de una red pública puede generar el tráfico entrante para operaciones de escritura (por ejemplo, cargar datos a OBS), así como tráfico saliente para operaciones de lectura (por ejemplo, descargar datos desde OBS). El tráfico entrante no incurre en tarifas, pero el tráfico saliente sí.

Para obtener más información sobre los precios del tráfico saliente a través de Internet, consulte [Detalles de precios del producto](#).

Si accede a OBS a través de una red pública, puede usar una dirección URL para especificar recursos en OBS. Una URL de OBS está estructurada de la siguiente manera:

Protocol://[BucketName.]Endpoint[:Port]/Object[?Param]

Tabla 3-2 Parámetros en una URL de OBS

| Parámetro | Descripción | Obligatoria u opcional |
|-----------|--|------------------------|
| Protocol | Indica el protocolo utilizado para enviar solicitudes, que pueden ser HTTP o HTTPS. HTTPS es un protocolo que garantiza un acceso seguro a los recursos. OBS soporta HTTP y HTTPS. | Obligatorio |

| Parámetro | Descripción | Obligatoria u opcional |
|------------|---|------------------------|
| BucketName | Nombre del bucket solicitado, que identifica de forma única un bucket en OBS. | Opcional |
| Endpoint | Nombre de dominio (endpoint) de la región donde se encuentra el bucket OBS. Para obtener más información sobre el nombre de dominio OBS de cada región, consulte Regiones y endpoints . | Obligatorio |
| Port | Indica el puerto habilitado para los protocolos utilizados para enviar solicitudes. El valor varía con la implementación del servidor de software. Si no se especifica ningún número de puerto, el protocolo utiliza el valor predeterminado. Cada protocolo de transmisión tiene su número de puerto predeterminado. En OBS, el número de puerto HTTP predeterminado es 80 y el de HTTPS es 443 . | Opcional |
| Object | Ruta de acceso del recurso de objeto solicitado. | Opcional |
| Param | Recurso específico contenido por un bucket u objeto. El valor predeterminado de este parámetro indica que se obtiene el bucket u objeto en sí. | Opcional |

Ejemplo: Tienes un bucket llamado **mybucket** en la región CN-Hong Kong **ap-southeast-1**. El bucket contiene un objeto llamado **myfolder/myfile.txt**. La dirección URL para acceder al objeto a través de la red pública es la siguiente:

<https://mybucket.obs.ap-southeast-1.myhuaweicloud.com/myfolder/myfile.txt>

NOTA

Todas las solicitudes de API, excepto la de los objetos de enumeración, deben contener el **BucketName**. Teniendo en cuenta el rendimiento y la fiabilidad de la resolución de DNS, OBS requiere que el nombre del bucket preceda al **Endpoint** cuando se construye una solicitud con un nombre del bucket para formar un nombre de dominio de tres niveles, también mencionado como nombre de dominio de acceso de alojamiento virtual.

Acceso a OBS a través de la Intranet

El acceso a OBS a través de una red privada se refiere al acceso a OBS a través de la red de comunicación interna entre diferentes servicios de Huawei Cloud. El acceso a OBS a través de una red privada puede generar tráfico entrante para operaciones de escritura (por ejemplo, cargar datos a OBS), así como tráfico saliente para operaciones de lectura (por ejemplo, descargar datos desde OBS). En este caso, ni el tráfico entrante ni saliente incurre en tarifas.

Por ejemplo, puede acceder a OBS desde un Elastic Cloud Server (ECS) a través de una red privada. Dicho acceso no es susceptible de problemas de calidad de la red pública, y también reduce los costos.

OBS le proporciona las prácticas recomendadas para la configuración. Para obtener más información, consulte [Acceso a OBS a través de Intranet](#).

Comprobación de la versión de OBS (OBS 2.0 u OBS 3.0)

La arquitectura OBS ha pasado por dos generaciones: OBS 2.0 y OBS 3.0. Un bucket recién creado se almacena en OBS 3.0 por defecto, y la versión del bucket es OBS 3.0. Sin embargo, los bucket creados anteriormente todavía están en OBS 2.0.

Las características y funciones básicas de OBS son compatibles con OBS 3.0 y OBS 2.0. Algunas características nuevas solo son compatibles con OBS 3.0, como el procesamiento de imágenes y la replicación entre regiones.

Puedes comprobar la versión del bucket en la consola de OBS o usar la API **Head Bucket** para comprobar si tu bucket está en OBS 2.0 u OBS 3.0. Dichos aspectos se detallan a continuación:

Método 1: Inicie sesión en la consola de OBS y compruebe la información básica del bucket.

Si la **versión del bucket** es **3.0**, el bucket se almacena en OBS 3.0. Si no es así, el bucket se almacena en OBS 2.0.

Método 2: Usa la API Head Bucket para comprobar la versión del bucket.

Solicitud de muestra:

```
HEAD / HTTP/1.1
Host: bucketname.obs.ap-southeast-1.myhuaweicloud.com
Accept: */*
Date: WED, 01 Jul 2015 02:23:25 GMT
Authorization: auth string
```

Respuesta de la muestra:

```
HTTP/1.1 200 OK
Server: OBS
x-obs-request-id: BF2600000163D80E4C5F20FDD5BD0085
Content-Type: application/xml
x-obs-version: 3.0
x-obs-id-2: 32AAAQAAEAABAAAQAAEAABAAAQAAEAABCS8ws9100114oMWndniV7XmdAvfewrQq
Date: WED, 01 Jul 2015 02:23:25 GMT
Content-Length: 0
```

En el mensaje de respuesta, **x-obs-version: 3.0** indica que el bucket está almacenado en OBS 3.0. Si este encabezado no existe o el valor de este encabezado se muestra de otro modo, el bucket se almacena en OBS 2.0.

Para obtener más información sobre la API de **Head Bucket**, consulta [Obtención de metadatos de Bucket](#).

4 Descripción general de la función

Para obtener más información, consulte [Descripción general de la función de OBS](#).

5 Gestión de permisos

Escenario de la aplicación

De forma predeterminada, los recursos de OBS (buckets y objetos) son privados. Solo los propietarios de recursos pueden acceder a sus recursos de OBS. Sin autorización, otros usuarios no pueden acceder a los recursos de OBS. Control de permisos de OBS significa conceder permisos a otras cuentas o usuarios de IAM mediante la edición de políticas de acceso. Por ejemplo, si tiene un bucket, puede autorizar a otro usuario de IAM a cargar objetos en el bucket. También puedes abrir los bucket al público, para que cualquier persona pueda acceder a tus bucket a través de Internet. OBS ofrece varios métodos para ayudarle a asignar permisos de recursos a otros. Los propietarios de recursos pueden formular diferentes políticas de control de permisos basadas en los requisitos de servicio para garantizar la seguridad de los datos.

¿Cómo usar esta función?

Puede usar la consola OBS, API, SDK, OBS Browser+ u obsutil para configurar los permisos.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Control de permisos |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Permisos y acciones admitidas Configuración de la política de bucket Configuración de la ACL de Bucket Configuración de la ACL de objeto |
| OBS Browser+ | - |
| obsutil | Definición de las propiedades de bucket Establecimiento de las propiedades de objeto |

6 Clases de almacenamiento

Escenario de la aplicación

OBS proporciona tres clases de almacenamiento: Standard, Infrequent Access, y Archive.

Las diferentes clases de almacenamiento cumplen con diferentes requisitos de rendimiento y costes de almacenamiento.

- La clase de almacenamiento Standard cuenta con baja latencia de acceso y alto rendimiento. Por lo tanto, es ideal para almacenar cantidades masivas de archivos calientes (a los que se accede con frecuencia cada mes) o archivos pequeños (menos de 1 MB). Los escenarios de aplicación incluyen análisis de big data, aplicaciones móviles, videos calientes y aplicaciones sociales.
- La clase de almacenamiento Infrequent Access es ideal para almacenar datos a los que se accede con poca frecuencia (menos de 12 veces al año) pero con requisitos de respuesta rápida. Los escenarios de aplicación incluyen sincronización de archivos, uso compartido de archivos y copia de seguridad empresarial.
- La clase de almacenamiento Archive es ideal para archivar los datos a los que rara vez se accede (una vez al año en promedio). Los escenarios de aplicación incluyen archivo de datos y copias de seguridad de datos a largo plazo. La clase de almacenamiento Archive es segura, duradera y económica, y se puede utilizar para reemplazar las bibliotecas de cintas. Sin embargo, puede tardar horas en restaurar los datos de la clase de almacenamiento Archive.

Comparación de clases de almacenamiento

| Artículo Comparado | Estándar | Acceso poco frecuente | Archivados |
|--------------------|---|--|---|
| Característica | Rendimiento de primer nivel, altamente confiable y disponible | Acceso a un servicio de almacenamiento confiable, de bajo costo y en tiempo real | Almacenamiento a largo plazo para datos archivados de un costo muy bajo |

| Artículo Comparado | Estándar | Acceso poco frecuente | Archivados |
|--|--|--|--|
| Escenarios de la aplicación | Aplicaciones alojadas en la nube, uso compartido de datos, uso compartido de contenido y almacenamiento de datos activos | Aplicaciones de disco web, backup empresarial, archivo activo y supervisión de datos | Archivado, almacenamiento de imágenes médicas, almacenamiento de video y sustitución de bibliotecas de cintas |
| Durabilidad | 99.999999999% | 99.999999999% | 99.999999999% |
| Durabilidad (multi-AZ) | 99.999999999% | 99.999999999% | Multi-AZ no soportada |
| Disponibilidad | 99.99% | 99% | 99% |
| Disponibilidad (multi-AZ) | 99.995% | 99.5% | Multi-AZ no soportada |
| Unidad mínima de medida ^a | 64 KB | 64 KB | 64 KB |
| Duración mínima de almacenamiento ^b | N/A | 30 días | 90 Días |
| Restauración de datos | N/A | Se carga según la cantidad de datos restaurados (medido en GB) | Dos opciones de restauración: urgente y estándar Se carga según la cantidad de datos restaurados (medido en GB) |
| Procesamiento de imágenes | Se admite | Se admite | No se admite |

 **NOTA**

a: La unidad mínima de medida se refiere al límite de tamaño inferior para la facturación de objetos. Por ejemplo, si carga un objeto Standard de 32 KB (menor que la unidad de medida mínima de 64 KB), se le facturará un objeto de 64 KB.

b: La duración mínima de almacenamiento se refiere al límite inferior de almacenamiento para la facturación de objetos. Por ejemplo, si un objeto de acceso poco frecuente se ha almacenado en OBS durante 20 días (más corto que la duración mínima de almacenamiento de 30 días) y después se ha eliminado, se le facturará 30 días.

Clases de almacenamiento de bucket y clases de almacenamiento de objetos

Puede especificar la clase de almacenamiento de un bucket mientras que lo crea. También puede cambiar la clase de almacenamiento de un bucket después de crearlo.

Un objeto hereda la clase de almacenamiento del bucket donde se carga. Puede especificar una clase de almacenamiento para un objeto al cargarlo, o puede cambiar la clase de almacenamiento de objeto después de cargar el objeto.

El cambio de la clase de almacenamiento de un bucket no va a cambiar las clases de almacenamiento de los objetos existentes en el bucket, pero los objetos cargados recientemente heredan la nueva clase de almacenamiento de forma predeterminada.

NOTA

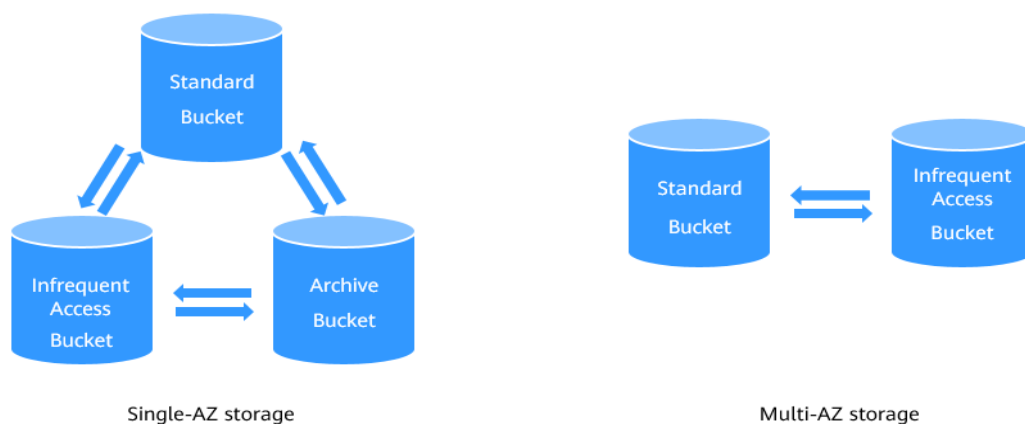
De forma predeterminada, los objetos de la clase de almacenamiento Archive deben restaurarse a la clase de almacenamiento Standard antes de poder descargarse. Puede habilitar la **Lectura directa** para descargar directamente objetos en la clase de almacenamiento Archive sin restaurarlos primero. Se cobrará la lectura directa.

Cambio de clases de almacenamiento de Bucket

La clase de almacenamiento de un bucket solo se puede cambiar manualmente.

Figura 6-1 enumera las reglas de transición de clase de almacenamiento de bucket. Las clases de almacenamiento Standard e Infrequent Access de OBS admiten las políticas de redundancia de datos de un solo AZ y de múltiples AZ, mientras que la clase de almacenamiento Archive solo admite la política de redundancia de un solo AZ. La política de redundancia de datos de un bucket no se puede modificar, incluso cuando se cambia su clase de almacenamiento.

Figura 6-1 Reglas de cambio de clase de almacenamiento de bucket



Nota:

- Cambiar la clase de almacenamiento de un bucket no cambia las clases de almacenamiento de los objetos existentes en el bucket. La clase de almacenamiento de un objeto cargado posteriormente heredará la nueva clase de almacenamiento del bucket de forma predeterminada. También puede **configurar reglas de ciclo de vida** para cambiar las clases de almacenamiento de objetos en un lote.

Por ejemplo, si el **bucket1** está en la clase de almacenamiento Standard y contiene **objeto1**, cuando la clase de almacenamiento del **bucket1** se pasa a Infrequent Access, **objeto1** todavía está en la clase de almacenamiento Standard. Si carga el **objeto2** en el **bucket1** después de la transición, el **objeto2** estará en la clase de almacenamiento Infrequent Access.

- Si la clase de almacenamiento de un bucket se cambia de Archive a Standard o Infrequent Access, los objetos del bucket no se restaurarán automáticamente desde la clase de almacenamiento Archive.

Cambio de clases de almacenamiento de objetos

La clase de almacenamiento de un objeto se puede cambiar manualmente o automáticamente. **Figura 6-2** enumera las reglas de transición de clase de almacenamiento de objetos.

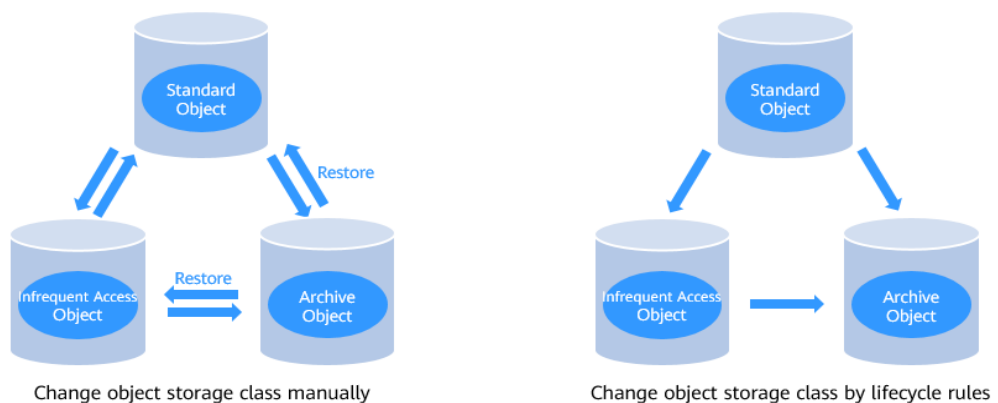
- Transición manual: Después de cargar un objeto, puede cambiar manualmente sus clases de almacenamiento.

Los objetos se pueden cambiar manualmente de la clase Standard a la de almacenamiento Infrequent Access o Archive, o de Infrequent Access a la clase Standard o Archive, pero los objetos de la clase de almacenamiento Archive deben restaurarse antes de cambiar manualmente a la clase de almacenamiento Standard o Infrequent Access. La transición de objetos en la clase de almacenamiento Infrequent Access o Archive conlleva tasas de recuperación. Seleccione una opción de transición adecuada en función de sus necesidades.

- Transición automática: OBS proporciona reglas de ciclo de vida para que pueda realizar la transición automática de objetos de una clase de almacenamiento a otra. Para más detalles, consulte **Gestión del ciclo de vida**.

Los objetos se pueden cambiar automáticamente de Standard a Infrequent Access o Archive, o de Infrequent Access a Archive.

Figura 6-2 Reglas de cambio de clase de almacenamiento de objetos



Nota:

- Después de que los objetos de la clase de almacenamiento Standard se transfieran a la clase de almacenamiento Infrequent Access o Archive, su estado de restauración es Sin restaurar.
- Los objetos de la clase Infrequent Access tienen un período de almacenamiento mínimo de 30 días y los de Archive tienen un período de almacenamiento mínimo de 90 días. Si un objeto se pasa a otra clase de almacenamiento antes de almacenarse durante el período de almacenamiento mínimo requerido, debe pagar por una duración de almacenamiento completa (30 días o 90 días).

¿Cómo usar esta función?

Puede usar la consola OBS, API, SDK, OBS Browser+ u obsutil para configurar las clases de almacenamiento para buckets y objetos.

| Herramienta | Referencia |
|----------------|---|
| Consola de OBS | <p>Especificación de una clase de almacenamiento durante la creación de bucket</p> <p>Especificación de una clase de almacenamiento durante las cargas de objetos</p> |
| SDKs | <p>OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK.</p> |
| APIs | <p>Configuración de la clase de almacenamiento predeterminada para un bucket</p> <p>Especificación de una clase de almacenamiento durante la creación del bucket (adición del encabezado <code>x-obs-storage-class</code>)</p> <p>Especificación de una clase de almacenamiento durante las cargas de objetos con PUT (adición del encabezado <code>x-obs-storage-class</code>)</p> <p>Especificación de una clase de almacenamiento durante cargas de objetos con POST (adición del encabezado <code>x-obs-storage-class</code>)</p> |
| OBS Browser+ | - |
| obsutil | <p>Especificación de una clase de almacenamiento durante la creación de bucket</p> <p>Especificación de una clase de almacenamiento durante la configuración de propiedades de bucket</p> <p>Especificación de una clase de almacenamiento durante las cargas de objetos</p> <p>Especificación de una clase de almacenamiento durante la configuración de propiedades de objeto</p> |

7 Gestión de buckets

7.1 Información general

Un bucket es un contenedor virtual utilizado para almacenar **objetos** en OBS. OBS proporciona un almacenamiento plano en forma de buckets y objetos. A diferencia de la estructura convencional por directorios multicapa de los sistemas de archivos, todos los objetos de un bucket se almacenan en la misma capa lógica.

Cada bucket tiene sus propios atributos, como las clases de almacenamiento (para obtener más información, consulte **Clases de almacenamiento**), permisos de acceso, y **regiones**. Puede especificar permisos de acceso, clases de almacenamiento y regiones al crear buckets. También puede configurar atributos avanzados para cumplir con los requisitos de almacenamiento en diferentes escenarios.

OBS proporciona las siguientes clases de almacenamiento: Standard, Infrequent Access, y Archive. Con soporte para diferentes clases de almacenamiento, OBS satisface diversos requisitos de rendimiento y costos de almacenamiento. Al crear un bucket, puede especificar una clase de almacenamiento para el bucket, que se puede cambiar más adelante.

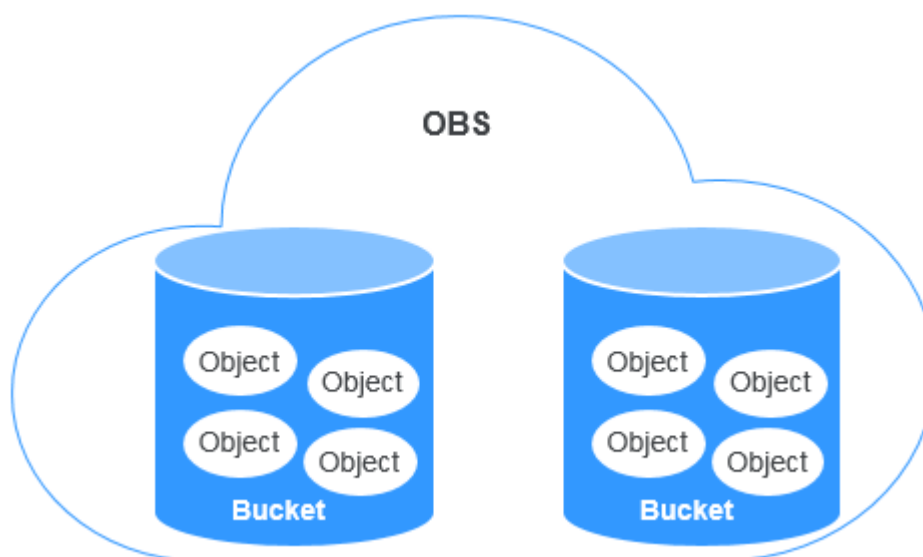
En OBS, cada nombre de bucket debe ser único y no se puede cambiar. La región en la que reside un bucket no se puede cambiar una vez creado el bucket. Cuando se crea un bucket, la lista de control de acceso de bucket (ACL) se genera de forma predeterminada. La ACL se utiliza para conceder los permisos de recursos a los usuarios. Solo los usuarios autorizados pueden realizar operaciones como crear, eliminar, consultar y configurar buckets.

Una cuenta (incluidos todos los usuarios de IAM de la cuenta) puede crear un máximo de 100 buckets. Puede aprovechar la capacidad de control de permisos de grano fino de OBS para planificar y usar buckets correctamente. Por ejemplo, puede crear carpetas en un bucket basado en prefijos de objeto y utilizar **control de permisos de grano fino** para aislar datos entre diferentes departamentos. No hay ninguna restricción sobre la cantidad y el tamaño total de los objetos en un bucket.

Como OBS se basa en una arquitectura RESTful sobre HTTP y HTTPS, puede usar localizadores uniformes de recursos (URL) para localizar recursos.

Figura 7-1 ilustra la relación entre buckets y objetos en OBS.

Figura 7-1 Buckets y objetos



Puede **usar diferentes métodos** para **crear buckets** basados en tus hábitos de uso y necesidades de almacenamiento. Después de crear un bucket, puede utilizar diferentes formas de **cargar archivos (datos) en el bucket**, donde estos archivos se almacenan como objetos. En OBS, los buckets y objetos se encuentran en **diferentes regiones**. Puede utilizar los diferentes métodos para acceder al mismo bucket y recursos en la misma región.

7.2 Creación de un bucket

Escenario de la aplicación

Los buckets son contenedores que almacenan objetos en OBS. Necesitas crear un bucket antes de poder comenzar a almacenar datos en OBS.

Prerrequisitos:

Para crear un bucket, asegúrate de tener una cuenta registrada, saldo de cuenta suficiente, claves de acceso (AK y SK) y la información del endpoint. Para obtener más información, consulte **Primeros pasos**.

Restricciones

- Después de crear un bucket, su nombre y región no se pueden cambiar. Especifique una región y un nombre de bucket adecuados al crear el bucket.
- Una cuenta (incluidos todos los usuarios de IAM de la cuenta) puede crear un máximo de 100 buckets. Puede aprovechar la capacidad de control de permisos de grano fino de OBS para planificar y usar buckets correctamente. Por ejemplo, puede crear carpetas en un bucket basado en prefijos de objeto y usar **control de permisos de grano fino** para implementar el aislamiento de permisos entre departamentos. Sin embargo, no hay ninguna restricción sobre el número y el tamaño total de los objetos en un bucket.
- Un nombre de bucket forma parte del nombre de dominio de acceso y debe resolverse. Por lo tanto, el nombre del bucket debe ajustarse a las **reglas de nombres de dominio**

DNS. Cuando recibe una solicitud de creación de bucket, la OBS comprueba estrictamente el nombre del bucket. Un nombre de bucket:

- Debe ser único en todas las cuentas y regiones.
- Debe contener entre 3 y 63 caracteres. Solo se permiten letras minúsculas, dígitos, guiones (-) y puntos (.). Si un nombre de bucket contiene un período de (.), se puede activar la verificación del certificado de seguridad al acceder al bucket o a sus objetos.
- El nombre no puede comenzar ni finalizar con un punto (.) ni un guion (-), y no puede contener dos puntos consecutivos (..) ni un punto (.) y un guion (-) adyacentes entre sí.
- No se puede formatear como una dirección IP.

AVISO

Si utiliza un bucket con (.) de períodos en su nombre para acceder a OBS, el cliente mostrará un mensaje indicando que el bucket es arriesgado, por ejemplo, puede mostrarse una alarma roja en la solicitud de seguridad del navegador. Esto se debe a que el certificado comodín SSL solo coincide con bucket sin períodos (.) en sus nombres cuando se usa HTTPS para el acceso OBS. Te recomendamos que evites usar (.) de puntos en los nombres de bucket.

¿Cómo usar esta función?

Puede usar la consola OBS, API, SDK, OBS Browser+ u obsutil para crear un bucket.

| Herramienta | Referencia |
|----------------|---|
| Consola de OBS | Creación de un bucket |
| SDK | OBS admite kits de desarrollo de software (SDK) en lenguajes de programación convencionales. Para obtener más información, consulte la guía para desarrolladores en la página Información general del SDK . |
| API | Creación de un bucket |
| OBS Browser+ | - |
| obsutil | Creación de un bucket |

¿Cómo acceder a un bucket?

Después de crear un bucket, puede usar el nombre de dominio para acceder al bucket. Puede ensamblar el nombre de dominio poniendo el nombre del bucket y el endpoint juntos, o puede obtenerlo [consultando la información básica del bucket](#) en OBS Console o OBS Browser+.

Un nombre de dominio de acceso se estructura de la siguiente manera:

[Structure] *BucketName.Endpoint*

[Example] *bucketname.obs.ap-southeast-1.myhuaweicloud.com*

Causas de los fallos en la creación de bucket y soluciones

Para obtener más información, consulte [Error al crear un bucket](#)

7.3 Enumeración de buckets

Escenario de la aplicación

Puede enumerar todos los bucket creados para consultar su información.

¿Cómo usar esta función?

Puede usar la consola OBS, API, SDK, OBS Browser+ u obsutil para enumerar un bucket.

| Herramienta | Referencia |
|----------------|---|
| Consola de OBS | Inicie sesión en la Consola de OBS y seleccione Object Storage . A continuación, se mostrarán todos los buckets de tu cuenta. |
| SDK | OBS admite kits de desarrollo de software (SDK) en lenguajes de programación convencionales. Para obtener más información, consulte la guía para desarrolladores en la página Información general del SDK . |
| API | Enumeración de buckets |
| OBS Browser+ | Iniciar sesión en OBS Browser+. Todos los buckets de tu cuenta se muestran en la lista de buckets. |
| obsutil | Enumeración de buckets |

7.4 Consulta de información de buckets

Escenario de la aplicación

Después de crear un bucket en la consola de OBS, puede ver la información del bucket, incluidas sus estadísticas y detalles básicos.

¿Cómo usar esta función?

Puede usar la consola de OBS, API, SDK, OBS Browser+ u obsutil para ver información básica del bucket.

| Herramienta | Referencia |
|----------------|---|
| Consola de OBS | Consulta de información básica de un bucket |

| Herramienta | Referencia |
|--------------|---|
| SDK | OBS admite kits de desarrollo de software (SDK) en lenguajes de programación convencionales. Para obtener más información, consulte la guía para desarrolladores en la página Información general del SDK . |
| API | Obtención de metadatos de bucket |
| OBS Browser+ | - |
| obsutil | Consulta de propiedades de bucket |

7.5 Gestión de cuotas de bucket

Escenario de la aplicación

De forma predeterminada, ni el sistema OBS completo ni ningún bucket individual tienen limitaciones sobre el tamaño total o la cantidad de objetos que se pueden almacenar. Puede establecer una cuota para un bucket para limitar el tamaño total de los objetos que se pueden cargar en el bucket. Si el tamaño total del objeto alcanza el límite superior, las cargas de objetos fallarán.

Una cuota de bucket solo puede controlar la carga de objetos después de establecerla. Si la cuota de bucket es menor que la capacidad de los objetos cargados, los objetos existentes no se eliminarán, pero no se podrán cargar nuevos objetos. En este caso, solo puede cargar nuevos objetos después de eliminar algunos objetos existentes hasta que la capacidad de almacenamiento utilizada sea inferior al límite de cuota.

NOTA

- Una cuota de bucket debe ser un entero no negativo, en bytes. El valor máximo es $2^{63} - 1$.
- OBS no proporciona una API para eliminar cuotas de bucket. Puede establecer la cuota de bucket en 0 para cancelar el límite de cuota.

¿Cómo usar esta función?

Puedes usar API y SDK para gestionar cuotas de bucket.

| Herramienta | Referencia |
|-------------|---|
| SDK | OBS admite kits de desarrollo de software (SDK) en lenguajes de programación convencionales. Para obtener más información, consulte la guía para desarrolladores en la página Información general del SDK . |
| API | Configuración de la cuota de almacenamiento de bucket |

7.6 Eliminación de buckets

Escenario de la aplicación

Puede eliminar buckets no deseados para liberar su cuota. Asegúrese de que el bucket esté vacío antes de eliminarlo.

Un bucket vacío debe cumplir con los siguientes requisitos:

- No hay ningún objeto ni ninguna versión histórica de un objeto en el bucket.
- No hay ninguna tarea de carga de varias partes sin fusionar en el bucket, es decir, no hay ningún fragmento en el bucket.

NOTA

- Si el control de versiones está habilitado para el bucket, asegúrese de que todas las versiones históricas y versiones con el **Delete Marker** (que también se consideran versiones históricas) se hayan eliminado.
- El nombre de un bucket eliminado se puede reutilizar al menos 30 minutos después de la eliminación.

¿Cómo usar esta función?

Puede usar la consola OBS, API, SDK, OBS Browser+ u obsutil para eliminar un bucket.

| Herramienta | Referencia |
|----------------|---|
| Consola de OBS | Eliminación del bucket |
| SDK | OBS admite kits de desarrollo de software (SDK) en lenguajes de programación convencionales. Para obtener más información, consulte la guía para desarrolladores en la página Información general del SDK . |
| API | Eliminación de los bucket |
| OBS Browser+ | - |
| obsutil | Eliminación del bucket |

Causas de los fallos en la eliminación de bucket y soluciones

Para obtener más información, consulte [Error al eliminar un bucket](#).

8 Gestión de objetos

8.1 Descripción general del objeto

Un objeto es la unidad básica de almacenamiento de datos en OBS. Consiste en los datos y metadatos de un objeto, los cuales describen los atributos del mismo. Los datos cargados en OBS se almacenan como objetos en **buckets**.

Un objeto consta de datos, metadatos y una clave.

- Una clave especifica el nombre de un objeto. Una clave de objeto es una cadena UTF-8 que tiene entre 0 y 1024 caracteres. Cada objeto se identifica de forma única dentro de un bucket mediante una clave.
- Metadatos: Especifican la descripción del objeto. Los metadatos incluyen los metadatos del sistema y los metadatos personalizados. Los metadatos se cargan en OBS como pares clave-valor.
 - Los metadatos del sistema son generados automáticamente por OBS y se utilizan para procesar datos de objetos, como la fecha, la longitud del contenido, la última modificación y el contenido MD5.
 - Puede especificar metadatos personalizados para describir el objeto al cargarlo en OBS.
- Datos: se refiere al contenido que contiene el objeto.

Por lo general, los objetos se gestionan como archivos. Sin embargo, OBS es un servicio de almacenamiento basado en objetos y no existe un concepto de archivos y carpetas. Para una fácil gestión de datos, OBS proporciona un método para simular carpetas. Al agregar una barra diagonal (/) en un nombre de objeto, por ejemplo, **test/123.jpg**, puede simular **test** como una carpeta y **123.jpg** como el nombre de un archivo bajo la carpeta **test**. Sin embargo, la clave del objeto sigue siendo **test/123.jpg**.

Al cargar un objeto, puede establecer una **clase de almacenamiento** para el objeto. Si no se especifica ninguna clase de almacenamiento, la clase de almacenamiento de objetos mantiene la misma que la del bucket. También puede cambiar la clase de almacenamiento de un objeto existente en un bucket.

Usted puede **Carga de un objeto** en **¿Cómo usar esta función?** según sus hábitos y escenarios de servicio. A continuación, OBS almacena los archivos como objetos en el bucket. En OBS, los bucket se encuentran en **diferentes regiones**. Independientemente del método que utilice, puede acceder al mismo bucket y sus recursos en la misma región.

Directrices sobre la asignación de nombres de claves de objetos

Aunque se puede utilizar cualquier carácter UTF-8 en un nombre de clave de objeto, nombrar las claves de objeto de acuerdo con las siguientes directrices puede ayudar a maximizar la compatibilidad de las claves de objeto con otras aplicaciones. Las formas de analizar caracteres especiales varían según las aplicaciones. Las siguientes directrices ayudan a que los nombres de claves de objeto cumplan considerablemente los requisitos de DNS, caracteres de seguridad web, analizadores XML y otras API.

Los siguientes conjuntos de caracteres se pueden utilizar de forma segura en los nombres de clave.

| | |
|--|---|
| Caracteres alfanuméricos (también conocidos como caracteres no reservados) | [0-9a-zA-Z] |
| Caracteres especiales (también conocidos como caracteres reservados) | Signo de exclamación (!) Guion (-) Guion bajo (_) Punto (.) Asterisco (*) Comillas simples (') Soporte izquierdo "(" Soporte derecho ")" |

Los siguientes son ejemplos de nombres de clave de objeto válidos:

```
4my-organization
my.great_photos-2014/jan/myvacation.jpg
videos/2014/birthday/video1.wmv
```

Codificación porcentual de caracteres reservados

Si un carácter reservado tiene un significado especial (conocido como propósito reservado) en un URI y el carácter debe usarse para otros propósitos en el URI, este carácter debe estar codificado porcentualmente. Utilice UTF-8 para codificar caracteres que no sean ASCII. De lo contrario, los nombres de los objetos que se suben a OBS pueden ser diferentes de lo que se espera. Por ejemplo, si el carácter reservado "/" se utiliza como delimitador de componentes de ruta en un URI, el carácter tiene un significado especial (separando un nombre de bucket de un nombre de objeto). Si se utiliza "/" en un componente de la ruta de acceso en un URI, use tres caracteres "%2F" o "%2f" para reemplazar a "/".

Caracteres que pueden requerir un procesamiento especial

Caracteres que requieren codificación en un nombre de clave

- Ampersandia (&)
- Signo de dólar (\$)
- Punto y coma (;)
- Dos puntos (:)

- La OBS decodifica los signos más (+) en un URI de solicitud en espacios. Si un nombre de clave de objeto original contiene signos más (+), debe codificarse en %2B antes de colocarse en el URI de solicitud.
- Espacio: En algunos casos se puede perder un gran número de espacios consecutivos.
- Señal de igualdad (=)
- Arroba (@)
- Coma (,)
- Signo de interrogación (?)
- Caracteres ASCII: 00-1F en forma hexadecimal (0-31 en forma decimal) y 7F (127 en forma decimal)

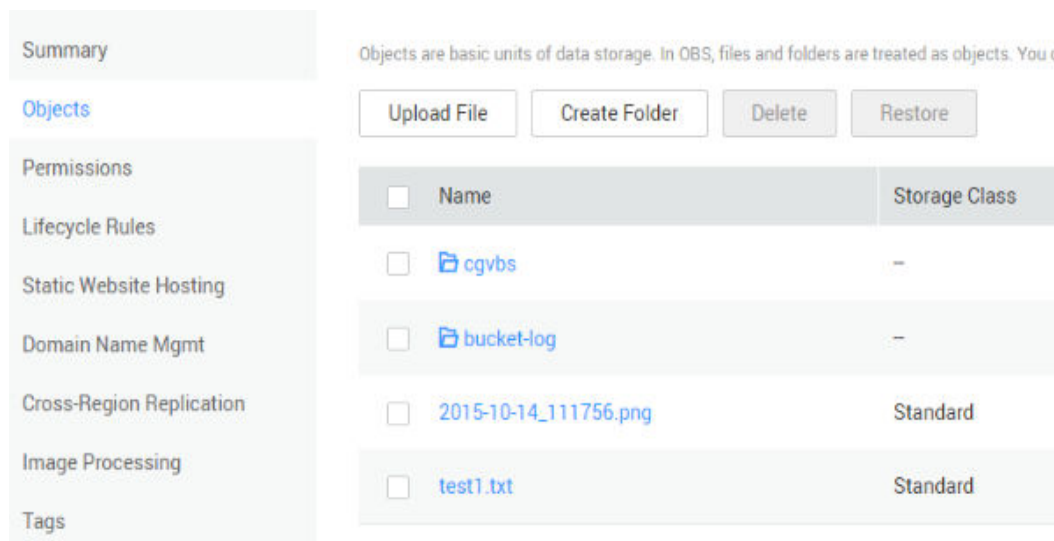
Evite utilizar los siguientes caracteres en los nombres de clave, ya que estos caracteres requieren mucho procesamiento especial para mantener la coherencia en todas las aplicaciones.

- Barra diagonal inversa (\)
- Corchete izquierdo ({)
- Caracteres ASCII no imprimibles (128-255 caracteres decimales)
- Símbolo de insertar (^)
- Corchete derecho (})
- Porcentaje de caracteres (%)
- Acento/Untick (`)
- Paréntesis cuadrado derecho (])
- Comillas
- Signo de mayor que (>)
- Paréntesis cuadrado izquierdo ([)
- La Línea de la Concepción (~)
- Signo de menos que (<)
- Hashtag (#)
- Barra vertical (|)

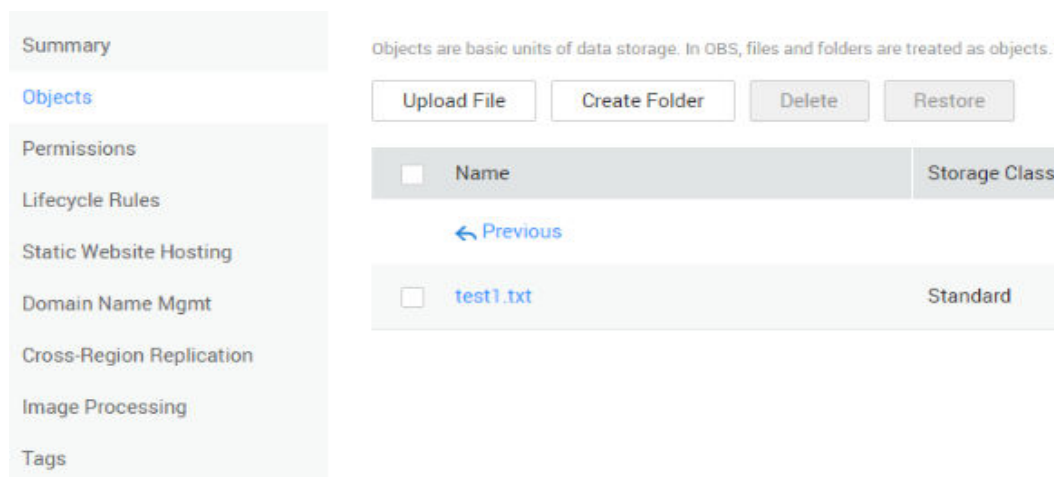
Tenga en cuenta que OBS adopta una estructura plana, donde se crean los bucket y se almacenan objetos en los bucket. No hay sub-bucket o sub-carpetas en la estructura. Sin embargo, puede usar prefijos de nombre de clave y delimitadores para deducir la estructura lógica como lo hace la consola de OBS. El concepto de carpeta está disponible en la consola de OBS. Suponga que su bucket (**companybucket**) contiene cuatro objetos con las siguientes claves de objeto:

- bucket-log/log01.txt
- cgvbs/test1.txt
- 2015-10-14_111756.png
- test1.txt

La Consola OBS utiliza el prefijo del nombre de la clave (**bucket-log/** y **cgvbs/**) y el separador (/) para mostrar la estructura de carpetas, como se muestra en la siguiente figura.



Las claves **2015-10-14_111756.png** y **key test1.txt** no tienen prefijos. Por lo tanto, los objetos aparecen en el nivel raíz del bucket. Si abre la carpeta **cgvbs/**, verá que contiene el objeto **test1.txt**.



Suponga que su bucket (**companybucket**) contiene dos objetos con las siguientes claves de objeto:

- obj
- 1/./obj

Si llama a una API (por ejemplo, usando el SDK) para obtener los objetos, puede obtenerlos correctamente. Si accede a los objetos desde la consola, el objeto **obj** se devolverá en función de la relación relativa. Debido a que **./** tiene una semántica especial en un URI, evite usarlo en claves de objeto.

8.2 Creación de la carpeta

Escenario de la aplicación

Puede crear una carpeta en un bucket para facilitar la clasificación y la gestión de los datos almacenados en OBS.

A diferencia de un sistema de archivos, OBS no implica los conceptos de archivo y carpeta. Para una fácil gestión de datos, OBS proporciona un método para simular carpetas. En OBS, un objeto se simula como una carpeta agregando una barra diagonal (/) al final del nombre del objeto en OBS Console. Cuando los objetos se enumeran llamando a la API, el nombre del objeto obtenido es la ruta del objeto, y el contenido que sigue a la última barra diagonal (/) es el nombre del objeto real. Si el trazado termina con una barra diagonal (/), indica que el objeto es una carpeta. La profundidad jerárquica del objeto no afecta al rendimiento del acceso al objeto.

¿Cómo usar esta función?

Puede usar la consola OBS, API, SDK, OBS Browser+ u obsutil para crear una carpeta.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Creación de una carpeta |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . NOTA Crear una carpeta en OBS es crear un objeto cuyo tamaño es 0 y cuyo nombre termina con una barra diagonal (/), en esencial. |
| API | - NOTA Crear una carpeta en OBS es crear un objeto cuyo tamaño es 0 y cuyo nombre termina con una barra diagonal (/), en esencial. |
| OBS Browser+ | - |
| obsutil | Creación de una carpeta |

8.3 Carga de un objeto

Escenario de la aplicación

Puede subir archivos o carpetas a un bucket OBS existente. Si desea clasificar los archivos que se van a cargar, cree una carpeta y suba los archivos relacionados a la carpeta.

Para obtener más información sobre cómo crear una carpeta, consulte [Creación de la carpeta](#).

Estos archivos pueden ser textos, imágenes, videos o cualquier otro tipo de archivos.

Restricciones

- La consola de OBS tiene restricciones sobre el tamaño y el número de archivos cargados.
 - En las regiones donde se admite la carga por lotes, se pueden cargar un máximo de 100 archivos a la vez, con un tamaño total de no más de 5 GB. Si carga un solo archivo en modo de carga por lotes, este archivo no puede exceder los 5 GB de tamaño.

- En las regiones donde no se admite la carga por lotes, solo se puede cargar un archivo a la vez, con un tamaño de no más de 50 MB.
- Puede usar OBS Browser+ y obsutil para cargar archivos con un tamaño total inferior a 48.8 TB. OBS Browser+ le permite subir un máximo de 500 archivos a la vez. No hay límite en el número de archivos que puedes subir usando obsutil a la vez.
- Mediante operaciones SDK/API PUT, POST u añadibles, puede cargar archivos con un tamaño total inferior a 5 GB.
- Mediante la carga multiparte SDK/API o la carga reanudable SDK, puede cargar archivos con un tamaño total inferior a 48.8 TB.
- La función de carga por lotes solo está disponible cuando se cumplen las siguientes condiciones:
 - a. La región donde reside el bucket admite la carga por lotes. Actualmente, las siguientes regiones soportan la carga por lotes: CN-Hong Kong, AP-Bangkok y AP-Singapore.
 - b. La versión del bucket debe ser **3.0**. Para obtener más información sobre cómo ver la versión del bucket, consulte [Comprobación de la versión de OBS \(OBS 2.0 u OBS 3.0\)](#).
- Puede cargar carpetas directamente a través de OBS Console, OBS Browser+ u obsutil.
- Puede establecer una cuota de bucket para limitar la capacidad de los objetos que se van a cargar en un bucket. Para más detalles, consulte [Gestión de cuotas de bucket](#).
- Si el control de versiones está deshabilitado y el nombre de un archivo recién cargado es el mismo que el de un archivo en el bucket, el archivo recién cargado sobrescribe automáticamente el archivo existente y no conserva la información de ACL del archivo existente. Si el nombre de la carpeta recién cargada es el mismo que el de una carpeta en el bucket, las dos carpetas se fusionarán y los archivos de la nueva carpeta sobrescribirán los archivos con nombre en la carpeta antigua.
- Si el control de versiones está habilitado y el nombre de un archivo recién cargado es el mismo que el de un archivo en el bucket, se agrega una nueva versión al archivo existente. Para obtener más información sobre el control de versiones, consulte [Control de versiones](#).

Métodos de carga

| Métodos de carga | Descripción |
|-------------------------------------|---|
| PUT Subida en streaming | Utilice el método PUT o POST cuando el tamaño del archivo que se va a cargar sea inferior a 5 GB. Para obtener más información sobre las diferencias entre los dos métodos de carga, consulte ¿Cuáles son las diferencias entre los métodos de carga PUT y POST? |
| POST (carga basada en navegador) | |

| Métodos de carga | Descripción |
|---|--|
| Carga multiparte | <p>Utilice este método cuando el tamaño del archivo que se va a cargar sea superior a 5 GB y inferior a 48.8 TB.</p> <p>Para más detalles, consulte Carga de varias partes.</p> <p>NOTA</p> <p>Si necesita cargar una gran cantidad de datos (más de 48.8 TB) a OBS, consulte Migración de datos locales a OBS.</p> |
| Transferencia reanudable | <p>La carga de archivos grandes a menudo falla debido a las malas condiciones de la red o a las averías del programa. Es un desperdicio de recursos reiniciar el proceso de carga tras un fallo de carga, y el proceso de carga reiniciado aún puede sufrir de la red inestable. Para resolver estos problemas, puedes usar la API para la carga reanudable, cuyo principio de funcionamiento es dividir el archivo que se va a cargar en varias partes y cargarlos por separado. Este método ahorra recursos y mejora la eficiencia al volver a cargar, y acelera el proceso de carga al cargar simultáneamente partes.</p> <p>Para obtener más información, consulte ¿OBS admite la transferencia de datos reanudable?</p> |
| Carga síncrona de objetos incrementales | <p>Este método sincroniza todo el contenido de la ruta de origen local con el bucket de destino especificado en OBS, asegurando que el contenido es consistente entre la ruta de acceso local y el bucket de destino. La sincronización incremental tiene los siguientes significados: 1) Incremento: Compare el archivo de origen con el objeto de destino y cargue solo el archivo de origen que tiene cambios. 2) Sincronización: Después de ejecutar el comando, asegúrese de que la ruta de origen local es un subconjunto del bucket de destino especificado por OBS. Es decir, cualquier archivo en la ruta de origen local tiene su objeto correspondiente en el bucket de destino en OBS.</p> <p>Para obtener más información, consulte Carga de objetos incrementales de forma sincrónica.</p> |
| Carga agregada | <p>La operación AppendObject agrega datos al final de un objeto en un bucket especificado. Si no hay ningún objeto con nombre en el bucket, se crea un nuevo objeto.</p> <p>Para obtener más información, consulte Adición de un objeto.</p> |

¿Cómo usar esta función?

Puede usar la consola OBS, API, SDK, OBS Browser+ u obsutil para cargar un objeto.

| Herramienta | Referencia |
|----------------|-------------------------------------|
| Consola de OBS | Carga de un archivo |

| Herramienta | Referencia |
|--------------|--|
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Operaciones en objetos |
| OBS Browser+ | Cargar por arrastre |
| obsutil | Carga de un objeto |

Operaciones relacionadas

Puede especificar su [clase de almacenamiento](#) al cargar un objeto o cambiar su clase de almacenamiento después de la carga.

- Los objetos se pueden cambiar de la clase Standard a la de almacenamiento Infrequent Access o Archive, o de Infrequent Access a la clase Standard o Archive, pero los objetos de la clase de almacenamiento Archive deben restaurarse antes de cambiar a la clase de almacenamiento Standard o Infrequent Access. La transición de objetos en la clase de almacenamiento Infrequent Access o Archive conlleva tasas de recuperación. Seleccione una opción de transición adecuada en función de sus necesidades.
- Los objetos de la clase Infrequent Access tienen un período de almacenamiento mínimo de 30 días y los de Archive tienen un período de almacenamiento mínimo de 90 días. Si un objeto se pasa a otra clase de almacenamiento antes de almacenarse durante el período de almacenamiento mínimo requerido, debe pagar por una duración de almacenamiento completa (30 días o 90 días).
- También puede configurar una regla de ciclo de vida para cambiar la clase de almacenamiento de un objeto. Para más detalles, consulte [Gestión del ciclo de vida](#).

Causas de las fallas y soluciones de carga de objetos

Para obtener más información, consulte [Error al cargar un objeto](#).

8.4 Descarga de un objeto

Escenario de la aplicación

Puede descargar objetos de OBS a un equipo local.

OBS admite la descarga por lotes de objetos. Para obtener más información, consulte [¿OBS admite la descarga por lotes?](#)

NOTA

Los objetos de la clase Almacenamiento de Archive sólo se pueden descargar cuando están en el estado **Restaurado**.

¿Cómo usar esta función?

Puede usar OBS Console, OBS Browser+, API, SDK u obsutil para descargar objetos y usar OBS Browser+ y obsutil para descargar carpetas.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Descarga de archivos |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Descarga de objetos |
| OBS Browser+ | - |
| obsutil | Descarga de un objeto |

8.5 Enumeración de objetos

Escenario de la aplicación

Puede ver la lista de objetos creados mediante un enumeración de objetos.

¿Cómo usar esta función?

Puede usar OBS Console, OBS Browser+, API, SDK u obsutil para enumerar objetos.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Inicie sesión en la consola OBS. Seleccione Object Storage , haga clic en el nombre de un bucket y elija Objects en el panel de navegación izquierdo para ver todos los objetos del bucket. |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Listado de objetos en un Bucket |
| OBS Browser+ | - |
| obsutil | Enumeración de objetos |

8.6 Copia de objetos

Escenario de la aplicación

Puede copiar objetos en OBS. Cada operación puede copiar un objeto de hasta 5 GB. Para copiar un objeto de más de 5 GB, utilice la API de carga multiparte. Específicamente, esta operación de copia le permite:

- Crear una copia para un objeto.
- Cambiar el nombre de un objeto creando una copia para él y eliminando el objeto de origen.
- Editar metadatos de objeto. Cada objeto tiene metadatos, que es un conjunto de pares nombre-valor. Puede establecer metadatos de objeto al cargar el objeto. Después de cargar el objeto, puede modificar sus metadatos mediante una API. Para obtener más información, consulte [Modificación de metadatos de objetos](#). También puede crear una copia de objeto y establecer los metadatos. En la operación de copia, establezca el objeto de destino para que sea el mismo que el objeto de origen.

Cada objeto contiene metadatos, incluidos los metadatos del sistema y los metadatos definidos por el usuario. Puede controlar algunos metadatos del sistema. Al copiar un objeto, también se copian los metadatos del sistema controlados por el usuario y los metadatos definidos por el usuario. OBS restablece los metadatos controlados por el sistema. Por ejemplo, al copiar un objeto, la OBS restablece la fecha de creación del objeto copiado. En la solicitud de copia, no es necesario establecer dicho valor.

Al copiar un objeto, es posible que desee actualizar algunos metadatos. Por ejemplo, si el objeto de origen está configurado para estar en el almacenamiento Standard, es posible que desee cambiar la clase de almacenamiento de la copia del objeto a Infrequent Access. También es posible que desee modificar algunos metadatos definidos por el usuario del objeto de origen. Si desea modificar metadatos, incluso un solo fragmento de metadatos, que pueden configurar los usuarios (definidos por los usuarios o el sistema), especifique todos los metadatos que pueden configurar los usuarios en el objeto de origen en la solicitud.

Si desea que los objetos se copien automáticamente en otra región, cree una regla de replicación entre regiones para garantizar la recuperación ante desastres fuera del sitio. Para más detalles, consulte [Replicación entre regiones](#).

NOTA

Cuando el control de versiones está deshabilitado para un bucket, si realiza una copia de **objetoa** y lo guarda como **objetob**, y ya existe un objeto llamado **objetob**, el nuevo **objetob** sobrescribirá el existente. Después de que la copia se ejecuta correctamente, solo se puede descargar el nuevo **objetob** porque se ha eliminado el antiguo **objetob**. Por lo tanto, antes de copiar un objeto, asegúrese de que no hay ningún objeto con el mismo nombre que la copia del objeto para evitar que los datos se eliminen por error. El objeto de origen **objetoa** no cambia durante la copia.

No se puede determinar si una solicitud se ejecuta correctamente solo con **status_code** en el encabezado devuelto por HTTP. Si se devuelve 200 en **status_code**, el servidor ha recibido la solicitud y comienza a procesar la solicitud. El cuerpo de la respuesta muestra si la solicitud se ejecuta correctamente. La solicitud se ejecuta correctamente solo cuando el cuerpo contiene ETag; de lo contrario, la solicitud no se puede ejecutar.

¿Cómo usar esta función?

Puede usar API, SDK, OBS Browser+ u obsutil para copiar objetos.

| Herramienta | Referencia |
|--------------|--|
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Copia de objetos |
| OBS Browser+ | - |
| obsutil | Copia de objetos |

8.7 Carga de varias partes

La carga de varias partes permite cargar un objeto único como un grupo de partes por separado. Cada parte es una parte de datos de objeto consecutivos. Puede cargar partes de objeto en cualquier secuencia o cargarlas de forma independiente. Una pieza se puede recargar después de un fallo de carga, sin afectar a otras piezas. Una vez cargadas todas las piezas, OBS fusiona estas piezas para crear el objeto. Generalmente, si el tamaño de un objeto alcanza los 100 MB, se recomienda cargar varias partes. Por ejemplo, desea cargar un objeto (500 MB) en un bucket de OBS. En este caso, puede utilizar la herramienta OBS Browser+ para cargar el objeto en varias partes. La herramienta divide automáticamente el objeto en varias partes para cargarlo. Alternativamente, puede hacer una llamada a la API para la carga multiparte, mejorando la eficiencia de la carga y reduciendo los fallos.

La carga de varias partes proporciona las siguientes ventajas:

- Mejorar el rendimiento: puede cargar piezas en paralelo para mejorar el rendimiento.
- Recuperación rápida de cualquier fallo de red: Las piezas de tamaño pequeño pueden minimizar el impacto de la carga fallida causada por errores de red.
- Cómoda suspensión y reanudación de la carga de objetos: Puede cargar piezas en cualquier momento. Una carga de varias partes no tiene un período de validez. Debe completar o cancelar explícitamente la carga de varias partes.
- Iniciar la carga antes de conocer el tamaño de un objeto: Puede cargar un objeto mientras lo crea.

La API de carga de varias partes permite cargar un objeto grande en varias partes. Puede cargar un nuevo objeto grande o crear una copia de un objeto existente mediante esta API.

El procedimiento para cargar varias secciones es el siguiente: iniciar la carga (inicializar la tarea de carga), cargar partes y completar la carga (unir las partes cargadas). Al recibir una solicitud de fusión de piezas, OBS fusiona las piezas cargadas para crear un nuevo objeto. Se puede acceder al objeto como otros objetos.

Puede enumerar todas las tareas de carga de varias partes en curso u obtener la lista de partes cargadas sobre una tarea especificada de carga de varias partes. A continuación se describen las operaciones detalladas.

Inicialización de una tarea de carga de varias partes

Cuando envía una solicitud para iniciar la carga multiparte, OBS devuelve una respuesta con el ID de carga, que es el identificador único de la carga de varias partes. Este ID debe incluirse en la solicitud para cargar piezas, enumerar las piezas cargadas, completar una carga de varias piezas o cancelar una carga de varias piezas.

Carga de una parte

Al cargar piezas, debe especificar el ID de carga y los números de pieza. Puede seleccionar cualquier número de pieza entre 1 y 10,000. Un número de pieza identifica de forma exclusiva una pieza y su ubicación en el objeto que está cargando. Si el número de una parte cargada se utiliza para cargar una nueva parte, la parte cargada se sobrescribirá. Cada vez que suba una parte, OBS devuelve el encabezado ETag en la respuesta. Para cada tarea de carga de piezas, debe registrar los números de pieza y los valores de ETag. Estos números de parte y valores de ETag son necesarios en operaciones posteriores de completar la tarea de carga de varias partes.

AVISO

Una vez inicializada la tarea de carga de varias partes y cargadas una o más partes, debe combinar las partes o cancelar la tarea de carga de varias partes. De lo contrario, tienes que pagar la tarifa de almacenamiento de las piezas cargadas. OBS libera el almacenamiento y deja de cobrar la tarifa de almacenamiento solo después de que se fusionen las partes cargadas o se cancele la tarea de carga de varias partes.

Cuando se realizan varias operaciones de carga simultánea para la misma parte de un objeto, el servidor cumple con la política Last Write Win, pero el tiempo al que se hace referencia en Last Write es el tiempo en que se crean los metadatos de la parte. Para garantizar la precisión de los datos, el cliente debe estar bloqueado durante la carga simultánea para la misma parte de un objeto. La carga simultánea para diferentes partes de un objeto no requiere que el cliente esté bloqueado.

Copia de una parte

Después de crear un trabajo de carga de varias partes, puede especificar ID de carga y cargar partes para la tarea de carga especificada. También puede llamar a la API para copiar piezas para agregar piezas. Una parte de un objeto o todo el objeto se puede copiar como una parte.

AVISO

No se puede determinar si una solicitud se ejecuta correctamente solo con **status_code** en el encabezado devuelto por HTTP. Si se devuelve 200 en **status_code**, el servidor ha recibido la solicitud y comienza a procesar la solicitud. El cuerpo de la respuesta muestra si la solicitud se ejecuta correctamente. La solicitud se ejecuta correctamente solo cuando el cuerpo contiene Etag. De lo contrario, la solicitud no se puede ejecutar.

Si copia el objeto de origen como una parte llamada part1 y otra part1 ya existe antes de la operación de copia, la part1 original será sobrescrita por la nueva después de la operación de copia. Una vez que la copia se realiza correctamente, sólo se muestra la nueva pieza1. Los

datos de la parte 1 anterior se eliminarán. Por lo tanto, asegúrese de que la pieza de destino no existe o no tiene valor al copiar una pieza. De lo contrario, los datos pueden ser eliminados por error. El objeto de origen no cambia durante la copia.

Fusión de piezas y cancelación de una tarea de carga de varias piezas

Al fusionar piezas, OBS crea un objeto estandarizando varias piezas en orden ascendente. Si se proporciona algún metadatos de objeto en la inicialización de una tarea de carga de artículos, OBS asocia los metadatos con el objeto. Una vez completada la carga de varias piezas, las piezas dejarán de existir. Una solicitud de fusión de piezas debe contener el ID de carga, los números de pieza y una lista de valores de ETag correspondientes. Las respuestas OBS incluyen la ETag que identifica de forma única datos de objetos compuestos. La ETag no es el valor hash MD5 de los datos del objeto. Puede cancelar una tarea de carga de varias partes. Después de cancelar una tarea de carga de varias partes, el ID de carga no se puede utilizar para cargar ninguna parte. Luego, OBS libera el almacenamiento de todas las partes cargadas. Si detienes una carga multiparte en curso, la carga seguirá completa (el resultado puede ser exitoso o fallado). Para liberar la capacidad de almacenamiento ocupada por todas las partes cargadas, cancele la carga de varias partes una vez completada toda la tarea.

AVISO

Si se cargan 10 piezas pero solo se seleccionan nueve piezas para combinar, las piezas que no se fusionan se eliminarán automáticamente por el sistema. Las piezas que no se fusionan no se pueden restaurar después de eliminarlas. Antes de fusionar las piezas, adopta la API utilizada para enumerar las piezas que se han cargado para comprobar todas las piezas para asegurarse de que no se pierda ninguna pieza.

Enumeración de piezas cargadas

Puede enumerar las partes de una tarea de carga de varias partes especificadas o las partes de todas las tareas de carga de varias partes en curso. La información sobre las piezas cargadas en una carga de varias piezas específica se devolverá para una solicitud de lista de las piezas cargadas. Para cada solicitud de listar las piezas cargadas, OBS devuelve información sobre las piezas cargadas en la carga multiparte específica. Se puede devolver información sobre un máximo de 1000 piezas. Si se cargan más de 1000 piezas en una carga de varias piezas, debe enviar varias solicitudes para listar todas las piezas cargadas. La lista de piezas cargadas no incluye las piezas combinadas.

AVISO

Una lista devuelta solo se puede utilizar para la verificación. Una vez completada la carga de varias partes, el resultado de la lista ya no es válido. Sin embargo, cuando se cargan los números de pieza y los valores ETag devueltos por OBS, se reservará la lista de números de pieza especificados por el usuario.

Listado de tareas de carga de varias partes

Puede obtener la lista de tareas de carga multiparte inicializadas enumerando las tareas de carga multiparte en el bucket. Las tareas de carga multiparte inicializadas se refieren a las

tareas de carga multiparte que no se fusionan o cancelan después de la inicialización. Se puede devolver un máximo de 1000 tareas de carga multiparte para cada solicitud. Si el número de tareas de carga multiparte en curso es superior a 1000, debe enviar más solicitudes para consultar las tareas restantes.

Tabla 8-1 enumera las restricciones en la lista de cargas de varias partes.

Tabla 8-1 Restricciones en la lista de cargas multiparte

| Artículo | Restricción |
|--|---|
| Tamaño del objeto | Hasta 48.8 TB |
| Número máximo de piezas para cada tarea de carga | 10,000 |
| Número de pieza | 1 - 10,000 (incluido) |
| Tamaño de la pieza | El tamaño de la pieza está entre 5 MB y 5 GB. El tamaño de la última parte está entre 0 bytes y 5 GB. |
| Número máximo de piezas cargadas devueltas en respuesta a la solicitud de enumeración de piezas cargadas. | 1000 |
| Número máximo de tareas de carga multiparte inicializadas devueltas en respuesta a la solicitud de enumeración de tareas multiparte inicializadas. | 1000 |

Operaciones y permisos de carga de varias partes

Puede realizar la carga de varias partes solo después de que se le haya concedido el permiso. Puede usar ACL, políticas de bucket o políticas de usuario para conceder a los usuarios el permiso. En la siguiente tabla se enumeran las operaciones de carga de varias partes y los permisos necesarios que pueden conceder las ACL, las políticas de bucket o las políticas de usuario.

| Operación | Permisos requeridos |
|---|---|
| Inicialización de una tarea de carga de varias partes | Para realizar esta operación, necesita tener el permiso PutObject. El propietario de un bucket puede permitir a los administradores realizar la operación PutObject. |

| Operación | Permisos requeridos |
|---|---|
| Carga de piezas | <p>Para realizar esta operación, necesita tener el permiso PutObject.</p> <p>Solo el iniciador de una carga de varias partes puede cargar partes. El propietario del bucket debe conceder al iniciador de carga de varias partes el permiso PutObject para que el iniciador pueda cargar partes del objeto.</p> |
| Copia de piezas | <p>Para realizar esta operación, necesita tener el permiso PutObject así como el permiso GetObject sobre el objeto que se va a copiar.</p> <p>Solo el iniciador de una carga de varias partes puede cargar partes. El propietario del bucket debe conceder al iniciador de carga de varias partes el permiso PutObject para que el iniciador pueda cargar partes del objeto.</p> |
| Fusión de piezas | <p>Para realizar esta operación, necesita tener el permiso PutObject.</p> <p>Solo el iniciador de una carga de varias partes puede combinar las partes. El propietario del bucket debe conceder al iniciador de carga multiparte el permiso PutObject para que el iniciador pueda completar la carga multiparte.</p> |
| Cancelación de tareas de carga multiparte | <p>Para realizar esta operación, necesita tener el permiso AbortMultipartUpload.</p> <p>De forma predeterminada, solo el propietario del bucket y el iniciador de carga de varias partes tienen este permiso. Además de la configuración predeterminada, el propietario del bucket puede permitir que los administradores realicen esta operación. El propietario del cubo también puede negar a los administradores que realicen esta operación.</p> |

| Operación | Permisos requeridos |
|---|---|
| Enumeración de piezas cargadas | <p>Para realizar esta operación, necesita tener el permiso ListMultipartUploadParts.</p> <p>De forma predeterminada, el propietario del bucket puede enumerar las partes cargadas de cualquier carga de varias partes al bucket. El iniciador de carga multiparte puede enumerar las partes cargadas de una carga multiparte específica.</p> <p>Además de la configuración predeterminada, el propietario del bucket puede permitir que los administradores realicen esta operación. El propietario del cubo también puede negar a los administradores que realicen esta operación.</p> |
| Listado de tareas de carga de varias partes | <p>Para enumerar las tareas de carga de varias partes en el bucket, necesita tener el permiso ListBucketMultipartUploads.</p> <p>Además de la configuración predeterminada, el propietario del bucket puede permitir que los administradores realicen esta operación.</p> |

REST APIs aplicables a la carga de varias partes

Las siguientes secciones de la *Referencia de la API de servicio de almacenamiento de objetos* describen las REST API relevantes para la carga de varias partes.

- ListBucketMultipartUpload
- InitiateMultipartUpload
- UploadPart
- UploadPart-Copy
- ListParts
- CompleteMultipartUpload
- AbortMultipartUpload

8.8 Consulta de información de objeto

Escenario de la aplicación

Después de cargar un objeto, puede ver la información o las propiedades del objeto.

¿Cómo usar esta función?

Puede utilizar la consola OBS, API, SDK, OBS Browser+ u obsutil para ver la información de un objeto.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | - |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Consulta de metadatos de objetos |
| OBS Browser+ | - |
| obsutil | Consulta de propiedades de objeto |

8.9 Compartir un archivo

Escenario de la aplicación

Puede compartir archivos con otros usuarios de forma temporal o permanente.

- **Compartición temporal:**

Compartir un archivo con otros usuarios a través de la URL temporal del archivo. Todas las URL compartidas son temporales con un período de validez.

Una URL temporal consiste en el nombre de dominio de acceso y la información de autenticación temporal de un archivo. Ejemplo:

```
https://bucketname.obs.ap-southeast-1.myhuaweicloud.com:443/image.png?
AccessKeyId=xxx&Expires=xxx&response-content-disposition=xxx&x-obs-security-
token=xxx&Signature=xxx
```

La información de autenticación temporal contiene los parámetros **AccessKeyId**, **Expires**, **x-obs-security-token**, y **Signature**. Los parámetros **AccessKeyId**, **x-obs-security-token** y **Signature** se utilizan para la autenticación. El parámetro **Expires** especifica el período de validez de la autenticación.

Para obtener más información sobre el método y los parámetros de autenticación temporal, consulte [Autenticación de firma en una URL](#) en *Referencia de API de servicio de almacenamiento de objetos*.

Una URL temporal también contiene el parámetro **response-content-disposition**, que define si un objeto se descarga directamente o se previsualiza en el navegador web cuando se accede a él. El valor lo determina el explorador web basado en el **Content-Type** del objeto compartido.

- **Compartición permanente:**

Si establece el permiso de un objeto para permitir que los usuarios anónimos lo lean, los usuarios anónimos pueden tener acceso al objeto a través de la dirección URL que compartió. Para obtener más información acerca de cómo configurar permisos, consulte [Concesión de permisos públicos de lectura sobre objetos a usuarios anónimos](#).

El método de uso de un navegador para acceder a objetos varía en función del tipo de objeto. Puede abrir directamente archivos **.txt** y **.html** usando un navegador. Sin embargo, cuando abre archivos **.exe** y **.dat** con un explorador, los archivos se descargan automáticamente en el equipo local.

Para obtener más información sobre cómo obtener la URL de acceso al objeto, consulte [¿Cómo obtengo la ruta de acceso a un objeto?](#)

Si no pudo acceder a un objeto, solucione el problema según [Error al acceder a un objeto a través de su URL](#).

Restricciones

- Los objetos cifrados no se pueden compartir.
- Es necesario restaurar un objeto Archive antes de compartirlo con otros usuarios.
- El código de autorización no es necesario para compartir archivos temporales, pero es necesario para compartir carpetas temporales. Para más detalles, consulte [Compartir una carpeta](#).
- Solo los bucket de la versión 3.0 o posterior admiten el uso compartido temporal de archivos. Para obtener más información sobre cómo ver la versión del bucket, consulte [Comprobación de la versión de OBS \(OBS 2.0 u OBS 3.0\)](#).
- En la siguiente tabla se describe el período de validez de la dirección URL de los archivos que se comparten temporalmente con diferentes herramientas.

| Herramienta | Período de validez de la dirección URL |
|----------------|--|
| Consola de OBS | La URL es válida durante 1 minuto a 18 horas. Después de compartir un objeto, el sistema generará una URL que contiene la información de autenticación temporal, válida durante cinco minutos desde su generación de forma predeterminada. Cada vez que cambia el período de validez de una URL, OBS obtiene la información de autenticación de nuevo para generar una nueva URL para compartir. La nueva URL pasa a ser válida en el momento en que se cambia el período de validez. |
| SDK | Configurar parámetro Expires para especificar cuándo expira una autorización temporal. La autorización temporal expira en 24 horas. |
| API | Configurar parámetro Expires para especificar cuándo expira una autorización temporal. La autorización temporal expira en 24 horas. |
| OBS Browser+ | Cuando inicie sesión en OBS Browser+ con una cuenta y una contraseña, una URL compartida será válida durante 24 horas como máximo. El periodo de validez predeterminado es de 10 horas. Si se requiere un período de validez más largo, utilice el AK/SK permanente para iniciar sesión. |
| obsutil | Utilice el parámetro adicional e para especificar cuándo caduca una URL de descarga de objeto. El valor mínimo es 60s y el predeterminado es 300s. |

¿Cómo usar esta función?

Puede utilizar la consola OBS, SDK, API, OBS Browser+ u obsutil para compartir archivos.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Compartir un archivo |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Autenticación de la firma en una URL |
| OBS Browser+ | - |
| obsutil | Generación del enlace de descarga de un objeto |

Operaciones relacionadas

Para descargar una URL temporal generada con wget, utilice "" y -O para especificar el nombre del archivo, o se producirá un error.

Ejemplo:

```
[root@ecs-test ~]# wget "Temporary URL" -O abc
```

En el comando anterior, *abc* indica un nuevo nombre de archivo.

8.10 Compartir una carpeta

Escenario de la aplicación

Puede compartir carpetas con otros usuarios de forma temporal o permanente.

- **Compartición temporal:**

Compartir una carpeta con otros usuarios a través de la URL temporal del archivo. Todas las URL compartidas son temporales con un período de validez.

Necesitas preparar un código de extracción de seis dígitos antes de compartir una carpeta. Después de crear el enlace para compartir de la carpeta, OBS agrega automáticamente los enlaces de descarga de todos los objetos de la carpeta a un sitio web estático que está alojado en un bucket OBS público. A continuación, los usuarios que tienen la URL temporal y el código de extracción pueden acceder al sitio web estático y descargar archivos.

Una URL temporal consiste en el nombre de dominio de acceso y la información de autenticación temporal de la carpeta. Ejemplo:

```
https://e-share.obs-website.ap-southeast-1.myhuaweicloud.com:443/image.png?token=xxx
```

Para obtener más información sobre el método y los parámetros de autenticación temporal, consulte [Autenticación de firma en una URL](#) en *Referencia de API de servicio de almacenamiento de objetos*.

- **Compartición permanente:**

Si establece el permiso de un objeto para permitir que los usuarios anónimos lo lean, los usuarios anónimos pueden tener acceso al objeto a través de la dirección URL que

compartió. Para obtener más información acerca de cómo configurar permisos, consulte [Concesión de permisos públicos de lectura sobre objetos a usuarios anónimos](#).

Para obtener más información sobre cómo obtener la URL de acceso al objeto, consulte [¿Cómo obtengo la ruta de acceso a un objeto?](#)

Si no pudo acceder a un objeto, solucione el problema según [Error al acceder a un objeto a través de su URL](#).

Restricciones

- Es necesario restaurar un objeto Archive antes de compartirlo con otros usuarios.
- La función de uso compartido de carpetas solo está restringida a algunas regiones.
- Solo los bucket de la versión 3.0 o posterior admiten el uso compartido temporal de carpetas. Para obtener más información sobre cómo ver la versión del bucket, consulte [Comprobación de la versión de OBS \(OBS 2.0 u OBS 3.0\)](#).
- En la siguiente tabla se describe el período de validez de la dirección URL de los archivos que se comparten temporalmente con diferentes herramientas.

| Herramienta | Período de validez de la dirección URL |
|----------------|--|
| Consola de OBS | La URL es válida durante 1 minuto a 18 horas. Después de compartir un objeto, el sistema generará una URL que contiene la información de autenticación temporal, válida durante cinco minutos desde su generación de forma predeterminada. Cada vez que cambia el período de validez de una URL, OBS obtiene la información de autenticación de nuevo para generar una nueva URL para compartir. La nueva URL pasa a ser válida en el momento en que se cambia el período de validez. |
| API | Configurar parámetro Expires para especificar cuándo expira una autorización temporal. La autorización temporal expira en 24 horas. |
| OBS Browser+ | Cuando inicie sesión en OBS Browser+ con una cuenta y una contraseña, una URL compartida será válida durante 24 horas como máximo. El periodo de validez predeterminado es de 10 horas. Si se requiere un período de validez más largo, utilice el AK/SK permanente para iniciar sesión. |
| obsutil | Utilice el parámetro adicional vp para especificar cuánto tiempo es válido un código de autorización. El período de validez predeterminado es un día. Las unidades permitidas incluyen m (meses), w (semanas), d (días), h (horas), min (minutos) y s (segundos). Si no se especifica ninguna unidad de tiempo, el valor se calcula en segundos. |

¿Cómo usar esta función?

Puede utilizar la consola OBS, las API, OBS Browser+ u obsutil para compartir carpetas.

| Herramienta | Referencia |
|----------------|---------------------------------------|
| Consola de OBS | Compartir una carpeta |

| Herramienta | Referencia |
|--------------|--|
| API | Autenticación de la firma en una URL |
| OBS Browser+ | - |
| obsutil | Creación de un código de autorización para uso compartido de directorios |

8.11 Gestión de metadatos de objetos

Escenario de la aplicación

Los metadatos de objeto son un conjunto de pares nombre-valor que describen el objeto y se utilizan para la gestión de objetos.

Actualmente, solo se admiten los metadatos definidos por el sistema.

Los metadatos definidos por el sistema se clasifican en los siguientes tipos: controlados por el sistema y controlados por el usuario. Por ejemplo, los metadatos como **Last-Modified** son controlados por el sistema y no pueden ser modificados. Puede modificar los metadatos como **ContentLanguage** a través de la API. Los metadatos que se pueden modificar se describen de la siguiente manera:

Tabla 8-2 Metadatos de OBS

| Nombre | Descripción |
|--------------------|---|
| ContentDisposition | <p>Proporciona un nombre de archivo predeterminado para el objeto que se está solicitando. Cuando se descarga o se accede a un objeto, el archivo con el nombre de archivo predeterminado se muestra directamente en el explorador o se muestra un cuadro de diálogo de descarga si se accede al archivo.</p> <p>Por ejemplo, seleccione ContentDisposition como nombre de metadatos e introduzca attachment;filename="testfile.xls" como valor de metadatos de un objeto. Si tiene acceso al objeto a través de un vínculo, se muestra directamente un cuadro de diálogo para descargar objetos y el nombre del objeto se cambia a testfile.xls.</p> <p>Para obtener más información, consulte la definición de ContentDisposition en HTTP.</p> |
| ContentLanguage | <p>Indica el idioma o idiomas destinados a la audiencia. Por lo tanto, un usuario puede diferenciarse de acuerdo con el idioma preferido del usuario. Para obtener más información, consulte la definición de ContentLanguage en HTTP.</p> |

| Nombre | Descripción |
|-------------------------|---|
| WebsiteRedirectLocation | <p>Redirecciona un objeto a otro objeto o a una URL externa. La función de redirección se implementa utilizando el alojamiento web estático.</p> <p>Por ejemplo, puede realizar las siguientes operaciones para implementar la redirección de objetos:</p> <ol style="list-style-type: none"> 1. Establezca los metadatos del objeto testobject.html en el directorio raíz del bucket testbucket. Seleccione WebsiteRedirectLocation en Name y escriba http://www.example.com en Value. <p>NOTA OBS solo admite la redirección de objetos en el directorio raíz de un bucket.</p> <ol style="list-style-type: none"> 2. Configure el alojamiento de sitios web estáticos para el bucket testbucket y establezca el objeto testobject.html en el bucket como la página de inicio predeterminada del sitio web estático alojado. 3. Si accede al objeto testobject.html a través del enlace URL proporcionado en la página Configure Static Website Hosting, la solicitud de acceso se redirige a http://www.example.com. |
| ContentEncoding | <p>Formato de codificación de contenido cuando se descarga un objeto. Las opciones son las siguientes:</p> <ul style="list-style-type: none"> ● Standard: compress, deflate, exi, identity, gzip, y pack200-gzip ● Otros: br, bzip2, lzma, peerdist, sdch, xpress, xz |
| CacheControl | <p>Comportamiento de caché de la página web cuando se descarga el objeto especificado.</p> <ul style="list-style-type: none"> ● Capacidad de caché: public, private, no-cache, and only-if-cached ● Tiempo de caducidad: max-age=<seconds>, s-maxage=<seconds>, max-stale[=<seconds>], min-fresh=<seconds>, stale-while-revalidate=<seconds>, stale-if-error=<seconds> ● Re-verificación y recarga: must-revalidate, proxy-revalidate, immutable ● Otros: no-store, no-transform |
| Expires | <p>Tiempo de caducidad de caché (GMT)</p> |
| ContentType | <p>Tipo de archivo de un objeto.</p> <p>Para más detalles, consulte Tipo de contenido de metadatos de objeto.</p> |

 **NOTA**

- Cuando el control de versiones está habilitado para un bucket, puede establecer metadatos para objetos que son de la **Versión más reciente**, pero no puede establecer metadatos para objetos que son de la **Versión histórica**.
- No se pueden establecer metadatos de objeto para un objeto Archive.

Tipo de contenido de metadatos de objeto

Cuando se carga un objeto en OBS a través de la Consola OBS o una herramienta, el sistema hace coincidir automáticamente el valor de **Content-Type** basándose en la extensión de nombre de archivo del objeto. Cuando se accede a un objeto a través de un explorador Web, el sistema especifica una aplicación para abrir el objeto de acuerdo con el valor de Content-Type. Puede modificar el Content-Type de un objeto basándose en su extensión de nombre de archivo.

 **NOTA**

Si carga un objeto llamando a una API, especifique el valor de Content-Type porque el sistema no coincide automáticamente con el valor de Content-Type.

Tabla 8-3 Valores comunes de tipo de contenido

| Extensión de nombre de archivo | Content-Type | Extensión de nombre de archivo | Content-Type |
|---|--------------------------------|--------------------------------|------------------------|
| * (corriente binaria, que no conoce el tipo de archivo que se va a descargar) | application/octet-stream | .tif | image/tiff |
| .001 | application/x-001 | .301 | application/x-301 |
| .323 | text/h323 | .906 | application/x-906 |
| .907 | drawing/907 | .a11 | application/x-a11 |
| .acp | audio/x-mei-aac | .ai | application/postscript |
| .aif | audio/aiff | .aifc | audio/aiff |
| .aiff | audio/aiff | .anv | application/x-anv |
| .asa | text/asa | .asf | video/x-ms-asf |
| .asp | text/asp | .asx | video/x-ms-asf |
| .au | audio/basic | .avi | video/avi |
| .awf | application/vnd.adobe.workflow | .biz | text/xml |
| .bmp | application/x-bmp | .bot | application/x-bot |
| .c4t | application/x-c4t | .c90 | application/x-c90 |

| Extensión de nombre de archivo | Content-Type | Extensión de nombre de archivo | Content-Type |
|--------------------------------|----------------------------|--------------------------------|-------------------------------|
| .cal | application/x-cals | .cat | application/vnd.ms-pki.seccat |
| .cdf | application/x-netcdf | .cdr | application/x-cdr |
| .cel | application/x-cel | .cer | application/x-x509-ca-cert |
| .cg4 | application/x-g4 | .cgm | application/x-cgm |
| .cit | application/x-cit | .class | java/* |
| .cml | text/xml | .cmp | application/x-cmp |
| .cmx | application/x-cmx | .cot | application/x-cot |
| .crl | application/pkix-crl | .crt | application/x-x509-ca-cert |
| .csi | application/x-csi | .css | text/css |
| .cut | application/x-cut | .dbf | application/x-dbf |
| .dbm | application/x-dbm | .dbx | application/x-dbx |
| .dcd | text/xml | .dcx | application/x-dcx |
| .der | application/x-x509-ca-cert | .dgn | application/x-dgn |
| .dib | application/x-dib | .dll | application/x-msdownload |
| .doc | application/msword | .dot | application/msword |
| .drw | application/x-drw | .dtd | text/xml |
| .dwf | Model/vnd.dwf | .dwf | application/x-dwf |
| .dwg | application/x-dwg | .dxb | application/x-dxb |
| .dxf | application/x-dxf | .edn | application/vnd.adobe.edn |
| .emf | application/x-emf | .eml | message/rfc822 |
| .ent | text/xml | .epi | application/x-epi |
| .eps | application/x-ps | .eps | application/postscript |
| .etd | application/x-ebx | .exe | application/x-msdownload |
| .fax | image/fax | .fdf | application/vnd.fdf |

| Extensión de nombre de archivo | Content-Type | Extensión de nombre de archivo | Content-Type |
|--------------------------------|-------------------------------|--------------------------------|-------------------------------|
| .fif | application/fractals | .fo | text/xml |
| .frm | application/x-frm | .g4 | application/x-g4 |
| .gbr | application/x-gbr | . | application/x- |
| .gif | image/gif | .gl2 | application/x-gl2 |
| .gp4 | application/x-gp4 | .hgl | application/x-hgl |
| .hmr | application/x-hmr | .hpg | application/x-hpgl |
| .hpl | application/x-hpl | .hqx | application/mac-binhex40 |
| .hrf | application/x-hrf | .hta | application/hta |
| .htc | text/x-component | .htm | text/html |
| .html | text/html | .htt | text/webviewhtml |
| .htx | text/html | .icb | application/x-icb |
| .ico | image/x-icon | .ico | application/x-ico |
| .iff | application/x-iff | .ig4 | application/x-g4 |
| .igs | application/x-igs | .iii | application/x-iphone |
| .img | application/x-img | .ins | application/x-internet-signup |
| .isp | application/x-internet-signup | .IVF | video/x-ivf |
| .java | java/* | .jfif | image/jpeg |
| .jpe | image/jpeg | .jpe | application/x-jpe |
| .jpeg | image/jpeg | .jpg | image/jpeg |
| .jpg | application/x-jpg | .js | application/x-javascript |
| .jsp | text/html | .la1 | audio/x-liquid-file |
| .lar | application/x-laplayer-reg | .latex | application/x-latex |
| .lavs | audio/x-liquid-secure | .lbm | application/x-lbm |
| .lmsff | audio/x-la-lms | .ls | application/x-javascript |
| .ltr | application/x-ltr | .m1v | video/x-mpeg |

| Extensión de nombre de archivo | Content-Type | Extensión de nombre de archivo | Content-Type |
|--------------------------------|-------------------------------|--------------------------------|----------------------------|
| .m2v | video/x-mpeg | .m3u | audio/mpegurl |
| .m4e | video/mpeg4 | .mac | application/x-mac |
| .man | application/x-troff-man | .math | text/xml |
| .mdb | application/msaccess | .mdb | application/x-mdb |
| .mfp | application/x-shockwave-flash | .mht | message/rfc822 |
| .mhtml | message/rfc822 | .mi | application/x-mi |
| .mid | audio/mid | .midi | audio/mid |
| .mil | application/x-mil | .mml | text/xml |
| .mnd | audio/x-musicnet-download | .mns | audio/x-musicnet-stream |
| .mocha | application/x-javascript | .movie | video/x-sgi-movie |
| .mp1 | audio/mp1 | .mp2 | audio/mp2 |
| .mp2v | video/mpeg | .mp3 | audio/mp3 |
| .mp4 | video/mp4 | .mpa | video/x-mpg |
| .mpd | application/vnd.ms-project | .mpe | video/x-mpeg |
| .mpeg | video/mpg | .mpg | video/mpg |
| .mpga | audio/rn-mpeg | .mpp | application/vnd.ms-project |
| .mps | video/x-mpeg | .mpt | application/vnd.ms-project |
| .mpv | video/mpg | .mpv2 | video/mpeg |
| .mpw | application/vnd.ms-project | .mpx | application/vnd.ms-project |
| .mtx | text/xml | .mxx | application/x-mmxp |
| .net | image/pnetvue | .nrf | application/x-nrf |
| .nws | message/rfc822 | .odc | text/x-ms-odc |
| .out | application/x-out | .p10 | application/pkcs10 |

| Extensión de nombre de archivo | Content-Type | Extensión de nombre de archivo | Content-Type |
|--------------------------------|---------------------------------|--------------------------------|----------------------------------|
| .p12 | application/x-pkcs12 | .p7b | application/x-pkcs7-certificates |
| .p7c | application/pkcs7-mime | .p7m | application/pkcs7-mime |
| .p7r | application/x-pkcs7-certreqresp | .p7s | application/pkcs7-signature |
| .pc5 | application/x-pc5 | .pci | application/x-pci |
| .pcl | application/x-pcl | .pcx | application/x-pcx |
| .pdf | application/pdf | .pdf | application/pdf |
| .pdx | application/vnd.adobe.pdx | .pfx | application/x-pkcs12 |
| .pgl | application/x-pgl | .pic | application/x-pic |
| .pko | application/vnd.ms-pki.pko | .pl | application/x-perl |
| .plg | text/html | .pls | audio/scpls |
| .plt | application/x-plt | .png | image/png |
| .png | application/x-png | .pot | application/vnd.ms-powerpoint |
| .ppa | application/vnd.ms-powerpoint | .ppm | application/x-ppm |
| .pps | application/vnd.ms-powerpoint | .ppt | application/vnd.ms-powerpoint |
| .ppt | application/x-ppt | .pr | application/x-pr |
| .prf | application/pics-rules | .prn | application/x-prn |
| .prt | application/x-prt | .ps | application/x-ps |
| .ps | application/postscript | .ptn | application/x-ptn |
| .pwz | application/vnd.ms-powerpoint | .r3t | text/vnd.rn-realtxt3d |
| .ra | audio/vnd.rn-realaudio | .ram | audio/x-pn-realaudio |
| .ras | application/x-ras | .rat | application/rat-file |

| Extensión de nombre de archivo | Content-Type | Extensión de nombre de archivo | Content-Type |
|--------------------------------|-------------------------------------|--------------------------------|-------------------------------------|
| .rdf | text/xml | .rec | application/vnd.rn-recording |
| .red | application/x-red | .rgb | application/x-rgb |
| .rjs | application/vnd.rn-realsystem-rjs | .rjt | application/vnd.rn-realsystem-rjt |
| .rlc | application/x-rlc | .rle | application/x-rle |
| .rm | application/vnd.rn-realmedia | .rmf | application/vnd.adobe.rmf |
| .rmi | audio/mid | .rmj | application/vnd.rn-realsystem-rmj |
| .rmm | audio/x-pn-realaudio | .rmp | application/vnd.rn-rn_music_package |
| .rms | application/vnd.rn-realmedia-secure | .rmvb | application/vnd.rn-realmedia-vbr |
| .rmx | application/vnd.rn-realsystem-rmx | .rnx | application/vnd.rn-realplayer |
| .rp | image/vnd.rn-realpix | .rpm | audio/x-pn-realaudio-plugin |
| .rsml | application/vnd.rn-rsml | .rt | text/vnd.rn-realtxt |
| .rtf | application/msword | .rtf | application/x-rtf |
| .rv | video/vnd.rn-realvideo | .sam | application/x-sam |
| .sat | application/x-sat | .sdp | application/sdp |
| .sdw | application/x-sdw | .sit | application/x-stuffit |
| .slb | application/x-slb | .sld | application/x-sld |
| .slk | drawing/x-slk | .smi | application/smil |
| .smil | application/smil | .smk | application/x-smk |
| .snd | audio/basic | .sol | text/plain |
| .sor | text/plain | .spc | application/x-pkcs7-certificates |
| .spl | application/futuresplash | .spp | text/xml |

| Extensión de nombre de archivo | Content-Type | Extensión de nombre de archivo | Content-Type |
|--------------------------------|-------------------------------|--------------------------------|----------------------------------|
| .ssm | application/streamingmedia | .sst | application/vnd.ms-pki.certstore |
| .stl | application/vnd.ms-pki.stl | .stm | text/html |
| .sty | application/x-sty | .svg | text/svg+xml |
| .swf | application/x-shockwave-flash | .tdf | application/x-tdf |
| .tg4 | application/x-tg4 | .tga | application/x-tga |
| .tif | image/tiff | .tif | application/x-tif |
| .tiff | image/tiff | .tld | text/xml |
| .top | drawing/x-top | .torrent | application/x-bittorrent |
| .tsd | text/xml | .txt | text/plain |
| .uin | application/x-icq | .uls | text/iuls |
| .vcf | text/x-vcard | .vda | application/x-vda |
| .vdx | application/vnd.visio | .vml | text/xml |
| .vpg | application/x-vpeg005 | .vsd | application/vnd.visio |
| .vsd | application/x-vsdx | .vss | application/vnd.visio |
| .vst | application/vnd.visio | .vst | application/x-vst |
| .vsw | application/vnd.visio | .vsx | application/vnd.visio |
| .vtx | application/vnd.visio | .vxml | text/xml |
| .wav | audio/wav | .wax | audio/x-ms-wax |
| .wb1 | application/x-wb1 | .wb2 | application/x-wb2 |
| .wb3 | application/x-wb3 | .wbmp | image/vnd.wap.wbmp |
| .wiz | application/msword | .wk3 | application/x-wk3 |
| .wk4 | application/x-wk4 | .wkq | application/x-wkq |
| .wks | application/x-wks | .wm | video/x-ms-wm |
| .wma | audio/x-ms-wma | .wmd | application/x-ms-wmd |
| .wmf | application/x-wmf | .wml | text/vnd.wap.wml |

| Extensión de nombre de archivo | Content-Type | Extensión de nombre de archivo | Content-Type |
|--------------------------------|---|--------------------------------|---------------------------------|
| .wmv | video/x-ms-wmv | .wmx | video/x-ms-wmx |
| .wmz | application/x-ms-wmz | .wp6 | application/x-wp6 |
| .wpd | application/x-wpd | .wpg | application/x-wpg |
| .wpl | application/vnd.ms-wpl | .wq1 | application/x-wq1 |
| .wrl | application/x-wrl | .wri | application/x-wri |
| .wrk | application/x-wrk | .ws | application/x-ws |
| .ws2 | application/x-ws | .wsc | text/scriptlet |
| .wsdl | text/xml | .wvx | video/x-ms-wvx |
| .xdp | application/vnd.adobe.xdp | .xdr | text/xml |
| .xfd | application/vnd.adobe.xfd | .xdfd | application/vnd.adobe.xdfd |
| .xhtml | text/html | .xls | application/vnd.ms-excel |
| .xls | application/x-xls | .xlw | application/x-xlw |
| .xml | text/xml | .xpl | audio/scpls |
| .xq | text/xml | .xql | text/xml |
| .xquery | text/xml | .xsd | text/xml |
| .xsl | text/xml | .xslt | text/xml |
| .xwd | application/x-xwd | .x_b | application/x-x_b |
| .sis | application/vnd.symbian.install | .sisx | application/vnd.symbian.install |
| .x_t | application/x-x_t | .ipa | application/vnd.iphone |
| .apk | application/vnd.android.package-archive | .xap | application/x-silverlight-app |
| .zip | application/zip | .rar | application/x-rar-compressed |
| .7z | application/x-7z-compressed | - | - |

Metadatos de objetos definidos por el usuario

Puede agregar los metadatos definidos por el usuario cuyo nombre comienza con **x-obs-meta-** para una fácil gestión de objetos. Cuando recupera o consulta los metadatos del objeto, los metadatos definidos por el usuario agregados se devolverán en el encabezado de respuesta. El tamaño de los metadatos definidos por el usuario está limitado a 8 KB. Para medir el tamaño total de los metadatos definidos por el usuario, sume el número de bytes en la codificación UTF-8 de cada clave y valor.

Las claves de metadatos definidas por el usuario no distinguen entre mayúsculas y minúsculas, pero OBS las almacena en minúsculas. Los valores clave distinguen entre mayúsculas y minúsculas.

A continuación se presenta un ejemplo.

```
PUT /key HTTP/1.1
Host: bucket01.obs.myhuaweicloud.com
x-obs-meta-Test1: Test Metal

HEAD /Key HTTP/1.1
Host: bucket01.obs.myhuaweicloud.com
x-obs-meta-test1: Test Metal
```

Tanto las claves de metadatos definidas por el usuario como sus valores deben ajustarse a los caracteres US-ASCII. Si son necesarios caracteres non-ASCII o irreconocibles, deben codificarse o decodificarse en URL o Base64 en el lado del cliente, porque el lado del servidor no realiza ninguna decodificación.

Si hay caracteres non-US-ASCII o irreconocibles en un valor y el cliente no codifica el valor, el lado del servidor codificará el valor en Base64 y encapsulará el valor usando **?UTF-8?B?<base64(str)>=?**. Se toma **x-obs-meta-nonascii: nonasciiÄÄ** como ejemplo. **nonasciiÄÄ** se codifica como **bm9uYXNjaWnDhMOE** en Base64 y la respuesta devuelta es **x-obs-meta-nonascii: =?UTF-8?B?bm9uYXNjaWnDhMOE=?**.

A continuación se presenta un ejemplo.

```
PUT /key HTTP/1.1
Host: bucket01.obs.myhuaweicloud.com
x-obs-meta-nonascii: nonasciiÄÄ

HEAD /Key HTTP/1.1
Host: bucket01.obs.myhuaweicloud.com
x-obs-meta-nonascii: =?UTF-8?B?bm9uYXNjaWnDhMOE=?
```

¿Cómo usar esta función?

Puede utilizar la consola OBS, las API, los SDK, el navegador OBS+ u obsutil para personalizar los metadatos de los objetos.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Configuración de metadatos de objeto |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Modificación de metadatos de objetos |

| Herramienta | Referencia |
|--------------|------------------------------------|
| OBS Browser+ | - |
| obsutil | Carga de un objeto |

8.12 Restaurar objetos desde el almacenamiento de archivos

Escenario de la aplicación

Debe restaurar un objeto Archive antes de descargarlo, acceder a él mediante su URL o establecer permisos de ACL o metadatos de objeto para él.

Para obtener más información sobre la duración y los precios de la restauración de datos, consulte [Detalles de precios del producto](#).

Restricciones

- Si se está restaurando un objeto Archive, no puede suspender ni eliminar la tarea de restauración.
- No se puede volver a restaurar un objeto que esté en estado **Restauración**.
- Después de restaurar un objeto, se generará una copia de objeto en la clase de almacenamiento Standard. De esta manera, hay un objeto en la clase de almacenamiento Archive y también una copia en la clase de almacenamiento Standard en el bucket. Después de restaurar un objeto Archive, el estado del objeto muestra **Restored** y la copia de objeto generada en la clase de almacenamiento Standard no se muestra en el bucket.
 Durante el período de validez de la restauración, se le cobrará por el espacio de almacenamiento ocupado tanto por el objeto como por su copia. La copia se eliminará automáticamente al expirar su período de validez.

Métodos de uso de esta función

Puede utilizar la consola OBS, API, SDK, OBS Browser+ u obsutil para restaurar objetos Archive.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Restaurar un archivo de archivo almacenado en OBS |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Restaurar objetos de archivo |
| OBS Browser+ | - |
| obsutil | Restaurar objetos desde el archivo OBS |

8.13 Lectura directa

Escenario de la aplicación

Puede habilitar la lectura directa si desea obtener objetos Archive inmediatamente o si hay problemas de compatibilidad de interfaz cuando OBS está interconectado con otros sistemas.

La lectura directa permite que se descarguen directamente los objetos de la clase de almacenamiento de Archive sin restaurarse de antemano.

La lectura directa es una función facturable. Para obtener información detallada, consulte [Detalles de los precios de productos](#).

Puede habilitar la lectura directa al crear un bucket o habilitarlo para un bucket existente.

¿Cómo usar esta función?

Puede utilizar la consola de OBS, API, SDK u OBS Browser+ para configurar la lectura directa.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Lectura directa |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Configuración de la política de lectura directa para objetos archivados en un bucket |
| OBS Browser+ | - |

8.14 Eliminación de un objeto

Escenario de la aplicación

Puede eliminar archivos o carpetas no deseados para ahorrar espacio de almacenamiento y reducir costos.

AVISO

Si el control de versiones no está habilitado para un bucket, los objetos eliminados no se pueden recuperar. Tenga cuidado cuando realice esta acción.

Eliminación manual o automática de objetos

Puede eliminar objetos manualmente en cualquier momento. También puede usar [reglas de ciclo de vida](#) para eliminar periódicamente objetos de un bucket o borrar todos los objetos de un bucket a la vez.

Eliminación de un objeto o un lote de objetos

Puede eliminar uno o más objetos a la vez. Para obtener más información acerca de cómo eliminar objetos por lotes, consulte [¿Soporta OBS la eliminación por lotes de objetos?](#)

Mecanismo de eliminación de objetos cuando se habilita el control de versiones

Cuando el control de versiones está habilitado, OBS utiliza diferentes métodos de eliminación para diferentes objetos.

- Al eliminar un archivo o carpeta no se elimina de forma permanente. El archivo o carpeta eliminado se conservará en la lista **Deleted Objects** y se marcará con el **Delete Marker**.
 - Si desea eliminar el archivo o carpeta de forma permanente, debe eliminarlo de la lista **Deleted Objects**.
 - Para recuperar un archivo eliminado, puede cancelar la eliminación mediante la operación **Undelete**. Para más detalles, consulte [Recuperación de un objeto](#).
- Al eliminar una versión de un objeto se eliminará permanentemente esa versión. Si la versión eliminada es la más reciente, la siguiente última versión se convierte en la última versión.

Cuando el control de versiones está habilitado, los archivos de la lista **Deleted Objects** también tienen varias versiones. Tenga en cuenta los siguientes puntos al eliminar diferentes versiones de archivos:

- Si elimina una versión con el **Delete Marker**, en realidad recupera esa versión específica en lugar de eliminarla permanentemente. Para más detalles, consulte [Recuperación de un objeto](#).
- Si elimina una versión sin el **Delete Marker**, esa versión específica se eliminará permanentemente. Incluso si el objeto se recupera más tarde, esta versión no se recuperará.

Para obtener más información, consulte [Mecanismo de recuperación de objetos cuando se habilita el control de versiones](#).

¿Cómo usar esta función?

Puede usar la consola OBS, API, SDK, OBS Browser+, obsutil u obsfs para eliminar objetos.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Eliminación de un archivo o carpeta |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |

| Herramienta | Referencia |
|--------------|--|
| API | Eliminación de un objeto |
| OBS Browser+ | - |
| obsutil | Eliminación de un objeto |

8.15 Recuperación de un objeto

Escenario de la aplicación

Si el control de versiones no está habilitado para un bucket, los objetos eliminados no se pueden recuperar. Si el control de versiones está habilitado, puede recuperar objetos eliminados a través de la operación Undelete. Para obtener más información sobre el control de versiones, consulte [Control de versiones](#).

Mecanismo de recuperación de objetos cuando se habilita el control de versiones

Cuando un bucket tiene activada la función de control de versiones, eliminar un archivo desde la lista **Objects** no significa que lo elimina permanentemente. El archivo eliminado se conservará con el **Delete Marker** en la lista **Deleted Objects**. Puede recuperar un objeto eliminado mediante la operación Undelete.

Al realizar la operación Undelete, tenga en cuenta los siguientes puntos:

1. Solo puede recuperar archivos borrados, pero no carpetas.
Después de recuperar un archivo eliminado, el archivo se recupera y aparecerá en la lista **Objects**. A continuación, puede realizar operaciones básicas en el archivo como normalmente lo hace con otros objetos. Si el archivo se almacenó en una carpeta antes de la eliminación, se recuperará a su ruta original después de recuperarlo.
2. Los archivos eliminados de **Deleted Objects** también tienen varias versiones. Al eliminar diferentes versiones de archivos, tenga en cuenta los siguientes puntos:
 - Si elimina una versión con el **Delete Marker**, en realidad recupera esa versión específica en lugar de eliminarla permanentemente.
 - Si elimina una versión sin el **Delete Marker**, esa versión específica se eliminará permanentemente. Incluso si el objeto se recupera más tarde, esta versión no se recuperará.
3. Existe al menos una versión sin el **Delete Marker** en la lista **Deleted Objects**. De lo contrario, la eliminación no se puede cancelar.

¿Cómo usar esta función?

Puede usar la consola OBS para recuperar objetos.

| Herramienta | Referencia |
|----------------|--------------------------------------|
| Consola de OBS | Recuperar un archivo |

8.16 Gestión de fragmentos

Escenario de la aplicación

Los fragmentos se refieren a datos incompletos en un bucket. Los datos se cargan en OBS en modo multiparte. En los siguientes escenarios comunes y otros, una carga falla y se generan fragmentos. Necesita borrar estos fragmentos para liberar espacio de almacenamiento.

- La red se encuentra en malas condiciones y la conexión con el servidor de OBS se interrumpe con frecuencia.
- La tarea de carga se suspende manualmente.
- El dispositivo presenta fallas.
- El dispositivo se apaga repentinamente.

Los fragmentos en OBS consumen espacio de almacenamiento y se facturan en función de los precios del espacio de almacenamiento. Si los fragmentos se generan debido a interrupciones o fallos de las tareas de carga de varias partes, puede reanudar dichas tareas para que los fragmentos se eliminen, o puede borrar directamente los fragmentos para ahorrar espacio de almacenamiento.

Para obtener más información sobre cómo administrar fragmentos, consulte [¿Cómo gestiono fragmentos?](#)

NOTA

- Si un bucket no se elimina, compruebe si se han eliminado todos los fragmentos. Si no, elimine todos los fragmentos del bucket.
- Si no existe ningún objeto en el bucket pero la tarifa aún se está deduciendo, compruebe si hay fragmentos en el bucket. En caso afirmativo, elimine los fragmentos para reducir los costos de almacenamiento.

9 Sistema de archivos paralelo

Sistema de archivos paralelo (Parallel File System, PFS) es un sistema de archivos de alto rendimiento proporcionado por OBS, con latencia de acceso en milisegundos. PFS admite ancho de banda de nivel TB/s y millones de IOPS, lo que lo hace ideal para procesar cargas de trabajo de cómputo de alto rendimiento (HPC).

Puede acceder a los datos en un sistema de archivos paralelo mediante API estándar de OBS. También soporta lectura y escritura de datos a través de obsfs, un cliente PFS que soporta POSIX. obsfs se puede implementar en un ECS, y luego puede usar los obsfs para montar un sistema de archivos paralelo al sistema operativo Linux que se ejecuta en el ECS. El sistema de archivos paralelo montado funciona como un sistema de archivos local. Puede gestionar el sistema de archivos paralelo montado en línea, incluida la creación, eliminación, cambio de nombre de archivos y carpetas, así como la modificación de archivos.

Para obtener más información, consulte la [Guía de características del sistema de archivos paralelo](#).

10 Procesamiento de imágenes

Integrada en OBS, la función de procesamiento de imágenes proporciona servicios de procesamiento de imágenes estables, seguros, eficientes y de bajo costo. Mediante el uso de esta función, puede adelgazar (reducir el tamaño), recortar, comprimir y marcas de agua imágenes, así como convertir los formatos de las imágenes.

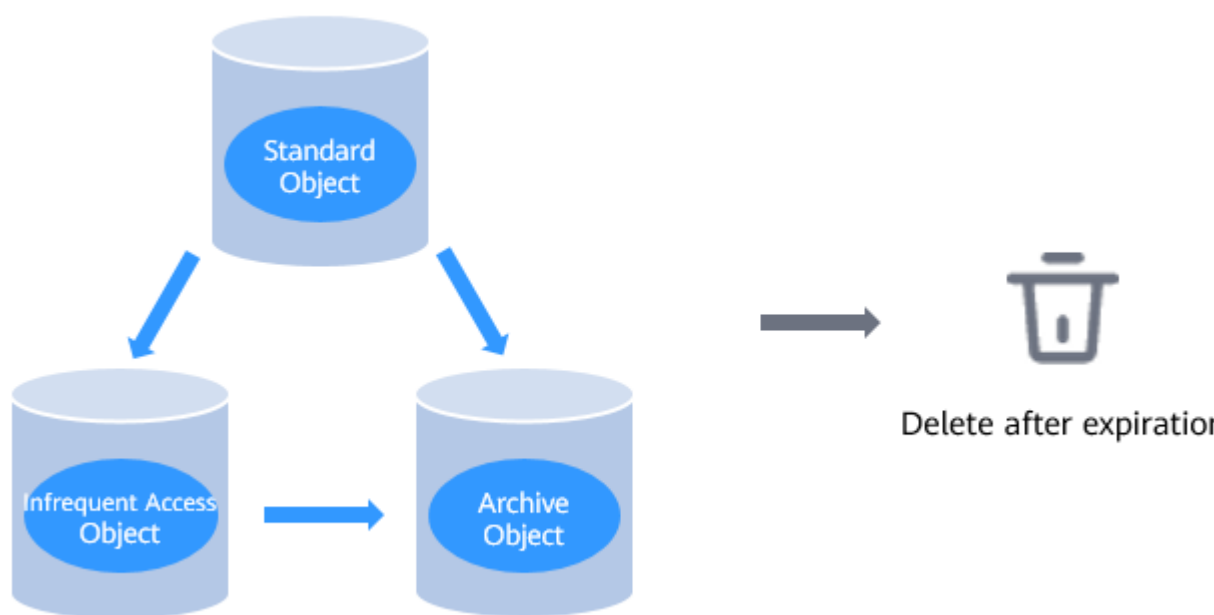
Esta característica aprovecha las API de REST y la Consola de OBS. Puede procesar las imágenes almacenadas en OBS de varias maneras en cualquier momento y en cualquier lugar, y obtener las imágenes procesadas de inmediato.

Para obtener más información, consulte la [Guía de funciones de procesamiento de imágenes](#).

11 Gestión del ciclo de vida

La gestión del ciclo de vida para OBS incluye la eliminación periódica de objetos desde bucket y la transición entre clases de almacenamiento de objetos según las reglas predefinidas.

Figura 11-1 Gestión del ciclo de vida



Escenario de la aplicación

Puede configurar las reglas de gestión del ciclo de vida para realizar lo siguiente:

- Eliminación periódica de archivos que solo están destinados a ser retenidos durante periodos de tiempo especificados
- Transición de documentos a los que rara vez se accede a la clase de almacenamiento Infrequent Access o Archive o eliminarlos
- Almacenamiento de varios tipos de datos en OBS con fines de archivo, como medios digitales, registros financieros y médicos, datos originales de secuencias del genoma,

respaldo de bases de datos a largo plazo y datos que deben conservarse para el cumplimiento normativo.

- Programar la eliminación de todos los archivos de un bucket. La eliminación manual de objetos requiere mucho tiempo y solo se puede eliminar un número limitado de objetos a la vez.

Puede definir las reglas de ciclo de vida para identificar objetos y gestionar los ciclos de vida de los objetos basándose en las reglas.

Los objetos a los que ya no se accede con frecuencia se pueden pasar a Infrequent Access o Archive, lo que reduce los costos. En resumen, transición significa básicamente que la clase de almacenamiento de objetos se altera sin copiar el objeto. También puede cambiar manualmente la clase de almacenamiento de un objeto en la página **Objects**. Para obtener más información, consulte [Carga de un archivo](#).

La configuración del ciclo de vida se puede agregar a un bucket con el control de versiones habilitado o deshabilitado. De forma predeterminada, el control de versiones está deshabilitado. Puede habilitarlo. Si el control de versiones está habilitado para un bucket, el bucket mantendrá la versión de objeto actual y cero o más versiones de objeto no actuales. Puede reducir los costos de almacenamiento de objetos mediante reglas de control de versiones y de ciclo de vida. Las acciones de configuración del ciclo de vida predefinidas pueden facilitar la gestión durante el ciclo de vida de las versiones de objetos actuales y no actuales.

Precauciones

- No hay límite en el número de reglas del ciclo de vida de un bucket, pero el tamaño total de las descripciones XML de todas las reglas del ciclo de vida de un bucket no puede superar los 20 KB.
- La duración mínima de almacenamiento del almacenamiento de Archive es de 90 días. Después de la transición, si el período de almacenamiento del objeto Archive es más corto que la duración mínima de almacenamiento, deberá pagar los días restantes.
- Restricciones en la transición de clase de almacenamiento:
 - Las reglas de ciclo de vida sólo pueden hacer la transición de objetos de la clase de almacenamiento Standard a la clase de almacenamiento Infrequent Access. Se requieren operaciones manuales para realizar la transición de objetos de la clase de almacenamiento Infrequent Access a la clase de almacenamiento Standard.
 - Los objetos sólo se pueden pasar de la clase de almacenamiento Standard o Infrequent Access a la clase de almacenamiento Archive. Si desea realizar la transición de objetos de la clase de almacenamiento Archive a la clase de almacenamiento Standard o Infrequent Access, primero debe restaurar objetos de la clase de almacenamiento Archive y, a continuación, realizar la transición manual de la clase de almacenamiento.
- Supresión de objetos al expirar:

OBS proporciona la acción **Expiration** que le permite establecer el tiempo de expiración para los objetos. Además, para los bucket con control de versiones habilitado, OBS proporciona la operación **NoncurrentVersionExpiration** que le permite establecer el tiempo de caducidad para las versiones de objetos no actuales.

 - Para los bucket con el control de versiones deshabilitado, la acción **Expiration** permite eliminar los objetos caducados de forma permanente.
 - Si el control de versiones está habilitado (o suspendido) para un bucket, la acción **Expiration** eliminará lógicamente la versión actual de un objeto creando un

marcador de eliminación para la versión actual y reconociendo el marcador de eliminación como la nueva versión actual. La acción

NoncurrentVersionExpiration elimina objetos de versiones no actuales de forma permanente.

- Además de los bucket, también puede configurar el tiempo de caducidad de los objetos durante las cargas de objetos. El tiempo de caducidad de los objetos prevalece contra el de los cubos.

Después de que un objeto expira, OBS agrega el objeto a la cola de eliminación y lo elimina asincrónicamente. Esto puede hacer que el tiempo de eliminación sea posterior al tiempo de expiración. Después de que un objeto caduque, no se le cobrará la tarifa de duración de almacenamiento relacionada.

Para consultar el tiempo de caducidad planificado de un objeto, puede llamar a la API de objeto GET o a la API de objeto HEAD. Estas API devuelven encabezados de respuesta que proporcionan información de caducidad sobre el objeto.

- Después de modificar una regla de ciclo de vida, la modificación no se aplica a los objetos que ya cumplen las condiciones configuradas. Por ejemplo, la regla de ciclo de vida original hace que los objetos que cumplen las condiciones configuradas expiren y, a continuación, se eliminen después de que se almacenen durante un día. Bajo esta regla, los objetos cargados el 1 de enero de 2021 se eliminarán el 3 de enero de 2021. El 3 de enero de 2021, si cambia la regla del ciclo de vida para eliminar objetos después de que se almacenen durante siete días, los objetos cargados el 1 de enero de 2021 se eliminarán el 3 de enero de 2021, pero esta modificación se aplicará a los cargados el 2 de enero de 2021 y después.

Reglas del ciclo de vida

Las reglas del ciclo de vida tienen los siguientes elementos clave:

- Política: configure una regla de ciclo de vida que surta efecto a los objetos especificados.
 - Por prefijo: puede especificar un prefijo de nombre de objeto, por lo que la regla de ciclo de vida tendrá efecto en los objetos que comparten el mismo prefijo.
 - Bucket completo: puede especificar un bucket completo, por lo que la regla del ciclo de vida tendrá efecto en todos los objetos del bucket.
- Hora: hora programada en la que se realiza la transición de la clase de almacenamiento de objetos

Puede especificar el número de días después de los cuales los objetos que se han actualizado recientemente y cumplen las condiciones especificadas se transfieren automáticamente a la clase de acceso infrecuente o de archivo; o los que han vencido, se eliminan automáticamente.

- Transición a Infrequent Access: puede especificar el número de días después de los cuales los objetos que se han actualizado por última vez y que cumplen las condiciones especificadas pasan automáticamente a Infrequent Access.
- Transición al Archive: puede especificar el número de días después de los cuales los objetos que se han actualizado por última vez y que cumplen las condiciones especificadas se pasan automáticamente al Archive.
- Eliminados al expirar: puede especificar el número de días después de los cuales los objetos que se han actualizado por última vez y que cumplen las condiciones especificadas se eliminan automáticamente.

El tiempo de la regla del ciclo de vida tiene las siguientes restricciones:

- Si solo se trata de una transición, el tiempo debe ser al menos un día después.
- Si se trata de transiciones múltiples, la última transición debe ser al menos un día más que la transición anterior.

Por ejemplo, si desea que una regla de ciclo de vida cambie la clase de almacenamiento de objetos y elimine objetos caducados, puede establecer el tiempo de transición en 23 días después y el tiempo de eliminación en al menos 24 días después.

NOTA

El período mínimo de facturación es de 30 días para el almacenamiento de Infrequent Access y de 90 días para el almacenamiento de archivos. Si los objetos se eliminan mediante una regla de ciclo de vida antes de que se hayan almacenado durante este período mínimo, se le cobrará el período completo (30 o 90 días, según el servicio).

Ejemplo: se carga un objeto a la clase de almacenamiento Standard y no se realiza ninguna otra operación en el objeto. El bucket, donde reside el objeto, tiene una regla de ciclo de vida que cambiará la clase de almacenamiento de objetos desde Standard hacia la de Infrequent Access tres días más tarde, luego de Infrequent Access a Archive y, finalmente, eliminará los objetos a medida que caduquen siete días más tarde. En este caso, se le cobrará tres días de almacenamiento Standard, 30 días de almacenamiento Infrequent Access y 90 días de almacenamiento Archive.

Elementos de configuración del ciclo de vida

Puede establecer el formato de configuración del ciclo de vida en XML. La configuración contiene una o más reglas de ciclo de vida.

Cada regla consta de los siguientes contenidos:

- Metadatos, especificando el ID de regla y si la regla está activada o desactivada. Si la regla está deshabilitada, OBS no realiza las acciones especificadas en la regla.
- Criterios de filtrado, identificando los objetos a los que se aplican las reglas del ciclo de vida. Puede establecer el prefijo de nombre de objeto para que sea el criterio de filtrado.
- Cuándo (una fecha o un período de tiempo) se realizará una acción de transición o expiración en los objetos del ciclo de vida.

Ejemplos de configuración:

Ejemplo 1: Configuración del ciclo de vida de un bucket con el control de versiones deshabilitado

De forma predeterminada, el control de versiones está deshabilitado para los bucket. Cada objeto tiene una sola versión.

Supongamos que el control de versiones está deshabilitado para el bucket y que desea que los objetos cuyo nombre comience con **documents/** pasen de la clase de almacenamiento Standard a la clase de almacenamiento Infrequent Access 30 días después de que se hayan cargado, luego de la clase de almacenamiento Infrequent Access a la clase de almacenamiento de archivo 60 días después de que se cargan, y finalmente se eliminan un año después de que se cargan. Para lograr esto, puede agregar la siguiente configuración de ciclo de vida para su bucket. Esta regla del ciclo de vida incluye dos operaciones (**Transition** y **Expiration**). Esta regla se aplica a los objetos cuyo prefijo clave es **documents** (especificado en el elemento **Prefix**).

```
<LifecycleConfiguration>
  <Rule>
    <ID>sample-rule</ID>
```

```
<Prefix>documents/</Prefix>
<Status>Enabled</Status>
<Expiration>
  <Days>365</Days>
</Expiration>
<Transition>
  <Days>30</Days>
  <StorageClass>WARM</StorageClass>
</Transition>
<Transition>
  <Days>60</Days>
  <StorageClass>COLD</StorageClass>
</Transition>
</Rule>
</LifecycleConfiguration>
```

Ejemplo 2. Configuración del ciclo de vida de un bucket con control de versiones habilitado

Puede habilitar el control de versiones para los bucket. Si el control de versiones está habilitado para un bucket, el bucket conservará la versión de objeto actual y sus versiones de objeto no actuales. Para más detalles, consulte [Control de versiones](#). El control de versiones le permite mantener los registros históricos de objetos y la gestión del ciclo de vida le permite controlar la retención de versiones de objetos, así como la transición de clases de almacenamiento.

Para un bucket con control de versiones habilitado, la configuración del ciclo de vida proporciona varias acciones predefinidas que se pueden utilizar para administrar versiones de objetos no actuales. En este ejemplo, la configuración del ciclo de vida tiene una regla que especifica dos operaciones **NoncurrentVersionTransition** y **NoncurrentVersionExpiration** para los objetos cuyo prefijo clave es **logs/**. Cuando los objetos se convierten en versiones no actuales, la acción **NoncurrentVersionTransition** los pasa a Infrequent Access después de 30 días y a Archive después de 60 días. La acción **NoncurrentVersionExpiration** elimina permanentemente los objetos 180 días después de que se conviertan en versiones no actuales.

```
<LifecycleConfiguration>
  <Rule>
    <ID>sample-rule</ID>
    <Prefix>logs/</Prefix>
    <Status>Enabled</Status>
    <NoncurrentVersionExpiration>
      <NoncurrentDays>180</NoncurrentDays>
    </NoncurrentVersionExpiration>
    <NoncurrentVersionTransition>
      <NoncurrentDays>30</NoncurrentDays>
      <StorageClass>WARM</StorageClass>
    </NoncurrentVersionTransition>
    <NoncurrentVersionTransition>
      <NoncurrentDays>60</NoncurrentDays>
      <StorageClass>COLD</StorageClass>
    </NoncurrentVersionTransition>
  </Rule>
</LifecycleConfiguration>
```

Puede usar las acciones predefinidas de **NoncurrentVersionTransition** y **NoncurrentVersionExpiration** para gestionar las versiones no actuales en su bucket.

Por lo general, cada regla del ciclo de vida consta de las siguientes partes:

- Metadatos: especificando el ID de regla (elemento <ID>) y si la regla está habilitada o deshabilitada (elemento <Status>). Si la regla está deshabilitada, OBS no realiza las acciones especificadas en la regla.
- Prefix (elemento <Prefix>), que identifica los objetos a los que se aplica la regla.
- Acciones que desea realizar en los objetos especificados (como **NoncurrentVersionTransition** y **NoncurrentVersionExpiration** mencionados en el ejemplo

anterior). Cada acción incluye la fecha en que o período de tiempo después del cual se realizará la acción.

Los elementos de la regla de configuración del ciclo de vida se describen de la siguiente manera:

- Elemento **ID**

Una configuración de ciclo de vida puede tener un máximo de 1000 reglas. Cada regla tiene el ID único.

- Elemento **Status**

El valor puede ser **Enabled** o **Disabled**. Si una regla está deshabilitada, OBS no realiza ninguna acción definida en la regla.

- Elemento **Prefix**

Una regla de ciclo de vida se aplica a uno o varios objetos con el prefijo de nombre especificado en la regla. Suponga que tenga los siguientes objetos:

- logs/day1
- logs/day2
- logs/day3
- ExampleObject.jpg

Si establece **Prefix** en **ExampleObject.jpg**, la regla sólo se aplica al **ExampleObject.jpg**. Si establece **Prefix** en **logs/**, la regla se aplica a los objetos cuyo nombre comience por **logs/**. Si deja **Prefix** nulo, la regla se aplica a todos los objetos del bucket.

- Elemento **Action**

Puede especificar acciones predefinidas en la regla de ciclo de vida para realizarlas en los depósitos del ciclo de vida. Las acciones predefinidas incluyen **Transition**, **Expiration**, **NoncurrentVersionTransition**, y **NoncurrentVersionExpiration**. El efecto de acción depende del estado del control de versiones de un bucket.

NOTA

De forma predeterminada, el control de versiones está deshabilitado para los bucket. Puede habilitar el control de versiones de bucket, de modo que cada objeto tenga una versión actual y pueda tener una o más versiones no actuales. Además, puede deshabilitar el control de versiones. Para más detalles, consulte [Control de versiones](#).

- Acción **Transition**

Esta acción archiva los objetos cambiando la clase de almacenamiento de objetos a Infrequent Access o Archive. Cuando llega la fecha o el período de tiempo especificado, el sistema cambia la clase de almacenamiento de objetos a Infrequent Access o Archive.

- Para los bucket con el control de versiones deshabilitado, la acción **Transition** cambia la clase de almacenamiento de objetos a Infrequent Access o Archive.
- Para los bucket con el control de versiones habilitado y suspendido, la acción **Transition** cambia la clase de almacenamiento de la versión de objeto actual a Infrequent Access o Archive. Esta acción no afecta a las versiones de objetos no actuales.

- Acción **Expiration**

Esta acción establece el tiempo en que caducan los objetos. Los objetos no están disponibles una vez que caducan. Si los objetos caducados se eliminarán permanentemente depende del estado de control de versiones del bucket.

NOTA

- La acción **Expiration** no eliminará las partes cargadas de una carga de varias partes incompleta.
- Para los bucket con el control de versiones deshabilitado, la acción **Expiration** elimina los objetos permanentemente y los objetos eliminados no se pueden restaurar.
 - Para los bucket con control de versiones habilitado, esta acción se aplica solo a las versiones de objetos actuales, en lugar de a las versiones de objetos no actuales. Esta acción no elimina permanentemente las versiones de objetos actuales. Retiene la versión actual como una versión no actual al agregarle un marcador de eliminación. Esta acción no se realizará en las versiones de objetos actuales que ya tengan los marcadores de eliminación. Si la versión actual del objeto es la única versión del objeto y tiene un marcador de eliminación, OBS eliminará la versión actual del objeto. El borrado de un marcador de eliminación puede tardar un tiempo, ya que OBS necesita confirmar que el marcador de eliminación es la única versión del objeto.

Si inicia una solicitud GET en un objeto cuya versión actual tiene el marcador de eliminación sin especificar el ID de versión, OBS identificará el objeto como eliminado y devolverá el error 404 Object Not Found. Pero puede especificar el ID de versión en la solicitud GET para recuperar el objeto eliminado.

Por ejemplo, puede establecer una regla para que el objeto llamado photo.gif caduque 5 días después de su carga. Si photo.gif se crea a las 10:30 UTC del 1 de enero de 2016, la regla de vencimiento se ejecutará en un punto de tiempo después de las 00:00 UTC (cinco días después de la creación del objeto) del 7 de enero de 2016. La hora no será posterior a las 23:59 UTC del 7 de enero de 2016. Para un bucket con el control de versiones deshabilitado, una operación de eliminación elimina permanentemente photo.gif. Para un bucket con control de versiones habilitado, después de ejecutar la regla de caducidad, photo.gif (versión 111111) todavía se almacena en el bucket y se puede acceder si es necesario, pero la versión actual (versión 4857693) del objeto tiene un marcador de eliminación. El objeto original photo.gif se convierte en una versión no actual. Para obtener más información sobre cómo funciona un marcador de eliminación, consulte [Control de versiones](#).

Para los bucket con versiones suspendidas, OBS creará marcadores de eliminación para objetos caducados cuyo ID de versión sea **null**. Cualquier versión **null** existente será sobrescrita por nuevas versiones **null**, y los datos asociados con esta versión no se pueden restaurar.

Acciones específicas para los bucket con control de versiones habilitado (o suspendido)

Las acciones del ciclo de vida de **Transition** y **Expiration** pueden administrar el ciclo de vida de las versiones de objetos actuales. Las acciones **NoncurrentVersionTransition** y **NoncurrentVersionExpiration** pueden administrar el ciclo de vida de las versiones de objetos no actuales.

Las siguientes acciones de configuración del ciclo de vida sólo se pueden realizar en los bucket con control de versiones habilitado (o suspendido). En un bucket con control de versiones habilitado, un objeto puede tener varias versiones, incluyendo una versión actual y cero o más versiones no actuales. Puede utilizar estas acciones para solicitar a OBS que realice operaciones específicas en versiones de objetos no actuales. Estas acciones no afectan a las versiones de objetos actuales.

NoncurrentVersionTransition: especifica el período de tiempo tras el cual las versiones no actuales se transitarán desde Standard hacia Infrequent Access o Archive.

`NoncurrentVersionExpiration`: Especifica el período de tiempo tras el cual las versiones de objetos no actuales se eliminarán permanentemente. No se puede restaurar un objeto eliminado.

Por ejemplo, si desea habilitar un período de cinco días para corregir cualquier eliminación o sobrescritura accidental, puede configurar una regla de caducidad para que el objeto se pueda eliminar 5 días después de que se haya convertido en una versión no actual.

En 1/1/2016 10:30 (hora UTC), creó un objeto llamado `photo.gif` cuyo ID de versión es 111111. En 2/1/2016 10:30 (hora UTC), eliminó accidentalmente este objeto y OBS creó un marcador de eliminación para este objeto con un nuevo ID de versión, 4857693. En los próximos cinco días, se le permitió restaurar el objeto original `photo.gif` cuyo ID de versión es 111111.

En 8/1/2016 00:00 (hora UTC), la acción `NoncurrentVersionExpiration` eliminó permanentemente el objeto `photo.gif` cuyo ID de versión es 111111 (después de que se convirtió en una versión no actual durante 5 días.)

¿Cómo verifica OBS durante cuánto tiempo un objeto se ha convertido en una versión no actual?

En un bucket con control de versiones habilitado, un objeto puede tener varias versiones, incluida una versión actual y cero o más versiones no actuales. Cada vez que se carga un objeto, la versión actual existente se guarda como una versión no actual, mientras que la versión cargada se convierte en la versión actual. Para comprobar los días en que una versión de objeto se ha convertido en una versión no actual, OBS comprueba la hora de creación del objeto correspondiente de la versión actual. OBS utiliza los días en que el objeto de la versión actual ha sido creado como los días en que el objeto de una versión no actual ha existido.

Restaurar una versión no actual mediante la configuración del ciclo de vida

Puede utilizar cualquiera de los métodos siguientes para restaurar un objeto a una versión no actual:

1. Copie una versión de objeto no actual en el mismo bucket. La versión copiada se convertirá en la versión actual y todas las versiones de objeto están reservadas.
2. Elimine la versión de objeto actual de forma permanente. Después de eliminar la versión actual de un objeto, una versión no actual del objeto se convierte en la versión actual.

Se recomienda utilizar el primer método para copiar la versión de objeto no actual en el mismo bucket. Debido a la sintaxis de coherencia de OBS, antes de cambiar el modo de comunicación, una versión actual que se elimina permanentemente no puede desaparecer (OBS puede no estar al tanto de esta acción de eliminación). Además, la acción de caducidad para las versiones no actuales, si ha configurado, puede eliminar las versiones de objetos no actuales de forma permanente, incluidas las que desea restaurar. Por lo tanto, el método de copiar una versión no actual es más seguro.

Tabla 11-1 enumera la relación entre la regla de configuración del ciclo de vida y el estado de control de versiones de un bucket.

Tabla 11-1 Acciones de configuración y estado de control de versiones de los bucket

| Operación | Bucket con control de versiones deshabilitado | Bucket con control de versiones habilitado | Bucket con control de versiones suspendido |
|---|--|---|--|
| Transición (Realizado en un objeto cuando llega la fecha o el período de tiempo especificado por el ciclo de vida se agota.) | Los objetos se pueden cambiar a Infrequent Access o Archive. | Si la versión actual no tiene un marcador de eliminación y es la última versión, la versión actual se puede cambiar a Infrequent Access o Archive. | Si la versión actual no tiene un marcador de eliminación y no es la última versión, la versión actual se puede cambiar a Infrequent Access o Archive. |
| Caducidad (Realizado en un objeto cuando llega la fecha o el período de tiempo especificado por el ciclo de vida se agota.) | La acción Expiración elimina los objetos y los objetos eliminados no se pueden restaurar. | Si la versión actual no tiene un marcador de eliminación, se crea un marcador de eliminación que se considera la versión actual. La versión actual existente se convierte en una versión no actual. | Para los bucket con versiones suspendidas, OBS creará marcadores de eliminación para objetos caducados cuyo ID de versión sea null . Cualquier versión null existente será sobrescrita por nuevas versiones null , y los datos asociados con esta versión no se pueden restaurar. |
| NoncurrentVersion-Transition (Se realiza en un objeto después de que se convierta en una versión no actual durante un número determinado de días). | La acción NoncurrentVersion-Transition no es válida. | Si la versión actual no es un marcador de eliminación y no es la última versión, la versión actual se puede transitar a Infrequent Access o Archive. | Si la versión actual no tiene un marcador de eliminación y no es la última versión, la versión actual se puede cambiar a Infrequent Access o Archive. |

| Operación | Bucket con control de versiones deshabilitado | Bucket con control de versiones habilitado | Bucket con control de versiones suspendido |
|--|---|--|---|
| NoncurrentVersionExpiration (Se realiza en un objeto después de que se convierta en una versión no actual durante un número determinado de días). | La acción NoncurrentVersionExpiration no es válida. | Los objetos no actuales se eliminan y no se pueden restaurar una vez eliminados. | Si la versión actual no tiene un marcador de eliminación y no es la última versión, la versión actual se puede cambiar a Infrequent Access o Archive. |

Reglas de ciclo de vida basadas en fechas

Puede especificar las fechas de ejecución de las acciones **Transition** y **Expiration**. Las fechas deben cumplir con las normas ISO 8601 y la hora exacta es siempre 00:00 (hora UTC). Si especifica la hora hasta una fecha pasada, todos los objetos aplicables se ejecutarán inmediatamente.

No se puede crear una regla de ciclo de vida basada en fechas en OBS Console.

ADVERTENCIA

Una acción de ciclo de vida con una fecha especificada no es una acción única. Incluso si la fecha ha pasado, OBS adoptará esta acción siempre que el ciclo de vida esté habilitado.

Suponga que haya especificado una fecha para realizar la acción **Expiration** para eliminar todos los objetos (sin establecer ningún criterio de filtrado). En la fecha especificada, OBS hace que todos los objetos del bucket expiren. Además, OBS continúa haciendo que todos los objetos nuevos creados en el bucket expiren. Para finalizar la acción **Expiration**, debe eliminar esta acción de la configuración del ciclo de vida, deshabilitar la regla o eliminarla de la configuración del ciclo de vida.

Regla de ciclo de vida basada en periodos de tiempo

Puede especificar cuántos días después de crear un objeto se realizará la acción **Transition** o **Expiration** en el objeto. Después de especificar el número de días, OBS comienza a calcular la hora a partir de las 00:00 (hora UTC) del día siguiente. Por ejemplo, creó un objeto en 15/1/2016 10:30 (hora UTC) y especificó que los objetos se cambiarían 3 días después de la creación, el objeto se transitaría en 19/1/2016 00:00 (hora UTC).

AVISO

OBS sólo registra la última fecha de modificación de cada objeto. En la consola OBS, puede ver la última hora de modificación (**LastModified**) de un objeto en la página de propiedades del objeto. Después de crear un objeto, la fecha es la fecha de creación. Si se reemplaza el objeto, la fecha también cambiará.

Al utilizar la acción `NoncurrentVersionTransition` o `NoncurrentVersionExpiration`, puede especificar después de cuántos días desde que un objeto cambie a una versión no actual (debido a sobrescribir o eliminar) se realizará la acción en el objeto.

Cuando se utilizan las operaciones `NoncurrentVersionTransition` y `NoncurrentVersionExpiration` para especificar el número de días, OBS agrega el número de días especificado en la regla al momento en que la versión del objeto se convierte en una versión no actual. A continuación, se calcula el tiempo de operación a partir de las 00:00 (hora UTC) del día siguiente. Por ejemplo, la versión actual de un objeto en un bucket se crea a las 10:30 UTC del 1 de enero de 2016. Si la versión del objeto se convierte en una versión no actual a las 10:30 UTC del 15 de enero de 2016 y especifica tres días en la regla de conversión, la fecha para cambiar la clase de almacenamiento de objetos se calcula como 00:00 UTC del 19 de enero de 2016.

 **NOTA**

Al configurar las reglas del ciclo de vida, dentro de una regla y para las reglas cuyos prefijos tienen una relación de inclusión, **Date** o **Days** de **Transition** y **Expiration** deben ser coherentes.

¿Cómo usar esta función?

Puede utilizar la consola de OBS, las API o los SDK para configurar las reglas del ciclo de vida.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Configuración de una regla de ciclo de vida |
| SDKs | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| APIs | Configuración de reglas del ciclo de vida del bucket Obtención de la configuración del ciclo de vida del cucharón Eliminación de reglas del ciclo de vida |

12 Replicación entre regiones

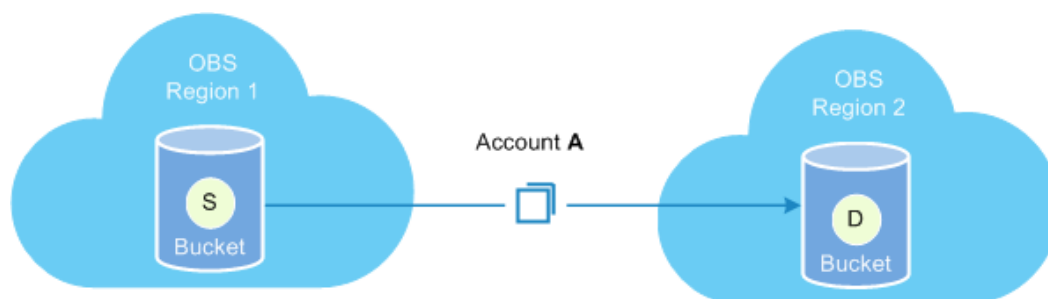
Escenario de la aplicación

La replicación entre regiones proporciona la capacidad de recuperación ante desastres en todas las regiones, lo que le permite configurar una solución de backup remoto.

La replicación entre regiones se refiere al proceso de replicación automática y asíncrona de datos desde un bucket (bucket de origen) a otro bucket (bucket de destino) entre regiones mediante la creación de una regla de replicación entre regiones. El bucket de origen y el bucket de destino deben pertenecer a la misma cuenta. No se admite la replicación entre cuentas.

Para una regla de replicación entre regiones, puede configurarla para que coincida con un prefijo de objeto predefinido de modo que se replicarán todos los objetos con este prefijo. Alternativamente, puede configurar la regla para que se aplique a todo el bucket de modo que se replicarán todos los objetos del bucket. Los objetos replicados en el bucket de destino son copias precisas de los objetos en el bucket de origen. Tienen los mismos nombres de objeto, metadatos, contenido, tamaño, hora de última modificación, creador, ID de versión, metadatos definidos por el usuario y ACL. De forma predeterminada, la clase de almacenamiento de una copia de objeto es la misma que la del objeto de origen. También puede especificar una clase de almacenamiento diferente para una copia de objeto.

Figura 12-1 Replicación entre regiones



- Cumplimiento de normativas

OBS almacena datos a través de AZs que están relativamente lejos entre sí. Sin embargo, el cumplimiento normativo puede requerir distancias adicionales. La replicación entre regiones permite a la OBS replicar datos entre regiones para el cumplimiento normativo.

- Latencias minimizadas
Es posible que sea necesario acceder a los mismos recursos de OBS desde diferentes ubicaciones. Para minimizar la latencia de acceso, puede utilizar la replicación entre regiones para crear copias de objetos en la región más cercana a los usuarios finales.
- Replicación de datos
La replicación entre regiones le permite migrar fácilmente los datos almacenados en OBS de una región a otra.
- Copia de seguridad de datos y recuperación ante desastres
Para garantizar la seguridad y disponibilidad de los datos, debe crear copias de seguridad explícitas para todos los datos escritos en OBS en el centro de datos de otra región, de modo que esté disponible una copia de seguridad segura en caso de que los datos de origen se dañen irrevocablemente.
- Fácil mantenimiento
Tiene un clúster de cómputo entre regiones para analizar la misma colección de objetos. Es necesario mantener réplicas de objetos en las dos regiones.

Contenidos Replicados

Una vez activada la regla de replicación entre regiones, los objetos que cumplen las siguientes condiciones se copian en el bucket de destino:

- Objetos recién cargados (excluyendo los objetos de la clase Almacenamiento Archive)
- Objetos actualizados. Por ejemplo, se actualiza el contenido del objeto o se actualiza la información de ACL de un objeto copiado.
- Objetos históricos de un bucket si está habilitada **Synchronizing Existing Objects** (excluyendo los objetos de la clase Almacenamiento Archive)

NOTA

La replicación entre regiones no replica objetos cifrados mediante SSE-C.

Restricciones

- Una regla de replicación entre regiones puede no surtir efecto inmediatamente después de su configuración. Por consiguiente, los objetos a los que se aplica esta regla pueden no replicarse inmediatamente después de configurar la regla.
- Actualmente, solo los bucket de la versión 3.0 admiten replicación entre regiones. Para comprobar la versión del bucket, vaya a la página **Overview** del bucket en OBS Console. A continuación, puede ver la versión del bucket en el área **Basic Information**.
- De forma predeterminada, los objetos cargados antes de que se active la replicación entre regiones no se copian en el bucket de destino a menos que esté habilitada la función para sincronizar objetos existentes.
- El bucket de origen y el bucket de destino deben pertenecer a diferentes regiones por separado. Los datos no se pueden copiar entre los bucket de la misma región.
- Los objetos no se pueden copiar desde el bucket de origen al bucket de destino si están en la clase de almacenamiento Archive.
- Si la región donde reside el bucket de destino no admite diferentes clases de almacenamiento de datos, las copias de objetos se almacenarán en la clase de almacenamiento estándar.

- El estado de control de versiones del bucket de origen debe ser el mismo que el del bucket de destino.
- Los objetos de un bucket de origen sólo se pueden copiar en un bucket de destino y no se pueden copiar de nuevo desde el bucket de destino a otro bucket. Por ejemplo, el bucket A y el bucket B están en dos regiones diferentes. Puede copiar datos del bucket A al bucket B o al revés. Sin embargo, las copias de datos en el bucket A o el bucket B ya no se pueden replicar.
- Si el control de versiones está habilitado o suspendido para los bucket de origen y destino, y la replicación entre regiones también está habilitada para ambos los bucket, eliminar un objeto sin especificar su versión en el bucket de origen también eliminará el objeto en el bucket de destino.
- Si cambia el estado de control de versiones del bucket de destino cuando está habilitada la replicación entre regiones, se producirá un error en la replicación de objetos. Si desea cambiar el estado de control de versiones del bucket de origen, deshabilite primero la replicación entre regiones y, a continuación, realice el cambio.
- Asegúrese de que los propietarios de los bucket de origen y destino tengan los permisos de lectura y escritura para los dos los bucket. De lo contrario, los datos no se pueden sincronizar. Si el sistema no tiene los permisos para leer el bucket de origen o escribir el bucket de destino debido a errores de permiso de lectura/escritura, los objetos no se pueden copiar correctamente y dicha replicación no se reanuda incluso si se rectifica el error de permiso.
- Para un bucket de origen, sólo puede crear una regla de replicación entre regiones que se aplique a todo el bucket para la replicación de todos los objetos del bucket. Sin embargo, puede crear un máximo de 100 reglas de replicación entre regiones basadas en prefijos de objeto para la replicación de objetos que coincidan con los prefijos.
- Actualmente, OBS solo admite la replicación entre un bucket de origen y un bucket de destino. No se admite la replicación de un bucket de origen a varios los bucket de destino. El bucket de destino se puede modificar. Sin embargo, al modificar el bucket de destino se cambiará el bucket de destino de todas las reglas existentes.
- Si elimina la configuración de la agencia OBS en una replicación entre regiones, el estado de replicación se convierte en **Failed**.
- No elimine, sobrescriba réplicas de objetos en el bucket de destino ni modifique sus ACL, lo que puede causar incoherencia de las últimas versiones de objetos o configuración de control de permisos entre el bucket de destino y el bucket de origen.
- Si la función para sincronizar objetos existentes está habilitada, la modificación de la configuración de replicación entre regiones puede provocar errores en la sincronización de objetos existentes. Por lo tanto, no modifique la configuración de replicación entre regiones antes de que finalice la sincronización.
- Si está habilitada la replicación entre regiones, no se pueden agregar datos al final de los objetos en el bucket de origen.
- Una vez completada una replicación con **Synchronize Existing Objects** habilitado, si la política de replicación no cambia, cualquier cambio de ACL de los objetos de origen se sincronizará con las copias de objetos. Sin embargo, los cambios de ACL de los objetos históricos de origen no se sincronizarán con las copias de los objetos históricos.
- Antes de replicar los datos, asegúrese de que las regiones de origen y destino puedan replicarse entre sí. [Figura 12-2](#) enumera las regiones admitidas. ✓ indica que los datos se pueden replicar entre regiones. x indica que los datos no se pueden replicar entre regiones.

Figura 12-2 Replicación entre regiones

| | CN North-Beijing4 | CN East-Shanghai1 | CN East-Shanghai2 | CN North-Beijing1 | CN North-Beijing2 | CN North-Ulanqab1 | CN Southwest-Guiyang1 | CN South-Guangzhou | CN-Hong Kong | AP-Bangkok | LA-Sao Paulo1 | AF-Johannesburg | LA-Mexico City1 | AP-Singapore |
|-----------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-----------------------|--------------------|--------------|------------|---------------|-----------------|-----------------|--------------|
| CN North-Beijing4 | / | √ | √ | √ | √ | √ | √ | √ | x | x | x | x | x | x |
| CN East-Shanghai1 | √ | / | √ | √ | √ | √ | √ | √ | x | x | x | x | x | x |
| CN East-Shanghai2 | √ | √ | / | √ | √ | √ | √ | √ | √ | x | x | √ | x | x |
| CN North-Beijing1 | √ | √ | √ | / | √ | √ | √ | √ | x | x | x | √ | x | x |
| CN North-Beijing2 | √ | √ | √ | √ | / | √ | √ | √ | x | x | x | x | x | x |
| CN North-Ulanqab1 | √ | √ | √ | √ | √ | / | √ | √ | x | x | x | x | x | x |
| CN Southwest-Guiyang1 | √ | √ | √ | √ | √ | √ | / | √ | x | x | x | x | x | x |
| CN South-Guangzhou | √ | √ | √ | √ | √ | √ | √ | / | x | x | x | x | x | x |
| CN-Hong Kong | x | x | √ | x | x | x | x | x | / | x | x | √ | √ | x |
| AP-Bangkok | x | x | x | x | x | x | x | x | x | / | x | x | x | x |
| LA-Sao Paulo1 | x | x | x | x | x | x | x | x | x | x | / | x | x | x |
| AF-Johannesburg | x | x | √ | √ | x | x | x | x | √ | x | x | / | x | x |
| LA-Mexico City1 | x | x | x | x | x | x | x | x | √ | x | x | x | / | x |
| AP-Singapore | x | x | x | x | x | x | x | x | x | x | x | x | x | / |

¿Cómo usar esta función?

Puede usar la consola OBS, API, SDK u obsutil para configurar la replicación entre regiones.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Configuración de la replicación entre regiones |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Configuración de la replicación entre regiones para un bucket |
| obsutil | Copia de objetos |

13 Encriptación del lado del servidor

Escenario de la aplicación

Después de habilitar el cifrado del lado del servidor, los objetos cargados en OBS se cifrarán y luego se almacenarán en el servidor. Al descargar los objetos encriptados, los datos cifrados se desenscriptan en el servidor y se muestran a los usuarios como texto no encriptado.

OBS admite los siguientes dos modos de cifrado del lado del servidor. Ambos modos utilizan el algoritmo de cifrado AES-256 estándar de la industria.

- La encriptación del lado del servidor mediante claves alojadas en KMS (SSE-KMS)
Debe crear una clave a través de Key Management Service (KMS) o utilizar la clave predeterminada proporcionada por KMS. La clave KMS se utiliza para realizar el cifrado del lado del servidor cuando se cargan objetos en OBS.
En el modo SSE-KMS, puede habilitar el cifrado predeterminado al crear un bucket. A continuación, todos los objetos cargados en el bucket se cifran. También puede habilitar el cifrado predeterminado para un bucket después de su creación y, a continuación, los objetos recién cargados se cifran.
OBS cifra solo los objetos cargados después de que la función de cifrado predeterminada esté activada. El estado de encriptación de los objetos existentes en el bucket permanece sin cambios. La deshabilitación de la encriptación por defecto no cambia el estado de encriptación de los objetos existentes en un bucket. Después de que se deshabilita esta función, todavía se pueden encriptar manualmente los objetos al cargarlos.
Puede usar la consola OBS, API, SDK o el navegador OBS+ para configurar SSE-KMS.
- La encriptación del lado del servidor con claves proporcionadas por el cliente (SSE-C)
OBS utiliza las claves y los valores MD5 proporcionados por los clientes para el cifrado del lado del servidor.
Puede usar API o SDK para configurar SSE-C.

Información de referencia

En el modo SSE-KMS, KMS utiliza un módulo de seguridad de hardware (HSM) para proteger la seguridad de las claves, lo que le ayuda a crear y controlar fácilmente las claves de encriptación. Las claves no se muestran en texto sin formato fuera de los HSM, lo que impide la divulgación de claves. Todas las operaciones realizadas en las claves se controlan mediante permisos de acceso y se registran, cumpliendo con los requisitos de cumplimiento normativo.

¿Cómo usar esta función?

Puede utilizar la consola OBS, API, SDK o el navegador OBS+ para establecer el cifrado del servidor.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Carga de un archivo con cifrado del lado del servidor Encriptación predeterminada de un bucket |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Encriptación del lado del servidor (SSE-KMS) Encriptación del lado del servidor (SSE-C) Configuración de la encriptación de bucket |
| OBS Browser+ | - |

14 Alojamiento de sitio web estático

Consideraciones

Por motivos de seguridad y cumplimiento, utilizar el alojamiento de sitios web estático a través del nombre de dominio OBS predeterminado. (**nombre de dominio de bucket o nombre de dominio de sitio web estático**) será prohibido por OBS. Cuando utiliza dicho nombre de dominio para acceder a páginas web a través de un navegador, no se mostrará ningún contenido, en su lugar, el contenido se descargará como un archivo adjunto.

Esta prohibición entrará en vigor en diferentes regiones en los dos momentos siguientes:

January 1, 2022: CN North-Beijing1, CN North-Beijing4, CN East-Shanghai1, CN East-Shanghai2, and CN South-Guangzhou

March 25, 2022: CN-Hong Kong, AP-Bangkok, AP-Singapore, AF-Johannesburg, LA-Mexico City1, LA-Mexico City2, LA-Sao Paulo1, and LA-Santiago

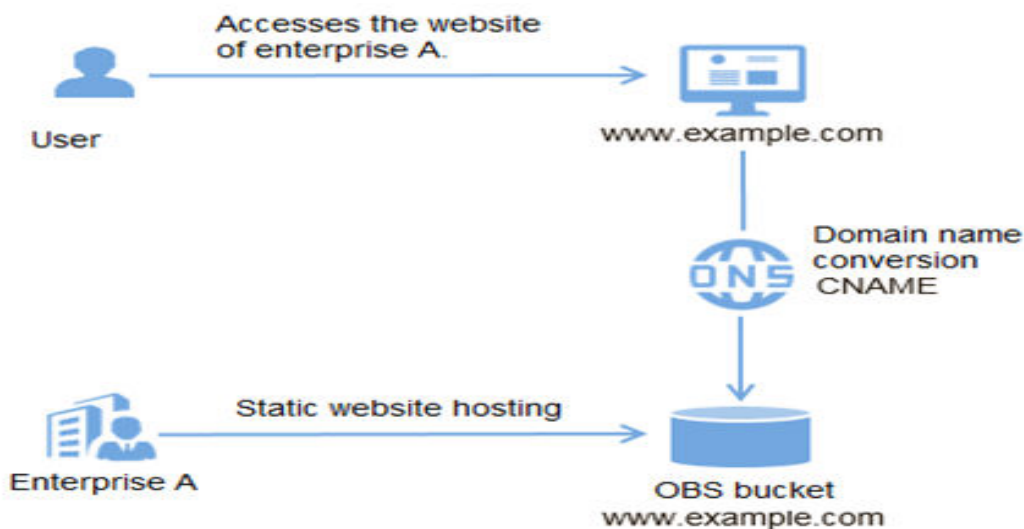
Todavía puede utilizar el alojamiento de sitios web estático a través de un nombre de dominio definido por el usuario. De esta manera, el contenido aún se puede previsualizar. Para obtener más información, consulte [¿Cómo previsualizo objetos en OBS a través de un explorador?](#)

Escenario de la aplicación

Puede usar OBS para alojar su sitio web estático, para el que también puede configurar un documento de índice, un documento de error y una dirección de página. Puede cargar los archivos de contenido del sitio web estático en el bucket en OBS y configurar un permiso de lectura para usuarios anónimos para estos archivos y, a continuación, configurar el modo de alojamiento de sitio web estático para que el bucket aloje los sitios web estáticos en OBS.

Los sitios web estáticos contienen páginas estáticas y algunos scripts que pueden ejecutarse en clientes, como JavaScript y Flash. A diferencia de los sitios web estáticos, los sitios dinámicos dependen de los servidores para procesar los scripts, entre ellos los scripts PHP, JSP y ASP.NET. OBS no admite scripts que se ejecutan en servidores.

Figura 14-1 Alojamiento de sitio web estático



Después de habilitar el alojamiento de sitios web estático, puede acceder a los objetos de un bucket utilizando cualquiera de los siguientes métodos:

- **Gestión de recursos:** puede acceder a los objetos de un bucket a través de un nombre de dominio común. Puede usar API o SDK y nombres de dominio OBS predeterminados con endpoints contenidos para realizar operaciones comunes en los bucket y objetos, como cargar, descargar, eliminar y enumerar.
- **Sitio web estático:** puede acceder a objetos en un bucket mediante un nombre de dominio específico. En este modo, la visualización de la página de índice, páginas de error y páginas solicitadas se implementa de acuerdo con las reglas de alojamiento de nombres de dominio estáticos configuradas.

La configuración del alojamiento web estático tiene efecto en dos minutos. Una vez que surta efecto, puede acceder a recursos estáticos utilizando los siguientes nombres de dominio de acceso:

```
https://bucketname.OBS static website hosting domain name/Object name
http://bucketname.OBS static website hosting domain name/Object name
```

NOTA

Evite usar (.) de períodos en el nombre del bucket de destino. De lo contrario, pueden producirse errores en el certificado de autenticación de cliente cuando los usuarios usan HTTPS para el acceso.

Para permitir que sus clientes accedan al contenido en el nodo del terminal del sitio web, debe hacer que todo su contenido sea público y accesible. Puede utilizar políticas de bucket o ACL en objetos para conceder permisos.

En la siguiente tabla se enumeran las diferencias entre el modo de administración de recursos y el modo de sitio web estático.

| Principal diferencia | Gestión de recursos | Sitio web estático |
|----------------------|---|--------------------------------------|
| Control de acceso | Tanto el contenido público como el privado son compatibles. | Solo se admite el contenido público. |

| Principal diferencia | Gestión de recursos | Sitio web estático |
|--|---|--|
| Procesamiento de mensajes de error | Se devuelve una respuesta de error en formato XML. | Se devuelve un documento HTML. |
| Soporte de redirección | N/A | Se admite la redirección a nivel de objeto y a nivel de depósito. |
| Solicitud soportada | Se admiten operaciones en todos los bucket y objetos. | Sólo se admiten las solicitudes GET y HEAD en objetos. |
| Respuesta a solicitudes GET y HEAD de nivel raíz de cubo | Se devuelve una lista de claves de objeto en un bucket. | Se devuelve el archivo de índice especificado en la configuración. |

Visión general de redirección

Al utilizar el alojamiento de sitios web estático, puede configurar la redirección para redirigir solicitudes específicas o todas.

Si se cambia la estructura, la dirección o la extensión del nombre de archivo de un sitio web, los usuarios no podrán acceder al sitio web utilizando la dirección anterior. (como la dirección guardada en la carpeta de favoritos) y se devolverá el mensaje de error 404. En este caso, puede configurar la redirección para que el sitio web redirija las solicitudes de acceso de los usuarios a la página especificada en lugar de devolver la página de error 404.

Las configuraciones típicas incluyen:

- Redirigir todas las solicitudes a otro sitio web.
- Redirigir solicitudes específicas basadas en reglas de redirección.

Configurar el alojamiento de sitios web estáticos

Información general

Si desea usar un bucket para alojar sitios web estáticos, agregue la configuración del sitio web al bucket. La configuración incluye la siguiente información:

Documento de índice

Cuando ingresas una URL como `http://example.com`, no estás solicitando una página específica. En este caso, el servidor web proporcionará una página predeterminada que contiene el directorio que almacena el contenido del sitio web solicitado. Esta página predeterminada se denomina documento de índice, y en la mayoría de los casos se denomina `index.html`. Cuando configura un bucket para el alojamiento de sitios web, debe especificar un documento de índice. Cuando se envía una solicitud al dominio raíz o a cualquier subcarpeta, OBS devuelve este documento de índice.

Documento de error

Si se produce un error, OBS devuelve un documento de error HTML. Para errores `4XX`, puede proporcionar su propio documento de error personalizado o proporcionar otras guías a los usuarios en este documento.

Redirección de todas las solicitudes

Si el dominio raíz es **example.com** y necesita responder a las solicitudes de **http://example.com** y **http://www.example.com**, puede crear dos los bucket llamados **example.com** y **www.example.com**. A continuación, puede conservar el contenido del sitio web en un solo bucket (como **example.com**) y configurar el otro bucket para redirigir todas las solicitudes al bucket de **example.com**.

Redirección basada en condiciones avanzadas

Puede redirigir las solicitudes basándose en el nombre o prefijo del objeto específico de la solicitud, o basándose en el código de respuesta. Por ejemplo, suponga que elimine o cambie el nombre de un objeto de un bucket. Puede agregar una regla de enrutamiento que redirija las solicitudes a otros objetos.

El formato sintáctico es el siguiente al establecer reglas de redirección de solicitudes específicas:

```
<WebsiteConfiguration>
  <IndexDocument>
    <Suffix>index.html</Suffix></IndexDocument>
  <ErrorDocument>
    <Key>SomeErrorDocument.html</Key>
  </ErrorDocument>
  <RoutingRules>
    RoutingRules.....
  </RoutingRules>
</WebsiteConfiguration>
```

El formato sintáctico de **RoutingRules** es el siguiente. El contenido entre corchetes es opcional.

```
<RoutingRules> =
  <RoutingRules>
    <RoutingRule>...</RoutingRule>
    [ <RoutingRule>...</RoutingRule> ... ]
  </RoutingRules>

<RoutingRule> =
  <RoutingRule>
    [ <Condition>...</Condition> ]
    <Redirect>...</Redirect>
  </RoutingRule>

<Condition> =
  <Condition>
    [ <KeyPrefixEquals>...</KeyPrefixEquals> ]
    [ <HttpErrorCodeReturnedEquals>...</HttpErrorCodeReturnedEquals> ]
  </Condition>

<Redirect> =
  <Redirect>
    [ <HostName>...</HostName> ]
    [ <Protocol>...</Protocol> ]
    [ <ReplaceKeyPrefixWith>...</ReplaceKeyPrefixWith> ]
    [ <ReplaceKeyWith>...</ReplaceKeyWith> ] [
    <HttpRedirectCode>...</HttpRedirectCode> ]
  </Redirect>
```

Para los elementos de solicitud necesarios para redirigir todas las solicitudes enviadas a un sitio web específico y para establecer reglas de redirección, consulte [Configuración de Alojamiento Estático de Sitios Web para un Bucket](#)

Ejemplos:

Ejemplo 1: Modificación del prefijo de nombre de objeto para la redirección

Suponga que el bucket contenga los siguientes objetos:

index.html

docs/article1.html

docs/article2.html

Usted decide cambiar el nombre de la carpeta de **docs/** a **documents/**. Después de la modificación, la solicitud de un objeto con el prefijo **/docs** debe ser redirigida a otro con **documents/**. Por ejemplo, la solicitud de **docs/article1.html** debe redirigirse a **documents/article1.html**.

En este caso, puede agregar las siguientes reglas de enrutamiento a la configuración del sitio web:

```
<RoutingRules>
  <RoutingRule>
    <Condition>
      <KeyPrefixEquals>docs/</KeyPrefixEquals>
    </Condition>
    <Redirect>
      <ReplaceKeyPrefixWith>documents/</ReplaceKeyPrefixWith>
    </Redirect>
  </RoutingRule>
</RoutingRules>
```

Ejemplo 2: Redirección de solicitudes enviadas a carpetas eliminadas a una página especificada

Supongamos que ha eliminado la carpeta **images/**, es decir, que ha eliminado todos los objetos cuyo prefijo de nombre de objeto es **images/**. Puede agregar una regla de enrutamiento que redirija las solicitudes de todos los objetos cuyo prefijo es **images/** a la página denominada **folderdeleted.html**.

```
<RoutingRules>
  <RoutingRule>
    <Condition>
      <KeyPrefixEquals>images/</KeyPrefixEquals>
    </Condition>
    <Redirect>
      <ReplaceKeyWith>folderdeleted.html</ReplaceKeyWith>
    </Redirect>
  </RoutingRule>
</RoutingRules>
```

Ejemplo 3: Redireccionamiento cuando se producen errores HTTP

Supongamos que el objeto solicitado no se encuentra y que la solicitud debe ser redirigida a **www.example.com**. Puede agregar reglas de redirección para que los visitantes del sitio puedan redirigir a **www.example.com** cuando se devuelva el código de estado HTTP 404 (no encontrado). En el ejemplo siguiente también se inserta el prefijo de nombre de objeto **report-404/** en la redirección. Por ejemplo, si solicita la página **ExamplePage.html** y causa un error HTTP 404, la solicitud será redirigida al informe de página **report-404/ExamplePage.html** en el **www.example.com**. Si no hay regla de encaminamiento y se produce el error 404 de HTTP, se devuelve el documento de error especificado en la configuración.

```
<RoutingRules>
  <RoutingRule>
    <Condition>
      <HttpErrorCodeReturnedEquals>404</HttpErrorCodeReturnedEquals >
    </Condition>
    <Redirect>
```



```
<HostName>www.example.com</HostName>
<ReplaceKeyPrefixWith>report-404</ReplaceKeyPrefixWith>
</Redirect>
</RoutingRule>
</RoutingRules>
```

Soporte de documentos de índice

Un documento de índice es una página web que se devuelve cuando se envía una solicitud a la raíz de un sitio o a cualquier subcarpeta. Por ejemplo, si el usuario introduce **http://www.example.com** en el navegador, el usuario no solicita ninguna página específica. En este caso, OBS proporciona documentos de índice, que a veces se denominan páginas predeterminadas.

Cuando configure su bucket como sitio web, proporcione el nombre del documento de índice. Debe cargar el objeto con este nombre y configurarlo para que sea público y accesible.

La barra diagonal final en la URL raíz es opcional. Por ejemplo, si configura un sitio web **index.html** como un documento de índice, **index.html** se devolverá a cualquiera de las siguientes URL:

```
http://bucketname.obs-website.example.com/
http://bucketname.obs-website.example.com
```

En OBS, los objetos se almacenan horizontalmente en los bucket. No proporciona ninguna organización jerárquica como sistema de archivos en un equipo. Puede crear una jerarquía lógica utilizando el nombre del objeto que representa la estructura de carpetas. Por ejemplo, considere un bucket con tres objetos y los siguientes nombres de objeto.

- sample1.jpg
- photos/2006/Jan/sample2.jpg
- photos/2006/Feb/sample3.jpg

Aunque no se almacenan como ninguna organización jerárquica física, puede inferir la siguiente estructura lógica de carpetas a partir del nombre del objeto.

- El objeto **sample1.jpg** se encuentra en el nivel raíz del bucket.
- El objeto **sample2.jpg** se encuentra en la subcarpeta **photos/2006/Jan**.
- El objeto **sample3.jpg** se encuentra en la subcarpeta **photos/2006/Feb**.

Soporte de documento de error personalizado

Tabla 14-1 enumera el subconjunto del código de respuesta HTTP devuelto por OBS cuando se produce un error.

Tabla 14-1 Lista de códigos de error de OBS

| Código de error HTTP | Descripción |
|-----------------------|--|
| 301 Moved Permanently | Cuando un usuario envía una solicitud a un endpoint de OBS, se devuelve una respuesta 301 Moved Permanently . |

| Código de error HTTP | Descripción |
|--------------------------|--|
| 302 Found | Cuando OBS recibe una solicitud de clave <i>x</i> que no termina con una barra diagonal, busca el objeto cuyo nombre de clave es <i>x</i> . Si no se encuentra el objeto, OBS determina que la solicitud se envía para la subcarpeta <i>x</i> . A continuación, OBS redirige la solicitud añadiendo una barra al final de la solicitud, y devuelve 302 Found . |
| 304 Not Modified | Los usuarios de OBS solicitan los encabezados If-Modified-Since y If-Unmodified-Since para determinar si el objeto solicitado es el mismo que la copia almacenada en caché en el cliente. Si los objetos son los mismos, el nodo terminal de sitio web devuelve una respuesta 304 Not Modified . |
| 403 Forbidden | Cuando la solicitud de usuario se convierte en un objeto que no puede leerse públicamente, la respuesta del nodo terminal de sitio web incluye 403 Forbidden . El propietario del objeto debe usar políticas de bucket o ACL para que el objeto sea público y accesible. |
| 404 Not Found | La respuesta del nodo terminal del sitio web contiene 404 Not Found debido a las siguientes razones: <ul style="list-style-type: none"> ● OBS determina que se hace referencia a una clave de objeto no válida en la dirección URL del sitio web. ● La OBS determina que la solicitud es para un documento de índice que no existe. ● El bucket especificado en la URL no existe. ● El bucket especificado en la URL existe pero no está configurado como sitio web. Puede crear documentos personalizados devueltos para 404 Not Found. Asegúrese de que el documento se ha cargado en el bucket configurado como sitio web y de que la configuración de alojamiento del sitio web se ha establecido para usar el documento. |
| 500 Service Error | Cuando se produce un error de servidor interno, la respuesta del nodo de terminal de sitio web contiene 500 Service Error . |

| Código de error HTTP | Descripción |
|-------------------------|---|
| 503 Service Unavailable | Cuando OBS determina que necesita reducir la frecuencia de solicitud, la respuesta del nodo terminal de sitio web contiene 503 Service Unavailable . |

Permisos requeridos para el acceso al sitio web

Cuando configuras un bucket como sitio web, debes hacer público y accesible el objeto que se va a proporcionar. Para realizar esta operación, debe escribir una política de bucket que conceda permisos de GetObject a todos. Si el objeto solicitado por el usuario no existe en el nodo terminal del sitio web, OBS devuelve el código de respuesta HTTP **404 Not Found**. Si el objeto existe pero no concede permiso de lectura para el objeto, el nodo de terminal del sitio web devuelve el código de respuesta HTTP **403 Access Denied**. Puede utilizar este código de respuesta para inferir si existe un objeto concreto. Si no necesita esta función, no habilite el soporte del sitio web del bucket.

NOTA

En el escenario de alojamiento de sitios web estáticos, los usuarios anónimos deben tener acceso al archivo de sitio web estático alojado. Durante su acceso al archivo alojado, se incurrirán en tarifas por el tráfico de Internet saliente y las solicitudes.

La política de bucket de ejemplo siguiente otorga a cada persona el permiso para tener acceso a los objetos de una carpeta especificada. Para obtener más información acerca de las políticas de bucket, consulte [Política de bucket](#).

```
{
  "Statement": [{
    "Sid": "PublicReadGetObject",
    "Effect": "Allow",
    "Principal": {"ID": "*"},
    "Action": ["GetObject"],
    "Resource": ["example-bucket/*" ]
  }
]
```

AVISO

Una política de bucket solo se aplica a los objetos propiedad de un propietario de bucket. Si el bucket contiene objetos que no son propiedad del propietario del bucket, utilice las ACL de objetos para conceder el permiso de lectura pública a los objetos.

Puede usar políticas de bucket o ACL de objetos para conceder permisos de lectura pública a los objetos. Para utilizar las ACL para hacer que los objetos sean públicos y accesibles, puede conceder el permiso de lectura a todos, como se muestra en los siguientes elementos de autorización. Puede agregar el elemento de autorización a las ACL de objeto. Para obtener más información acerca de cómo gestionar ACL, vea [ACL](#).

```
<Grant>
  <Grantee>
    <Canned>Everyone</Canned>
  </Grantee>
```

```
<Permission>READ</Permission>
</Grant>
```

Funciones relacionadas

| Función | Descripción de la relación | Referencia |
|--|--|---|
| CORS | De forma predeterminada, los sitios web estáticos alojados en un bucket OBS solo pueden responder a las solicitudes de acceso de sitios web en el mismo dominio. Puede configurar el uso compartido de recursos de origen cruzado (CORS) para el bucket de modo que se pueda acceder a los sitios web estáticos mediante solicitudes entre dominios. | CORS |
| Vinculación de nombre de dominio definido por el usuario | OBS le permite usar un nombre de dominio definido por el usuario para acceder a sitios web estáticos alojados en un bucket OBS. Puede utilizar el nombre de dominio original para acceder al sitio web sin modificar el código del sitio web. | Vinculación de nombre de dominio definido por el usuario Uso de un nombre de dominio definido por el usuario para alojar un sitio web estático |
| Validación de URL | OBS utiliza la validación de URL para evitar que sus sitios web alojados en un bucket OBS sean robados. OBS verifica las URL basadas en el campo de referencia en el encabezado HTTP. | Validación de URL |

¿Cómo usar esta función?

Puede utilizar la consola OBS, las API o los SDK para configurar CORS.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Configurar el alojamiento de sitios web estáticos |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Configuración de Alojamiento Estático de Sitios Web para un Bucket |

15 CORS

El uso compartido de recursos de origen cruzado (Cross-origin resource sharing, CORS) es un mecanismo estándar de navegador proporcionado por el World Wide Web Consortium (W3C). Define los métodos de interacción entre las aplicaciones web del lado del cliente en un origen y los recursos en otro. Para las solicitudes generales de páginas web, los scripts de sitios web y los contenidos de un origen no pueden interactuar con los de otro debido a las políticas del mismo origen (Same Origin Policies, SOP).

OBS soporta las reglas CORS y permite el acceso a los recursos en OBS a través de los orígenes. La configuración de CORS entra en vigor en dos minutos.

OBS admite **Alojamiento de sitio web estático**. Los sitios web estáticos almacenados en OBS pueden responder a solicitudes de sitios web de otro origen solo cuando CORS está configurado para el bucket.

AVISO

De forma predeterminada, el sistema OBS está configurado para admitir el acceso entre dominios mediante el nombre de dominio raíz. Esto permite el acceso desde todos los dominios, y es probable que los clientes sean atacados.

Para solucionar este problema, puede crear un archivo **crossdomain.xml** con reglas específicas en el bucket para cada cliente, y agregar **Security.loadPolicyFile("https://bucket.obs.ap-southeast-1.myhuaweicloud.com/crossdomain.xml")** en el código flash del archivo para evitar ataques. Es necesario reemplazar **bucket.obs.ap-southeast-1.myhuaweicloud.com** con el nombre de dominio de acceso a bucket actual que está utilizando.

Información de referencia

Cross-domain se refiere al acceso entre diferentes dominios.

Restringir el acceso entre dominios es una política de navegador por motivos de seguridad, es decir, la política del mismo origen. Debido a esta política JavaScript del mismo origen, las JavaScript bajo el dominio A no pueden operar objetos bajo el dominio B o C.

El mismo protocolo, nombre de dominio (o dirección IP) y puerto se consideran como el mismo dominio. Si los protocolos, nombres de dominio y puertos (si se especifican) de las dos páginas web son iguales, las dos páginas web tienen el mismo origen. Para entender

mejor la política del mismo origen, puede ver el análisis sobre el acceso a la dirección de ejemplo <https://support.huaweicloud.com/dir/test.html> en [Tabla 15-1](#).

Tabla 15-1 Análisis de ejemplos

| URL | Resultado del acceso | Motivo |
|---|----------------------|---|
| https://support.huaweicloud.com/dir/other.html | Exitoso | El mismo protocolo, nombre de dominio y puerto |
| https://support.huaweicloud.com/dir/inner/other.html | Exitoso | El mismo protocolo, nombre de dominio y puerto |
| http://support.huaweicloud.com/dir/test.html | Error | El mismo nombre de dominio y puerto, pero protocolos diferentes |
| https://support.huaweicloud.com:81/dir/test.html | Error | El mismo protocolo y nombre de dominio, pero puertos diferentes |
| https://help.huaweicloud.com/dir/test.html | Error | El mismo protocolo y puerto, pero nombres de dominio diferentes |

Escenarios de la aplicación

Los escenarios de aplicación típicos de CORS son los siguientes:

- Permite utilizar JavaScript y HTML5 para establecer aplicaciones web que pueden acceder directamente a recursos en OBS. No se requieren servidores proxy para la transferencia.
- Habilita la función de arrastre de HTML5 para ser utilizado para subir archivos a OBS (con el progreso de la carga mostrado) o actualizar contenidos OBS usando aplicaciones web.
- Las páginas web externas, las hojas de estilo y las aplicaciones HTML5 alojadas en diferentes orígenes pueden acceder a fuentes web o imágenes almacenadas en OBS, implementando el uso compartido de recursos.

¿Cómo usar esta función?

Puede configurar CORS mediante la consola OBS, las API o los SDK.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Configuración de CORS |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |

| Herramienta | Referencia |
|-------------|--|
| API | Configuración de Bucket CORS |

16 Vinculación de nombre de dominio definido por el usuario

Escenario de la aplicación

Si los usuarios desean migrar archivos de un sitio web a OBS y aún así desean usar su propio enlace de sitio web para acceder a los archivos almacenados en OBS, pueden vincular un nombre de dominio definido por el usuario a un bucket OBS.

Por ejemplo, si el nombre de dominio de su sitio web es `www.example.com` y el archivo de su sitio web es `abc.html`, entonces la URL para acceder a este archivo es `http://www.example.com/abc.html`. El procedimiento de configuración es el siguiente:

1. Cree un bucket en OBS y suba el archivo del sitio web `abc.html` al bucket.
2. En OBS Console, vincule el nombre de dominio de tu sitio web `www.example.com` al bucket creado.
3. En el servidor DNS, agregue **una regla CNAME** y asigne `www.example.com` al nombre de dominio del bucket.
4. Después de que la solicitud de `http://www.example.com/abc.html` llega a OBS, OBS encuentra la asignación entre el `www.example.com` y el nombre de dominio del bucket, y lo redirige al archivo `abc.html` almacenado en el bucket. Es decir, OBS redirige la solicitud de acceso a `http://www.example.com/abc.html` a nombre de dominio `http://bucket/abc.html`.

Restricciones

- Solo los bucket de la versión 3.0 o posterior admiten la vinculación de nombres de dominio definidos por el usuario. Para comprobar la versión del bucket, vaya a la página **Overview** del bucket en OBS Console. A continuación, puede ver la versión del bucket en el área **Basic Information**.
- De forma predeterminada, se puede enlazar un máximo de 20 nombres de dominio definidos por el usuario a un bucket. In some regions (for example, CN South-Guangzhou), a bucket can have up to 30 user-defined domain names bound. For the maximum number allowed in each region, see the requirements on OBS Console.
- Actualmente, los nombres de dominio de usuario enlazados a OBS solo permiten solicitudes de acceso a través de HTTP.

Si desea utilizar un nombre de dominio de usuario para acceder a OBS a través de HTTPS, debe habilitar CDN para administrar certificados HTTPS.

Para obtener más información sobre cómo administrar certificados HTTPS en la consola de administración de CDN, consulte [Configuración HTTPS](#).

- Un nombre de dominio definido por el usuario se puede vincular a un solo bucket.
- Actualmente, el sufijo de un nombre de dominio definido por el usuario puede contener de 2 a 6 letras mayúsculas y minúsculas.
- Como lo requiere el MIIT, debe completar el archivo ICP, si el bucket al que está enlazado su nombre de dominio definido por el usuario está en cualquiera de las siguientes regiones:
CN North-Beijing1, CN North-Beijing4, CN East-Shanghai1, CN East-Shanghai2, and CN South-Guangzhou

NOTA

Si se utiliza la aceleración de CDN, para evitar que los objetos en los bucket de OBS se descarguen directamente al acceder, debe realizar las operaciones necesarias después de que se haya asociado un nombre de dominio definido por el usuario. Para obtener más información, consulte [Con la aceleración de CDN activada, ¿Por qué se descargan directamente los objetos de mi cubo OBS cuando accedo a ellos?](#)

¿Cómo usar esta función?

Puede configurar el enlace de nombres de dominio definidos por el usuario mediante la consola OBS, las API o los SDK.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Configuración del enlace de nombres de dominio definidos por el usuario |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Configuración de un nombre de dominio personalizado para un bucket |

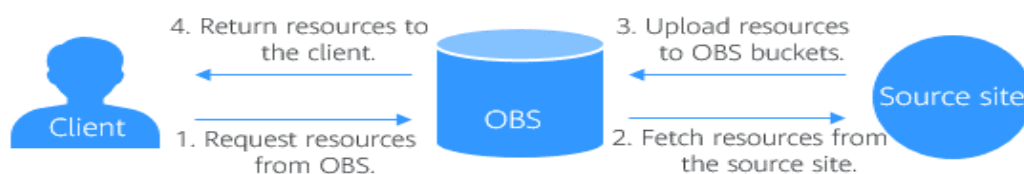
17 Vuelta a la fuente

Cuando un cliente no accede a los datos solicitados en OBS, se devuelve el error 404. Sin embargo, OBS proporciona la función de retorno al origen para ayudarle a obtener los datos solicitados de su sitio de origen si no se encuentran en OBS.

Regreso a la fuente por Mirroring

Si se configura una regla de retorno a origen de mirroring para un bucket OBS y los datos solicitados no se encuentran en el bucket, el sistema recuperará los datos, cuando la regla de retorno a origen se aplique a los datos, desde el servidor de origen, los cargará al bucket, y luego devolverlo al cliente solicitante. Este proceso no interrumpe los servicios. Por lo tanto, puede utilizar esta función para migrar sin problemas datos desde el servidor de origen a OBS, o migrar servicios a OBS sin ser detectados por los usuarios, a bajo costo. [Figura 17-1](#) ilustra el proceso de duplicación de regreso al origen.

Figura 17-1 Regreso a la fuente por Mirroring



Restricciones

- Esta función está actualmente disponible solo en las regiones CN North-Beijing4, CN East-Shanghai1, CN East-Shanghai2, CN-Hong Kong y AP-Singapore.
- Los usuarios anónimos no pueden configurar la duplicación de reglas de origen para un bucket.
- Los sistemas de archivos paralelos no admiten la duplicación de reglas de regreso al origen.
- Una regla de duplicación de back-to-source no es compatible con la función de alojamiento de sitios web estáticos. Específicamente, si se produce un error 404 cuando se descargan objetos desde un dominio de sitio web estático alojado en OBS, no activa el proceso de duplicación de regreso al origen.

- El bucket, en el que está configurada una regla de retorno al origen, no se puede especificar como el sitio de origen.
- No se puede especificar un bucket privado como sitio de origen en una regla de copia de seguridad de origen. Cambie la política de bucket a Lectura pública si necesita usar el bucket como sitio de origen.
- El servidor de origen no puede transferir datos en el modo **Transfer-Encoding: chunked**. Es decir, la respuesta a la solicitud de descarga de un objeto desde el servidor de origen debe contener el encabezado **Content-Length** para especificar el tamaño del objeto de origen.
- Un objeto no puede coincidir con dos reglas diferentes de duplicación de regreso al origen.
- Cuando se configuran las reglas de espejado y redirección de retorno a origen y los datos solicitados coinciden con ambas reglas, la OBS procede primero de acuerdo con la regla de espejado. Si los datos solicitados no se encuentran en el sitio de origen, entonces OBS procesa con la regla de redirección.
- Solo los bucket de la versión 3.0 o posterior admiten la función de duplicación de regreso al origen.
- Una regla de vuelta al origen de mirroring entra en vigor cinco minutos después de cualquier cambio en la regla.
- Se puede configurar un máximo de 10 reglas de vuelta a origen de mirroring para un bucket.
- La función de vuelta a origen de mirroring se ofrece de forma gratuita.

Creación de una regla de vuelta a origen de mirroring

Puedes crear una regla de vuelta a origen de mirroring en la consola de OBS o llamando a las API.

| Herramienta | Reference |
|----------------|--|
| Consola de OBS | Creación de una regla de vuelta a la fuente |
| API | Configuración de las reglas de vuelta a la fuente de mirroring |

18 Validación de URL

Escenario de la aplicación

Algunos sitios fraudulentos pueden robar enlaces de otros sitios para enriquecer su contenido sin ningún costo. El robo de enlaces perjudica los intereses de los sitios web originales y también es una tensión en sus servidores. OBS proporciona validación de URL para resolver este problema.

En HTTP, el campo **Referer** permite que los sitios web y los servidores web identifiquen desde dónde los visitan las personas. La validación de URL de OBS utiliza este campo Referer. La idea es que una vez que encuentres que una solicitud a tu recurso no se origina desde una fuente autorizada (por ejemplo, una URL), puedes bloquear o redirigir la solicitud a una página web específica. De esta manera, OBS evita el acceso no autorizado a los datos almacenados en los bucket.

Dicha autorización se controla mediante una lista blanca y una lista negra.

Reglas de configuración de referencia:

- La longitud de una lista blanca o negra no puede exceder los caracteres de 1,024.
- Formato de referencia:
 - Puede introducir varios referentes, cada uno en una línea.
 - El parámetro referer admite asteriscos (*) y signos de interrogación (?). Un asterisco funciona como un comodín que puede reemplazar cero o varios caracteres, y un signo de interrogación (?) puede reemplazar a un solo carácter.
 - Si el campo de encabezado de referencia contiene http o https durante la descarga, el referente debe contener http o https.
- Si los Referentes de la lista blanca se dejan en vacío pero los Referentes de la lista negra no lo hacen, todos los sitios web, excepto los especificados en la lista negra, pueden acceder a los datos del bucket de destino.
- Si los Referentes de la Lista Blanca no se dejan en vacío, solo los sitios web especificados en la lista blanca pueden acceder al bucket de destino sin importar si los Referentes de la Lista Negra se dejan en blanco o no.

NOTA

Si los Referentes de la lista blanca se configuran de la misma manera que los Referentes de la lista negra, la lista negra tiene efecto. Por ejemplo, si tanto los Referentes de la lista blanca como los Referentes de la lista negra están configurados en <https://www.example.com>, se bloquearán las solicitudes de acceso desde esta dirección.

- Si tanto los Referentes de la lista blanca como los Referentes de la lista negra se dejan en blanco, todos los sitios web pueden acceder a los datos del bucket de destino de forma predeterminada.
- Antes de determinar si un usuario tiene los cuatro tipos de permisos (**Lectura**, **Escritura**, **Vista de ACL** y **Edición de ACL**) para un bucket u objetos en el bucket, compruebe si este usuario cumple con los principios de validación de URL del campo **Referer**.

Métodos de configuración de listas blancas y negras:

- Configuración de la lista blanca

Al establecer una lista blanca, puede permitir solicitudes de los sitios web en la lista blanca, pero denegar las de los sitios web que no están en la lista blanca.

Para las solicitudes que se inicializan desde el cuadro de dirección de un navegador, puede agregar el campo **\${null}** a **Referer** of **Condition** para especificar si se permiten las solicitudes HTTP con un **referer** en blanco.

Para configurar una lista blanca, consulte la siguiente política:

```
"Statement": [
  {
    "Sid": "1",
    "Effect": "Allow",
    "Principal": {"ID": ["*"]},
    "Action": "*",
    "Resource": ["bucket/*"],
  },
  {
    "Sid": "2",
    "Effect": "Deny",
    "Principal": {"ID": ["*"]},
    "Action": "*",
    "Resource": ["bucket/*"],
    "Condition": {
      "StringNotEquals": {
        "Referer": ["http://www.example01.com", "${null}"]
      }
    }
  }
]
```

Si establece una lista blanca de esta manera, solo las solicitudes cuyo **referer** está establecido en `www.example01.com` o dejado en blanco pueden trabajar en los recursos de los bucket.

- Configuración de la lista negra

Para configurar una lista negra, consulte la siguiente política:

```
"Statement": [
  {
    "Sid": "1",
    "Effect": "Deny",
    "Principal": {"ID": ["*"]},
    "Action": "*",
    "Resource": ["bucket/*"],
    "Condition": {
      "StringEquals": {
        "Referer": ["http://www.example01.com", "http://www.example02.com"]
      }
    }
  }
]
```

Si establece una lista negra de esta manera, las solicitudes cuyo **referer** está establecido en `www.example01.com` o `www.example02.com` no pueden trabajar en recursos en los bucket.

¿Cómo usar esta función?

Puede usar la consola OBS o las API para configurar la validación de URL.

| Herramienta | Referencia |
|-------------|--|
| Consola | Configuración de la validación de URL |
| API | Configuración de una lista blanca de validación de URL |

19 Etiquetas

Escenario de la aplicación

Si su sistema de servicios utiliza varios servicios en la nube de Huawei Cloud, puede establecer etiquetas para identificar mejor las instancias de recursos para diferentes servicios en la nube. Para OBS, una instancia de recurso es un bucket. Estas etiquetas se incluirán en los registros de datos de cobro (CDR) generados para esos servicios y sus instancias. Si el sistema de servicio está compuesto por varias aplicaciones, establecer la misma etiqueta para todas las instancias de recursos utilizadas para cada aplicación le ayuda a analizar fácilmente el uso y los costos de los recursos.

En OBS, las etiquetas se usan para identificar y clasificar los bucket. Si agrega etiquetas a un bucket, los CDR generados para las solicitudes enviadas a este bucket incluirán estas etiquetas, por lo que puede usar las etiquetas para clasificar los CDR para un análisis detallado de costos. Por ejemplo, si tiene una aplicación que carga datos en tiempo de ejecución en un bucket, puede etiquetar el bucket con el nombre de la aplicación. De esta manera, los costes de la aplicación pueden analizarse usando etiquetas en CDR.

Una etiqueta se describe mediante un par clave-valor. Un bucket puede tener un máximo de 10 etiquetas. Cada etiqueta tiene solo una clave y un valor. La clave y el valor pueden existir en cualquier orden en una etiqueta. Cada clave es única entre todas las etiquetas de un bucket, mientras que los valores pueden ser repetidos o estar vacíos. Se tarda aproximadamente tres minutos para que la etiqueta surta efecto.

¿Cómo usar esta función?

Puede configurar etiquetas mediante la consola OBS, las API o los SDK.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Configuración de etiquetas |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Configuración de etiquetas para un bucket |

20 Inventario de buckets

Escenario de la aplicación

Un inventario de buckets puede enumerar objetos de un bucket, guardar la información de objeto relacionada en archivos CSV y entregar los archivos CSV al bucket especificado para almacenar archivos de inventario de buckets. De esta manera, puede gestionar fácilmente objetos en un bucket. Un bucket de origen también puede ser el bucket de destino.

- Un archivo de inventario de bucket puede contener la siguiente información relacionada con el objeto: versiones, tamaños, clases de almacenamiento, etiquetas, estados de cifrado y tiempo de última modificación.
- Puede cifrar los archivos de inventario de buckets en el modo SSE-KMS.
- Puede establecer la frecuencia (diaria o semanal) para generar archivos de inventario de buckets.
- También puede especificar un bucket para almacenar los archivos de inventario de bucket generados.

Restricciones

- Los inventarios de bucket solo se pueden generar para cubetas OBS 3.0, pero se pueden almacenar en OBS 3.0 u OBS 2.0.
- Un bucket puede tener un máximo de 10 reglas de inventario.
- El bucket de origen (para el que se configura una regla de inventario de bucket) y el bucket de destino (donde se almacenan los archivos de inventario generados) deben pertenecer a la misma cuenta.
- El bucket de origen y el bucket de destino deben estar en la misma región.
- Los archivos de inventario deben estar en formato CSV.
- OBS puede generar archivos de inventario para todos los objetos de un bucket o un grupo de objetos cuyos nombres comienzan con el mismo prefijo.
- Si un bucket tiene varias reglas de inventario, estas reglas no deben solaparse.
 - Si un bucket ya tiene una regla de inventario para todo el bucket, no se pueden crear nuevas reglas de inventario que filtren objetos por prefijos. Si necesita una regla de inventario que cubra sólo un subconjunto de objetos del bucket, elimine primero la regla de inventario configurada para todo el bucket.
 - Si ya existe una regla de inventario que filtra objetos por un prefijo especificado, no puede crear una regla de inventario para todo el bucket. Para crear una regla de

inventario para todo el bucket, asegúrese de que el bucket no tiene otras reglas de inventario que filtren los objetos por prefijos especificados.

- Si un bucket ya tiene una regla de inventario que filtra objetos por el prefijo de nombre de objeto **ab**, el filtro de una nueva regla de inventario no puede comenzar con **a** o **abc**. O bien, puede eliminar la regla de inventario existente y crear una nueva que filtre los objetos según sus necesidades.
- Los archivos de inventario de cucharón solo se pueden cifrar en el modo SSE-KMS.
- La función de inventario de cubeta se ofrece de forma gratuita, pero los archivos de inventario se facturan por el espacio de almacenamiento que utilizan.
- No se puede habilitar el cifrado predeterminado para el bucket de destino configurado para almacenar archivos de inventario.
- Los archivos de inventario son entregados al bucket de destino por un usuario del sistema OBS. Por lo tanto, debe autorizar al usuario del sistema el permiso para escribir el bucket de destino.

¿Cómo se configura un inventario de bucket?

Antes de la configuración, debe comprender brevemente qué es un bucket de origen o un bucket de destino.

- Bucket de origen: Un bucket de origen es el bucket para el que se configura un inventario. El inventario muestra los objetos almacenados en el bucket de origen.
- Bucket de destino: Un bucket de destino es donde se almacenan los archivos de inventario generados. Un bucket de origen también puede ser el bucket de destino. Puede especificar un prefijo de nombre para un inventario. A continuación, los archivos de inventario generados serán nombrados con el prefijo y guardados en el directorio con el prefijo. Si no especifica ningún prefijo de nombre para el inventario, los archivos de inventario generados se almacenan en el directorio raíz del bucket.
 - Restricciones en el bucket de destino
 - El bucket de destino y el bucket de origen deben pertenecer al mismo inquilino.
 - El bucket de destino y el bucket de origen deben estar en la misma región.
 - La política del bucket de destino debe conceder a los usuarios del sistema OBS permiso para escribir objetos en el bucket. Para obtener detalles sobre cómo autorizar dicho permiso, consulte [1](#).
 - El bucket de destino contiene los archivos siguientes:
 - Una lista de archivos de inventario
 - El archivo **Manifest**, que contiene la lista de todos los archivos de inventario bajo una determinada configuración de inventario. Para obtener más información sobre el archivo Manifest, consulte [Archivo de manifiesto](#).

Configuración de un inventario de bucket

Puedes usar OBS Console o llamar a la API para configurar un inventario de buckets. Si configura un inventario de bucket en OBS Console, se genera automáticamente una política de bucket con la configuración de permisos requerida para el bucket de destino. Si llama a la API para configurar el inventario del bucket, debe configurar manualmente la política del bucket para el bucket de destino.

1. Agregue una política de bucket para el bucket de destino.

Se debe configurar una política de bucket para el bucket de destino, para conceder a los usuarios del sistema OBS el permiso para escribir objetos en el bucket de destino. El formato de la política de bucket es el siguiente. Reemplace **destbucket** por el nombre real del bucket de destino.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Sid": "1",
      "Principal": {"Service": "obs"},
      "Resource": ["destbucket/*"],
      "Action": ["PutObject"]
    }
  ]
}
```

2. Configura el inventario de bucket.

Proporcionamos múltiples herramientas para configurar un inventario de cubetas. Para más detalles, consulte [¿Cómo usar esta función?](#)

Contenido de un archivo de inventario

El contenido de un archivo de inventario se puede configurar al crear el inventario. Para obtener más información sobre todos los campos posibles, consulte [Tabla 20-1](#).

Tabla 20-1 Metadatos de objeto enumerados en un archivo de inventario

| Metadato | Descripción |
|------------------|--|
| Bucket | Nombre del bucket de origen |
| Key | El nombre de un objeto. Cada objeto de un bucket tiene una clave única. (Los nombres de objeto en el archivo de inventario están codificados por URL usando el conjunto de caracteres UTF-8 y solo se pueden usar después de decodificar.) |
| VersionId | ID de versión de un objeto. Si el valor de IncludedObjectVersions en la configuración de inventario es Current , este campo no se incluye en el archivo de inventario. |
| IsLatest | Si la versión del objeto es la última, este parámetro es True . Si el valor de IncludedObjectVersions en la configuración de inventario es Current , este campo no se incluye en el archivo de inventario. |
| IsDeleteMarker | Cuando el control de versiones está habilitado para el bucket de origen, si se elimina un objeto, se genera un nuevo metadatos de objeto para el objeto y el IsDeleteMarker de los metadatos se establece en true . Si el valor de IncludedObjectVersions en la configuración de inventario es Current , este campo no se incluye en el archivo de inventario. |
| Size | Tamaño del objeto, en bytes. |
| LastModifiedDate | Fecha de creación del objeto o fecha de la última modificación |

| Metadato | Descripción |
|---------------------|---|
| Etag | Digestión hexadecimal del objeto MD5. ETag es el identificador único del contenido del objeto. Se puede utilizar para identificar si se ha cambiado el contenido del objeto. Por ejemplo, si el valor ETag es A cuando se carga un objeto y el valor ETag ha cambiado a B cuando se descarga el objeto, indica que se cambia el contenido del objeto. |
| StorageClass | Clase de almacenamiento de un objeto |
| IsMultipartUploaded | Indica si se carga un objeto en el modo de varias partes. |
| ReplicationStatus | Estado de replicación entre regiones de un objeto |
| EncryptionStatus | Estado de cifrado de un objeto |

Nombre de archivo de inventario

El nombre de un archivo de inventario tiene el siguiente formato:

```
destinationPrefix/sourceBucketName/inventoryId/yyyy-MM-dd'T'HH-mm'Z'/files/  

    UUID_index.csv
```

- **destinationPrefix**: el prefijo de nombre de archivo de inventario configurado al crear la regla de inventario. Los archivos de inventario generados bajo la regla reciben el nombre del prefijo, lo que puede facilitar la clasificación de los archivos de inventario. Si no se especifica ningún prefijo, el prefijo predeterminado es "BucketInventory".
- **sourceBucketName**: Nombre del bucket de origen para el que está configurado un inventario. Este campo se puede utilizar para diferenciar archivos de inventario de diferentes los bucket de origen, si esos archivos de inventario se guardan en el mismo bucket de destino.
- **inventoryId**: si un bucket de origen tiene varias reglas de inventario cuyos archivos de inventario se guardan en el mismo bucket de destino, este campo se puede utilizar para identificar diferentes reglas de inventario.
- **yyyy-MM-dd'T'HH-mm'Z'**: Hora y fecha de inicio para escanear el bucket de destino cuando se genera un archivo de inventario. Es posible que los objetos cargados en el bucket de origen después de este tiempo no se muestren en el archivo de inventario.
- **UUID_index.csv**: uno de los archivos de inventario

Archivo de manifiesto

Si hay un gran número de objetos en un bucket, se pueden generar múltiples archivos de inventario para una única configuración de inventario. Se necesita algún tiempo para generar estos archivos. Por ejemplo, si 200,000 hay objetos en un bucket, se tarda aproximadamente 1.5 minutos en generar todos los archivos de inventario. Una o dos horas después de que se hayan generado todos los archivos de inventario, se generará un archivo **manifest.json**. El archivo **manifest.json** contiene información acerca de todos los archivos de inventario generados esta vez, incluyendo:

- **sourceBucket**: nombre del bucket fuente
- **destinationBucket**: nombre del bucket de destino

- **version**: versión del inventario
- **fileFormat**: formato de archivos de inventario
- **fileSchema**: campos de metadatos de objetos contenidos en los archivos de inventario
- **files**: lista de todos los archivos de inventario
- **key**: nombre del archivo de inventario
- **size**: tamaño de un archivo de inventario, en bytes
- **inventoriedRecord**: número de registros contenidos en un archivo de inventario

El siguiente es un ejemplo de un archivo simple de **manifest.json**.

```
{
  "sourceBucket": "user001",
  "destinationBucket": "bucket001",
  "version": "2019-01-03",
  "fileFormat": "CSV",
  "fileSchema": "Bucket,Key,Size,LastModifiedDate,ETag,StorageClass,IsMultipartUpload,ReplicationStatus,EncryptionStatus",
  "files": [
    {
      "key": "inventory/user001/test_id/2019-01-03T12-28Z/files/0000016813AF58E66806C1E2D7F15155_1.csv",
      "size": 6705647390,
      "inventoriedRecord": 70585762,
    }
  ]
}
```

El nombre de un archivo de **manifest** es el siguiente (para detalles sobre cada campo, vea [Nombre de archivo de inventario](#)):

```
destinationPrefix/sourceBucketName/inventoryId/yyyy-MM-dd'T'HH-mm'Z'/manifest.json
```

¿Cómo se notifica a un usuario cuando se generan archivos de inventario?

Puede habilitar SMN (el servicio de notificación de mensajes) para el bucket de destino. Al hacerlo, puede recibir mensajes SMS o correos electrónicos cada vez que se generen los archivos de inventario y el archivo de **manifest**. Para obtener más información acerca de SMN, consulte [Notificación de evento](#).

El siguiente es un ejemplo sencillo de configuración SMN. **destinationPrefix/sourceBucketName** indica el prefijo del archivo de **manifest**. **destinationPrefix** es el prefijo de nombre configurado para los archivos de inventario y **sourceBucketName** es el bucket de origen para el que está configurado el archivo de inventario. El **manifest.json** es el sufijo del archivo **manifest**.

```
<NotificationConfiguration>
  <TopicConfiguration>
    <Id>01</Id>
    <Filter>
      <Object>
        <FilterRule>
          <Name>prefix</Name>
          <Value>destination-prefix/source-bucket</Value>
        </FilterRule>
        <FilterRule>
          <Name>suffix</Name>
          <Value>manifest.json</Value>
        </FilterRule>
      </Object>
    </Filter>
  </TopicConfiguration>
</NotificationConfiguration>
```

```
<Event>ObjectCreated:Put</Event>  
</TopicConfiguration>  
</NotificationConfiguration>
```

¿Cómo usar esta función?

Puede configurar inventarios de buckets mediante la consola OBS, las API o los SDK.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Configuración del inventario de bucket |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Configuración de inventarios de bucket |

21 Notificación de evento

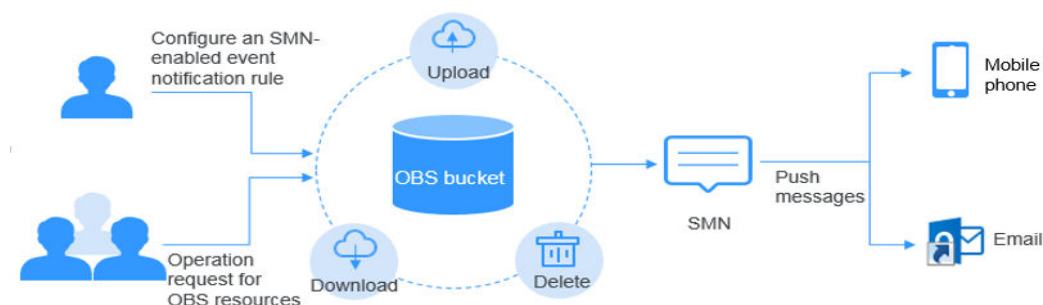
Notificación de eventos habilitados para SMN

Simple Message Notification (SMN) es un servicio de notificación de mensajes fiable y ampliable que puede manejar una gran cantidad de mensajes. SMN simplifica significativamente el acoplamiento del sistema. Puede enviar mensajes automáticamente a los suscriptores a través de correos electrónicos y mensajes de texto.

OBS utiliza SMN para proporcionar la función de notificación de eventos. Puede usar SMN para enviar notificaciones de eventos a los suscriptores especificados para informarles en tiempo real de las operaciones críticas (como la carga y la eliminación) que se producen en los bucket especificados. Por ejemplo, puede configurar una regla de notificación de evento para enviar mensajes a través de SMN a la dirección de correo electrónico especificada siempre que se produzca una operación de carga en el bucket especificado.

Puede configurar una regla de notificación de evento para filtrar objetos por el prefijo o sufijo de nombre de objeto. Por ejemplo, puede agregar una regla de notificación de evento para enviar notificaciones cada vez que se suba un objeto con el sufijo **.jpg** al bucket especificado. También puede agregar una regla de notificación de evento para enviar notificaciones cada vez que se carga un objeto con el prefijo **images/** en el bucket especificado.

Figura 21-1 Notificación de eventos habilitados para SMN



Tipos de eventos soportados por OBS

OBS puede publicar eventos de los siguientes tipos. Debe especificar estos tipos de eventos en la configuración de notificación.

Tabla 21-1 Tipos de eventos compatibles con OBS

| Tipo de evento | Descripción |
|---|---|
| <p>ObjectCreated:* (todas las operaciones de carga)</p> <p>ObjectCreated:Put (subida de objetos)</p> <p>ObjectCreated:Post (subir objetos a través de un navegador web)</p> <p>ObjectCreated:Copy (copiar objetos)</p> <p>ObjectCreated:CompleteMultipartUpload (combinar partes cargadas)</p> | <p>OBS puede usar API como PUT, POST y COPY para crear objetos y configurar los tipos de eventos correspondientes. Recibirá una notificación cuando se cree un objeto usando una API específica. Además, puede utilizar el tipo de evento ObjectCreated:* para solicitar todas las notificaciones de creación de objetos.</p> <p>NOTA No recibirá notificaciones de eventos de operaciones fallidas.</p> |
| <p>ObjectRemoved:* (todas las operaciones de eliminación)</p> <p>ObjectRemoved>Delete (eliminar objetos)</p> <p>ObjectRemoved>DeleteMarkerCreated (Se crea un objeto deletemarker)</p> | <p>Mediante el uso de los tipos de eventos ObjectRemoved puede habilitar la notificación cuando se elimina un objeto o un lote de objetos de un bucket.</p> <p>Puede solicitar una notificación cuando se elimina un objeto o se elimina permanentemente un objeto versionado mediante el tipo de evento ObjectRemoved>Delete. Como alternativa, puede solicitar una notificación cuando se crea un marcador de eliminación para un objeto versionado mediante ObjectRemoved>DeleteMarkerCreated. También puede utilizar ObjectRemoved:* para solicitar una notificación cada vez que se elimina un objeto.</p> <p>NOTA No recibirá notificaciones de eventos de eliminaciones automáticas de políticas de ciclo de vida o de operaciones fallidas.</p> |

Restricciones

- Una configuración sin atributos de filtrado contenidos coincide con todos los prefijos y sufijos de forma determinada.
- Cuando hay dos sufijos configurados, si una cadena dada puede terminar con ambos sufijos, esta configuración no es válida debido a la superposición de sufijos (por ejemplo, los sufijos jpg y pg). Esto también funciona para prefijos.

Destinos apoyados por OBS

OBS puede enviar mensajes de notificación de eventos a temas [SMN](#). Debe conceder a OBS los permisos para enviar mensajes a estos destinos. Además, debe especificar los valores URN de estos destinos en la configuración de notificación.

Cómo habilitar notificaciones de eventos

La habilitación de notificaciones es una operación a nivel de bucket. OBS almacena la configuración de notificación de eventos en subrecursos de bucket en el formato XML. De forma predeterminada, la notificación no está habilitada para ningún tipo de evento. Es decir, la configuración de notificación de evento inicial de cada bucket está vacía.

Para habilitar las notificaciones de eventos de tipos específicos, debe agregar la configuración XML correspondiente que identifique los tipos de eventos que desea que OBS publique y el destino donde desea que se publiquen las notificaciones.

- En este ejemplo, debe publicar mensajes de evento en un tema SMN. Para establecer un tema de SMN como destino de notificación para tipos de eventos específicos, agregue el `TopicConfiguration`.

```
<NotificationConfiguration>
  <TopicConfiguration>
    <Id>optional-id-string</Id>
    <Topic>topic-urn</Topic>
    <Event>event-type</Event>
    <Event>event-type</Event>
    ...
  </TopicConfiguration>
  ...
</NotificationConfiguration>
```

Para eliminar todas las configuraciones de notificación de un bucket, establezca el elemento `<NotificationConfiguration>` en null.

Uso de nombres de clave de objeto para filtrar notificaciones de eventos

Puede establecer la notificación de evento mediante el prefijo y el sufijo de un nombre de clave de objeto. Por ejemplo, puede configurar una configuración para que una notificación se publique solo cuando se agreguen objetos con extensión de **.jpg** a un bucket.

OBS almacena la configuración de notificación como XML. Puede utilizar el elemento `Filter` de la estructura XML para definir las reglas para las notificaciones que se van a filtrar por el prefijo y/o sufijo de un nombre de clave de objeto. Las configuraciones de notificación que utilizan el filtro no pueden definir reglas de filtrado con prefijos superpuestos, sufijos superpuestos o prefijos y sufijos superpuestos. Los siguientes son ejemplos de configuraciones de notificación con filtrado de nombres de clave de objeto:

- Ejemplo de configuración de notificación válida con filtrado de nombre de clave de objeto

La siguiente información contiene una configuración que identifica un tema SMN en el que OBS publica eventos del tipo **ObjectCreated:Put**. Los eventos se publicarán cada vez que un objeto que tiene un prefijo de **imagen** y un sufijo **jpg** se PUT a un bucket.

```
<NotificationConfiguration>
  <TopicConfiguration>
    <Id>01</Id>
    <Filter>
      <Object>
        <FilterRule>
          <Name>prefix</Name>
          <Value>image</Value>
        </FilterRule>
        <FilterRule>
          <Name>suffix</Name>
          <Value>jpg</Value>
        </FilterRule>
      </Object>
    </Filter>
  </TopicConfiguration>
</NotificationConfiguration>
```



```
</Filter>
<Topic>urn:smn:southchina:11aa22bb:notification</Topic>
<Event>ObjectCreated:Put</Event>
</TopicConfiguration>
</NotificationConfiguration>
```

La siguiente configuración de notificación tiene varios prefijos que no se superponen. La configuración define lo siguiente: Cuando los objetos que tienen un prefijo de **images** se cargan en los bucket, las notificaciones de eventos se publicarán en el tema-A; cuando los objetos que tienen un prefijo de **videos** se cargan en los bucket, las notificaciones de eventos se publicarán en el tema-B.

```
<NotificationConfiguration>
  <TopicConfiguration>
    <Id>01</Id>
    <Filter>
      <Object>
        <FilterRule>
          <Name>prefix</Name>
          <Value>images</Value>
        </FilterRule>
      </Object>
    </Filter>
    <Topic>urn:smn:southchina:11aa22bb:topic-A</Topic>
    <Event>ObjectCreated:Put</Event>
  </TopicConfiguration>
  <TopicConfiguration>
    <Id>02</Id>
    <Filter>
      <Object>
        <FilterRule>
          <Name>prefix</Name>
          <Value>videos</Value>
        </FilterRule>
      </Object>
    </Filter>
    <Topic>urn:smn:southchina:11aa22bb:topic-B</Topic>
    <Event>ObjectCreated:Put</Event>
  </TopicConfiguration>
</NotificationConfiguration>
```

La siguiente configuración de notificación tiene varios sufijos que no se superponen. La configuración define lo siguiente: Las notificaciones se publicarán en el tema-A para todos los objetos.**.jpg** subidos (PUT) en los bucket, y las notificaciones se publicarán en el tema-B para todos los objetos.**.png**. Los sufijos.png y.jpg no se superponen aunque tengan la misma última letra.

```
<NotificationConfiguration>
  <TopicConfiguration>
    <Id>01</Id>
    <Filter>
      <Object>
        <FilterRule>
          <Name>suffix</Name>
          <Value>.jpg</Value>
        </FilterRule>
      </Object>
    </Filter>
    <Topic>urn:smn:southchina:11aa22bb:topic-A</Topic>
    <Event>ObjectCreated:Put</Event>
  </TopicConfiguration>
  <TopicConfiguration>
    <Id>02</Id>
    <Filter>
      <Object>
        <FilterRule>
          <Name>suffix</Name>
          <Value>.png</Value>
        </FilterRule>
      </Object>
    </Filter>
    <Topic>urn:smn:southchina:11aa22bb:topic-B</Topic>
    <Event>ObjectCreated:Put</Event>
  </TopicConfiguration>
</NotificationConfiguration>
```

```
</Object>
</Filter>
<Topic>urn:smn:southchina:11aa22bb:topic-B</Topic>
<Event>ObjectCreated:Put</Event>
</TopicConfiguration>
</NotificationConfiguration>
```

- Ejemplos de configuración de notificación con prefijo/sufijo no válidos superpuestos

En la mayoría de los casos, las configuraciones de notificación que utilizan el filtro no pueden definir reglas de filtrado con prefijos superpuestos, sufijos superpuestos o combinaciones superpuestas de prefijos y sufijos para los mismos tipos de eventos. (Puede tener prefijos superpuestos si los sufijos no se superponen.) Puede utilizar reglas de filtrado de nombres de clave de objeto superpuestas con diferentes tipos de eventos. Por ejemplo, puede crear una configuración de notificación que utilice el prefijo **image** para el tipo de evento **ObjectCreated:Put** y el prefijo **image** para el tipo de evento **ObjectDeleted:***.

Una configuración sin atributos de filtrado contenidos coincide con todos los prefijos y sufijos de forma predeterminada. La siguiente configuración de notificación no es válida porque contiene prefijos superpuestos. (Lo mismo sería cierto si se usa sufijo en lugar de prefijo en este ejemplo.)

```
<NotificationConfiguration>
  <TopicConfiguration>
    <Topic>urn:smn:southchina:11aa22bb:topic-A</Topic>
    <Event>ObjectCreated:*</Event>
  </TopicConfiguration>
  <TopicConfiguration>
    <Filter>
      <Object>
        <FilterRule>
          <Name>prefix</Name>
          <Value>abc</Value>
        </FilterRule>
      </Object>
    </Filter>
    <Topic>urn:smn:southchina:11aa22bb:topic-B</Topic>
    <Event>ObjectCreated:*</Event>
  </TopicConfiguration>
</NotificationConfiguration>
```

La siguiente configuración de notificación no es válida porque contiene sufijos superpuestos. Dos sufijos se consideran superpuestos si una cadena dada puede terminar con ambos sufijos. Una cadena puede terminar con jpg y pg. Por lo tanto, los sufijos se superponen. (Lo mismo es cierto para los prefijos.)

```
<NotificationConfiguration>
  <TopicConfiguration>
    <Filter>
      <Object>
        <FilterRule>
          <Name>suffix</Name>
          <Value>jpg</Value>
        </FilterRule>
      </Object>
    </Filter>
    <Topic>urn:smn:southchina:11aa22bb:topic-A</Topic>
    <Event>ObjectCreated:*</Event>
  </TopicConfiguration>
  <TopicConfiguration>
    <Filter>
      <Object>
        <FilterRule>
          <Name>suffix</Name>
          <Value>pg</Value>
        </FilterRule>
      </Object>
    </Filter>
  </TopicConfiguration>
</NotificationConfiguration>
```

```
</Filter>
<Topic>urn:smn:southchina:11aa22bb:topic-B</Topic>
<Event>ObjectCreated:Put</Event>
</TopicConfiguration>
</NotificationConfiguration>
```

Estructura de mensaje de evento

Un mensaje de notificación enviado por OBS para publicar un evento es un mensaje JSON con la siguiente estructura.

```
{
  "Records": [
    {
      "eventVersion": "", //Version number. The current version is 3.0.
      "eventSource": "", //Message source. The value is fixed to OBS.
      "eventRegion": "", //Region where the event occurs
      "eventTime": "", //Time when an event occurs, in the ISO-8601 format,
for example, 2020-07-10T09:24:11.418Z
      "eventName": "", //Name of the event that triggers the notification
      "userIdentity": {
        "ID": "" //Billing ID of the user who triggers the event
      },
      "requestParameters": {
        "sourceIPAddress": "" //Source IP address of the request
      },
      "responseElements": {
        "x-obs-request-id": "", //ID of the request
        "x-obs-id-2": "" ///Special characters for locating problems
      },
      "obs": {
        "Version": "1.0",
        "configurationId": "", //Name of the event notification rule in
OBS that matches the event
        "bucket": {
          "name": "examplebucket",
          "ownerIdentity": {
            "ID": "" //Account ID of the bucket owner
          },
          "bucket": "" //Bucket name
        },
        "object": {
          "key": "", //Object name
          "eTag": "", //ETag of the object
          "size": , //Object size
          "versionId": "null", //Version ID of the object
          "sequencer": "" //Identifier that defines the event sequence
of a specific object
        }
      }
    }
  ]
}
```

Tenga en cuenta lo siguiente:

- Puede utilizar el valor clave de **responseElements** para rastrear las solicitudes OBS. Tanto **x-obs-request-id** como **x-obs-id-2** pueden usarse para rastrear una sola solicitud. Su valor es el devuelto por OBS en la respuesta a la solicitud.
- El valor clave **obs** contiene información sobre el bucket y el objeto involucrados en el evento. Tenga en cuenta que el nombre de la clave de objeto está codificado por URL. Por ejemplo, **TEST/Chinese.jpg** se cambia a **TEST%2F%E4%B8%AD+%E6%96%87%2F.jpg**.

AVISO

Durante el desarrollo secundario, si utiliza el SDK de OBS para descargar el objeto, debe decodificar el nombre del objeto codificado por URL y luego llamar a la API para su descarga. Si se utiliza un navegador web para acceder al objeto, no se requiere decodificación.

- El valor clave **sequencer** determina la secuencia de eventos. En general, las notificaciones de eventos no llegan en el orden en que se produjeron los eventos. Sin embargo, las notificaciones de eventos para crear objetos (PUT) y eliminar objetos (DELETE) contienen un secuenciador, que puede usarse para determinar el orden de eventos para una clave de objeto dada. Si compara las cadenas de secuenciador hexadecimal de dos notificaciones de evento en la misma clave de objeto, la notificación de evento con el mayor valor de secuenciador es el evento que ocurrió más tarde.

AVISO

1. El secuenciador no se puede utilizar para determinar la secuencia de eventos en diferentes claves de objeto.
2. La secuencia de eventos indicada por el secuenciador es solo para referencia, lo que no funciona para sistemas altamente fiables

Ejemplo de mensaje:

- Cuando configura una notificación de evento en un bucket, OBS envía el siguiente mensaje de prueba:

```
{
  "Service": "OBS",
  "Event": "TestEvent",
  "Time": "1970-01-01T00:00:00.000Z",
  "Bucket": "examplebucketname",
  "RequestId": "0002F4BCF6000001563B064B17B2094D",
  "HostId": "2Zf+b9AmbaBgNQ+YE8XU2j87DZaBNxu4TaMiOCTqpmkC2SA9ouf8TpB2SY5j3i4"
}
```

- Mensaje de ejemplo cuando se crea un objeto mediante la solicitud de la subida (PUT), A continuación se muestra un ejemplo de un mensaje enviado por OBS para publicar un evento ObjectCreated:Put:

```
{
  "Records": [
    {
      "eventVersion": "3.0",
      "eventSource": "OBS",
      "eventRegion": "region",
      "eventTime": "2018-06-26T14:37:05.468Z",
      "eventName": "ObjectCreated:Put",
      "userIdentity": {
        "ID": "71f3901173514e6988115ea2c26d1999"
      },
      "requestParameters": {
        "sourceIPAddress": "104.55.173.69"
      },
      "responseElements": {
        "x-obs-request-id": "9006000001643C86D03C300BE8860FA7",
        "x-obs-id-2": "2+/Ucr6uinCJAbUejWyQ+rhxkuF/K/9uoaXuewIi/SE9j4tU5LwaXTTlD1gvMv2o"
      },
      "obs": {
        "Version": "1.0",

```

```
"configurationId": "ConfigurationId",
"bucket": {
  "name": "examplebucket",
  "ownerIdentity": {
    "ID": "b4bf1b36d9ca43d984fbc9491b6fce9"
  },
  "bucket": "examplebucket"
},
"object": {
  "key": "object0001.txt",
  "eTag": "3b9680702b9a12733c5490d1b15c7607",
  "size": 538,
  "versionId": "null",
  "sequencer": "000000001643C86D06576F5320000000"
}
}
]
```

 **NOTA**

Debido a los problemas de Internet o restricciones en el envío de correo electrónico, es posible que las notificaciones no se envíen a HTTP o terminales de correo electrónico.

¿Cómo usar esta función?

Puede configurar las notificaciones SMN mediante la consola OBS, las API o los SDK.

| Herramienta | Referencia |
|-------------|--|
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Configuración de Notificación de Evento para un Bucket |

22 Registro

Escenario de la aplicación

Puede habilitar el registro para facilitar el análisis o la auditoría según sea necesario. Los registros de acceso permiten a un propietario del bucket analizar en profundidad la propiedad, el tipo o la tendencia de las solicitudes al bucket. Con el registro habilitado, OBS registra automáticamente las solicitudes de acceso para el bucket y escribe los archivos de registro generados en el bucket especificado.

Debe especificar un bucket para almacenar archivos de registro al habilitar el registro de un bucket. Los archivos de registro se pueden almacenar en cualquier bucket de la región donde reside el bucket registrado, incluido el bucket registrado en sí. Para gestionar mejor los registros, se recomienda almacenar los archivos de registro en un bucket que no sea el bucket registrado. Si los archivos de registro se almacenan en el bucket registrado, OBS crea registros adicionales para escribir archivos de registro en el bucket, lo que ocupa espacio de almacenamiento adicional que aumentará sus costos y le dificultará localizar los registros necesarios.

AVISO

La carga de registros de bucket en el bucket de destino conlleva costes para las solicitudes de PUT. Para obtener detalles sobre los precios de solicitud de PUT, consulte [Facturación de OBS](#).

OBS puede registrar las solicitudes de acceso al bucket en los registros para el análisis de solicitudes y la auditoría de registros.

Los registros ocupan algún espacio de almacenamiento OBS alquilado por los usuarios, incurriendo en tarifas adicionales. Por esta razón, la política predeterminada es que OBS no recopila registros de acceso al bucket.

Los archivos de registro son generados y cargados por OBS en el bucket donde se almacenan los registros. Por lo tanto, OBS requiere la autorización para cargar los archivos de registro generados. Por lo tanto, antes de configurar el registro para un bucket, debe crear una agencia de IAM para OBS y agregar esta agencia de IAM al configurar el registro para el bucket. De forma predeterminada, al configurar permisos para una agencia de IAM, solo tiene que conceder a la agencia de IAM el permiso para cargar archivos de registro en el bucket donde se almacenan los archivos de registro. En el siguiente ejemplo, **mybucketlogs** es el nombre

del bucket para almacenar archivos de registro. Si la función de cifrado predeterminada está habilitada para el bucket de almacenamiento de registros, la agencia IAM también requiere los permisos de administrador de KMS en la región donde reside el bucket de almacenamiento de registros.

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Action": [
        "obs:object:PutObject"
      ],
      "Resource": [
        "OBS:*:*:object:mybucketlogs/*"
      ],
      "Effect": "Allow"
    }
  ]
}
```

Después de configurar el registro, puede ver los registros de operaciones en el bucket que almacena los registros en aproximadamente quince minutos.

A continuación se muestra un registro de acceso de ejemplo:

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

El registro de acceso de cada bucket contiene la siguiente información.

Tabla 22-1 Formato de los registros de acceso al bucket

| Nombre | Ejemplo | Descripción |
|-------------|----------------------------------|--|
| BucketOwner | 787f2f92b20943998a4fe2ab75eb09b8 | ID de cuenta del propietario del bucket |
| Bucket | Bucket | Nombre del bucket |
| Time | [13/Ago/2015:01:43:42 +0000] | Marca de tiempo de la solicitud (UTC) |
| Remote IP | xx.xx.xx.xx | Solicitud de la dirección IP |
| Requester | 787f2f92b20943998a4fe2ab75eb09b8 | ID del solicitante <ul style="list-style-type: none"> ● Cuando una cuenta inicia una solicitud, éste es el ID de cuenta. Cuando un usuario de IAM inicia una solicitud, este es el ID de la cuenta a la que pertenece el usuario de IAM. ● Cuando un usuario anónimo inicia una solicitud, el valor de este parámetro es Anonymous. |

| Nombre | Ejemplo | Descripción |
|--------------------|--------------------------------------|--|
| RequestID | 281599BACAD9376ECE141 B842B94535B | ID de solicitud |
| Operation | REST.GET.BUCKET.LOCATION | Operación |
| Key | - | Nombre del objeto |
| Request-URI | GET /bucket?location HTTP/1.1 | Solicitud de URI |
| HTTPStatus | 200 | Código de respuesta |
| ErrorCode | - | Código de error |
| BytesSent | 211 | Tamaño de la respuesta HTTP, expresado en bytes |
| ObjectSize | - | Tamaño del objeto (bytes) |
| TotalTime | 6 | Tiempo de procesamiento en el servidor (ms) |
| Turn-AroundTime | 6 | Tiempo total para procesar la solicitud (ms) |
| Referer | - | Encabezado de referencia de la solicitud |
| User-Agent | HttpClient | Encabezado User-Agent de la solicitud |
| VersionID | - | ID de versión contenido en una solicitud |
| STSLogUrn | - | Autenticación federada e información de agencia |
| StorageClass | STANDARD_IA | Clase de almacenamiento actual del objeto |
| TargetStorageClass | GLACIER | Clase de almacenamiento a la que se transitará el objeto |

| Nombre | Ejemplo | Descripción |
|------------|----------------|---|
| DentryName | 12456/file.txt | <ul style="list-style-type: none"> ● Para un sistema de archivos paralelo, este campo indica un identificador interno de un archivo o directorio. Su valor consiste en un número de inodo de directorio padre y un nombre de archivo o directorio. ● Para un bucket, el valor de este campo es -. |

¿Cómo usar esta función?

Puede configurar el registro en la consola OBS, mediante API o SDK.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Configuración del registro de acceso para un bucket |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Configuración del registro de un bucket |

23 Control de versiones

Escenario de la aplicación

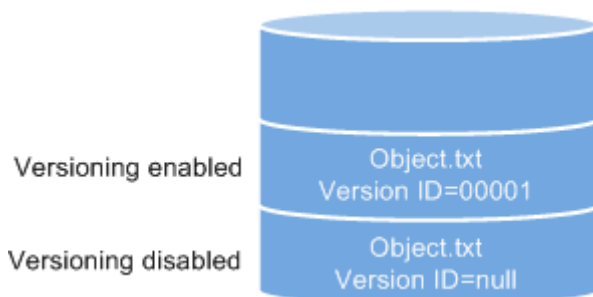
OBS puede almacenar varias versiones de un objeto. Puede buscar y restaurar las diferentes versiones rápidamente, o restaurar los datos en caso de que se realicen operaciones incorrectas o se presenten fallas en la aplicación.

De forma predeterminada, el control de versiones está deshabilitado para los nuevos los bucket de OBS. Los nuevos objetos sobrescribirán los objetos existentes en caso de que tengan los mismos nombres.

Habilitación del control de versiones

- Al habilitar el control de versiones no se cambian las versiones ni el contenido de los objetos existentes en el bucket. El ID de versión de un objeto es nulo antes de habilitar el control de versiones. Si se carga un objeto homónimo después de habilitar el control de versiones, se asignará un ID de versión al objeto. Para más detalles, consulte [Figura 23-1](#).

Figura 23-1 Control de versiones (habilitado o deshabilitado)



- Con el control de versiones habilitado, OBS asigna automáticamente un ID de versión único a un objeto recién cargado. Cuando se vuelve a cargar un objeto con el mismo nombre que un objeto existente, ambos objetos se almacenan en OBS con el mismo nombre pero diferentes ID de versión. Para más detalles, consulte [Figura 23-2](#).

Figura 23-2 Versiones (diferentes ID de versión para los objetos del mismo nombre)

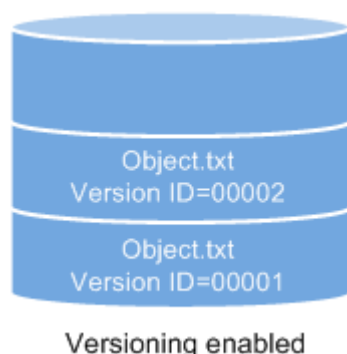


Tabla 23-1 Descripción de la versión

| Versión | Descripción |
|----------------------|---|
| Versión más reciente | Después de habilitar el control de versiones, cada operación en un objeto dará como resultado el almacenamiento del objeto con un nuevo ID de versión. El ID de versión generado en la última operación se denomina versión más reciente. |
| Versiones históricas | Después de habilitar el control de versiones, cada operación en un objeto dará como resultado el almacenamiento del objeto con un nuevo ID de versión. Los ID de versión generados en operaciones distintas de la última operación se denominan versiones históricas. |

- Los objetos más recientes de un bucket se devuelven por defecto después de una solicitud GET.
- Los objetos se pueden descargar por ID de versión. Por defecto, se descarga el último objeto si no se especifica el ID de la versión.
- Puede seleccionar un objeto y hacer clic en **Delete** a la derecha para borrarlo. Una vez eliminado el objeto, OBS genera un **Delete Marker** con un ID de versión único para el objeto eliminado, y el objeto eliminado se muestra en la lista **Deleted Objects**. Si intenta acceder al objeto eliminado, se mostrará un error 404.

Figura 23-3 Objeto con un delete marker



- Puede recuperar un objeto eliminado mediante la eliminación de la versión del objeto que tiene el **Delete Marker**.
- Después de eliminar un objeto, puede especificar el número de versión en Objetos eliminados para eliminar permanentemente el objeto de la versión especificada.
- Un objeto se muestra en la lista de objetos o en la lista de objetos eliminados. Nunca se mostrará en ambas listas simultáneamente.

Por ejemplo, después de cargar y eliminar el objeto A, se mostrará en la lista **Deleted Objects**. Si vuelve a cargar un objeto de nombre A, el objeto A se mostrará en la lista **Objects** y el objeto A eliminado anteriormente ya no se mostrará en la lista **Deleted Objects**. Para más detalles, consulte [Figura 23-4](#).

Figura 23-4 Subir un objeto homónimo después de eliminar el objeto original

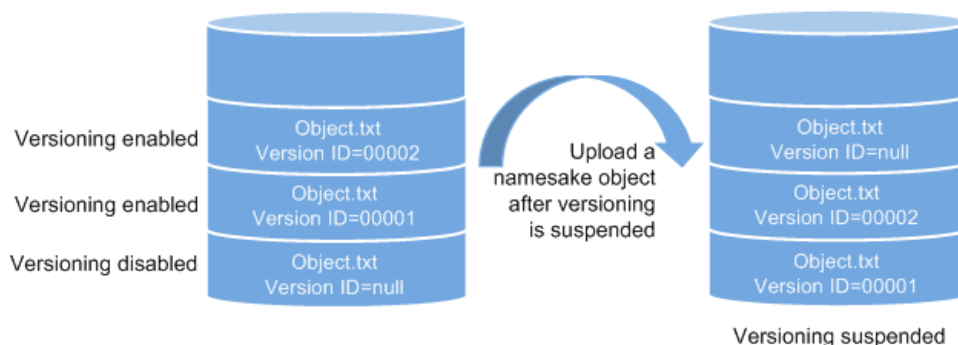


- Se cobran todas las versiones de objetos, excepto aquellas con el **Delete Marker** almacenado en OBS.

Suspensión del control de versiones

Una vez habilitada la función de control de versiones, se puede suspender pero no se puede deshabilitar. Una vez que se suspende el control de versiones, los identificadores de versión ya no se asignarán a los objetos recién cargados. Si ya existe un objeto con el mismo nombre y no tiene un ID de versión asignado, el objeto será sobrescrito.

Figura 23-5 Versiones de objetos en el escenario cuando se suspende el control de versiones



Si no es necesario controlar las versiones de objetos de un bucket, puede suspender la función de control de versiones.

- Las versiones históricas se conservarán en OBS. Si no necesita estas versiones históricas, elimínelas manualmente.
- Los objetos se pueden descargar por ID de versión. Por defecto, se descarga el último objeto si no se especifica el ID de la versión.
- Todas las versiones históricas de objetos almacenados en OBS, excepto las que tienen el **Delete Marker**, se cobran.

Diferencias entre escenarios cuando el control de versiones está suspendido y desactivado

Si se elimina un objeto cuando el control de versiones está suspendido, se genera una versión con ID nulo y **Delete Marker**, independientemente de si el objeto tiene versiones históricas. Pero, si el control de versiones está deshabilitado, la misma operación no generará una versión con el **Delete Marker**.

NOTA

Después de habilitar el control de versiones, cada versión histórica de un objeto se almacena y ocupa espacio de almacenamiento. OBS cobra tarifas de almacenamiento para todas las versiones. Tenga cuidado para evitar cargos adicionales de almacenamiento.

¿Cómo usar esta función?

Puede configurar el control de versiones a través de la consola OBS, las API y los SDK.

| Herramienta | Referencia |
|----------------|--|
| Consola de OBS | Configuración de Versiones |
| SDK | OBS admite kits de desarrollo de software (SDK) en varios idiomas. Para obtener más información, consulte la guía para desarrolladores correspondiente en la página de Descripción general del SDK . |
| API | Configuración del control de versiones para un bucket |

24 Monitoreo

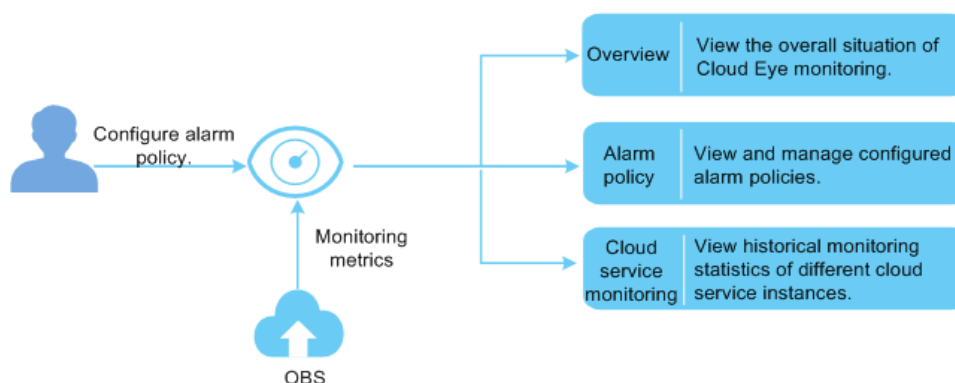
Escenario de la aplicación

Puede enviar solicitudes PUT y GET continuamente cuando utilice OBS, que genera tráfico de carga y descarga. También puede recibir respuestas de error del servidor. Huawei Cloud proporciona el servicio Cloud Eye para ayudarlo a monitorear OBS y comprender mejor los estados de sus buckets. Cloud Eye puede realizar un monitoreo automático y en tiempo real de sus los bucket. Activa alarmas y notificaciones sobre las operaciones del usuario basadas en políticas predefinidas, lo que le permite vigilar de cerca las solicitudes de acceso al bucket, el tráfico y las respuestas a errores.

No es necesario suscribirse por separado al servicio de Cloud Eye. Se inicia automáticamente una vez que se crea un recurso (un bucket, por ejemplo) en OBS.

Para obtener más información acerca de Cloud Eye, consulte [¿Qué es Cloud Eye?](#)

Figura 24-1 Monitoreo de Cloud Eye



Configuración de reglas de alarma

Además de la supervisión automática y en tiempo real, puede configurar reglas de alarma en Cloud Eye para que se envíen notificaciones de alarma cuando ocurran situaciones específicas.

Para obtener más información sobre cómo configurar una regla de alarma para la supervisión de Cloud Eye a través de OBS, consulte [Creación de una regla de alarma](#).

Consulta de Métricas de Monitoreo OBS

Cloud Eye monitoriza el uso de OBS en tiempo real. Puedes ver estadísticas detalladas de cada métrica en la consola de Cloud Eye.

Para obtener más información sobre cómo ver las métricas de supervisión de OBS, consulte [Consulta de métricas de monitoreo de servicios en la nube](#).

Métricas de monitoreo

Para obtener más información, consulte [Métricas de monitoreo de OBS](#).

25 Auditoría

Escenario de la aplicación

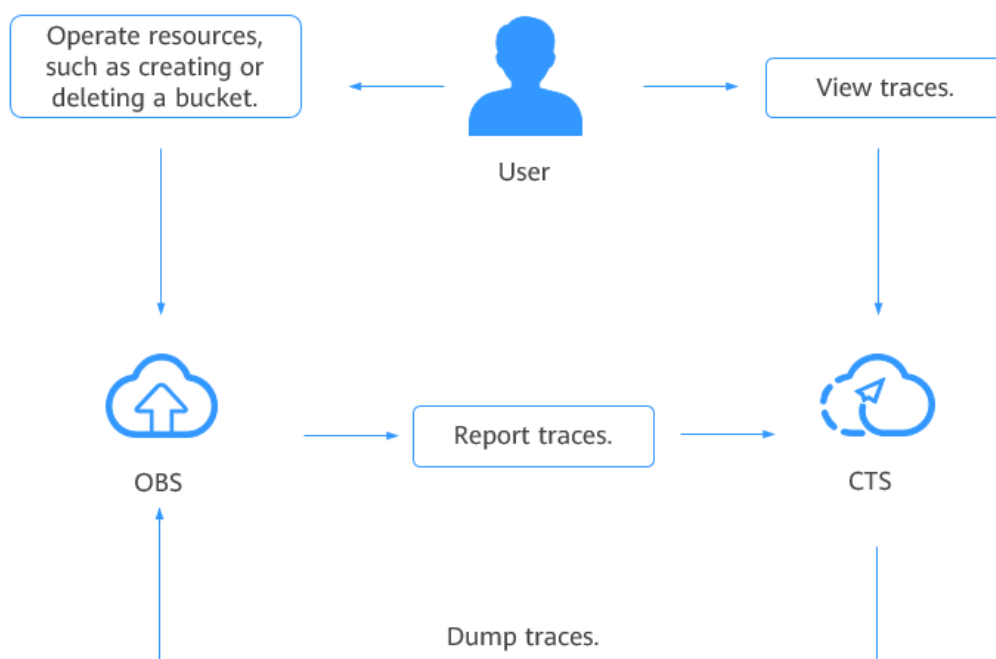
Cloud Trace Service (CTS) registra las operaciones en los recursos de la nube de su cuenta. Puede utilizar los registros generados por CTS para realizar análisis de seguridad, realizar un seguimiento de los cambios de recursos, auditar el cumplimiento y localizar fallos.

Después de habilitar CTS y configurar un rastreador, CTS puede registrar la gestión y las trazas de datos de OBS para su auditoría.

Para obtener más información acerca de cómo habilitar y configurar CTS, consulte [Habilitación de CTS](#).

Para obtener más información sobre la gestión de OBS y los seguimientos de datos que puede rastrear CTS, consulte [Cloud Trace Service](#).

Figura 25-1 Cloud Trace Service



26 Descompresión en línea (OBT)

Puede comprimir varios archivos en un paquete ZIP y subirlo a OBS.

OBS descomprime automáticamente los paquetes ZIP después de que se cargan.

Escenario de la aplicación

- Sube un paquete ZIP a un bucket OBS y luego haz que el paquete se descomprima automáticamente y se almacene en un directorio específico.
- Sube una gran cantidad de archivos pequeños a la vez usando un paquete ZIP para ahorrar tiempo y esfuerzos. Para las restricciones de carga por lotes, consulte [Carga por lotes de OBS](#).

Prerrequisitos

Se le ha asignado la función de **Tenant Administrator**.

Restricciones

- Actualmente, la descompresión en línea solo está disponible en las regiones CN North-Beijing4, CN South-Guangzhou y CN East-Shanghai2.
- Actualmente, solo se admiten paquetes ZIP.
- Un solo paquete ZIP no puede exceder 1 GB de tamaño.
- Un solo paquete ZIP puede contener un máximo de archivos 65,536.
- Un solo archivo descomprimido no puede exceder los 40 GB de tamaño.
- La descompresión de un paquete ZIP tarda 10 minutos como máximo.
- Los nombres de paquetes ZIP no pueden contener signos de puntuación chinos, caracteres especiales o códigos especiales.
- Para descomprimir el paquete ZIP que contiene otros paquetes ZIP, el tipo de evento de la política de descompresión en línea debe establecerse en **ObjectCreated:*** o **ObjectCreated:CompleteMultipartUpload**.
- Actualmente, no se enviará ninguna notificación a los usuarios después de completar las tareas de descompresión.
- Actualmente, solo se pueden descomprimir paquetes ZIP desinflado-comprimidos. Deflate es diferente de los algoritmos de compresión (como Store y Normal) proporcionados por WinRAR.

- La longitud total de la ruta de descompresión más el nombre de cualquier archivo descomprimido no puede exceder de 1024 caracteres.
- Actualmente, los paquetes ZIP cifrados no se pueden descomprimir.

Precauciones

- Se recomienda establecer un prefijo preciso para una política de descompresión. En el mismo bucket, las condiciones de activación (incluidos eventos, prefijos o sufijos) de diferentes políticas de descompresión no pueden solaparse entre sí.
 Por ejemplo, hay dos políticas de descompresión **event-0001** y **event-0002** en un bucket. Si el prefijo del **event-0001** es **aa**, el prefijo del **event-0002** no puede ser **aaaa**, porque **aa** está contenido en **aaaa**.
- Si el prefijo se deja en blanco, la política de descompresión se aplica a todos los paquetes ZIP del bucket de forma predeterminada. Esto puede activar la descompresión cíclica si un paquete contiene otros paquetes.
 Por ejemplo, el paquete **AA.zip** contiene otro paquete **BB.zip**. Si el prefijo se deja en blanco, el sistema continúa descomprimiendo **BB.zip** después de descomprimir **AA.zip**. Este problema no se producirá si se establece un prefijo en la política.
- Debe establecer un directorio para almacenar los archivos descomprimidos. Si el directorio no está definido, los archivos descomprimidos se almacenarán en el directorio principal del bucket actual.
- Se recomienda codificar los nombres de archivos o carpetas usando UTF-8. De lo contrario, los nombres de los archivos o carpetas descomprimidos pueden contener caracteres confusos, o la descompresión puede ser interrumpida.
- Si desea descomprimir paquetes ZIP en la clase de almacenamiento Archive, asegúrese de que la Lectura directa se ha habilitado para el bucket.
- Una descompresión de paquete ZIP fallará si tarda más de 10 minutos.

¿Cómo usar esta función?

Configure las políticas para descomprimir paquetes ZIP en línea a través de la consola OBS o API.

| Herramienta | Referencia |
|----------------|---|
| Consola de OBS | Creación de una política de descompresión en línea |
| API | Configuración de una política de descompresión en línea |

27 Historial de revisiones

| Fecha | ¿Qué hay de nuevo? |
|-----------|--|
| 31/8/2020 | Este es el primer lanzamiento oficial. |