Object Storage Service(OBS)

Product Introduction

Issue	01
Date	2024-10-17

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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

Notice

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

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

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: https://www.huaweicloud.com/intl/en-us/

Contents

1 About OBS

OBS Overview

Object Storage Service (OBS) is a scalable service that provides secure, reliable, and cost-effective cloud storage for massive amounts of data.

OBS provides unlimited storage capacity for objects of any format, catering to the needs of common users, websites, enterprises, and developers. There is no limitation on the storage capacity of the entire OBS system or of a single bucket, and any number of objects can be stored. As a web service, OBS supports APIs over Hypertext Transfer Protocol (HTTP) and Hypertext Transfer Protocol Secure (HTTPS). You can use OBS Console or OBS tools to access and manage data stored in OBS anytime, anywhere. With OBS SDKs and APIs, you can easily manage data stored in OBS and develop upper-layer applications.

OBS infrastructures are deployed in multiple regions across the globe, which delivers high scalability and reliability. You can deploy OBS in specific regions for faster access at an affordable price.

Product Architecture

OBS basically consists of **buckets** and **objects**.

A bucket is a container for storing objects in OBS. Each bucket is specific to a region and has specific storage class and access permissions. A bucket is accessible through its **access domain name** over the Internet.

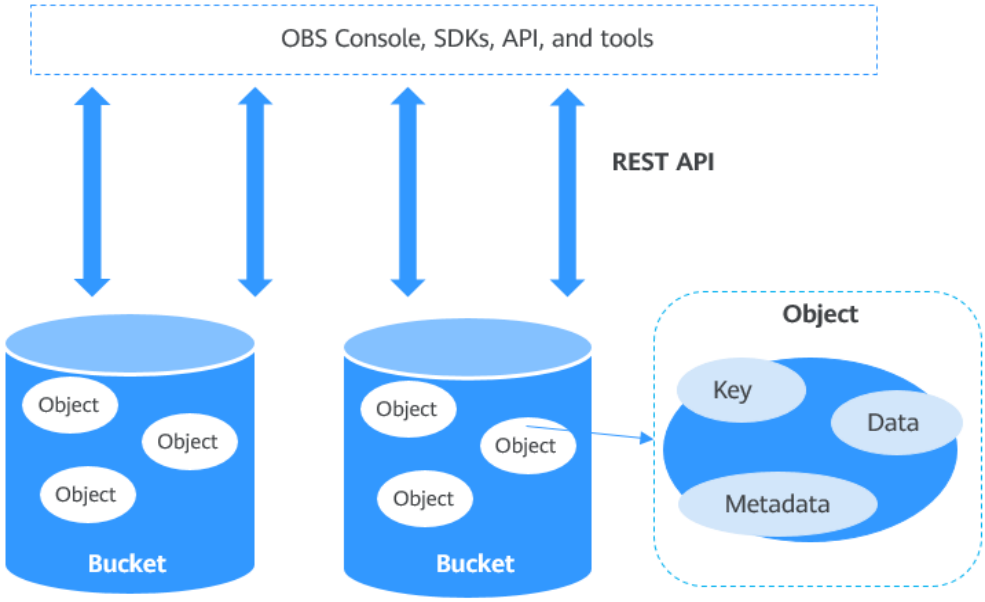
An object is the fundamental storage unit in OBS. An object consists of the following:

- A key that specifies the name of an object. An object key is a UTF-8 string up to 1,024 characters long. Each object is uniquely identified by a key within a bucket.
- Metadata that describes an object. The metadata is a set of key-value pairs that are assigned to objects stored in OBS. There are two types of metadata: system-defined metadata and custom metadata.
  - System-defined metadata is automatically assigned by OBS for processing objects. Such metadata includes Date, Content-Length, Last-Modified, ETag, and more.

  - You can specify custom metadata to describe the object when you upload an object to OBS.
- Data that refers to the content of an object.

By means of secondary development based on OBS REST APIs, OBS Console, SDKs, and a variety of tools are provided for you to use OBS. You can also use OBS SDKs and APIs to develop applications customized for your business needs.

Figure 1-1 Product architecture



Storage Classes

OBS offers the storage classes below to meet your requirements for storage performance and costs. You can **change buckets and objects between storage classes**. To learn billing for different storage classes, see **Storage Space**.

- Standard: The Standard storage class features low latency and high throughput. It is therefore good for storing frequently (multiple times per month) accessed files or small files (less than 1 MB). Its application scenarios include big data analytics, mobile apps, hot videos, and social apps.
- Infrequent Access: The Infrequent Access storage class is for storing data that is infrequently (less than 12 times per year) accessed, but when needed, the access has to be fast. It can be used for file synchronization, file sharing, enterprise backups, and many other scenarios. This storage class has the same durability, low latency, and high throughput as the Standard storage class, with a lower cost, but its availability is slightly lower than the Standard storage class.
- Archive: The Archive storage class is ideal for storing data that is rarely (once per year) accessed. Its application scenarios include data archive and long-term backups. This storage class is secure, durable, and inexpensive, so it can be used to replace tape libraries. To keep cost low, it may take hours to restore data from the Archive storage class.

An object uploaded to a bucket inherits the storage class of the bucket by default. You can also specify a storage class for an object when you upload it.

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

Table 1-1 Comparison of storage classes

Item	Standard	Infrequent Access	Archive	Deep Archive (Under Limited Beta Testing)
Feature	Top-notch performance, high reliability and availability	Reliable, inexpensive storage with real-time access	Long-term, inexpensive storage for Archive data	Long-term storage for Deep Archive data, with a lower unit price than Archive storage
Application scenarios	Cloud applications, data sharing, content sharing, and hot data storage	Web disk applications, enterprise backup, active archiving, and data monitoring	Storage of archives, medical imaging data, and videos, as well as replacement of tape libraries	Archive data that is barely accessed
Designed durability	99.999999999%	99.999999999%	99.999999999%	99.999999999%
Designed durability (multi-AZ)	99.999999999%	99.999999999%	Not supported	Not supported
Designed availability	99.99%	99%	99%	99%
Designed availability (multi-AZ)	99.995%	99.5%	Not supported	Not supported
Minimum storage duration	N/A	30 days	90 days	180 days
Minimum billable object size[a]	64 KB	64 KB	64 KB	64 KB

Item	Standard	Infrequent Access	Archive	Deep Archive (Under Limited Beta Testing)
Data restore	N/A	Billed for each GB restored.	Data can be restored at a standard or an expedited speed. Billed for each GB restored.	Data can be restored at a standard or an expedited speed. Billed for each GB restored.
Image processing	Supported	Supported	Not supported	Not supported

NOTE

The minimum storage duration is the minimum billable storage duration. This means that objects will be billed for the minimum storage duration even if they are not stored for that long. For example, if an Infrequent Access object is deleted after being stored in OBS for 20 days, it will be billed for the storage of 30 days (the minimum storage duration for Infrequent Access).

Accessing OBS

OBS provides various resource management tools. You can use any of the tools listed in Table 1-2 to access and manage resources in OBS.

Table 1-2 OBS resource management tools

Tool	Description	How to Use
OBS Console	OBS Console is a web-based GUI for you to easily manage OBS resources.	Console Operation Guide
OBS Browser (abandoned)	OBS Browser has been taken offline since April 15, 2020. Its functions are inherited by the new client tool OBS Browser+ that provides you with better experience. Download the latest OBS Browser+. We apologize for any inconvenience and appreciate your understanding.	-
OBS Browser +	OBS Browser+ is a Windows client that lets you easily manage OBS resources from your desktop.	OBS Browser+ Tool Guide

Tool	Description	How to Use
obsutil	obsutil is a command line tool for you to perform common configuration and management operations on OBS. If you are comfortable using the command line interface (CLI), obsutil is recommended for batch processing and automated tasks.	**obsutil Tool Guide**
obsfs	obsfs is an OBS tool based on Filesystem in Userspace (FUSE). It helps you mount parallel file systems to Linux, so that you can easily access virtually unlimited storage space of OBS the same way as you would use a regular local file system.	**obsfs Tool Guide**
SDKs	OBS SDKs encapsulate the REST API provided by OBS to simplify development. You can call API functions provided by the OBS SDKs to enjoy OBS capabilities.	**SDK Reference**
APIs	OBS offers the REST API for you to access it from web applications with ease. By making API calls, you can upload and download data anytime, anywhere over the Internet.	**API Reference**

2 Advantages

Comparison Between OBS and On-Premises Storage Servers

In this information era, it becomes increasingly difficult for conventional on-premises storage servers to deal with the fast-growing data of enterprises. **Table 2-1** compares OBS with on-premises storage servers.

Table 2-1 Comparison between OBS and on-premises storage servers

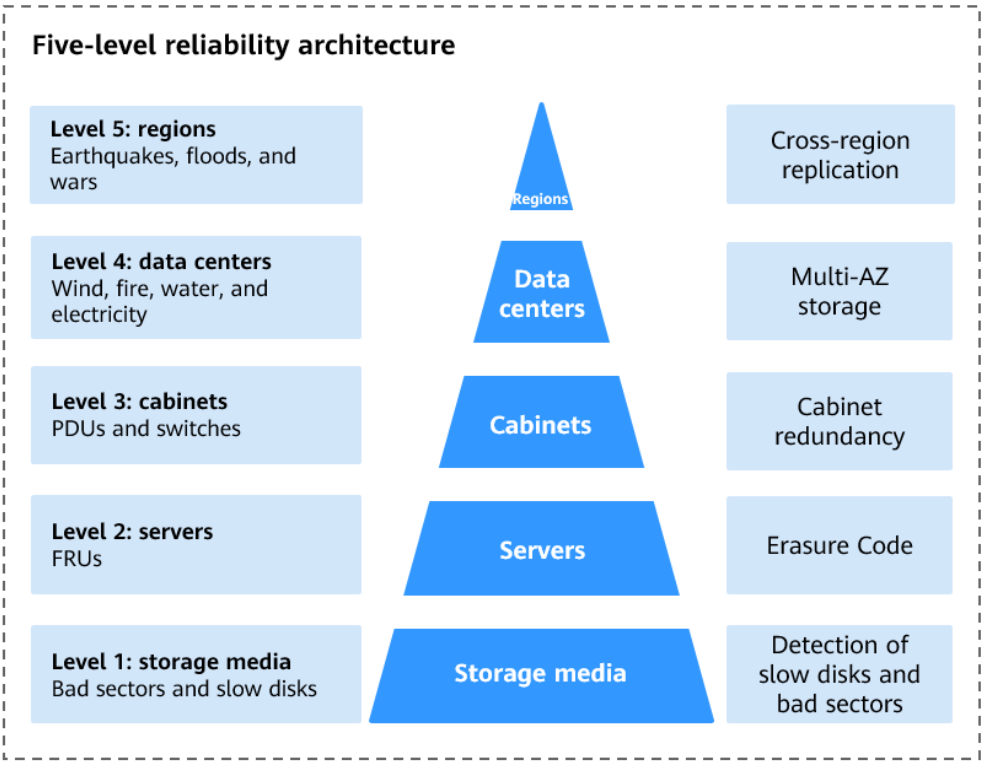
Item	OBS	On-Premises Storage Server
Storage capacity	OBS provides unlimited storage capacity, with data centers deployed across the world. All services and storage nodes are deployed in distributed clusters. You can expand each node or cluster separately, and you never have to worry about running out of space.	Such servers provide confined storage space due to the limited capacity of the hardware devices they use. When the storage space is not sufficient, you need to buy extra disks for manual expansion.
Security	OBS uses HTTPS and SSL protocols and encrypts data during uploads. To keep data in transit and at rest safe, OBS uses access key IDs (AKs) and secret access keys (SKs) to authenticate user identities and adopts a range of approaches including IAM permissions, bucket policies, access control lists (ACLs), and uniform resource locator (URL) validation.	The owner and users are exposed to security risks from cyber attacks, technical vulnerabilities, and accidental operations.

Item	OBS	On-Premises Storage Server
Reliability	The OBS five-level reliability architecture ensures up to 99.9999999999% of durability and up to 99.995% of continuity, much higher than those of the conventional architecture.	Due to limited investment, on-premises storage servers cannot ensure reliability at all levels of media, servers, cabinets, data centers, and regions. Once there is a failure or disaster, it may cause irreversible data loss to enterprises.
Costs	OBS is an out-of-the-box service that has no initial capital investment or time or labor costs and frees you from O&M. You only need to pay as you go. OBS offers tiered-pricing, meaning the more you use, the more you will save.	The initial deployment of on-premises servers requires high investments and a long construction period, but it quickly lags behind as enterprise businesses change so fast. Additional expenditures are required to ensure security.

OBS Advantages

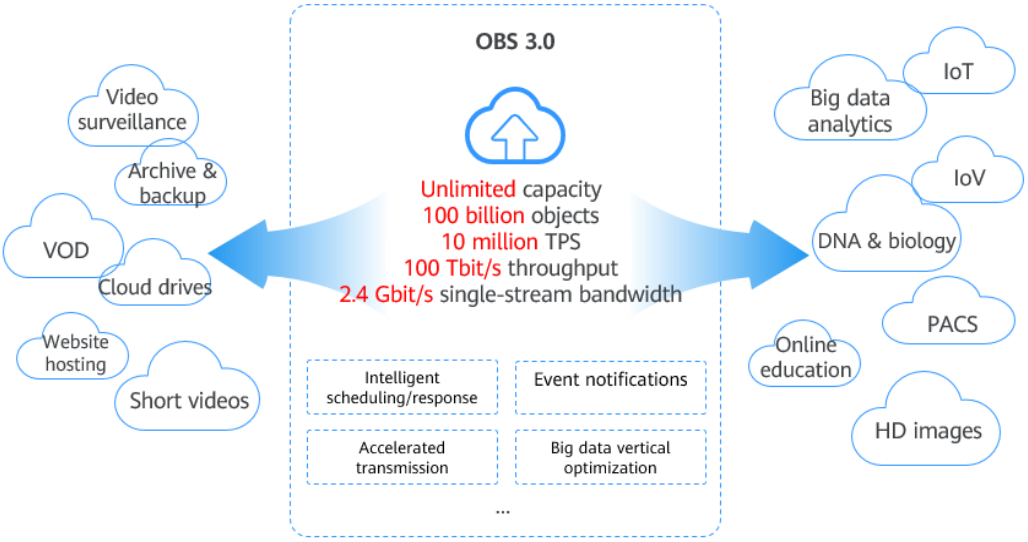
- **Data durability and service continuity**: OBS provides storage for cloud albums of mobile phones to support access of hundreds of millions of users. It delivers a data durability of up to 99.9999999999% and service continuity of up to 99.995% by using cross-region replication, cross-AZ disaster recovery, device and data redundancy in an AZ, slow disk or bad sector detection, and other technologies.

Figure 2-1 Five-level reliability architecture of OBS



- **Multi-level protection and authorization management**: OBS has passed the Trusted Cloud Service (TRUCS) certification. Measures, including versioning, server-side encryption, URL validation, virtual private cloud (VPC)-based network isolation, access log audit, and fine-grained access control are provided to keep data secure and trusted.
- **Highly concurrent access for hundreds of billions of objects**: With intelligent scheduling and response, optimized access paths, and technologies such as transmission acceleration and big data vertical optimization, you can store hundreds of billions of objects in OBS and still experience smooth concurrent access with ultra-high bandwidth and low latency.

Figure 2-2 Access to numerous objects at high-level concurrency



- **Easy use and management**: OBS provides standard REST APIs, SDKs in different programming languages, and data migration tools to help you quickly move your workloads to cloud. Storage resources are linearly, infinitely scalable, without compromising performance. You do not have to plan storage capacity beforehand or worry about expansion or reduction. When needed, you can ask Huawei Cloud to perform online upgrade or capacity expansion on your behalf.
- **Tiered storage and on-demand use**: Both pay-per-use and yearly/monthly billing are available for OBS. Data in each of the Archive, Infrequent Access, and Standard storage classes is separately metered and billed, which reduces storage costs.

3 Application Scenarios

Big Data Analytics

Description

OBS offers unlimited affordable storage for big data solutions that are designed for a wide range of scenarios, including:

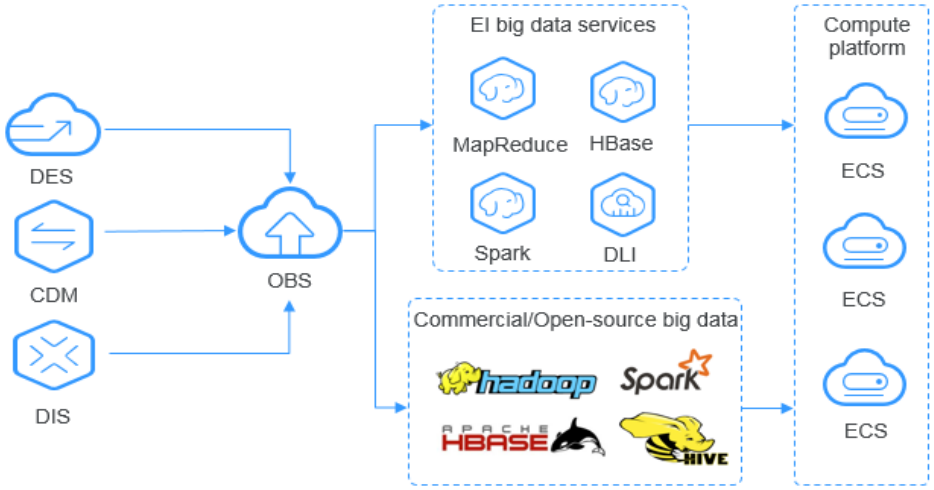
- Massive data storage and analysis, such as storage of petabytes of data, batch data analysis, and query in milliseconds
- Historical data query, such as transaction auditing, device energy consumption analysis, track playback, driving behavior analysis, and refined monitoring
- Massive log analysis, such as analysis of learning habits and operation logs
- Public transaction analysis, such as crime tracking, correlation analysis of criminal behaviors, traffic congestion analysis, and scenic spot popularity analysis

You can use Data Express Service (DES) to migrate data to OBS and then use big data services like MapReduce Service (MRS) or open-source frameworks such as Hadoop and Spark to analyze the data stored in OBS. The analysis results will be sent to your programs or applications running on Elastic Cloud Servers (ECSs).

Related Services

MRS, ECS, and DES

Figure 3-1 Big data analytics



Static Website Hosting

Description

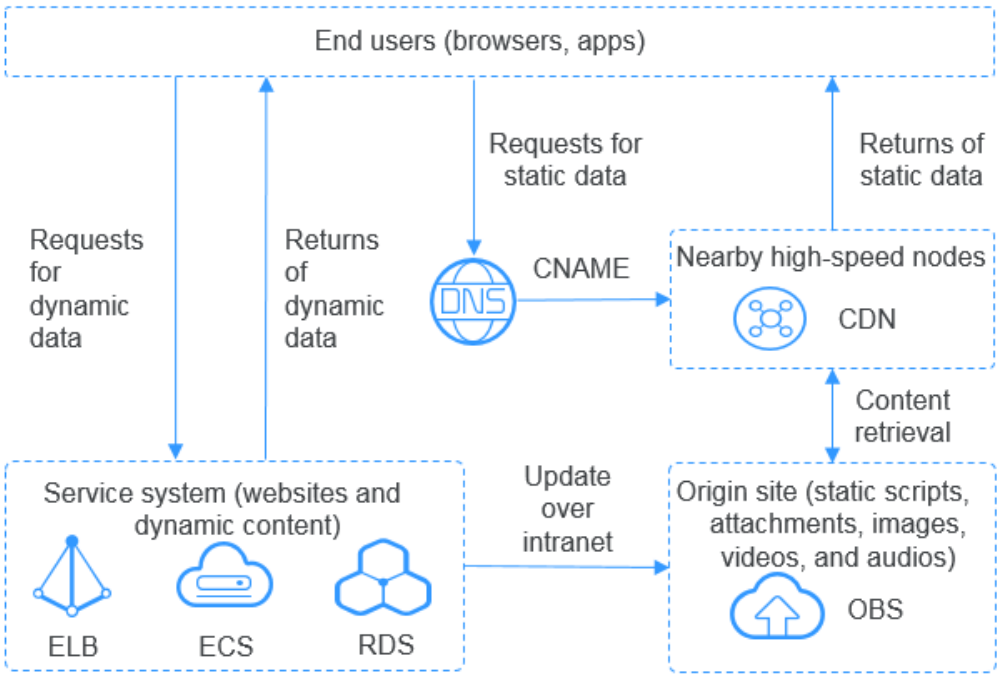
OBS provides a cost-effective, highly available, and scalable website hosting solution. By using the static website hosting together with Content Delivery Network (CDN) and ECS, you can quickly build a website or an application with static content separated from dynamic content.

End users send dynamic data from their web browsers or apps to service systems on Huawei Cloud for procesing. They then receive the results. Static data is stored in OBS. The service system processes the static data over the intranet. End users can request the static data from OBS through nearby high-speed nodes.

Related Services

CDN and ECS

Figure 3-2 Static website hosting



Video on Demand (VOD)

Description

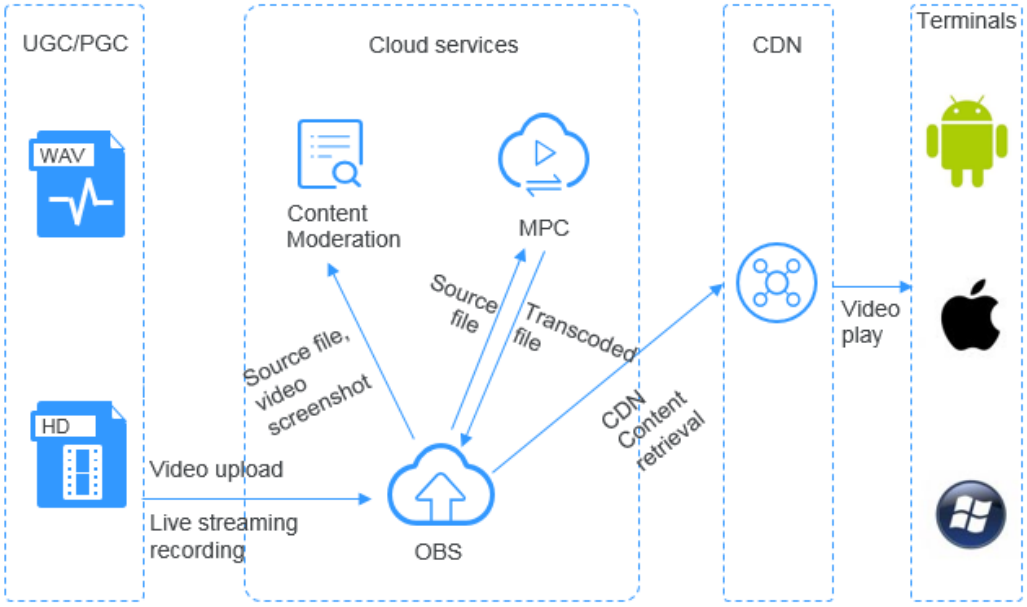
You can use OBS with Media Processing Center (MPC), Content Moderation, and CDN to set up a fast secure, and highly available VOD platform.

OBS serves as the origin server for VOD services. Internet users, especially content creators, can upload their video files to OBS for storage. Then, they can use Content Moderation to review the video content, use MPC to transcode the video files, and use CDN to speed up the content delivery from OBS.

Related Services

CDN, MPC, and Content Moderation

Figure 3-3 VOD



Genome Sequencing

Scenario

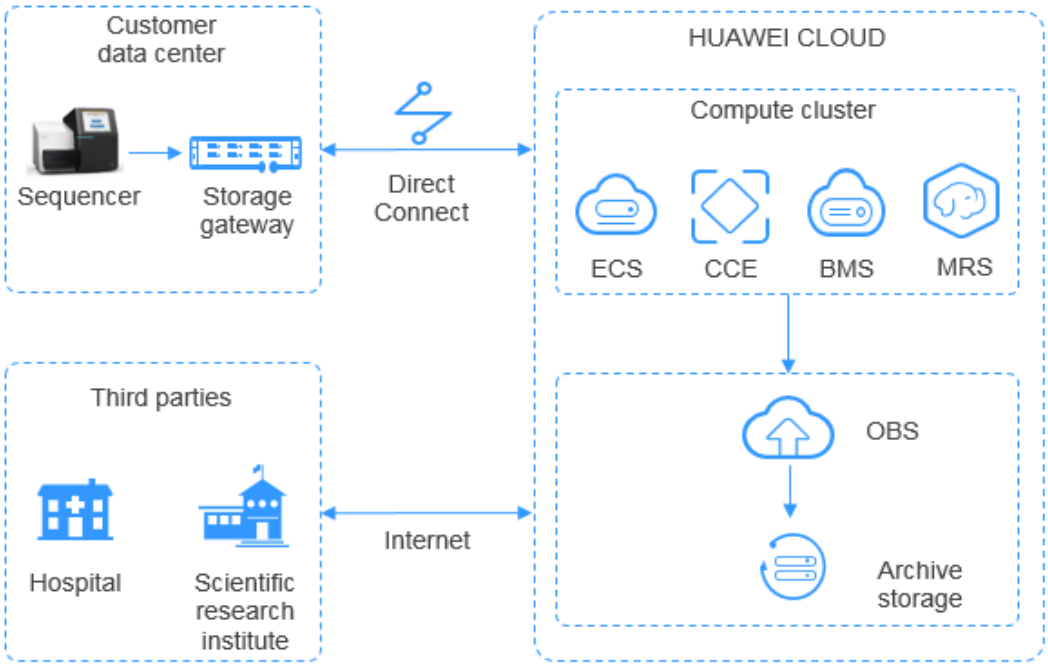
You can use OBS with compute services to build a genome sequencing platform.

You can use Direct Connect to upload data from your data center's sequencers to Huawei Cloud. The data will then be analyzed by the compute cluster (comprising ECS, CCE, and MRS). After the analysis is completed, the results will be stored in OBS and distributed to hospitals or research institutes over the Internet. The source genome data in OBS will be moved to the Archive storage class to reduce costs.

Related Services

ECS, Bare Metal Server (BMS), MRS, Cloud Container Engine (CCE), and Direct Connect

Figure 3-4 Genome sequencing



Intelligent Video Surveillance

Scenario

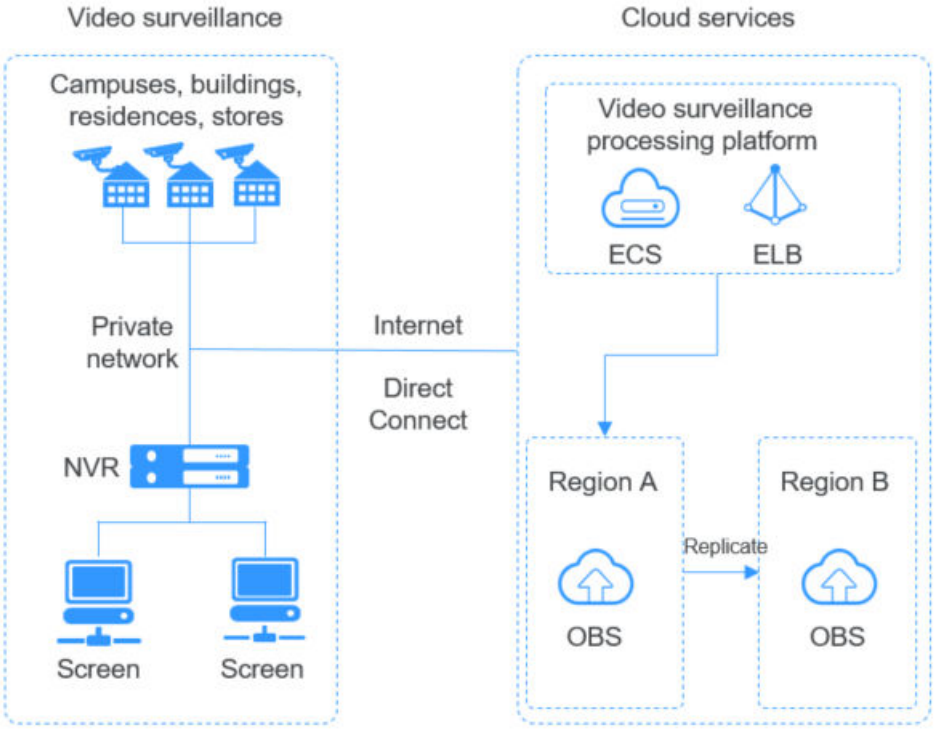
OBS offers unlimited storage for video surveillance solutions. These solutions can be tailored for individuals and enterprises alike and meet their needs for device management and video processing.

You can upload surveillance videos to Huawei Cloud over the Internet or a **Direct Connect** connection. The processing platform that consists of ECS and ELB slices the video streams and stores them in OBS. Later, you can download the video files from OBS to play on terminals. Video files in OBS can also be backed up using **Cross-Region Replication** for secure storage.

Related Services

Elastic Load Balance (ELB) and ECS

Figure 3-5 Video surveillance



Backup and Archiving

Description

You can use OBS to archive huge amounts of unstructured data of applications and databases.

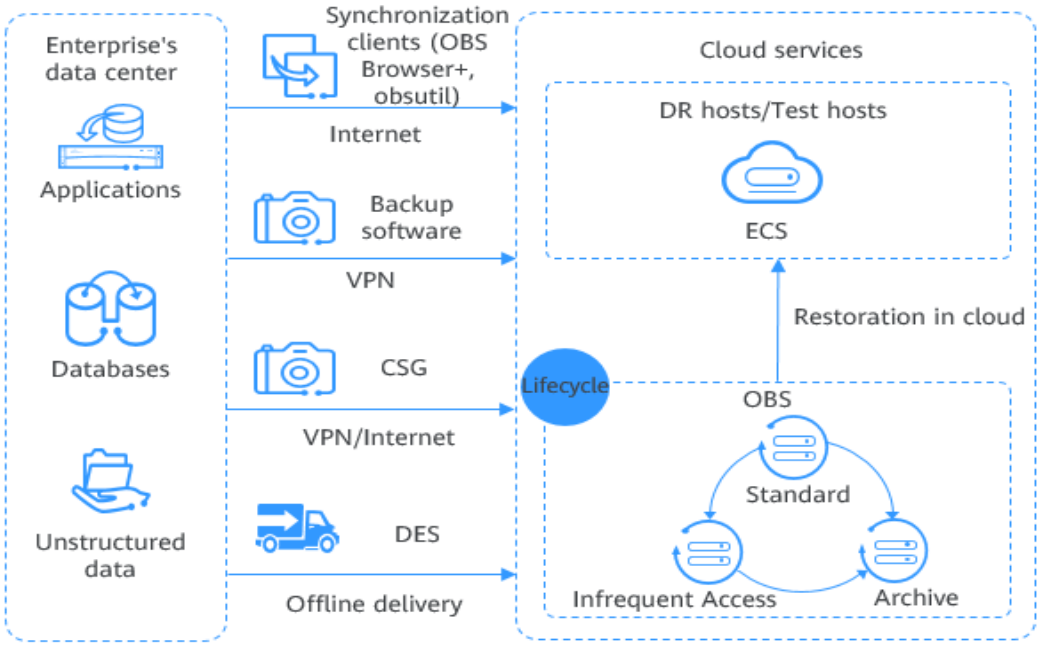
To back up on-premises data to OBS, you can use synchronization clients (such as OBS Browser+ and obsutil), Cloud Storage Gateway (CSG), DES, and popular backup software. OBS also allows you to configure lifecycle rules to transition objects between storage classes to reduce storage costs. If needed, you can restore data from OBS to a cloud DR or test host.

- Synchronization clients are good for manual backup of a single database or program.
- Backup software has strong compatibility. It is ideal for automatic backup of multiple applications or hosts.
- CSG can seamlessly work with on-premises backup systems.
- DES is ideal for archiving massive volumes of data. It uses physical devices to deliver data from on-premises data centers to the cloud.

Related Services

Data Express Service (DES) and ECS

Figure 3-6 Backup and archiving



High-Performance Computing

Description

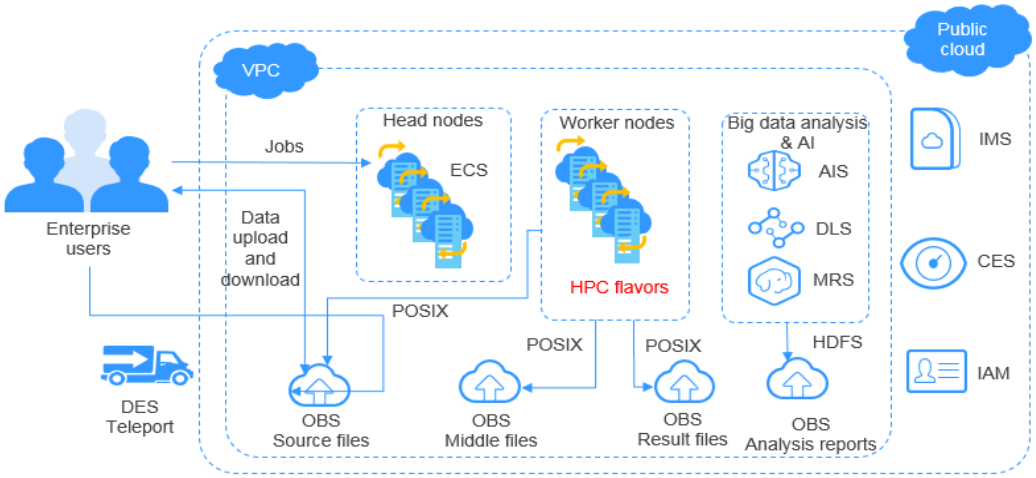
OBS can work with cloud services such as ECS, AS, EVS, IMS, IAM, and Cloud Eye to provide high-performance computing (HPC) solutions. These solutions require huge capacity and large single-stream bandwidth.

In HPC scenarios, enterprises can upload data to OBS directly or by using DES. The POSIX and HDFS of OBS allow you to mount buckets to HPC nodes, as well as big data and AI applications. This enables fast data reads and writes and efficient storage for high-performance computing.

Related Services

DES, ECS, AS, IMS, IAM, and Cloud Eye

Figure 3-7 High-performance computing



Enterprise Cloud Boxes (Web Disks)

Description

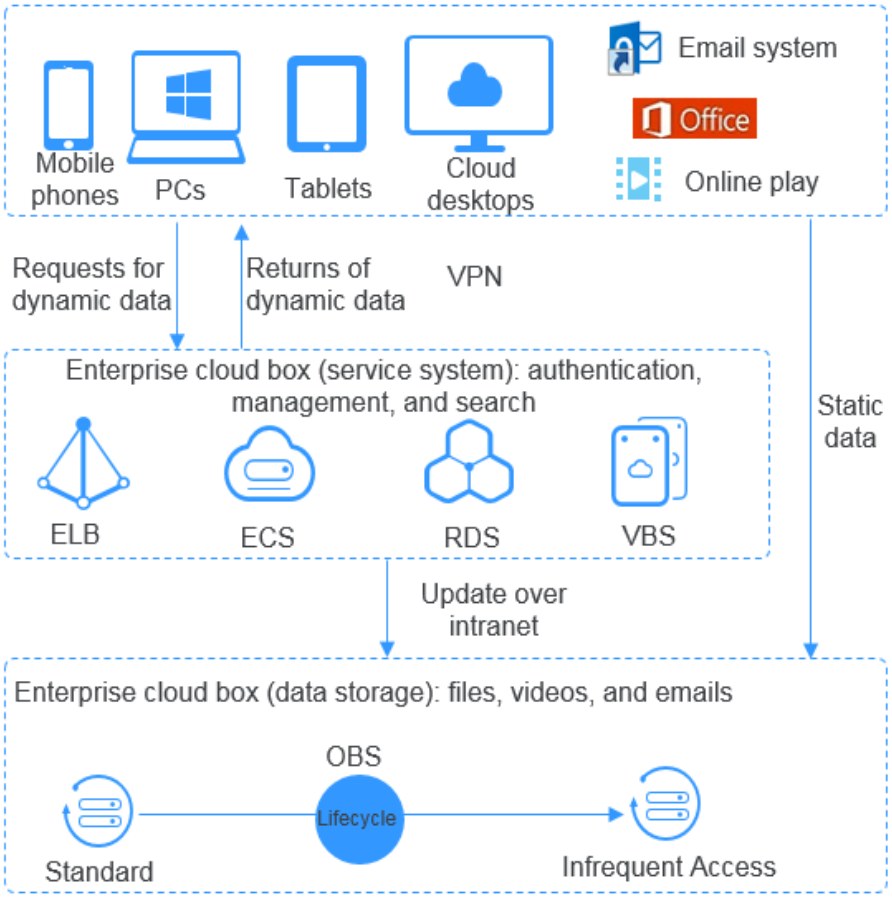
OBS can work with cloud services such as ECS, ELB, RDS, and VBS to provide enterprise web disks with a reliable, inexpensive storage system featuring low latency and high concurrency. The storage capacity automatically scales as the volume of stored data grows.

Dynamic data on devices such as mobile phones, PCs, and tablets interacts with the enterprise cloud disk service system built on Huawei Cloud. Requests for dynamic data are sent to the service system for processing and then returned to devices, and the static data is stored in OBS. Service systems can process static data over the intranet. End users can directly request and read the static data from OBS. In addition, OBS provides the lifecycle management function to automatically change storage classes for objects, reducing storage costs.

Related Services

Elastic Cloud Server (ECS), Elastic Load Balance (ELB), Relational Database Service (RDS), and Volume Backup Service (VBS)

Figure 3-8 Enterprise cloud boxes (web disks)



4 Functions

Table 4-1 lists the basic functions of OBS.

It is recommended that you get familiar with the basic concepts of OBS before using OBS.

Table 4-1 OBS functions

Function	Description	Region Availability	OBS 2.0	OBS 3.0
Storage classes	OBS offers the following storage classes: Standard, Infrequent Access, Archive, and Deep Archive (under limited beta testing), to meet different requirements for storage performance and cost.	All	Supported	Supported
Bucket management	Buckets are containers that store objects in OBS. OBS provides easy bucket management. You can conveniently create, list, search for, view, and delete buckets.	All	Supported	Supported
Object management	Objects are the fundamental entities stored in OBS. You can perform the following operations on objects: upload, download, listing, searching, resumable transfer, and multipart uploads.	All	Supported	Supported

Function	Description	Region Availability	OBS 2.0	OBS 3.0
Permissions management	OBS uses IAM permissions, bucket policies, object policies, and ACLs for access control. You can grant access to different accounts and users, and also configure policies or ACLs for buckets and objects to control read and write permissions for them.	All	Supported	Supported
Server-side encryption	To enhance data security, OBS uses server-side encryption to encrypt data before storing it. The encryption methods include SSE-KMS, SSE-OBS, and SSE-C.	See **Function Overview**. **NOTE** To find out the regions that support SSE-KMS or SSE-OBS, see **Server-Side Encryption**.	Supported	Supported
WORM	You can use a write-once-read-many (WORM) model to protect objects from being deleted or tampered with within a specified period.	See **Function Overview**.	Not supported	Supported
Lifecycle management	You can configure lifecycle rules to automatically delete objects or transition objects between storage classes.	All	Supported	Supported
Static website hosting	You can upload static website files to your OBS bucket, grant the read permission for these files to anonymous users, and configure static website hosting for the bucket to host them.	All	Supported	Supported

Function	Description	Region Availability	OBS 2.0	OBS 3.0
CORS	Cross-origin resource sharing (CORS) is a browser-standard mechanism defined by the World Wide Web Consortium (W3C). It allows a web client in one origin to interact with resources in another one. For general web page requests, website scripts and contents in one origin cannot interact with those in another because of Same Origin Policies (SOPs). OBS supports CORS rules for resources in it to be accessed across origins.	All	Supported	Supported
URL validation	URL validation protects your data in OBS from being stolen using the **Referer** field in HTTP requests. Such authorization is controlled using whitelists and blacklists.	All	Supported	Supported
Bucket tags	Tags are provided for you to identify and classify OBS buckets. If you add tags to a bucket, service detail records (SDRs) generated for it will be labeled with these tags. You can classify SDRs by tag for cost analysis.	All	Supported	Supported

Function	Description	Region Availability	OBS 2.0	OBS 3.0
User-defined domain names	You can bind a domain name to an OBS bucket and then use this domain name to access data in the bucket. For instance, if you need to migrate files from a website to OBS while keeping the website address unchanged, you can bind the website domain name to an OBS bucket.	All	Not supported	Supported
Cross-region replication	You can create a cross-region replication rule to automatically, asynchronously replicate objects from a source bucket in one region to a destination bucket in another region, as long as both buckets are under your account. This enables cross-region data disaster recovery, catering to your needs for remote backup.	All	Not supported	Supported
Image processing	You can this function to quickly process images stored in OBS, including compression, cropping, resizing, watermarking, and format conversion.	See **Function Overview**.	Not supported	Supported
Bucket inventories	Bucket inventories help you manage objects. You can configure a bucket inventory rule for OBS to periodically scan the specified objects, list the objects with their properties (such as metadata, size, modification time, and storage class) in CSV files, and store the files into the specified bucket.	See **Function Overview**.	Not supported	Supported

Function	Description	Region Availability	OBS 2.0	OBS 3.0
Parallel file systems	Parallel File System (PFS) is a high-performance file system with access latency in milliseconds. It supports TB/s-level bandwidth and millions of IOPS, which is ideal for processing high-performance computing (HPC) workloads. You can call standard OBS APIs to read data in a parallel file system, or use obsfs, an OBS tool, to mount a parallel file system to a Linux server in the cloud. Migrating files and directories in a parallel file system is just like operating a local file system.	See **Function Overview**.	Not supported	Supported
Logging	With logging, you can obtain the bucket access data. After logging is enabled for a bucket, OBS automatically logs every access request for the bucket, packs multiple log records into a log file, and saves the log file to the specified bucket. Using the stored logs, you can analyze or audit logs.	All	Supported	Supported
Versioning	When versioning is enabled for a bucket, OBS can keep multiple versions of an object in the bucket. That way you can quickly retrieve and restore every object version as needed, or recover data from both accidental actions and application failures.	All	Supported	Supported

Function	Description	Region Availability	OBS 2.0	OBS 3.0
Appending data to objects	You can call the AppendObject API to write additional data to an appendable object in a specified bucket. Objects created by calling the AppendObject API are appendable, while those created by calling the PutObject API are normal ones.	See **Function Overview**.	Not supported	Supported
Customizing metadata	You can add, modify, or delete metadata of uploaded objects.	All	Supported	Supported
Bucket storage quota	You can set the bucket space quota to limit the maximum amount of data that can be stored in a bucket. The maximum value is 2^{63}-1, in bytes. By default, the quota of a newly created bucket is not limited.	All	Supported	Supported
Direct reading	With direct reading enabled, you can download objects in the Archive storage class without restoring them in advance. Direct reading is a billable function.	See **Function Overview**.	Not supported	Supported
Object sharing	You can share a file or folder stored in OBS with all users by using a temporary URL. All shared URLs are valid for only the specified period of time.	See **Function Overview**.	Supported	Supported
Fragment management	You can clear fragments that are generated during multipart uploads to save storage space in a bucket.	All	Supported	Supported

Function	Description	Region Availability	OBS 2.0	OBS 3.0
Enterprise projects	When you create a bucket, you can specify an enterprise project for it, to facilitate bucket resource and permission management.	See **Function Overview**.	Not supported	Supported
Bucket encryption	You can enable server-side encryption for a bucket when creating it. Then all objects uploaded to this bucket will be encrypted by default.	See **Function Overview**.	Not supported	Supported
Multi-AZ storage	When creating a bucket, you can choose multi-AZ storage to store your data in multiple AZs for a higher data reliability. OBS uses the Erasure Code (EC) algorithm, instead of multiple copies, to ensure data redundancy.	See **Function Overview**.	Not supported	Supported
Back to source	With a back-to-source rule, if the data you requested is not found in OBS, OBS automatically pulls the data from its origin server and returns the data to you.	See **Function Overview**.	Not supported	Supported
Online decompression	Online decompression allows you to compress multiple files into a ZIP package and upload it to OBS for auto decompression.	See **Function Overview**.	Not supported	Supported
Bucket settings replication	You can replicate the settings of an existing bucket to the bucket you are creating, including bucket policies, CORS rules, back-to-source rules, image processing styles, online decompression rules, and lifecycle rules.	All	Not supported	Supported

Function	Description	Region Availability	OBS 2.0	OBS 3.0
Agencies	You can create an IAM agency to authorize other cloud services or Huawei Cloud accounts to manage your OBS resources.	All	Not supported	Supported
Monitoring	You can monitor the traffic statistics and requests of buckets on OBS Console and Cloud Eye, so that you are able to properly use your buckets.	See Function Overview.	Supported	Supported
Audit	CTS keeps track of operations on buckets and objects in OBS. You can query the records from CTS for security analysis, compliance audit, resource tracking, and fault locating.	All	Supported	Supported
Tools	OBS offers a range of tools, including OBS Browser+, obsfs, and obsutil, for data migration and management in different scenarios.	All	Supported	Supported
API	OBS provides REST APIs that support HTTP and HTTPS. You can call these APIs to create, modify, and delete buckets, as well as to upload, download, or delete objects.	All	Supported	Supported
SDKs	OBS SDKs help you perform secondary development. The SDKs are available in the following programming languages: Java, Python, C, Go, BrowserJS, .NET, Android, iOS, PHP, and Node.js.	All	Supported	Supported

5 Security

5.1 Shared Responsibilities

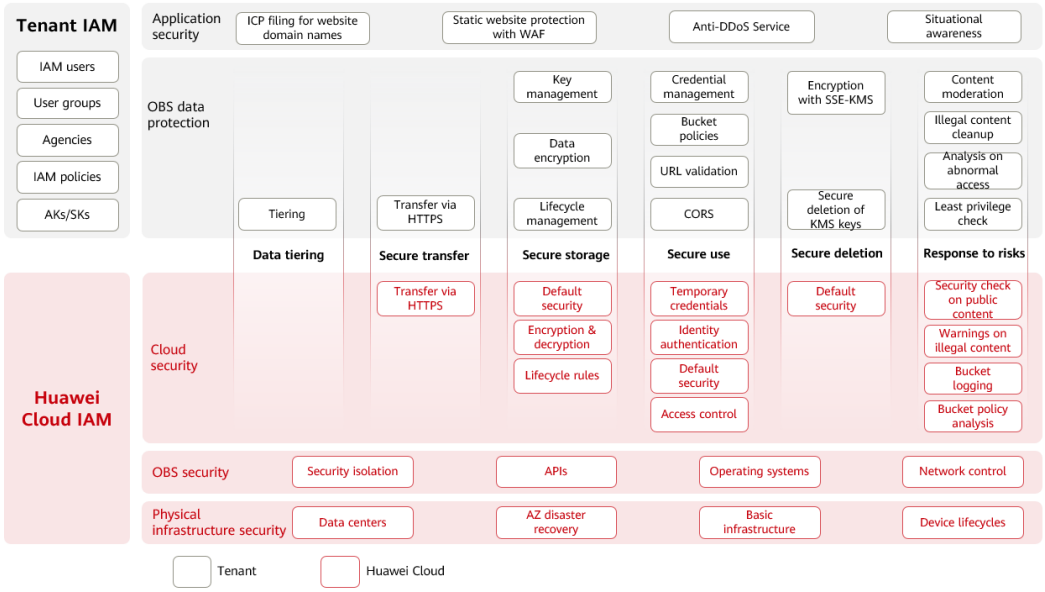
Huawei guarantees that its commitment to cyber security will never be outweighed by the consideration of commercial interests. To cope with emerging cloud security challenges and pervasive cloud security threats and attacks, Huawei Cloud builds a comprehensive cloud service security assurance system for different regions and industries based on Huawei's unique software and hardware advantages, laws, regulations, industry standards, and security ecosystem.

Security is a shared responsibility between Huawei Cloud and you. **Figure 5-1** illustrates how the security responsibilities are shared.

- **Huawei Cloud**: Ensure the security of cloud services and provide secure clouds. Huawei Cloud's security responsibilities include ensuring the security of our IaaS, PaaS, and SaaS services, as well as the physical environments of the Huawei Cloud data centers where our IaaS, PaaS, and SaaS services operate. Huawei Cloud is responsible for not only the security functions and performance of our infrastructure, cloud services, and technologies, but also for the overall cloud O&M security and, in the broader sense, the security and compliance of our infrastructure and services.
- **Tenant**: Use the cloud securely. Tenants of Huawei Cloud are responsible for the secure and effective management of the tenant-customized configurations of cloud services including IaaS, PaaS, and SaaS. This includes but is not limited to virtual networks, the OS of virtual machine hosts and guests, virtual firewalls, API Gateway, advanced security services, all types of cloud services, tenant data, identity accounts, and key management.

Huawei Cloud Security White PaperHuawei Cloud Security White Paper elaborates on the ideas and measures for building Huawei Cloud security, including cloud security strategies, the shared responsibility model, compliance and privacy, security organizations and personnel, infrastructure security, tenant service and security, engineering security, O&M security, and ecosystem security.

Figure 5-1 Huawei Cloud shared security responsibility model



5.2 Identity Authentication and Access Control

5.2.1 Identity Authentication and Access Control

Identity Authentication

You can use OBS Console, OBS Browser+ (a client), obsutil (a command line tool), APIs, and SDKs to access OBS. No matter which method you use, you are accessing OBS over the REST API.

OBS REST APIs support both authenticated and anonymous requests. There will usually be anonymous requests in the scenarios that require public access, for example, accessing a hosted static website. In most cases, requests for OBS resources must be authenticated. An authenticated request must include a signature. The signature is calculated based on the requester's access keys (a pair of AK and SK) that are used as the encryption factor and the specific information included in the request body. OBS uses an access key ID (AK) and a secret access key (SK) together to authenticate the identity of a requester. For more information, see **Access Keys (AK/SK)**.

Other OBS access scenarios include:

- **Accessing OBS Using Permanent Access Keys**
- **Accessing OBS Using Temporary Access Keys**
- **Accessing OBS Using a Temporary URL**
- **Accessing OBS Using an IAM Agency**

Access Control

OBS access control can be implemented based on IAM permissions, bucket policies, ACLs, URL validation, and CORS.

Table 5-1 OBS access control

Method		Description	Reference
Permission control	IAM permissions	IAM permissions define which actions on your cloud resources are allowed or denied. After creating an IAM user, the administrator needs to add it to a user group and grant the permissions required by OBS to the user group. Then, all users in this group automatically inherit the granted permissions.	**IAM Permissions**
	Bucket policies	A bucket policy applies to an OBS bucket and the objects in it. A bucket owner can use bucket policies to grant IAM users or other accounts the permissions required to operate the bucket and the objects in it. Bucket policies supplement, and in many cases, replace ACLs of buckets and objects.	**Bucket Policies**
	ACLs	An access control list (ACL) defines grantees and their granted permissions. Bucket and object ACLs are associated with accounts or user groups. When you create a bucket or an object, OBS creates a default ACL that authorizes the owner full control over the bucket or object. Bucket or object owners can configure ACLs to grant basic read and write permissions to specific accounts or user groups.	**ACLs**
URL validation		URL validation protects your data in OBS from being stolen using the **Referer** field in HTTP requests. Such authorization is controlled using whitelists and blacklists.	**URL Validation**
CORS		OBS allows you to configure cross-origin resource sharing (CORS) rules on buckets to allow or forbid cross-origin requests from certain websites.	**CORS**

5.3 Data Protection

OBS takes different measures to keep data stored in OBS secure and reliable.

Table 5-2 Data protection measures

Measure	Description	Reference
Transmission encryption (HTTPS)	OBS supports HTTP and HTTPS, but HTTPS is recommended to enhance the security of data transmission.	**Constructing a Request**
Data redundancy	OBS uses the Erasure Code (EC) algorithm, instead of multiple copies, to ensure data redundancy. Compared with the multi-copy redundancy, EC delivers a higher storage space utilization while maintaining the same reliability level. When creating a bucket on OBS, you can choose a data redundancy policy. Choosing the multi-AZ storage will make your data redundantly stored in multiple AZs in the same region. If one AZ becomes unavailable, data can still be properly accessed from the other AZs. The multi-AZ storage is ideal for scenarios that demand high reliability.	**Creating a Bucket**
Data integrity verification (MD5)	During object uploads or downloads, data may become inconsistent due to network hijacking, caching, and other reasons. OBS verifies data consistency by calculating the MD5 value when data is uploaded or downloaded.	**Data Consistency Verification**
Server-side encryption	With server-side encryption enabled, objects you upload to OBS will be encrypted into ciphertext before they are stored on the server. When objects are downloaded, they will be decrypted on the server first and then returned in plaintext to you.	**Server-Side Encryption**
Cross-region replication	You can configure cross-region replication rules to automatically, asynchronously replicate data from a source bucket to a destination bucket in another region. This provides you with the capability for disaster recovery across regions, catering to your needs for remote backup.	**Cross-Region Replication**

Measure	Description	Reference
Versioning	When versioning is enabled for a bucket, OBS can keep multiple versions of an object in the bucket. That way you can quickly retrieve and restore every object version as needed, or recover data from both accidental actions and application failures.	**Versioning**
Critical operation protection	With this function enabled, the system authenticates user's identity when they perform any risky operation like deleting a bucket. This enhances the protection for your data and configuration.	**Critical Operation Protection**

5.4 Audit and Logging

Audit

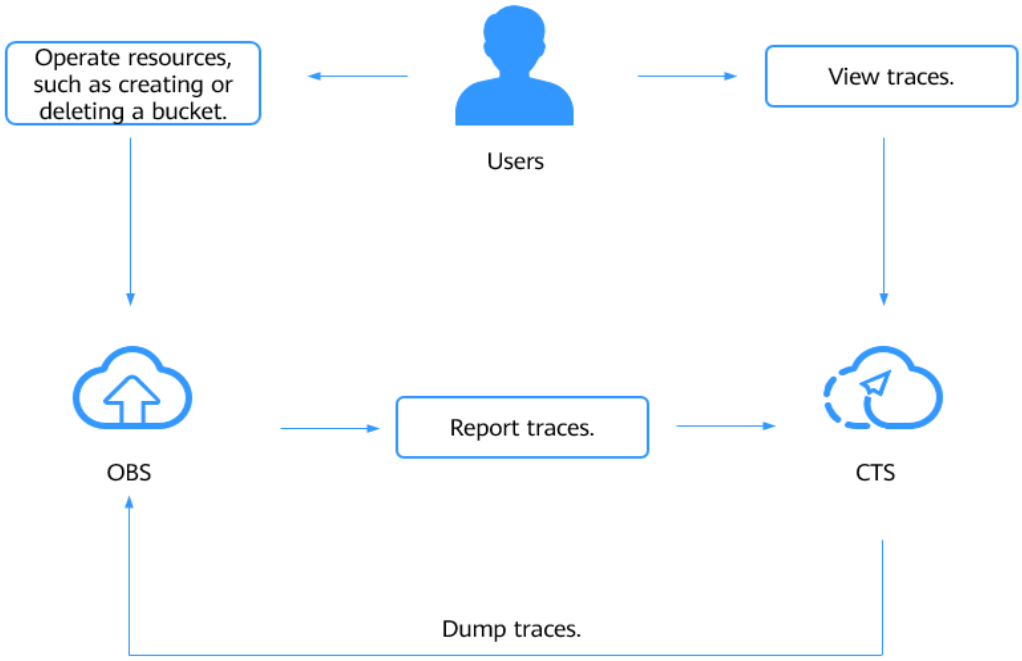
Cloud Trace Service (CTS) records operations on the cloud resources in your account. You can use the logs generated by CTS to perform security analysis, track resource changes, audit compliance, and locate faults.

After you enable CTS and configure a tracker, CTS can record management and data traces of OBS for auditing.

For details about how to enable and configure CTS, see **Enabling CTS**.

For details about OBS management and data traces that can be tracked by CTS, see **Cloud Trace Service**.

Figure 5-2 CTS



Logging

You can enable OBS logging for bucket analysis or audit. After logging is enabled for a bucket, OBS automatically logs access requests for the bucket and writes the generated log files into the specified bucket. With access logs, the bucket owner can deeply analyze the characteristics, types, or trends of requests sent to the bucket.

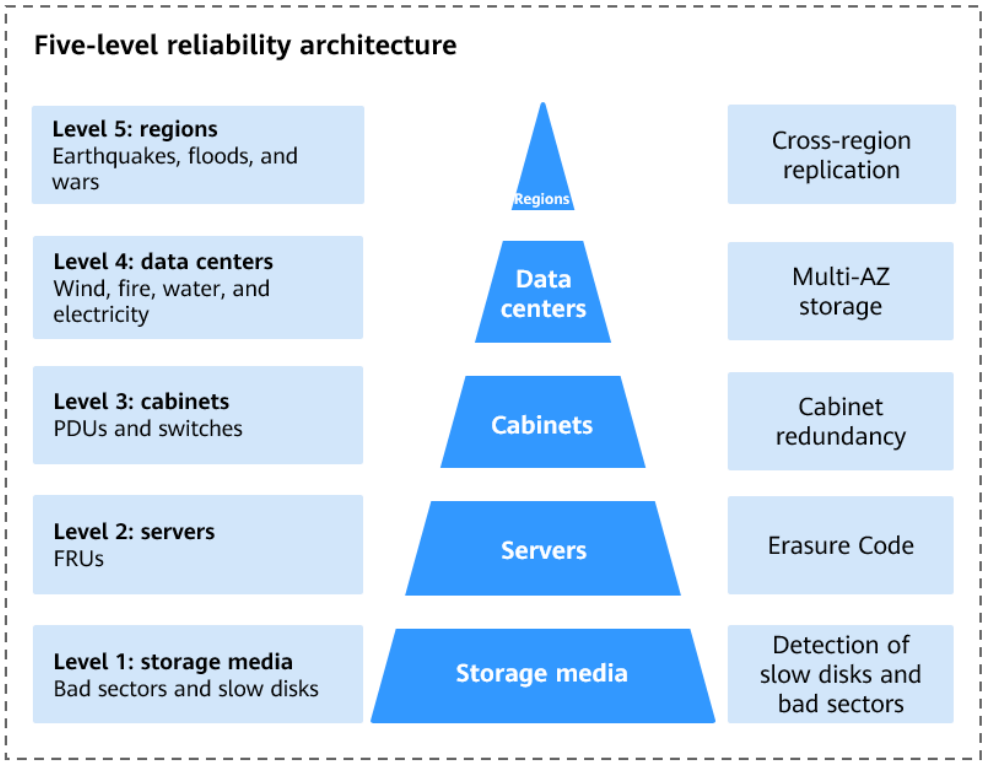
For the introduction and configuration of OBS logging, see **Logging**.

5.5 Resilience

OBS offers a five-level reliability architecture. It ensures data durability and availability by leveraging cross-region replication, disaster recovery across AZs, device and data redundancy in an AZ, and detection of slow disks and bad sectors.

OBS delivers up to 99.9999999999% (12 nines) of data durability, and up to 99.995% of data availability, far higher than a conventional architecture would offer.

Figure 5-3 Five-level reliability architecture of OBS



5.6 Risk Monitoring

OBS uses Cloud Eye to perform monitoring over resources and operations, helping you monitor your buckets and receive alarms and notifications in real time. You can get the details about requests, traffic, bandwidth, error responses, and storage usage of your buckets.

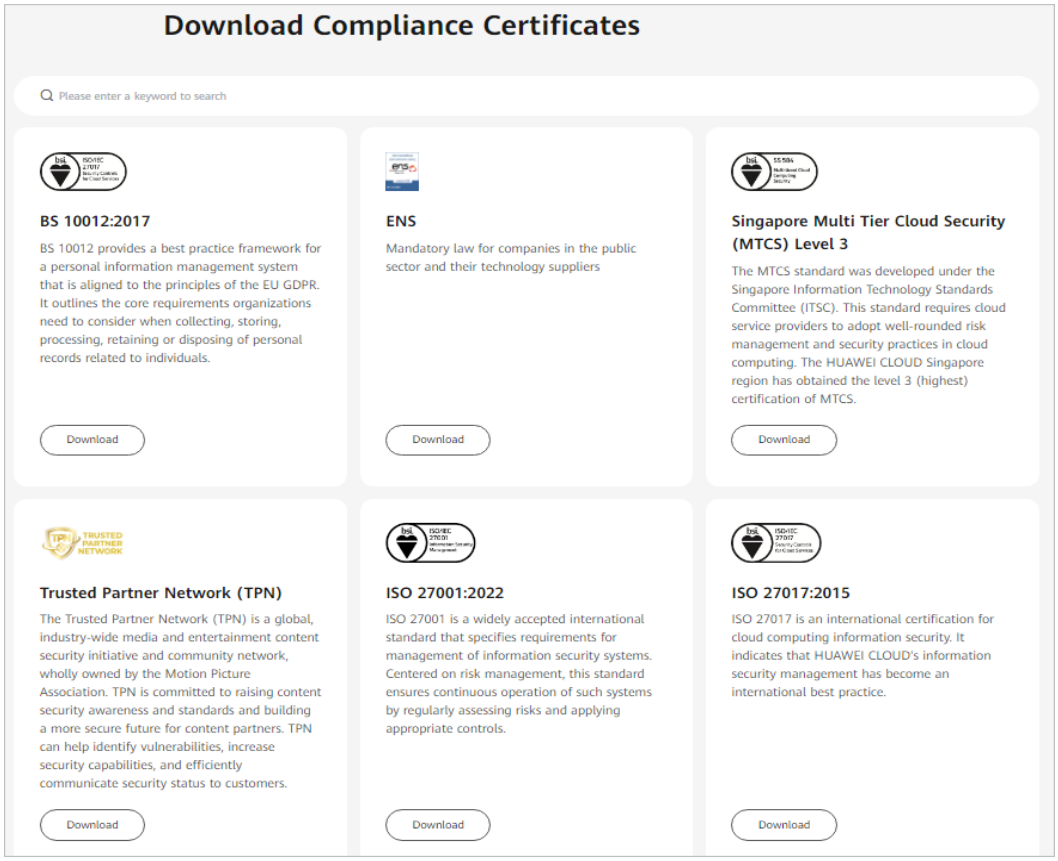
For details about OBS monitoring metrics and how to create alarm rules, see **Monitoring**.

5.7 Certificates

Compliance Certificates

Huawei Cloud services and platforms have obtained various security and compliance certifications from authoritative organizations, such as International Organization for Standardization (ISO). You can **download** them from the console.

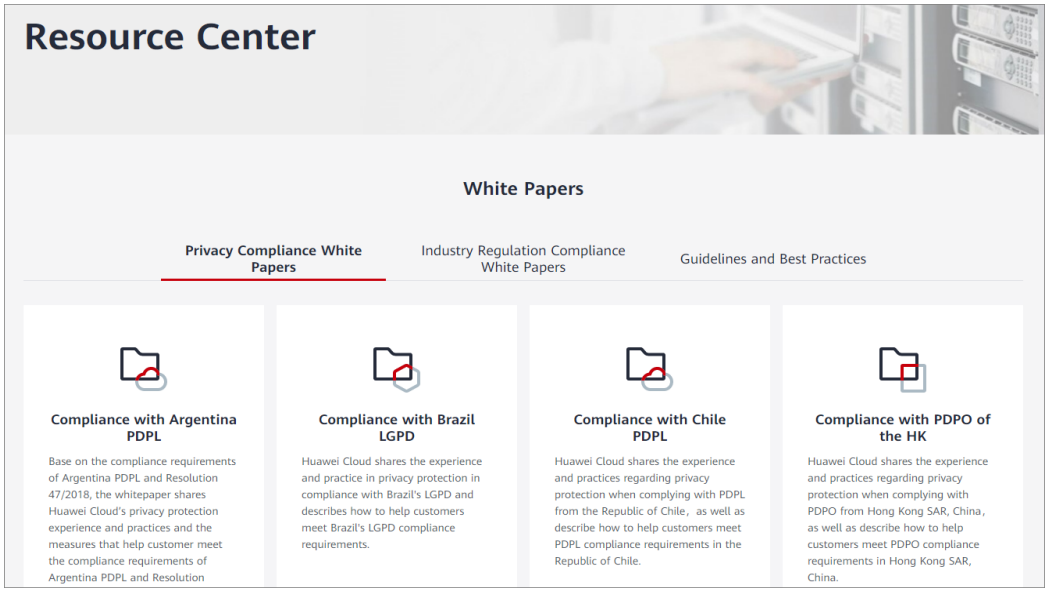
Figure 5-4 Downloading compliance certificates



Resource Center

Huawei Cloud also provides the following resources to help users meet compliance requirements. For details, see **Resource Center**.

Figure 5-5 Resource Center



6 Permissions Management

OBS Resource Permissions Management

Access to OBS buckets and objects can be controlled by IAM user permissions, bucket policies, and ACLs. This section describes how to use IAM to manage permissions.

For more information, see **OBS Permission Control**.

You can use Identity and Access Management (IAM) to manage OBS permissions and control access to your resources. IAM provides identity authentication, permissions management, and access control.

You can create IAM users for your employees, and assign permissions to these users on a principle of least privilege (PoLP) basis to control their access to specific resource types. For example, you can create IAM users for software developers and assign specific permissions to allow them to use OBS resources but prevent them from being able to delete resources or perform any high-risk operations.

If your Huawei Cloud account does not require individual IAM users for permissions management, skip this section.

IAM is a free service. You only pay for the resources in your account. For more information about IAM, see **What Is IAM?**

OBS Permissions

By default, new IAM users do not have any permissions assigned. You can assign permissions to these users by adding them to one or more groups and attaching policies or roles to the groups.

OBS is a global service deployed and accessed without specifying any physical region. OBS permissions are assigned to users in the global project, and users do not need to switch regions when accessing OBS.

You can grant users permissions by using roles or policies.

- Roles: A type of coarse-grained authorization mechanism that provides only a limited number of service-level roles. When using roles to grant permissions, you also need to assign dependency roles. However, roles are not an ideal choice for fine-grained authorization and secure access control.

- Policies: A type of fine-grained authorization mechanism that defines permissions required to perform operations on specific cloud resources under certain conditions. This mechanism allows for more flexible policy-based authorization for secure access control. For example, you can grant OBS users only the permissions for managing a certain type of OBS resources. Most policies define permissions based on APIs. For the API actions supported by OBS, see **Permissions and Supported Actions**.

NOTE

Due to data caching, a role and policy involving OBS actions will take effect 10 to 15 minutes after it is attached to a user, an enterprise project, and a user group.

Table 6-1 lists all system permissions of OBS.

Table 6-1 OBS system permissions

Role/Policy Name	Description	Type	Dependency
Tenant Administrator	Allows you to perform all operations on all services except IAM.	System-defined role	None
Tenant Guest	Allows you to perform read-only operations on all services except IAM.	System-defined role	None
OBS Administrator	Allows you to perform any operation on all OBS resources under the account.	System-defined policy	None
OBS Buckets Viewer	Allows you to list buckets, and obtain basic bucket information and bucket metadata.	System-defined role	None
OBS ReadOnlyAccess	Allows you to list buckets, obtain basic bucket information and bucket metadata, and list objects (excluding versioned objects). **NOTE** If a user with this permission fails to list objects on OBS Console, there may be multiple versions of objects in the bucket. In this case, you need to grant the user the **obs:bucket:ListBucketVersions** permission so that the user can view different versions of objects on OBS Console.	System-defined policy	None

Role/Policy Name	Description	Type	Dependency
OBS OperateAccess	Allows you to perform all operations defined in OBS ReadOnlyAccess and to perform basic object operations, such as uploading objects, downloading objects, deleting objects, and obtaining object ACLs. **NOTE** If a user with this permission fails to list objects on OBS Console, there may be multiple versions of objects in the bucket. In this case, you need to grant the user the **obs:bucket:ListBucketVersions** permission so that the user can view different versions of objects on OBS Console.	System-defined policy	None

Table 6-2 lists the common operations supported by each system-defined policy or role of OBS. Select the policies or roles as required.

Table 6-2 Permissions and the allowed operations on OBS resources

Operation	Tenant Administrator	Tenant Guest	OBS Administrator	OBS Buckets Viewer	OBS ReadOnlyAccess	OBS Operate Access
Listing buckets	Supported	Supported	Supported	Supported	Supported	Supported
Creating buckets	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Deleting buckets	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Obtaining basic bucket information	Supported	Supported	Supported	Supported	Supported	Supported
Controlling bucket access	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Managing bucket policies	Supported	Not supported	Supported	Not supported	Not supported	Not supported

Operation	Tenant Administrator	Tenant Guest	OBS Administrator	OBS Buckets Viewer	OBS ReadOnlyAccess	OBS Operate Access
Changing bucket storage classes	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Listing objects	Supported	Supported	Supported	Not supported	Supported	Supported
Listing objects with multiple versions	Supported	Supported	Supported	Not supported	Not supported	Not supported
Uploading files	Supported	Not supported	Supported	Not supported	Not supported	Supported
Creating folders	Supported	Not supported	Supported	Not supported	Not supported	Supported
Deleting objects	Supported	Not supported	Supported	Not supported	Not supported	Supported
Deleting folders	Supported	Not supported	Supported	Not supported	Not supported	Supported
Downloading objects	Supported	Supported	Supported	Not supported	Not supported	Supported
Deleting object versions	Supported	Not supported	Supported	Not supported	Not supported	Supported
Downloading object versions	Supported	Supported	Supported	Not supported	Not supported	Supported
Changing object storage classes	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Restoring objects	Supported	Not supported	Supported	Not supported	Not supported	Not supported

Operation	Tenant Administrator	Tenant Guest	OBS Administrator	OBS Buckets Viewer	OBS ReadOnlyAccess	OBS Operate Access
Undeleting objects	Supported	Not supported	Supported	Not supported	Not supported	Supported
Deleting fragments	Supported	Not supported	Supported	Not supported	Not supported	Supported
Controlling object access	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Configuring object metadata	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Obtaining object metadata	Supported	Supported	Supported	Not supported	Not supported	Supported
Managing versioning	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Managing logging	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Managing tags	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Managing lifecycle rules	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Managing static website hosting	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Managing CORS rules	Supported	Not supported	Supported	Not supported	Not supported	Not supported

Operation	Tenant Administrator	Tenant Guest	OBS Administrator	OBS Buckets Viewer	OBS ReadOnlyAccess	OBS Operate Access
Managing URL validation	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Managing domain names	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Managing cross-region replication	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Managing image processing	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Appending data to objects	Supported	Not supported	Supported	Not supported	Not supported	Supported
Configuring object ACL	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Configuring the ACL for an object version	Supported	Not supported	Supported	Not supported	Not supported	Not supported
Obtaining object ACL information	Supported	Supported	Supported	Not supported	Not supported	Supported
Obtaining the ACL of a specific object version	Supported	Supported	Supported	Not supported	Not supported	Supported
Initiating a multipart upload	Supported	Not supported	Supported	Not supported	Not supported	Supported

Operation	Tenant Administrator	Tenant Guest	OBS Administrator	OBS Buckets Viewer	OBS ReadOnlyAccess	OBS Operate Access
Listing uploaded parts	Supported	Supported	Supported	Not supported	Not supported	Supported
Canceling multipart uploads	Supported	Not supported	Supported	Not supported	Not supported	Supported
Configuring online decompression	Supported	Not supported	Not supported	Not supported	Not supported	Not supported

Permissions Required for OBS Console Operations

Table 6-3 Roles or policies that are required for performing operations on OBS Console

OBS Console Operation	Dependency	Role/Policy Required
Listing existing domain names (required when you configure a user-defined domain name or an acceleration domain name)	Domains	Domains:domains:getDetails
Configuring mirroring back-to-source rules	Object Storage Service (OBS)	• Tenant Administrator • **obs:object:PutObject** assigned by the IAM agency for OBS to pull data from its origin server • **kms:cmk:get**, **kms:cmk:list**, **kms:cmk:create**, **kms:dek:create**, **kms:dek:crypto**, and **kms:dek:crypto** configured for the agency of OBS when SSE-KMS is enabled for a bucket
Obtaining mirroring back-to-source rules	Object Storage Service (OBS)	Tenant Administrator

OBS Console Operation	Dependency	Role/Policy Required
Deleting mirroring back-to-source rules	Object Storage Service (OBS)	Tenant Administrator
Configuring online decompression policies	Object Storage Service (OBS)	Tenant Administrator
Obtaining online decompression policies	Object Storage Service (OBS)	Tenant Administrator
Deleting online decompression policies	Object Storage Service (OBS)	Tenant Administrator
Uploading or downloading encrypted objects	Key Management Service (KMS)	**kms:cmk:get**, **kms:cmk:list**, **kms:cmk:create**, **kms:dek:create**, **kms:dek:crypto**, and **kms:dek:crypto** for uploading or downloading objects encrypted with SSE-KMS

References

- **What Is IAM?**
- **IAM Basic Concepts**
- **Creating a User and Granting OBS Permissions**
- **IAM Policies and Supported Actions**

7 Notes and Constraints

This section describes the constraints on the use of OBS features.

Table 7-1 OBS use constraints

Item	Description
Bandwidth	By default, the maximum bandwidth for read/write (GET/PUT) requests of a single Huawei Cloud account is 16 Gbit/s. If the bandwidth reaches this upper limit, flow control will be triggered. If you require a bandwidth higher than 16 Gbit/s, submit **a service ticket**.
Queries per second (QPS)	Default maximum QPS allowed by a single Huawei Cloud account: • 6,000 write requests (PUT Object) per second • 10,000 read requests (GET Object) per second • 1,000 listing requests (LIST) per second **NOTE** If you use sequential prefixes (sorted by timestamp or in alphabetical order) for object naming, object access requests may be concentrated in a specific partition, resulting in access hotspots. This limits the request rate in the hot partition and increases access latency. Random prefixes are recommended for naming objects so that requests are evenly distributed across partitions, achieving horizontal expansion. For details about how to name objects with random prefixes, see **Suggestions on OBS Performance Optimization**. If you require a higher QPS, submit **a service ticket**.

Item	Description
Resource packages	• A resource package can be used only in the specified region and cannot be shared across regions. So select an appropriate region when purchasing a resource package. • OBS provides resource packages only for some billing items. For other billing items, the pay-per-use billing mode applies. For details, see **Resource Package Overview**. • Any resource usage beyond your package quotas in the current month is billed in the pay-per-use mode. A newly purchased resource package cannot cover the already generated resource usage. • A resource package must match your bucket's data redundancy policy (single-AZ storage or multi-AZ storage) and storage class (Standard, Infrequent Access, Archive), or the pay-per-use billing applies. • Your Standard storage, Archive storage, and Internet outbound traffic packages can cover the fees incurred by both your parallel file systems and buckets. The pull traffic and cross-region replication traffic packages are currently not available for parallel file systems.
Access rules	In consideration of the DNS resolution performance and reliability, OBS requires that the bucket name must precede the domain when a request carrying a bucket name is constructed to form a three-level domain name, also mentioned as virtual-hosted-style access domain name. For example, you have a bucket named **test-bucket** in the **ap-southeast-1** region, and you want to access the ACL of the **test-object** object in the bucket. The correct access URL is **https://test-bucket.obs.ap-southeast-1.myhuaweicloud.com/test-object?acl**.

Item	Description
Buckets	• Each bucket name must be unique and cannot be changed. • After you create a bucket, its namestorage redundancy policy, and region cannot be changed. • An account (including all IAM users under this account) can create a maximum of 100 buckets and parallel file systems. You can use the fine-grained access control of OBS to properly plan and use buckets. For example, you can create folders in a bucket for storing objects with different prefixes and use **fine-grained permission control** to implement permission isolation between departments. • By default, there is no limit on the storage capacity of the entire OBS system or a single bucket, and any number of objects can be stored. • A bucket can be deleted only after all objects in the bucket have been deleted. • The name of a deleted bucket can be reused for another bucket or parallel file system at least 30 minutes after the deletion.
Bucket inventories	See **Bucket Inventory Overview**.

Item	Description
Uploading objects	• OBS Console puts limits on the size and number of files you can upload. – In regions that support batch uploads, a maximum of 100 files can be uploaded at a time, with a total size of no more than 5 GB. If you upload only one file in a batch upload, it cannot exceed 5 GB in size. – In regions that do not support batch uploads, only one file can be uploaded at a time, with a size of no more than 50 MB. • If you use OBS Browser+, obsutil, an SDK, or an API, you can upload a single object of up to 48.8 TB. • Batch upload is available only when: 1. The region where the bucket resides supports batch upload. 2. The bucket version is 3.0. • If versioning is disabled for your bucket and you upload a new file with the same name as the one you previously uploaded to your bucket, the new file automatically overwrites the previous one and does not retain its ACL information. If you upload a new folder using the same name that was used with a previous folder in the bucket, the two folders will be merged, and files in the new folder will overwrite those with the same name in the previous folder. • After versioning is enabled for your bucket, if the new file you upload has the same name as the one you previously uploaded to the bucket, a new file version will be added in the bucket. • Though any UTF-8 characters can be used in object keys (object names), it is recommended that object keys be named according to the **object key naming guidelines**. These guidelines help object key names substantially meet the requirements of DNS, web security characters, XML analyzers, and other APIs.
Deleting objects	If versioning is not enabled for a bucket, deleted objects cannot be recovered.

Item	Description
Restoring Archive objects	• If an Archive object is being restored, you cannot suspend or delete the restore task. • You cannot restore an object in the **Restoring** state. • After an object is restored, an object copy in the Standard storage class will be generated. This way, there is an Archive or a Deep Archive object and a Standard object copy in the bucket at the same time. During the copy retention period, you will be billed for the storage space occupied by both the object and its copy. The Standard object copy will be automatically deleted upon its expiration.
Lifecycle management	There is no limit on the number of lifecycle rules in a bucket, but the total size of XML descriptions about all lifecycle rules in a bucket cannot exceed 20 KB.
Cross-region replication	See **Cross-Region Replication Overview**.
User-defined domain name binding	• Only buckets whose version is 3.0 or later support the binding of user-defined domain names. • By default, a maximum of 20 user-defined domain names can be bound to a 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. • By default, user-defined domain names allow requests for OBS over only HTTP. If you want to use a bound domain name to access OBS over HTTPS, configure an HTTPS certificate for the domain name on the CDN management console. For details, see **HTTPS Settings**. • A user-defined domain name can be bound to only one bucket. • Currently, the suffix of a user-defined domain name can contain 2 to 6 uppercase or lowercase letters.
Back to source	See **Back to Source Overview**.
ACLs	• A bucket ACL can have up to 100 grants. The total bucket ACL size cannot exceed 50 KB. • An object ACL can have up to 100 grants. The total object ACL size cannot exceed 50 KB.
Bucket policies	There is no limit on the number of bucket policies (statements) for a bucket, but the JSON descriptions of all bucket policies in a bucket cannot exceed 20 KB in total.

Item	Description
Parallel file systems	See the **Parallel File System Feature Guide**.
Image processing	See the **Image Processing Feature Guide**.

8 Related Services

Table 8-1 Related services

Interactive Function	Related Service	Reference
Migrate data to OBS using the related services.	Direct Connect	**Using a Direct Connect Connection to Migrate Data**
Access OBS from ECS over the intranet of Huawei Cloud.	Elastic Cloud Server (ECS)	**Accessing OBS over Intranet**
IAM provides the following functions: • User identity authentication • IAM user permission control • IAM agency configuration	Identity and Access Management (IAM)	**Permissions Management** **Configuring User Permissions**
Cloud Eye monitors OBS buckets, to collect statistics about the upload traffic, download traffic, number of GET and PUT requests, the average TTFB of GET requests, and the number of 4*xx* and 5*xx* errors.	Cloud Eye	**OBS Monitoring Metrics on Cloud Eye**

Interactive Function	Related Service	Reference
CTS collects records of operations on OBS resources, facilitating querying, audits, and backtracking.	Cloud Trace Service (CTS)	**Cloud Trace Service**
Tags are used to identify and organize buckets in OBS.	Tag Management Service (TMS)	**Tags**
KMS encrypts files uploaded to the OBS.	Data Encryption Workshop (DEW)	**Server-Side Encryption**
CDN accelerates the customized domain names bound to OBS buckets.	Content Delivery Network (CDN)	**User-Defined Domain Name Binding**
DNS resolves domain names configured for static website hosting in OBS.	Domain Name Service (DNS)	**Using a User-Defined Domain Name to Host a Static Website** **User-Defined Domain Name Binding**

OBS can be used as the storage resource pool for other cloud services such as Image Management Service (IMS) and Cloud Trace Service (CTS).

9 Basic Concepts

9.1 Objects

Objects are basic units stored in OBS. An object contains both data and the metadata that describes data attributes. Data uploaded to OBS is stored in buckets as objects.

An object consists of the following:

- A key that specifies the name of an object. An object key is a UTF-8 string up to 1,024 characters long. Each object is uniquely identified by a key within a bucket.
- Metadata that describes an object. The metadata is a set of key-value pairs that are assigned to objects stored in OBS. There are two types of metadata: system-defined metadata and custom metadata.
  - System-defined metadata is automatically assigned by OBS for processing objects. Such metadata includes Date, Content-Length, Last-Modified, ETag, and more.
  - You can specify custom metadata to describe the object when you upload an object to OBS.
- Data that refers to the content of an object.

Generally, objects are managed as files. However, OBS is an object-based storage service and there is no concept of files and folders. For easy data management, OBS provides a method to simulate folders. By adding a slash (/) to an object name, for example, **test/123.jpg**, you can specify **test** as a folder and **123.jpg** as the name of a file in the **test** folder. The key of the object is **test/123.jpg**.

When uploading an object, you can specify a storage class for it. If you do not specify a storage class, the object inherits the storage class of the bucket. You can also change the storage class of an existing object in a bucket.

On OBS Console and OBS Browser+, you can use folders the same way you use them in a file system.

For details about object operations, see **Managing Objects**.

9.2 Buckets

Buckets are containers for storing objects. OBS provides flat storage in the form of buckets and objects. Unlike the conventional multi-layer directory structure of file systems, all objects in a bucket are stored at the same logical layer.

Each bucket has its own attributes, such as access permissions, storage class, and the region. You can specify these attributes when creating buckets. You can also configure advanced attributes to meet storage requirements in different scenarios.

OBS provides the following storage classes for buckets: Standard, Infrequent Access, Archive, and Deep Archive (under limited beta testing). With support for these storage classes, OBS caters to diverse storage performance and cost requirements. When creating a bucket, you can specify a storage class for it, which can be changed later.

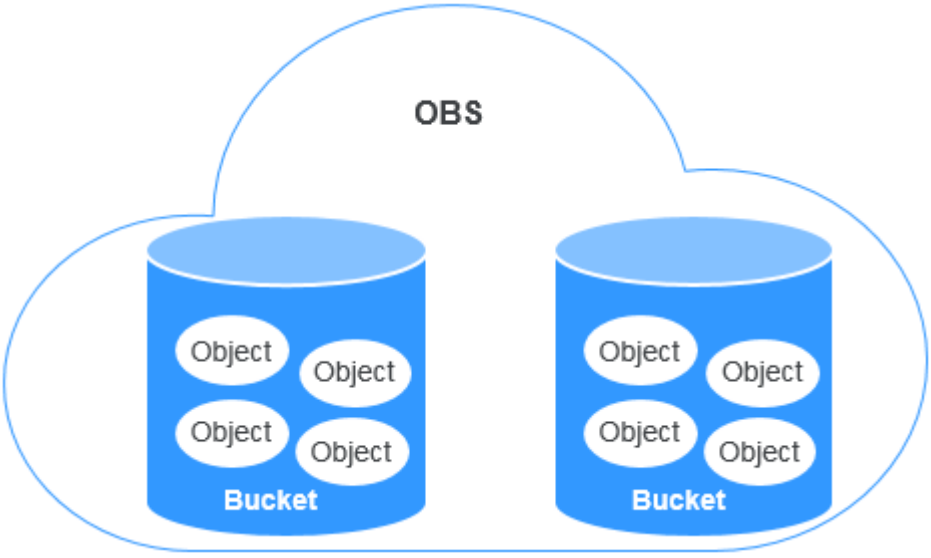
Each bucket name in OBS is globally unique and cannot be changed after the bucket is created. The region where a bucket resides cannot be changed once the bucket is created. When you create a bucket, OBS creates a default access control list (ACL) that grants users permissions (such as read and write permissions) on the bucket. Only authorized users can perform operations such as creating, deleting, viewing, and configuring buckets.

An account (including all IAM users under this account) can create a maximum of 100 buckets and parallel file systems. There is no limit on the number and total size of objects in a bucket.

OBS adopts the REST architectural style, and is based on HTTP and HTTPS. You can use URLs to locate resources.

Figure 9-1 illustrates the relationship between buckets and objects in OBS.

Figure 9-1 Relationship between objects and buckets



For details about bucket operations, see **Managing Buckets**.

9.3 Parallel File System

Parallel File System (PFS), a sub-product of OBS, is a high-performance file system, with only milliseconds of access latency in the case of suitable networking and computing packages. OBS PFS can support a TB/s-level bandwidth and handle millions of IOPS on the storage side, making it ideal for processing high-performance computing (HPC) workloads, especially in big data scenarios.

For details about PFS, see the **Parallel File System Feature Guide**.

9.4 Access Keys (AK/SK)

OBS uses access keys to authenticate the identity of a request sender.

Access keys comprise two parts: an access key ID (AK) and a secret access key (SK). They are long-term identity credentials for you to **sign requests for APIs**. AKs are used together with SKs to sign requests cryptographically, ensuring that the requests are confidential, complete, and correct.

When you use OBS APIs for secondary development and use an AK and SK pair for authentication, the signature must be calculated based on the algorithm defined by OBS and added to the request.

The authentication can be based on a permanent AK and SK pair, or based on a temporary AK/SK pair and security token.

Permanent AK/SK Pairs

You can create a pair of permanent AK and SK on the **My Credentials** page. For details, see **Obtaining Access Keys (AK and SK)**.

- Access key ID (AK): It is a unique identifier associated with a secret access key and is used to identify the sender of a request.
- Secret access key (SK): It is used in combination with the access key ID to sign requests. It can prevent requests from being tampered with and ensures the confidentiality and integrity of the requests.

Temporary AK/SK Pairs

A temporary AK/SK pair and security token assigned by OBS comply with the principle of least privilege and are for temporarily accessing OBS. They are valid from 15 minutes to 24 hours, and need to be obtained again once they expire. If the security token is missing from your request, a 403 error will be returned.

- Temporary access key ID (AK): It is a unique identifier associated with a temporary secret access key and is used to identify the sender of a request.
- Temporary secret access key (SK): It is used in combination with the temporary access key ID to sign requests. It can prevent requests from being tampered with and ensures the confidentiality and integrity of the requests.
- Security token: It is used together with the temporary AK and SK to access all resources of a specified account.

When using the following tools to access OBS resources, you need to use the AK/SK pair for security authentication.

Table 9-1 OBS resource management tools

Tool	AK/SK Configuration
OBS Browser+	Configure the AK and SK during login account configuration. For details, see **Logging In to OBS Browser+**.
obsutil	Configure the AK and SK when initializing the configuration. For details, see **Performing the Initial Configuration**.
obsfs	Configure the AK and SK when initializing the configuration. For details, see **Initializing obsfs**.
SDKs	Configure the AK and SK in the initialization phase. For details, see the **SDK Reference**.
APIs	Add the AK/SK pair to the request when computing the signature. For details, see **User Signature Authentication**.

References

Obtaining Permanent Access Keys

Obtaining a Temporary Access Key and Security Token

9.5 Endpoints and Domain Names

Endpoint: OBS provides an endpoint for each region. An endpoint is considered a domain name to access OBS in a region and is used to process requests of that region. For details about regions and endpoints, see **Regions and Endpoints**.

Bucket domain name: Each bucket in OBS has a domain name. A domain name is the address of a bucket and can be used to access the bucket over the Internet. It is applicable to cloud application development and data sharing.

A bucket domain name is in the format of *BucketName.Endpoint*, where *BucketName* indicates the name of the bucket, and *Endpoint* indicates the domain name of the region where the bucket is located.

Table 9-2 lists the bucket domain name and other domain names in OBS, including their structure and protocols.

Table 9-2 OBS domain names

Type	Structure	Description	Protocol
Regional domain name	[Structure] **Endpoint** [Example] obs.ap-southeast-1.myhuaweicloud.com	Each region has an endpoint, which is the domain name of the region. For regions and endpoints, see **Regions and Endpoints**. Each region corresponds to an OBS endpoint that keeps unchanged on both the internal and external networks. After **access over an intranet** is configured, you can access OBS over an internal network.	HTTPS HTTP
Bucket domain name	[Structure] **BucketName.Endpoint** [Example] bucketname.obs.ap-southeast-1.myhuaweicloud.com	After a bucket is created, you can use the domain name to access the bucket. You can compose the domain name according to the structure of bucket domain names, or you can obtain it from basic information of the bucket on OBS Console or OBS Browser+.	HTTPS HTTP
Object domain name	[Structure] **BucketName.Endpoint/ObjectName** [Example] bucketname.obs.ap-southeast-1.myhuaweicloud.com/object.txt	After an object is uploaded to a bucket, you can use the object domain name to access the object. You can spell out the domain name according to the structure of object domain names, or you can obtain it from the object details on OBS Console or OBS Browser+. Alternatively, you can call the GetObjectUrl API through the SDK to obtain the object domain name.	HTTPS HTTP

Type	Structure	Description	Protocol
Static website domain name	[Structure] **BucketName.obs-website.Endpoint** [Example] bucketname.obs-website.ap-southeast-1.myhuaweicloud.com	A static website domain name is a bucket domain name when the bucket is configured to host a static website.	HTTPS HTTP
User-defined domain name	Self-owned domain name registered with a domain name provider	You can bind a user domain name to a bucket so that you can access the bucket through the user domain name.	HTTP

9.6 Region and AZ

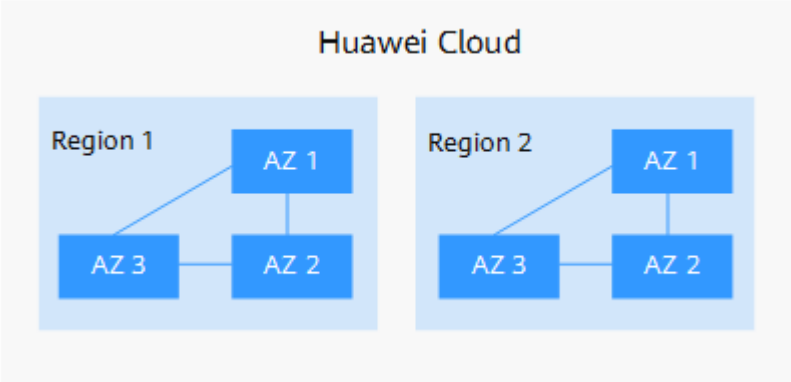
Concept

A region and availability zone (AZ) identify the location of a data center. You can create resources in a specific region and AZ.

- Regions are classified based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. Regions are classified as universal regions and dedicated regions. A universal region provides universal cloud services for common tenants. A dedicated region provides services of the same type or only provides services for specific tenants.
- An AZ contains one or more physical data centers. Each AZ has independent cooling, fire extinguishing, moisture-proofing, and electricity facilities. Within an AZ, computing, network, storage, and other resources are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow you to build cross-AZ high-availability systems.

Figure 9-2 shows the relationship between regions and AZs.

Figure 9-2 Regions and AZs



Huawei Cloud provides services in many regions around the world. You can select a region and AZ according to your requirement. For more information, see **Huawei Cloud Global Regions**.

How Do I Select a Region?

When selecting a region, consider the following factors:

- Location

  Select a region close to you or your target users. This reduces network latency and improves access speed. However, Chinese mainland regions provide the same infrastructure, BGP network quality, as well as resource operations and configurations. If you or your target users are in the Chinese mainland, you do not need to consider differences in network latency when selecting a region.

  - If you or your target users are in the Asia Pacific region (excluding the Chinese mainland), select regions such as AP-Bangkok and AP-Singapore.
  - If you or your target users are in Africa, select the AF-Johannesburg region.
  - If you or your target users are in Europe, select the EU-Paris region.
- Resource prices

  Resource prices may vary depending on different regions. For details, see **Product Pricing Details**.

How Do I Select an AZ?

When determining whether to deploy resources in the same AZ, consider your applications' requirements for disaster recovery (DR) and network latency.

- For high DR capability, deploy resources in different AZs in the same region.
- For low network latency, deploy resources in the same AZ.

Regions and Endpoints

Before using an API to call resources, you must specify its region and endpoint. For details about regions and endpoints, see **Regions and Endpoints**.

