Data Accelerator Goose FileSystem

GooseFSx

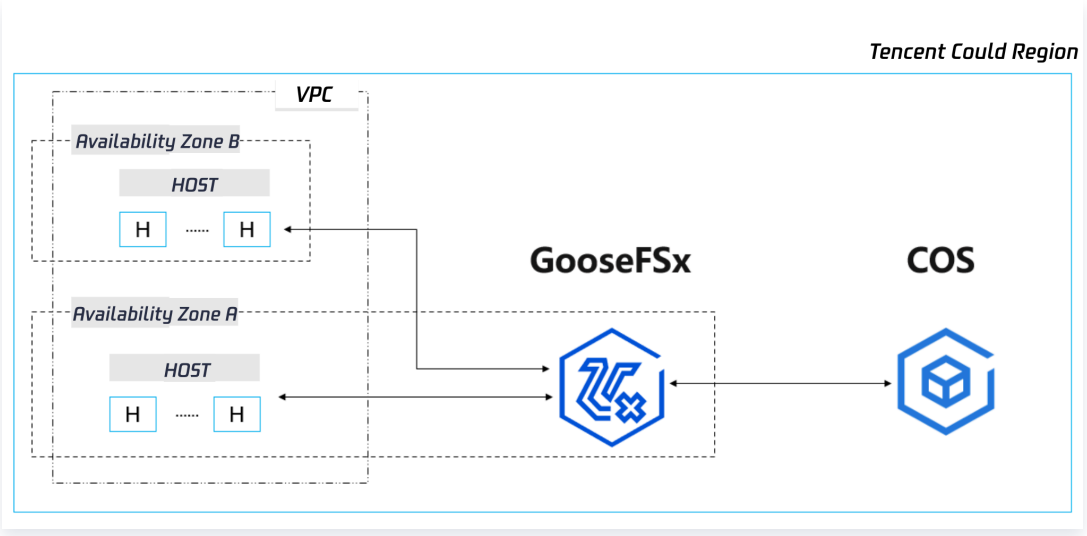
Product Documentation

Contents

GooseFSx
Product Introduction
Product Overview

Last updated：2025-07-17 17:16:39

GooseFSx (Data Accelerator Goose FileSystem extreme, GooseFSx) is a high-performance POSIX-compliant Data Accelerator introduced by Tencent Cloud. It can accelerate Cloud Object Storage (COS), deliver ultra-high performance and ultra-low latency for high-performance computing workloads, and enable users to flexibly manage cold and hot Data. Suitable for business scenarios such as high-performance computing, autonomous driving, and machine learning. GooseFSx is a fully hosted service that is easy to use, offers hourly billing, allows resources to be released after use, and persists Data through COS. As shown in the figure below:



GooseFSx can be auto-mounted as a local directory on the host (this directory is called the mount directory). When the host accesses the GooseFSx mount directory, it is the same as accessing GooseFSx on the local file system.
GooseFSx accelerates COS, enabling direct data loading from COS so that hosts can access cached data in GooseFSx at high speed; it can sink computation results generated on GooseFSx to COS for persistent, low-cost storage.
You can create and manage GooseFSx via Tencent Cloud API Platform or Tencent Cloud Console.

Super Directory

A super directory (Fileset) is a directory with features such as quota, file record deletion, and QoS. It is a sub-file system within the GooseFSx file system, possessing independent storage space and performance management capability. Compared with ordinary directories, super directories are more powerful, as shown in the table below.

Comparison Item	Super Directory	Ordinary Directory
Directory quota	Supported	Not supported
Delete file records	Supported	Not supported
Parent directory	Root directory or ordinary directory	Root directory or ordinary directory or super directory
Directory content	Ordinary directory or file	Ordinary directory or file or super directory
Quantity	A single instance supports a certain number of super directories. For details, see Specifications and Usage Limits of a Super Directory.	Single instance supports 10 billion-level ordinary directories

Super directory, used to achieve refined resource management and policy control within a single GooseFSx file system. Typical usage is to assign different super directories to different departments, users, or applications to prevent data mixing while sharing the same GooseFSx instance; assign different capacity quotas or file quotas to each super directory to prevent a single department, user, or application from exhausting global resources; record access logs based on super directory granularity to meet compliance requirements.

> Note:
> Specifications and usage limitations of the super directory. For details, see Specifications and Usage Limits of a Super Directory.

Quota

Quota is used to set storage capacity limits and file quantity limits for different super directories. It supports GooseFSx in allocating and managing storage resources across multiple departments, users, or application scenarios, facilitating refined management of resources and preventing a single department, user, or application from exhausting global resources.

- Capacity quota, the maximum writable capacity for a super directory. Once the quota upper limit is reached, no new data can be written.
- File number quota, the maximum number of files and directories that can be written to a super directory. Once the quota upper limit is reached, no new data can be written.

When a quota is set, during each write I/O operation, the used capacity and number of files of the quota are accumulated with the capacity and number of files added by this operation. The total usage is checked to see if it exceeds the quota. If the quota is exceeded, the write I/O operation fails with an error: Disk quota exceeded. If the quota is not exceeded, the operation is allowed. This ensures that usage always stays within the quota range and prevents exceeding the limit. It is recommended to reasonably set super directory quotas and configure alarm policies to receive timely alerts when capacity, number of files, etc., are close to the quota.

To use quotas for implementing refined resource management within a single GooseFSx file system. A typical usage is to assign different quotas to different departments, users, or applications, i.e., assigning different capacity quotas or file number quotas to each super directory, with different departments, users, or applications corresponding to different super directories.

> Note:
> Quota specifications and usage restrictions. For details, see Specifications and Usage Limits of a Super Directory.

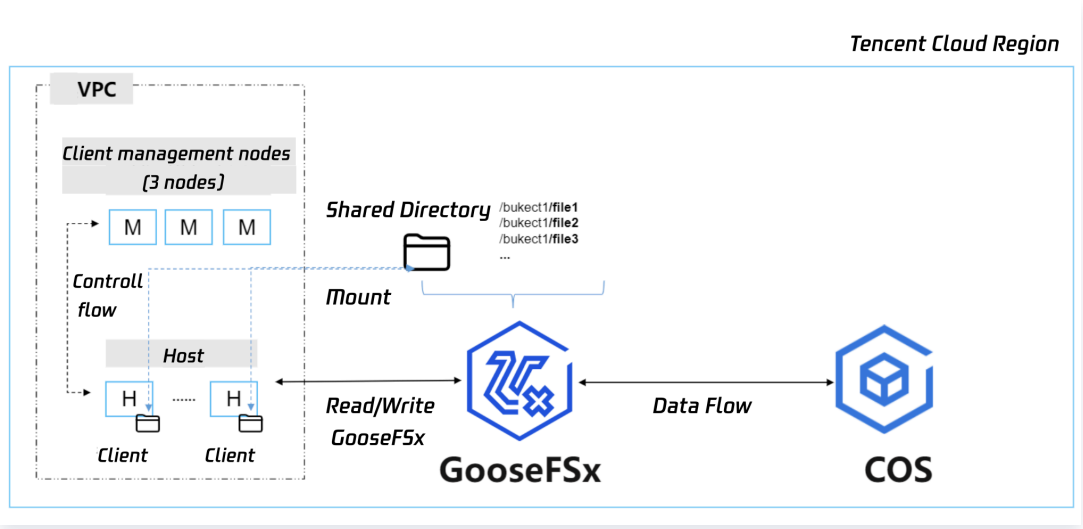
Delete File Records

Delete file records, which are operation records of deleting files or directories, including information such as path, filename, and time, is a feature provided by GooseFSx for auditing deletion operations.
Super directories have the ability to turn on or off delete file records, while ordinary directories do not support this. Once super directories enable delete file records, they will log all data deletion actions under the super directory.
Delete file records, used for quickly identifying and fixing issues based on delete file records in scenarios like accidental data deletion.

> Note:
> Specifications and usage limits for deleting file records. For details, see Specifications and Usage Limits of a Super Directory.

POSIX Client

The GooseFSx POSIX client is a host where the GooseFSx POSIX client software is deployed and GooseFSx is mounted as a local directory. GooseFSx can automatically deploy a host as a POSIX client. The implementation principle is to automatically deploy the GooseFSx POSIX client software on the host and mount GooseFSx as the host's local directory. Accessing the GooseFSx mounted directory from a POSIX client is the same as accessing GooseFSx in the local file system. As shown below:



When creating a GooseFSx instance, three Tencent Cloud Cloud server (CVM) instances will be automatically created under your Tencent Cloud account. The instance specifications are not less than 4C8G, deployed as POSIX client management nodes. The POSIX client management nodes are created and terminated along with the GooseFSx instance, and you do not need to manage them.

> **Note:**
> Note: Do not destroy or change the POSIX client management node, otherwise it will cause POSIX client working exception.

Configure 3 POSIX client management nodes for GooseFSx to ensure high availability. In extreme cases, if a POSIX client management node fails, it may result in the POSIX client being unable to access GooseFSx normally, but will not affect the data already written to GooseFSx. Once the POSIX client management node recovers from the fault, the POSIX client can normally access GooseFSx.
Manage POSIX client nodes, manage the deployment and deletion of POSIX clients. POSIX client management nodes do not participate in data streams, do not access data, and do not store data.
POSIX client management nodes need to interconnect with POSIX clients on the management plane. To ensure the information security of your POSIX clients, deploy POSIX client management nodes in your VPC to guarantee that management traffic does not flow out of your VPC.
GooseFSx supports automatic addition of POSIX clients, automatically adding specified hosts as GooseFSx POSIX clients, eliminating the need to manually execute commands step by step to add hosts as GooseFSx POSIX clients. GooseFSx can add POSIX clients in batches, automatically adding multiple POSIX clients at once. GooseFSx can automatically delete POSIX clients and can delete multiple POSIX clients in batches.
GooseFSx implements centralized management of POSIX clients, enables real-time query of POSIX clients using GooseFSx, allows timely deletion of POSIX clients no longer using GooseFSx, and supports immediate addition of POSIX clients requiring GooseFSx.
You can create and manage POSIX clients through the Tencent Cloud API Platform or the Tencent Cloud Console.

Data Flow

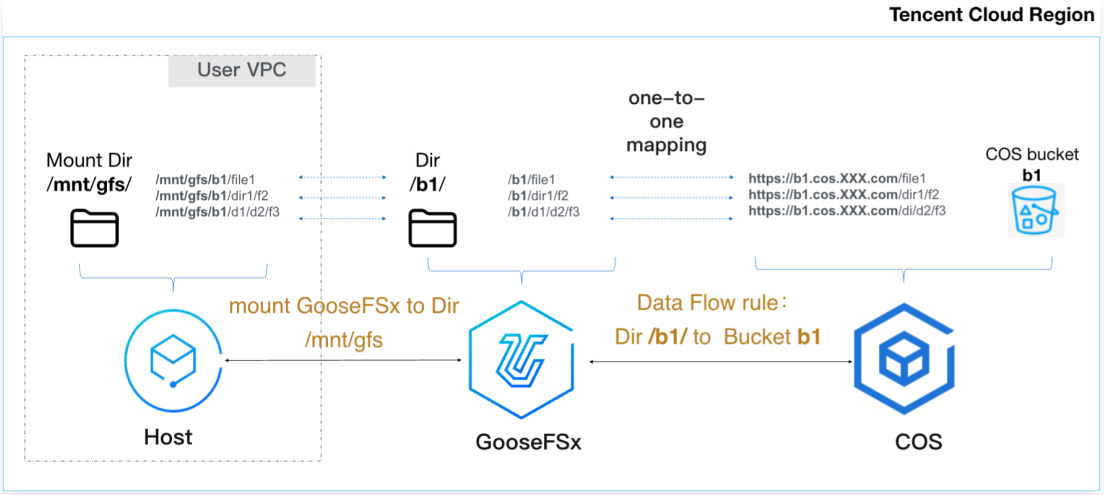
Data flow refers to the on-demand movement of data between the data accelerator GooseFSx and the object storage COS. GooseFSx preheats the required data from COS so that hosts can access the high-speed GooseFSx, achieving acceleration for COS. GooseFSx sinks the generated computation results to COS for persistent, low-cost storage or shares the computation results through COS's Internet distribution capability.

GooseFSx can simultaneously flow data with multiple intra-region or cross-regional COS storage buckets. For data flow with cross-regional COS storage buckets, it is recommended to use COS global acceleration domain names for fast speed, low cost, and straightforward usage. Alternatively, you can use a network connection interoperable with cross-regional COS storage buckets, which requires manual configuration and management.

Comparison Item	COS Global Acceleration Domain	Custom Access Point
How It Works	With the load balancing system leveraging Tencent's global traffic scheduling, utilizing globally distributed Cloud Data Centers, selecting the best network access linkage, and achieving proximity access for requests.	Users manually adopt cloud networking and other interconnection technologies to establish network connectivity between cross-regional COS buckets and associated VPCs.
rate	High flow performance. For details, please contact the Storage Architect.	Indeterminacy, dependent on cross-region interconnection network.
Costs	Low cost.	It is advisable to reuse the existing cross-region interconnection network. If a separate cross-region interconnection network is created, the cost will be high.
Usability	Fully automated, easy to use, only requires a COS storage bucket to enable global acceleration. GooseFSx will automatically obtain and use the global acceleration domain name of the COS storage bucket.	Manually manage, for example: create or configure a cross-region interconnection network, map a cross-region COS storage bucket to the IP of an associated VPC (the user VPC associated with GooseFSx), and manage the connectivity and rate of the cross-region interconnection network.

Data flow maintains a one-to-one correspondence between the data in GooseFSx directories and the data in COS storage buckets. During preheating, COS storage bucket objects are converted into files and

preheated into the corresponding GooseFSx directories. During settlement, files in GooseFSx directories are converted into objects and settled into the corresponding paths of COS storage buckets. As shown in the following figure:



Data Flow Rules

Data flow rules are the regulations for data flow between GooseFSx directories and COS storage buckets. Create data flow rules. Determine data flow rules between a certain directory of GooseFSx data accelerator and a certain bucket of COS, such as data flow mode and data flow bandwidth. GooseFSx can create multiple data flow rules and transfer data with multiple COS buckets simultaneously. For details, see Usage Limits of Data Flow.

Data flow modes, as shown in the table below.

Data Flow Mode	Description	Use Cases
manual mode	Manual mode is data pre-warming or data settlement triggered manually and executed automatically by the system.	Host high-performance access GooseFSx, avoid synchronous access to COS slowing down performance. Notes: Data pre-warming needs to be done in advance for GooseFSx.

Before creating data flow rules, please ensure the data flow bandwidth is not 0; conversely, if you need to reduce the data flow bandwidth to 0, please delete all data flow rules first.
Delete data flow rules Delete data flow rules between a certain directory of GooseFSx data accelerator and a certain bucket of COS Delete data flow (data pre-warming and data settlement) corresponding to

the deleted flow rules After deleting the flow rules, data flow (data pre-warming and data settlement) cannot be executed on the GooseFSx directory.

Data Flow Bandwidth

Data flow bandwidth is the bandwidth for data flow between GooseFSx data accelerator and COS. The higher the GooseFSx data flow bandwidth, the faster GooseFSx performs data flow; conversely, the slower it is. To match business requirements, you can expand or reduce data flow bandwidth at any time.

Example

The step length of data flow bandwidth is 600MB/s. After users purchase N steps, the data flow bandwidth will increase by N * 600MB/s. Conversely, it will decrease by N * 600MB/s.
The maximum value of data flow bandwidth is approximately equal to the bandwidth of the GooseFSx instance. Because data flow transmits data between GooseFSx and COS buckets, the data flow bandwidth will not exceed the maximum bandwidth of the GooseFSx instance and the maximum bandwidth of the COS bucket. Meanwhile, a GooseFSx instance can simultaneously flow data with multiple COS buckets. Therefore, the maximum value of data flow bandwidth is approximately equal to the bandwidth of the GooseFSx instance.

Status of data flow bandwidth: to be scaled out, scaling out, scaling in, running.

Status	Description	Data Flow
to be scaled out	Data flow bandwidth is 0. Scale it out first; otherwise, data cannot flow.	Cannot create data flow rules. Data cannot flow.
Expanding Shrinking	Adjusting data flow bandwidth. Cannot initiate scaling-out or scaling-in of data flow bandwidth again.	Data flow rules can be created, data can flow. For details, contact the Storage Architect.
Running	Functioning properly. Data flow bandwidth can be scaled-out or scaled-in.	Data flow rules can be created, data can flow.

> Note:
> Data flow bandwidth, which requires users to pay for use. For details, see Purchase Guide. During the process of using data flow, incurred fees may occur. For details, see Data Flow Costs.

Data Flow Cost

Data flow will incur GooseFSx data flow bandwidth fees; if global acceleration is enabled to access cross-regional COS buckets, global acceleration traffic fees will be incurred and charged by the COS product;

reading from COS will generate COS request fees, which are charged by the COS product.

Scenario	Fees	Description
Flow data with COS buckets in the same region	GooseFSx data flow bandwidth fee.	GooseFSx data flow bandwidth cost, charged by the GooseFSx product.
	Fees for COS requests.	COS request fees, collected by the COS product.
Flow data with COS buckets across regions	GooseFSx data flow bandwidth fee.	Data flow bandwidth cost for GooseFSx, charged by the GooseFSx product.
	COS global acceleration traffic cost.	COS global acceleration traffic cost, charged by the COS product.
	Fees for COS requests.	Fees for COS requests, charged by the COS product.

Data Pre-Warming

Data preheating is the process of preheating data from a COS storage bucket to a GooseFSx directory, automatically, completely, and incrementally preheating the specified data (entire directory, a subdirectory, or a list) into GooseFSx.
Prerequisite for data preheating: The corresponding data flow rule has been created. For details, see Data Flow Rule.
Data preheating maintains a one-to-one correspondence between GooseFSx data and COS storage bucket data, keeping directory and file permissions unchanged. For example, a COS storage bucket object: H1/big/test.dat, is preheated into a GooseFSx directory file: H1/big/test.dat.
Data preheating includes: single task and periodic task.

- One-time task, initiated by the user and executed immediately by the system.
- Periodic task, configured by the user with a periodic policy and automatically executed periodically by the system. For details, see Data Flow Periodic Policy.

Data Settlement

Data settlement involves settling GooseFSx directory data into a COS storage bucket, automatically, completely, and incrementally settling your specified data (entire directory, a certain subdirectory, or list) into COS.
Prerequisite for data settlement: Corresponding data flow rules have been created. For details, see Data Flow Rules.
Data settlement ensures a one-to-one correspondence between GooseFSx data and COS storage bucket data, while keeping directory and file permissions unchanged. For example, a file in the GooseFSx

directory: H1/big/test.dat, is settled as an object in the COS storage bucket: H1/big/test.dat, and directories are settled as special objects.
Data settlement is divided into: data settlement, data settlement and deletion of local data, and data settlement .

Type	Description	Use Cases
data settlement	Settle GooseFSx data to COS. GooseFSx and COS each have a piece of data.	When creating data flow rules, select manual mode for data flow mode. Settle calculation results to COS for long-term storage or distribute them over the Internet.
Settle data and delete local data	Settle GooseFSx data to COS, then delete GooseFSx data. GooseFSx does not have this data, COS has a piece of data.	When creating data flow rules, select manual mode for data flow mode. Settle calculation results to COS for long-term storage or distribute them over the Internet, then delete GooseFSx data and release GooseFSx space.

Data settlement includes: single task and periodic task.

- One-time task, initiated by the user and executed immediately by the system.
- Periodic task, configured by the user with a periodic policy and automatically executed periodically by the system. For details, see Data Flow Periodic Policy.

Data Flow Periodic Policy

Data flow periodic policy (abbreviated as periodic policy), used for periodically triggering data flow tasks, such as the periodic policy for settling COS every hour, triggering a settlement task every 1 hour, and settling specified datasets from GooseFSx to corresponding COS buckets. Suitable for scenarios such as regularly synchronizing data with COS buckets and promptly preheating data from COS buckets.
Data flow periodic policy, supports configuration of multiple periodic policies:

- Per-hour periodic policy: Triggers a data flow task at the specified minute every hour.
- Daily periodic policy: Triggers a data flow task at the specified hour every day.
- Weekly periodic policy: Triggers a data flow task on the specified day every week.
- Monthly periodic policy: Triggers a data flow task on the specified day every month.

Mount Multiple Cloud Disks

GooseFSx supports multi-mounting of cloud disks, allowing a single cloud disk to be mounted on multiple GooseFSx storage nodes simultaneously. Multi-mounting tolerates simultaneous failures of any number of nodes without interrupting the business or causing data loss, significantly improving product availability (from 99.9% to 99.9999999%).

The working mechanism of multi-mounting cloud disks is as follows:

- Under normal circumstances, the first node reads and writes to this cloud disk, while other nodes do not. This is the same as the single-mount working mechanism of the cloud disk.
- In case of an exception, the first node fails and will automatically switch to the second node for reading and writing this cloud disk. The switching process is as follows:
  - Automatic switching, host unaware, host requests during the switching process are sent to the second node via a retry mechanism.
  - Instant switching, because multiple nodes share the cloud disk, no data synchronization is required during the switching process, and the switch is completed in seconds.
  - The second node fails and will automatically switch to the third node, tolerating simultaneous faults of multiple nodes, thereby achieving extremely high availability.

Product type

Last updated：2025-07-17 17:16:39

The data accelerator GooseFSx types include GooseFSx C50, GooseFSx C60, and GooseFSx C70. This document introduces the GooseFSx data accelerator types.

GooseFSx Types

GooseFSx C50 Type

C50 has two specifications: C50.T3 and C50.T12. Except for capacity, the two specifications are the same in performance, price, features, and robustness.

Features

- High performance, performance scales linearly with capacity, 100MB/s – 120MB/s read bandwidth per TiB, 100MB/s – 120MB/s write bandwidth per TiB, 10,000 read IOPS per TiB, 10,000 write IOPS per TiB.
- Starting capacity, step length and usage recommendations as shown in the table below:

Specification	Start Capacity	Step Length Capacity	Usage recommendations
C50.T12	36TiB	12TiB	Data volume exceeds 200TiB or grows fast.
C50.T3	9TiB	3TiB	Data volume within 200TiB

If you need to further enhance performance, or if purchase failed due to resource insufficient in AZ, or have another requirement, you can contact Storage Architect or consult Online Customer Service to ask for help.

Applicable Scenarios

- AI training and inference scenarios: The data accelerator GooseFSx C50 accelerates AI training and reasoning performance. It enables seamless data flow with the data lake COS, providing immediate access to COS model data for AI computing power. This avoids expensive computing resources waiting for data and allows training results to be stored in COS as needed, reducing overall storage costs.
- Gene analysis scenario: The data accelerator GooseFSx C50 accelerates high-performance storage requirements for gene sequencing and local alignment stages. It enables free data flow with the data lake COS, implements immediate access to COS samples on the Omics platform, achieves automatic archiving of gene analysis results to COS, and delivers them to end users via COS Internet distribution capability.

- Autonomous driving scenario: The data accelerator GooseFSx C50, combined with data lakes COS and data migration, forms an end-to-end autonomous driving solution. It provides an integrated service for the entire process and full cycle, including data collection (road and vehicle data migration to the cloud and uploading to data lakes), computation (immediate training), and storage (long-term persistent storage).
- High-performance computing scenario: The data accelerator GooseFSx C50 offers high-performance, low-latency, high-throughput parallel file service, fully meeting the high-throughput, low-latency needs of HPC computation. Integrated with the data lakes COS, it provides ultra-high performance, hyperscale, low-cost storage service.

GooseFSx C60 Type

C60 has two specifications: C60.T2 and C60.T12. Except for capacity, the two specifications are the same in performance, price, features, and robustness.

Features

- High performance, performance scales linearly with capacity, 200MB/s read bandwidth per TiB, 200MB/s write bandwidth per TiB, 20,000 read IOPS per TiB, 20,000 write IOPS per TiB.
- Starting capacity, step length and usage recommendations as shown in the table below:

Specification	Start Capacity	Step Length Capacity	Usage recommendations
C60.T12	36TiB	12TiB	Data volume exceeds 100TiB or grows fast.
C60.T2	4.5TiB	1.5TiB	Data volume within 100TiB

If you need to further enhance performance, or if purchase failed due to resource insufficient in AZ, or have another requirement, you can contact Storage Architect or consult Online Customer Service to ask for help.

Applicable Scenarios

- AI training and inference scenarios: Data accelerator GooseFSx C60 accelerates AI training and inference performance. It enables data to flow freely with data lakes like COS, providing immediate access to model data in COS for AI computing power. This avoids expensive compute resources waiting for data and allows training results to be stored in COS as needed, reducing overall storage costs.
- Genetic Genetic analysis scenario: The data accelerator GooseFSx C60 speeds up high-performance storage needs for gene sequencing and local alignment stages. It enables seamless data flow with the data lake COS, allowing the Omics platform to immediately access COS samples, automatically

archive genetic analysis results to COS, and deliver them to end users via COS's Internet distribution capability.

- Autonomous driving scenario: The data accelerator GooseFSx C60, combined with data lakes COS, data migration, and other products, forms an end-to-end autonomous driving solution. It offers an integrated service covering the entire process and full cycle, including data collection (road and vehicle data uploading to the cloud and migration to data lakes), computation (immediate training), and storage (long-term persistent storage).
- High-performance computing scenario: The data accelerator GooseFSx C60 offers high-performance, low-latency, high-throughput parallel file service, fully meeting the high-throughput, low-latency needs of HPC computation. Integrated with the data lakes COS, it provides ultra-high-performance, hyperscale, low-cost storage service.
- CAE/CAD scenario: GooseFSx C60 accelerates small object read/write for CAE/CAD applications. Integrated with COS data lakes, it provides ultra-high-performance, hyperscale, low-cost storage service.
- Film and television rendering scene: Data accelerator GooseFSx C60 integrates with data lakes COS to provide integrated storage service. Rendering material is archived in COS at low cost. Data accelerator GooseFSx C60 is pulled up as needed for rendering, and final outputs are stored in COS for long-term maintenance. They are delivered to end users via COS Internet distribution capability.

GooseFSx C70 Type

Features

- High performance, performance scales linearly with capacity, 600MB/s read bandwidth per TiB, 200MB/s write bandwidth per TiB, 30,000 read IOPS per TiB, 20,000 write IOPS per TiB.
- The initial capacity is 13.9453125TiB (~14TiB), with a step length of 4.6484375TiB (~4.5TiB). Capacity specifications include 13.9453125TiB (~14TiB), 18.59375TiB (~18.5TiB), 23.2421875TiB (~23TiB), 27.890625TiB (~28TiB), 32.53906250TiB (~32.5TiB).

  If you have another requirement, or if purchase failed due to resource insufficient in AZ, you can consult Online Customer Service to ask for help.

Applicable Scenarios

- AI training and inference scenarios: The data accelerator GooseFSx C70 accelerates AI training and inference performance. It enables seamless data flow with the data lakes COS, providing immediate access to model data for AI computing power. This avoids expensive computing resources waiting for data and allows training results to be stored in COS as needed, reducing overall storage costs.
- Autonomous driving scenario: The data accelerator GooseFSx C70, combined with data lakes COS, data migration, and other products, forms an end-to-end autonomous driving solution. It offers an integrated service covering the entire process and full cycle, including collection (road and vehicle

data uploading to the cloud and entering the lake), computation (immediate training), and storage (long-term persistent storage).

- High-performance computing scenario: The data accelerator GooseFSx C70 offers high-performance, low-latency, high-throughput parallel file service, fully meeting the high-throughput, low-latency needs of HPC computation. Integrated with data lakes COS, it provides ultra-high performance, hyperscale, low-cost storage service.

GooseFSx Type Comparison

> Note:
> - Involved in resource reservation. If you have further needs, you can consult online customer service to ask for help.
> - Due to network reasons, performance may fluctuate by around 10%.

Main performance specifications comparison as follows:

Type	C70	C60	C50
Initial capacity (TiB)	13.95 (approx. 14)	• C60.T2： 4.5 • C60.T12： 36	C50.T3： 9 C50.T12： 36
Step length capacity (TiB)	4.65 (approx. 4.5)	• C60.T2：1.5 • C60.T12： 12	C50.T3： 3 C50.T12： 12
Read bandwidth MB/s per TiB [1]	600	200	100 – 120
Write bandwidth MB/s per TiB [1]	200	200	100 – 120
Read IOPS per TiB	30,000	20,000	10,000
Write IOPS per TiB	20,000	20,000	10,000
Latency ms	0.1~1	0.1~1	0.1~1

Note 1: If you need to further enhance performance, you can contact Storage Architect or consult Online Customer Service to ask for help.

Advantages

Last updated：2025-07-17 17:16:39

Flowing Data with COS on Demand

Data Accelerator GooseFSx accelerates Cloud Object Storage (COS). As the hot tier for COS, GooseFSx stores hot data while COS provides persistent storage for full data. GooseFSx and COS exchange data on demand. GooseFSx directly loads data from COS to enable hosts to quickly access cached data in GooseFSx, while avoiding repeated data loading from COS by different hosts, reducing COS request costs. GooseFSx sinks computed results into COS for persistent, low-cost storage. GooseFSx can be released when usage is completed, with data persistently stored via COS. GooseFSx can be pulled up as needed to load data from COS and accelerate COS access.

GooseFSx exchanges specified data with COS on demand, sinking data from an entire GooseFSx directory or a certain subdirectory into COS, or loading data from an entire COS bucket or a certain prefix into GooseFSx.

GooseFSx is deeply integrated with COS, automatically loading data from COS or automatically settling data into COS. Data flow is fully managed—you do not need to deploy data migration software for data transfer, no need to intervene in the execution process of data flow, and no need to worry about configuration, tuning, management, or operation and maintenance.

GooseFSx loads data from COS, escapes objects in the COS bucket into files for hosts to access as if accessing a local file system. GooseFSx escapes files into COS objects, settles them into COS, saves them in COS in the original data format, enabling direct access to the settled data in COS.

Ease of Use

The data accelerator GooseFSx is a fully hosted service—place order and start using it immediately, with no need to worry about deployment, configuration, tuning, management, or operation and maintenance.

GooseFSx is easy to use. With just a few clicks on the console within minutes, you can complete the process from purchase and creation to mounting for use. Mount GooseFSx as a local directory on the host and use it like a local file system.

GooseFSx is POSIX file semantics compatible, allowing your workloads to use it without any modification, just like a local file system.

High-Performance and Ultra-Low Latency

The data accelerator GooseFSx adopts a fully parallel architecture designed for high-performance workloads. All nodes are fully equivalent and access data in full parallel. Metadata and data are evenly distributed across all nodes. Performance scales linearly with node expansion, delivering hundreds of GBs of throughput per second, millions of IOPS, and sub-millisecond latency.

Seamless Integration with Computing Ecosystem, Supporting Linux and Windows

The data accelerator GooseFSx is POSIX file semantics compatible, can auto mount as a local directory on the host, and seamlessly integrates into the computing ecosystem. It simultaneously supports Linux and Windows hosts, addressing issues with many applications, such as the DAE tool, which only supports the Windows platform.

GooseFSx integrates with Tencent Cloud High Performance Computing (THPC) to enable rapid deployment of high-performance computing environments. Leveraging GooseFSx's high-performance and ultra-low latency, it accelerates computational performance and shortens calculation time, enhancing overall efficiency.

Elastic and Efficient

Storage/computing separation. The data accelerator GooseFSx is separated from computing, independently expanding and upgrading, auto scaling, independently pulling up and terminating as needed, and flexibly scheduling.

Hot and cold data layering; cold data is persisted to COS, and hot data is cached in GooseFSx. COS offers unlimited capacity to meet the long-term, reliable, and low-cost large-capacity storage needs for massive data. GooseFSx supports PB-level scalability, hundreds of GB per second throughput, millions of IOPS, and sub-millisecond latency to meet high-performance access requirements.
GooseFSx and COS are decoupled from each other, scaling elastically respectively. GooseFSx and COS are deeply integrated to achieve data flow on demand.

GooseFSx uses hourly billing and can be released at any time after usage is completed. It persists the calculation results generated on GooseFSx to COS. GooseFSx can be pulled up at any time, loads data from COS, directly accesses the cached data in GooseFSx, and accelerates access to COS.

Applicable Region

Last updated：2025-07-17 17:16:39

Data Accelerator GooseFSx is available in the regions as shown in the table below. If you have further requirements, you can consult Online Customer Service for help.

Type	Applicable Region	Applicable Scenario
GooseFSx C50	Beijing, Shanghai, Guangzhou, Nanjing, Chengdu, Chongqing, and Singapore.	Small- to medium-sized throughput requirement, suitable for AI training and reasoning, gene analysis, autonomous driving, high-performance computing scenario.
GooseFSx C60	Beijing, Shanghai, Guangzhou, Nanjing, Chengdu, Chongqing, and Singapore.	Medium to high scale throughput requirements, suitable for AI training and reasoning, genetic analysis, autonomous driving, high-performance computing, CAE/CAD, and film and television rendering scenes.
GooseFSx C70	Beijing, Shanghai, Guangzhou, Nanjing, Chengdu, Chongqing, and Singapore.	Large-scale throughput requirements, suitable for AI training and reasoning, autonomous driving, and high-performance computing.

Specifications and Limits
File Processing Capacity Specs and Limits

Last updated: 2025-07-17 17:16:39

The specifications and limits of file processing capability are as follows:

Project	Specification	Description
Upper Limit of Files Number	Min[40,000 * deployed capacity (GiB), 1.6B]	Each GooseFSx instance has an upper limit of 1.6 billion files. When the deployment capacity is below 40,000 GiB, it supports 40,000 files per GiB with linear scale-out. For example, the deployment capacity is 10,000 GiB, and the upper limit of files number is 400 million (400 million = 10,000 GiB * 40,000 per GiB). If you have further needs, you can consult Online Customer Service to contact us.
Maximum number of directories	Min[Upper Limit of Files Number, 100M]	The maximum number of directories for each GooseFSx instance: 100 million. If the upper limit of files number is insufficient for small-scale GooseFSx deployment, the maximum value of directories will be the upper limit of files number.
Maximum filename length/byte	255	The filename supports a maximum of 255 bytes.
Maximum length of absolute path/byte	4096	The absolute path supports up to 4096 bytes.
POSIX client cap	4096	The POSIX client cap for each GooseFSx instance: 4096.
Maximum number of open files	10,000	Number of open files per GooseFSx instance: 100,000. If you have further needs, you can consult Online Customer Service to contact us.

Specifications and Usage Limits of Super Directory

Last updated：2025-07-17 17:16:39

This document primarily introduces super directory and quota specifications with usage limits.

Super Directory Specifications and Usage Limits

Project	Specification	Description
Super directory maximum number	512	The maximum number of super directories for each GooseFSx instance is 512. If you have further needs, you can consult Online Customer Service to contact us.
Super directory name length cap/byte	255	The super directory name supports a maximum of 255 bytes.
Super directory hierarchy depth limit/byte	127	Ultra-large extreme value. 1 character per directory name
Super Directory Upper Limit of Files Number	Min[upper limit of files number for GooseFSx instance, 160 million]	The upper limit of files number for each super directory is 1.6 billion. If you have further needs, you can consult Online Customer Service to contact us.
Parent directory of the super directory	Ordinary directory or root directory	• The parent directory (or parent of parent) of a super directory only supports ordinary directories or the root directory, not super directories. If you have further needs, you can consult online customer service to contact us. • When creating a super directory, if its parent directory (or parent of parent) does not exist, it will be automatically generated with an ACL of 777 selected by default, which users can modify.
Subdirectory of the super directory	Ordinary directory	Subdirectories of the super directory (or subdirectories of subdirectories) only support ordinary directories and do not support super directories.

Access Control List (ACL) of the super directory	777	The super directory's access control list (ACL) is 777 by default and can be modified by the user.

Quota Specifications and Usage Limits

Project	Specification	Description
Maximum quota number	Super directory cap	Quota can be set for each super directory.
file quota range of values	[1, 1000 trillion]	• Minimum value is 1, maximum value is 1,000 trillion. • Step length is 1. • Supports dynamic modification.
capacity quota permissible range	[1GiB, 909.49PiB]	• Minimum value is 1GiB, maximum value is 909.49PiB. • Step length is 1GiB. • Dynamic modification supported.

Client Specifications and Limits

Last updated：2025-07-17 17:16:40

This document describes the specifications and limitations of the GooseFSx POSIX client.

Applicable Operating Systems

The POSIX client of GooseFSx supports Linux and Windows operating systems. The recommended operating system versions are as shown in the table below:

Operating System Types	Operating System	Edition	Kernel version
Linux	TencentOS Server	TencentOS Server 3.1 (TK4)	Deeply optimized based on the community 5.4 LTS kernel
		TencentOS Server 2.4(TK4)	Deeply optimized based on the community 5.4 LTS kernel
	RHEL or CentOS	8.4	4.18.0-305.19.1.el8_4
		8.3	4.18.0-240.22.1.el8_3
		8.2	4.18.0-193.28.1.el8_2
		8.1	4.18.0-147.8.1.el8_1
		8	4.18.0-80.11.2.el8_0
		7.9	3.10.0-1160.42.2.el7
		7.8	3.10.0-1127.19.1.el7
		7.7	3.10.0-1062.18.1.el7
		7.6	3.10.0-957.54.1.el7
		7.5	3.10.0-862.14.4.el7
		7.4	3.10.0-693.2.2.el7
		7.3	3.10.0-514.26.2.el7
		7.2	3.10.0-514.26.2.el7

	Ubuntu	Ubuntu 22.04	5.15.0-27-generic
	Ubuntu	Ubuntu 20.04	5.4.0-147-generic
Windows	Windows	Windows 2016	all
		Windows 2019	all

Note:
If your operating system version is not listed in this table, you can consult online customer service for help.

POSIX Client Specification

Project	Specification	Description
Single GooseFSx instance supports POSIX client count	4096	A single GooseFSx instance can be mounted by 4096 POSIX clients simultaneously.
Number of GooseFSx instances that a single POSIX client can mount	1	A single POSIX client can only mount one GooseFSx instance. Note: Since unmounting by the client is an asynchronous operation and takes some time to complete completely. Therefore, it takes more than 10 minutes for a POSIX client to unmount a GooseFSx instance before it can be remounted; otherwise, mounting fails.
Minimum configuration for a POSIX client	4 core 8 GB	Recommended minimum configuration for a POSIX client: 4 cores, 8GB memory. The available memory space of the client must be greater than 2GB to avoid deployment failure caused by insufficient available memory.

Note:

- If this table cannot meet your needs or you have other requirements, you can consult online customer service for help.
- Do not delete or overwrite the authorized_keys file of the POSIX client. Recommend using the `ssh copy` command to append write the authorized_keys file. Do not use the `scp` command to overwrite write the authorized_keys file.

POSIX Client Security Group Rules

The POSIX client needs to allow the following ports to communicate with GooseFSx. For relevant guidelines, please see Configure Security Group. If you also set up a security group with a denial policy for the POSIX client, please ensure that the POSIX client security group rules are not denied.

Protocol	Direction	Port	IP
TCP	inbound rule	22,1191,4242,4243,4739,8445,9085,9250,9251,9980,10080,55533,60000-61000	All IPs in the VPC to which GooseFSx belongs
UDP	inbound rule	4739	All IPs in the VPC to which GooseFSx belongs
ICMP	inbound rule	ALL	All IPs in the VPC to which GooseFSx belongs

Note:
Port 22 is the default port number used for SSH mutual trust communication between the POSIX client and the client management node. If it does not meet your requirements, customization is supported, and available ports can avoid the TCP ports allowed by the POSIX client's security group rules.
For example, if you customize the port to 2008, 22 will be replaced with 2008 and appear in the POSIX client security group rule.

GooseFSx Occupied VPC Subnet IP Address

To enable communication between hosts in the Customer VPC and GooseFSx, the GooseFSx file system will occupy a certain number of IP addresses in the VPC subnet. The larger the GooseFSx capacity, the more IP addresses are occupied, as follows:

Model	Number of Occupied IP Addresses	Description

C50.T3	1 IP address per 3 TiB	9 TiB capacity occupies 3 IPs
C50.T12	1 IP address per 12 TiB	36 TiB capacity occupies 3 IPs
C60.T2	1.5 TiB uses 1 IP address	4.5 TiB capacity occupies 3 IPs
C60.T12	1 IP address per 12 TiB	36 TiB capacity occupies 3 IPs
C70	1 IP address per 4.65 TiB	13.95 TiB capacity occupies 3 IPs

Note:
If the available IP addresses in this VPC are insufficient, it can cause GooseFSx expansion or creation failure. You can consult online customer service to ask for help.

Data Flow Specifications and Limits

Last updated：2025-07-17 17:16:40

GooseFSx Data Flow Rules

Project	Specification	Description
A single GooseFSx instance supports the count of data flow rules	16	A single GooseFSx instance supports 16 data flow rules, just flow data with 16 COS storage buckets simultaneously. If you have further requirements, you can consult Online Customer Service to ask for help.

GooseFSx Data Pre-Warming and Data Settlement

Project	Specification	Description
Delete tasks in progress	Not supported	Deleting data flow tasks in progress is not yet available, which can cause intermediate state data.
Delete completed task	Supported.	Delete completed tasks, you will not be able to know what tasks were executed, proceed with caution.
Number of parallel data flows in a single GooseFSx instance	10	A single GooseFSx instance supports up to 10 concurrent data flows. If you have further requirements, you can consult Online Customer Service to ask for help.

Quick Start

Last updated：2025-07-17 17:16:40

Overview

Data Accelerator GooseFSx console provides a simple and easy-to-use operation method. Users can directly use the GooseFSx service through the console without writing code or running programs.

Preparations

For first-time use of data accelerator GooseFSx, it is advisable to understand related terms and concepts. For details, see Glossary.

Step 1: Register a Tencent Cloud account.

Before using Tencent Cloud Data Accelerator GooseFSx service, you need to register a Tencent Cloud account. Click the button below to start registration. (If you have already registered, skip this step.)

Step 2: Complete Real-Name Authentication

After successful registration, log in to the Tencent Cloud console with this account and start real-name verification. For detailed instructions, see the Real-Name Verification Guide. (If you have already completed this step, please skip it.)

Step 3: Enable the Data Accelerator Service

In Tencent Cloud Console, select Cloud Products > Data Accelerator, enter the Data Accelerator console, Enable Data Accelerator Service, and follow the interface prompts to activate Data Accelerator service. (If you have already enabled it, skip this step.)

Step 4: Create a GooseFSx Instance

1. Log in to the Data Accelerator Console, click to create, and select GooseFSx.
2. On the GooseFSx configuration page, set according to the following configuration item descriptions.
   - Name: Customizable, modifiable after creation.
   - <Region: Select based on actual conditions, cannot be modified after creation.>
   - Availability zone: Select based on actual conditions. Cannot be modified after creation.
   - <Model: Select based on actual conditions, cannot be modified after creation.>
   - POSIX client's VPC and subnet, select based on actual conditions, cannot be modified after creation.
3. Click Buy Now.

Step 5: Client Mount GooseFSx

1. Purchase one cloud server as a host. Note down its host IP address (or host name) and root account password.
1. For recommended host configuration, please see Client Specifications and Limits. For purchase operation details, please see Instance Creation Guide.
2. Add security group rules to this host. For details, see Managing GooseFSx Client.
3. Log in to the Data Accelerator Console, and select the newly created GooseFSx instance from the instance list.
4. Click the instance ID or the Manage operation to enter the POSIX Client page.
5. In the POSIX client list, click Add POSIX Client.
6. In the pop-up dialog box for creating a POSIX client, manually input the host's IP or hostname, or use fuzzy matching to retrieve the host from the drop-down list, input the root account password, and click Confirm.

Step 6: POSIX Client Read-Write GooseFSx

Log in to the host, enter the mounted directory /goosefsx/x-c60-XXX-proxy (XXX is the GooseFSx instance ID), and read/write this directory means read/write GooseFSx.

Purchase Guide

Last updated：2025-07-17 17:16:40

Billing Mode

Pay-as-you-go

The pay-as-you-go (postpaid) method is the default billing mode for GooseFSx. Billing is based on the specific dosage of billing items, with fees deducted from the user account on an hourly basis.

Billing Formula

Storage capacity fee = storage capacity unit price × storage capacity × storage duration.
Data flow bandwidth cost = unit price x data flow bandwidth x duration.

Billing Cycle

Billed hourly, with settlement for the previous hour's generated costs. Less than 1 hr is billed as 1 hr.

Instance Creation Time	Terminate Instance Time	Billing duration	Description
11:28 (UTC+8)	11:58 (UTC+8) (same date as the instance creation date)	1 hour	1 transaction The usage start time is 11:00 and the usage end time is 11:59. Total billing duration: 1 hour
11:28 (UTC+8)	12:18 (UTC+8) (same date as the instance creation date)	2 hour	2 transactions • Transaction 1: The usage start time is 11:00 and the usage end time is 11:59. • The second transaction: usage start time is 12:00, usage end time is 12:59. • Billing duration: total 2 hours
11:28 (UTC+8)	13:08 (UTC+8) (same date as the instance creation date)	3 hours	3 transactions • The first transaction: usage start time is 11:00, usage end time is 11:59. • The second transaction: usage start time is 12:00, usage end time is 12:59. • The third transaction: usage start time is 13:00, usage end time is 13:59.

			Total billing duration: 3 hours

Billing Items and Pricing

GooseFSx Storage Capacity Fee

GooseFSx storage capacity refers to the storage space purchased by the user. The storage capacity fee is calculated based on storage capacity and duration.

Billing Mode	Billing Item	Unit Price	Applicable Region	Description
Pay-as-you-go	GooseFSx C50 storage capacity fee	$0.000114/GB/hour (approx. $0.082/GB/month)	Beijing, Shanghai, Guangzhou, and Nanjing	Hourly pay-as-you-go billing, hourly storage capacity fee = storage capacity unit price x storage capacity x hour (billing duration)
	GooseFSx C60 storage capacity fee	$0.000258/GB/hour (approx. $0.186/GB/month)	Beijing, Shanghai, Guangzhou, and Nanjing	
	GooseFSx C70 storage capacity fee	$0.000386/GB/hour (approx. $0.278/GB/month)	Beijing, Shanghai, Guangzhou, and Nanjing	

GooseFSx Data Flow Bandwidth Fee

GooseFSx data flow bandwidth refers to the data flow bandwidth purchased by the user. The data flow bandwidth fee is calculated based on data flow bandwidth and duration.

> Notes:
> GooseFSx C50, C60, and C70 have the same data flow bandwidth cost.

Billing Mode	Billing Item	Unit Price	Applicable Region	Billing Mode	Description
Pay-as-you-go	GooseFSx data flow	$0.000429/M	Beijing, Shanghai,	Hourly pay-as-you-go billing. Data flow bandwidth fee per hour	Cost per hour, rounded to

	bandwidth fee	B/s/hour	Guangzhou, and Nanjing	= unit price x data flow bandwidth x hr (billing duration).	six decimal places

Billing Example

User A in the Beijing region uses GooseFSx C60 with a capacity of 4.5TB. The billing duration is 1 hr, and the fee calculation is $0.000258/GB/hour × 4.5TB × 1024 × 1 hr = $1.189.
User A in the Beijing region uses GooseFSx data flow bandwidth of 600MB/s. The billing duration is 2 hr, and the fee calculation is $0.000429/MB/s/hour × 600MB/s × 2 hr = $0.515.

Querying Bills

You can view the payment information generated by your account's usage of the GooseFSx service in the Billing Center on the Tencent Cloud console.
GooseFSx is a subproduct of GooseFS and belongs to the product GooseStorage. GooseFSx appears in the order list of order management, with the product field registered as GooseStorage and the subproduct registered as GooseFS. Click Details to query specific information about GooseFSx.

Overdue Payment Instructions

When your account is in payment overdue status, the system will push an overdue notification. After receiving the arrears notification, you can go to the Payment Management in time to recharge to avoid affecting your business.

Service Status after Pay-As-You-Go Overdue

- The service can still be used normally within 24 hours of account arrears. You need to top up promptly during this period to ensure the account balance is at least 0, so as not to affect your business operation.
- Your account will be suspended 24 hours after falling into arrears. The service will stop running automatically, and data will not be readable or writable, during which time fees will be incurred.
- If the account remains in arrears for 8 days, it will be deemed as active abandonment of this service. Tencent Cloud does not guarantee to retain your data, and your data will be terminated and cannot be recovered.

Overdue Duration	Service Status	Whether to Bill	Restorable to Normal State
24 hours	normal usage	normal billing	recoverable
1–8 days	unavailable	normal billing	recoverable

8 days later	unavailable	free of charge	Terminate, cannot be restored

Client Management Node Billing Rules

Enable POSIX client will automatically create 3 Cloud Virtual Machine (CVM) instances under your Tencent Cloud account. The CVM instance specification is not less than 4C16G, deployed as POSIX client management nodes. The cost of POSIX client management nodes will be charged by CVM product. In the expense center, you can query the cost by the Host ID of the POSIX client management node. Retrieve the Host ID from the GooseFSx console (log in to GooseFSx console, enter the POSIX client page of the GooseFSx instance, and view the Host ID of the 3 POSIX client management nodes).

Console Guide
Using GooseFSx on TKE Containers

Last updated：2025-07-17 17:16:40

Overview

Kubernetes abstracts PV (PersistentVolume) and PVC (PersistentVolumeClaim) to mount and use storage. For details, refer to Kubernetes official documentation.

- PV (Persistent Volume): PV encapsulates storage resources, describing a persistent storage volume in a container cluster, which belongs to cluster-level resources.
- PVC (PersistentVolumeClaim): PVC applies for storage resources and consumes PV resources in the container cluster. If no PV resources exist in the cluster, it will dynamically create PV resources and underlying storage. By associating PVC with a Pod, the Pod can use the storage resources.
- StorageClass describes the type of PV and PVC in a container cluster. PV can be auto-created based on StorageClass to reduce the work of creating and maintaining PV. When dynamically creating PVC/PV, you must specify StorageClass.

Data Accelerator Goose FileSystem (GooseFS) supports two persistent volume (PV) types: Local PV and CSI PV.

- Local PV (Local Persistent Volume): Kubernetes directly uses the mounted GooseFSx directory on the host machine to persistently store container data.
- CSI PV: Kubernetes uses the CSI (Container Storage Interface) protocol to dynamically mount GooseFSx for persistent storage of container data.

Since GooseFSx has mounted the shared directory on the host machine of the container cluster, using Local PV is more direct and offers improved efficiency. Local PV is recommended for use.

Prerequisites

- A container cluster has been created, such as TKE (Tencent Kubernetes Engine) container cluster or self-built Kubernetes container cluster.
- The container cluster and GooseFSx instance are in the same VPC and same subnet.
- The host machine's operating system of the container cluster is compatible with GooseFSx. See Client Specifications and Limits.
- Container cluster host has mounted GooseFSx shared directory. See Managing GooseFSx Instances.

Use Limits

1. GooseFSx does not currently support TKE super node. Please use TKE Node Pool to achieve dynamic scaling.

2. GooseFSx does not currently support dynamically creating PV based on StorageClass.

Local PV Operation Steps

Defining a yaml File Sample for PV Persistent Volume `local_goosefsx_PV.yaml`

> **Note:**
> Replace the local path with the mount directory of GooseFSx on the host machine.

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: csi-goosefsx-local-pv
spec:
  accessModes:
  - ReadWriteMany
  capacity:
    storage: 10Gi
  volumeMode: Filesystem
  persistentVolumeReclaimPolicy: Delete
  storageClassName: local-storage
  local:
# Replace the path here with the host mounting path of GooseFSx, then
delete this reminder
    path: /goosefsx/x-c60-ow1j60r9-proxy
  nodeAffinity:
    required:
      nodeSelectorTerms:
      - matchExpressions:
        - key: kubernetes.io/arch
          operator: In
          values:
          - amd64
```

Key parameters are described as follows:

Parameter	Description

name: csi-goosefsx-local-pv	Define the persistent volume name and modify according to actual conditions.
accessModes: - ReadWriteMany	Define the access mode. "ReadWriteMany" means it can be mounted in read-write mode by multiple nodes.
storage: 10Gi	Define the storage capacity. "10Gi" indicates 10GiB storage capacity. This parameter does not limit the capacity provided by the file system. The actual storage capacity is the purchased GooseFSx capacity and dynamically scales with GooseFSx expansion. For example, if the purchased GooseFSx capacity is 4.5TiB, the storage capacity is 4.5TiB (not 10GiB). After expanding GooseFSx to 9TiB, the storage capacity becomes 9TiB.
volumeMode: Filesystem	Define the persistent volume mode as file system.
persistentVolumeReclaimPolicy: Delete	Define the recycling policy, delete.
storageClassName: local-storage	Define the persistent volume belonging to the class "local-storage". The persistent volume claim must belong to the same class "local-storage". The name "local-storage" should be consistent with the storage class file's name "local-storage".
local: path: /goosefsx/x-c60-ow1j60r9-proxy	Define the container's storage space as the host machine's directory path "/goosefsx/x-c60-ow1j60r9-proxy", which is GooseFSx's mount directory on the host machine. Modify according to actual conditions.
nodeAffinity	Define node affinity.

2. Sample yaml File for Defining a PVC Persistent Volume Claim `local_goosefsx_PVC.yaml`

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: local-goosefsx-pvc
spec:
  accessModes:
  - ReadWriteMany
  resources:
    requests:
```

```
      storage: 10Gi
  storageClassName: local-storage
```

Key parameters are described as follows:

Parameter	Description
name: local-goosefsx-pvc	Define the persistent volume claim name and modify according to actual conditions.
accessModes: - ReadWriteMany	Define access mode and persistent volume.
resources: requests: storage: 10Gi	Define the storage capacity. "10Gi" indicates 10GiB storage capacity. This parameter does not limit the capacity provided by the file system. The actual storage capacity is the capacity purchased for GooseFSx and dynamically scales with GooseFSx expansion. For example, if the purchased GooseFSx capacity is 4.5TiB, the storage capacity is 4.5TiB (not 10GiB). After expanding GooseFSx to 9TiB, the storage capacity becomes 9TiB.
storageClassName: local-storage	Define a persistent volume claim belonging to the class "local-storage". The persistent volume must belong to the same class "local-storage". The name "local-storage" should be consistent with the storage class file's name "local-storage".

3. Sample yaml File for Defining StorageClass Storage Class `local_goosefsx_StorageClass.yaml`

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: local-storage
provisioner: kubernetes.io/no-provisioner
volumeBindingMode: WaitForFirstConsumer
```

Key parameters are described as follows:

Parameter	Description

name: local-storage	Define a storage class named "local-storage". The yaml file for the PV persistent volume and the yaml file for the PVC persistent volume claim will use it.
provisioner: kubernetes.io/no-provisioner	Define the provisioner for the PV persistent volume. GooseFSx's PV persistent volume is relatively simple, and auto-creating PV persistent volumes by using StorageClass is not required.
volumeBindingMode:WaitForFirstConsumer	Define volume binding mode. This mode delays PersistentVolume binding and provisioning until the Pod using the PersistentVolumeClaim is created.

4. Run Command to Create StorageClass, PV, and PVC

Run the following command to create StorageClass:

```
kubectl apply -f local_goosefsx_storageclass.yaml
```

```
[root@VM-3-42-tlinux /home]# kubectl apply -f local_goosefsx_storageclass.yaml
storageclass.storage.k8s.io/local-storage created
```

Run the following command to create a PV:

```
kubectl apply -f local_goosefsx_pv.yaml
```

```
[root@VM-3-42-tlinux /home]# kubectl apply -f local_goosefsx_pv.yaml
persistentvolume/csi-goosefsx-local-pv created
```

Run the following command to create a PVC:

```
kubectl apply -f local_goosefsx_pvc.yaml
```

```
[root@VM-3-42-tlinux /home]# kubectl apply -f local_goosefsx_pvc.yaml
persistentvolumeclaim/local-goosefsx-pvc created
```

5. Deploying Pod and Mounting the PVC

A sample yaml file for the Pod mounting this PVC `local_goosefsx_Pod.yaml` is as follows:

```
apiVersion: apps/v1
```

```
kind: Deployment
metadata:
  labels:
    k8s-app: local-goosefsx-dp
  name: local-goosefsx-dp
spec:
  replicas: 1
  selector:
    matchLabels:
      k8s-app: local-goosefsx-pod
  template:
    metadata:
      labels:
        k8s-app: local-goosefsx-pod
    spec:
      containers:
        - image: nginx
          name: local-goosefsx-pod
          volumeMounts:
            - mountPath: /local-goosefsx
              name: local-goosefsx-pv
      volumes:
        - name: local-goosefsx-pv
          persistentVolumeClaim:
            claimName: local-goosefsx-pvc
```

Deploy Pod:

```
kubectl apply -f local_goosefsx_pod.yaml
```

```
[root@VM-3-42-tlinux /home]# kubectl apply -f local_goosefsx_pod.yaml
deployment.apps/local-goosefsx-dp created
```

Check if the Pod is in ready State:

```
kubectl get pod
```

```
[root@VM-3-42-tlinux /home]# kubectl exec -ti local-goosefsx-dpa-7fb9b9f877-qmtb4   -- /bin/sh
[root@VM-3-42-tlinux /home]# kubectl get pod
NAME                                  READY   STATUS    RESTARTS   AGE
kubernetes-proxy-74bf5df48-2zh2x      1/1     Running   0          98m
kubernetes-proxy-74bf5df48-vxjmc      1/1     Running   0          98m
local-goosefsx-dp-7fb9b9f877-fcttx    1/1     Running   0          67m
local-goosefsx-dpa-7fb9b9f877-qmtb4   1/1     Running   0          25m
```

Log in to the Pod, check whether the mount point is correct, and check whether the mount point is online:

```
kubectl exec -ti local-goosefsx-dp-7fb9b9f877-fcttx   -- /bin/sh
```

CSI PV Operation Steps

Static creation of PV and PVC, StorageClass is not required.
Additionally, three yaml files for CSI need to be defined. The CSI code is already built into the TKE mirror, so no need to worry.

1. Defining a PV yaml File

Sample YAML file for defining PV `pv.yaml`:

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: csi-goosefsx-pv
spec:
  accessModes:
  - ReadWriteMany
  capacity:
    storage: 10Gi
  csi:
    driver: com.tencent.cloud.csi.goosefsx
    volumeHandle: csi-goosefsx-pv
  storageClassName: ""
```

2. Defining a PVC yaml File

Sample yaml file for defining PVC `pvc.yaml`:

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
```

```
  name: csi-goosefsx-pvc
spec:
  accessModes:
  - ReadWriteMany
  resources:
    requests:
      storage: 10Gi
  volumeName: csi-goosefsx-pv
  storageClassName: ""
```

3. Define the yaml File for the CSI driver

Sample YAML file for defining CSI driver `csi-driver.yaml` :

```
apiVersion: storage.k8s.io/v1
kind: CSIDriver
metadata:
  name: com.tencent.cloud.csi.goosefsx
spec:
  attachRequired: false
  podInfoOnMount: false
  fsGroupPolicy: File
```

4. Defining a CSI node yaml File

Sample YAML file for defining CSI node `csi-node.yaml` :

> **Note:**
> Replace the fileSystemId with the file system ID for host mounting.

```
kind: DaemonSet
apiVersion: apps/v1
metadata:
  name: csi-goosefsx-node
  namespace: kube-system
spec:
  selector:
```

```
    matchLabels:
      app: csi-goosefsx-node
  template:
    metadata:
      labels:
        app: csi-goosefsx-node
    spec:
      serviceAccount: csi-goosefsx-node
      priorityClassName: system-node-critical
      hostNetwork: true
      hostPID: true
      containers:
        - name: driver-registrar
          image: ccr.ccs.tencentyun.com/tkeimages/csi-node-driver-
registrar:v2.0.1
          lifecycle:
            preStop:
              exec:
                command: ["/bin/sh", "-c", "rm -rf
/registration/com.tencent.cloud.csi.goosefex
/registration/com.tencent.cloud.csi.goosefsx-reg.sock"]
          args:
            - "--v=5"
            - "--csi-address=$(ADDRESS)"
            - "--kubelet-registration-
path=/var/lib/kubelet/plugins/com.tencent.cloud.csi.goosefsx/csi.sock"
          env:
            - name: ADDRESS
              value: /plugin/csi.sock
            - name: KUBE_NODE_NAME
              valueFrom:
                fieldRef:
                  fieldPath: spec.nodeName
          volumeMounts:
            - name: plugin-dir
              mountPath: /plugin
            - name: registration-dir
              mountPath: /registration
        - name: goosefsx
          securityContext:
```

```
            privileged: true
            capabilities:
              add: ["SYS_ADMIN"]
            allowPrivilegeEscalation: true
          image: ccr.ccs.tencentyun.com/qcloud_goosefsx/goosefsx-
csi:v1.0.1
          args:
            - "--v=5"
            - "--logtostderr=true"
            - "--nodeID=$(NODE_ID)"
            - "--endpoint=$(CSI_ENDPOINT)"
# Replace the fileSystemId here with the host mounting file system ID,
then delete this reminder
            - "--filesystemId=x-c60-s1bz66l4"
          env:
            - name: NODE_ID
              valueFrom:
                fieldRef:
                  fieldPath: spec.nodeName
            - name: CSI_ENDPOINT
              value: unix://plugin/csi.sock
          volumeMounts:
            - name: plugin-dir
              mountPath: /plugin
            - name: goosefsx-mount-dir
              mountPath: /goosefsx
              mountPropagation: "Bidirectional"
            - name: pods-mount-dir
              mountPath: /var/lib/kubelet/pods
              mountPropagation: "Bidirectional"
      volumes:
        - name: plugin-dir
          hostPath:
            path:
/var/lib/kubelet/plugins/com.tencent.cloud.csi.goosefsx
            type: DirectoryOrCreate
        - name: registration-dir
          hostPath:
            path: /var/lib/kubelet/plugins_registry
            type: Directory
```

```
        - name: pods-mount-dir
          hostPath:
            path: /var/lib/kubelet/pods
            type: Directory
        - name: goosefsx-mount-dir
          hostPath:
            path: /goosefsx
            type: Directory
```

5. Define the yaml File for CSI rbac

Sample YAML file for defining CSI RBAC `csi-rbac.yaml`:

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: csi-goosefsx-node
  namespace: kube-system

---
kind: ClusterRole
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: csi-goosefsx-node
rules:
  - apiGroups: [""]
    resources: ["persistentvolumeclaims/status"]
    verbs: ["patch", "update"]
  - apiGroups: [""]
    resources: ["configmaps", "events",
"persistentvolumes","persistentvolumeclaims"]
    verbs: ["get", "list", "watch", "update", "patch"]
  - apiGroups: [""]
    resources: ["pods", "nodes"]
    verbs: ["get", "list", "watch"]
  - apiGroups: ["storage.k8s.io"]
    resources: ["storageclasses"]
    verbs: ["get", "list", "watch"]
  - apiGroups: ["storage.k8s.io"]
```

```
    resources: ["volumeattachments"]
    verbs: ["get", "list", "watch", "update"]
  - apiGroups: [""]
    resources: ["events"]
    verbs: ["list", "watch", "create", "update", "patch"]
  - apiGroups: ["coordination.k8s.io"]
    resources: ["leases"]
    verbs: ["get", "list", "watch", "create", "update", "patch",
"delete"]

---
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: csi-goosefsx-node
subjects:
  - kind: ServiceAccount
    name: csi-goosefsx-node
    namespace: kube-system
roleRef:
  kind: ClusterRole
  name: csi-goosefsx-node
  apiGroup: rbac.authorization.k8s.io
```

6. Running Command to Create CSI, PV and PVC

Run the following command to configure rbac, driver, and node:

```
kubectl apply -f csi-rbac.yaml
kubectl apply -f csi-driver.yaml
kubectl apply -f csi-node.yaml
```

Run the following command to check whether it is functioning normally:

```
kubectl get ds -n kube-system
```

Run the following command to create a PV and PVC:

```
kubectl apply -f pv.yaml
kubectl apply -f pvc.yaml
```

7. Deploying Pod and Mounting the PVC

Mount the PVC to the Pod with the `pod.yaml` file as follows:

> **Note:**
> Replace the claimName with the appropriate PVC name, which is the name defined in the YAML file (for example, file sample pvc.yaml) as name: csi-goosefsx-pvc.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    k8s-app: csi-goosefsx-pod
  name: csi-goosefsx-pod
spec:
  replicas: 1
  selector:
    matchLabels:
      k8s-app: csi-goosefsx-pod
  template:
    metadata:
      labels:
        k8s-app: csi-goosefsx-pod
    spec:
      containers:
        - image: nginx
          name: csi-goosefsx-pod
          volumeMounts:
            - mountPath: /csi-goosefsx
              name: csi-goosefsx
      volumes:
        - name: csi-goosefsx
          persistentVolumeClaim:
            claimName: csi-goosefsx-pvc
```

Deploy Pod

```
kubectl apply -f pod.yaml
```

Check if the Pod is in ready State:

```
kubectl get pod
```

Managing GooseFSx Task Center

Last updated：2025-07-17 17:16:40

Overview

Scaling out a GooseFSx instance, adding clients, and other complex operations require certain execution time and asynchronously return the execution result. The GooseFSx task center provides detailed execution process. You can log in to the task center page in the GooseFSx console to check.

Operation Steps

1. Log in to the Data Accelerator GooseFS Console, and enter the Instance List.
2. Select the region where the target instance is located, select the target GooseFSx instance (the GooseFSx instance for creating POSIX clients). You can quickly find the target instance by checking availability zone, status, type, etc., or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Manage to enter the instance management page, then click Task Center.
4. Select the date range to be queried and choose the corresponding Task List. Optional parameters include:
   - Within a specific number of days: support querying task status within a certain period, such as on the day, 7-day, 15-day, or 30-day task list.
   - Selected time range: support querying for task list within the specified window.
5. In the task list, you can click the task ID or View Procedure button to view task steps via the console drawer.
6. In the task steps, you can click the Step Details button to view the execution status, creation time, and end time of each step's detailed process in the pop-up window.

Managing GooseFSx Monitoring Information

Last updated：2025-07-17 17:16:40

Overview

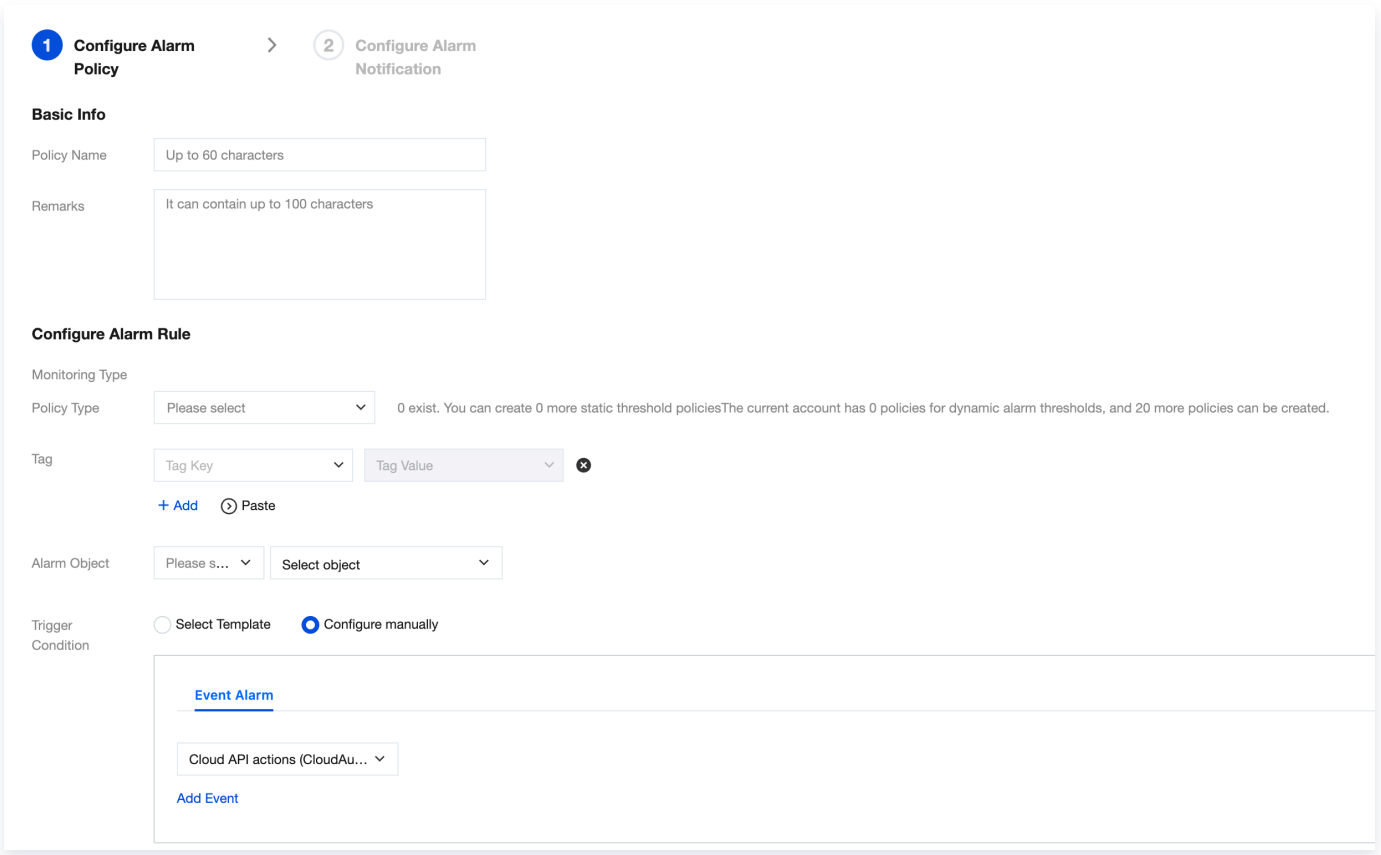
GooseFSx console provides a monitoring information page where you can view GooseFSx's purchased capacity, free capacity, and read/write bandwidth data.

Setting Monitoring Alarms

You can use the alarm policy of Tencent Cloud Observability Platform (TCOP) to set threshold alarms for GooseFSx monitoring metrics. Tencent Cloud Observability Platform is a service that provides real-time monitoring and alarms for cloud product resources. Users can set alarm rules, view alert history, and receive alarm notifications for GooseFSx monitoring metrics through TCOP. For more information about Tencent Cloud Observability Platform, please see Create Alarm Policy.

Operation Steps

1. Log in to the Data Accelerator Console.
2. Click GooseFSx > Instance List, then click the GooseFSx instance name.
3. Click the Monitoring Information on the left to view each monitoring module. Click the icon on the monitoring module to quickly jump to the alarm management page of the Tencent Cloud observability platform. For configuration instructions, please refer to the Create Alarm Policy on the Tencent Cloud observability platform. Examples of configured alarm policies are shown below:



Managing GooseFSx Data Settlement

Last updated：2025-07-17 17:16:40

Creating a Data Settlement One-Time Task

Create a one-time data settlement task to execute once between the GooseFSx directory and the COS storage bucket.

Prerequisites

- Creating GooseFSx Instances.
- Creating GooseFSx Data Flow Rules.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select a region, select the target GooseFSx instance. You can quickly find the target instance by Availability Zone, Status, Type, or search for the target instance by name keyword fuzzy matching.
3. Click the instance ID or Manage, select Data Flow > Data Settlement.
4. Under the single task tab, click Create One-time Task. In the pop-up dialog box, configure as follows:
    - Task name: Enter an easy-to-identify name, modifiable after creation.
    - GooseFSx directory: Select from directories that have established data flow rules with COS. Go to the data flow rules page to create.
    - Dataset: Specifies the flow of certain data in a GooseFSx directory. The GooseFSx directory can be specified by directory (a subdirectory or the entire directory) or by manifest (certain files). The data flow maintains a one-to-one correspondence between the data in the GooseFSx directory and the data in the COS storage bucket. The dataset also specifies the corresponding data flow of the COS storage bucket. For details, ,see Managing GooseFSx Data Pre-Warming.
    - Task type:
        - Data settlement: Settle specified data in a GooseFSx directory to a COS storage bucket. Suitable for storing a piece of data in COS or distributing via the Internet, among other scenarios.
        - Data settlement and delete local data: Settle specified data in a GooseFSx directory to a COS storage bucket and delete the specified data in the GooseFSx directory. Suitable for data cooling and other scenarios.
        - Data settlement and eliminate local data but reserve metadata: Settle specified data in a GooseFSx directory to a COS storage bucket and eliminate the specified data in the GooseFSx directory while retaining metadata. Suitable for scenarios such as GooseFSx building a global namespace.

5. Click Confirm, and backend automation will execute data settlement.

> Note:
> - GooseFSx directories in Lazy-load mode (auto mode L) do not currently support data settlement for single tasks.
> - Note: If the data flow mode of the GooseFSx directory is manual mode, data settlement is unsupported, and local data is eliminated while metadata is reserved.

Deleting a Data Settlement One-Time Task

> Note:
> You can delete completed or pending execution one-time tasks for data settlement. Note:
> - Delete the completed data settlement one-time task, you cannot query the task's relevant information.
> - Delete the pending execution data settlement one-time task, which cancels the task.
> - Deletion is not supported for executing one-time data settlement tasks.

Prerequisites

- Creating GooseFSx Instances.
- Creating a Data Settlement One-Time Task.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select a region, select the target GooseFSx instance. You can quickly find the target instance by Availability Zone, Status, Type, or search for the target instance by name keyword fuzzy matching.
3. Click the instance ID or Manage, select Data Flow > Data Settlement.
4. In the single task list, select the target task, or filter the target task by conditional filtering; or check multiple tasks and execute delete in batches.

Querying Data Settlement One-Time Tasks

Prerequisites

- Creating GooseFSx Instances.
- Creating a Data Settlement One-Time Task.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.

2. Select the region, select the target GooseFSx instance. You can quickly find the target instance by Availability Zone, Status, Type, or search for the target instance by name keyword fuzzy matching.
3. Click the instance ID or Manage, select Data Flow > Data Settlement.
4. In the single task list, select the target task, or filter the target task by conditional filtering, click Details to view the task details.

Replicating Data Settlement One-Time Task

Prerequisites

- Creating GooseFSx Instances.
- Creating a Data Settlement One-Time Task.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select a region, select the target GooseFSx instance. You can quickly find the target instance by Availability Zone, Status, Type, or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Manage, select Data Flow > Data Settlement.
4. In the single task list, select the target task, or filter the target task by conditions, click Copy Task to perform.
5. In the pop-up dialog box, you can modify the copied task.

Creating a Data Settlement Periodic Task

Periodic task, automatically triggered to execute periodically, such as settling data from GooseFSx to COS every 1-hour interval, suitable for syncing data to COS promptly in other scenarios.
In the periodic task list, you can see created periodic tasks and perform management operations such as add/delete/query.
Settlement tasks triggered by periodic tasks will appear in the one-time task list, all named after the periodic task name and distinguished by task ID.

Prerequisites

- Creating GooseFSx Instances.
- Created Managing GooseFSx Data Flow Rules.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select the region and target GooseFSx instance. You can quickly find the target instance by Availability Zone, Status, Type, or perform a keyword fuzzy matching search by name.

3. Click the instance ID or Manage, select Data Flow > Data Settlement.
4. Under the periodic task tab, click Create Periodic Task. In the pop-up dialog box, configure as follows:
   - Task Name: Enter an easy-to-identify name.
   - Trigger cycle:
     - per hour: execute once per interval of one hour.
     - per day: execute once per interval of one day.
     - Weekly: Execute once every other week.
     - per month: execute once per interval of one month.
   - GooseFSx directory: Select from directories that have established data flow rules with COS. Go to the instance details page to create.
   - Target path: Specify the directory path for settlement, such as a GooseFSx directory or a certain subdirectory under the GooseFSx directory.
   - Task type:
     - Data settlement: Settle specified data in a GooseFSx directory to a COS storage bucket. Suitable for storing a piece of data in COS or distributing via the Internet, among other scenarios.
     - Data settlement and delete local data: Settle specified data in a GooseFSx directory to a COS storage bucket and delete the specified data in the GooseFSx directory. Suitable for data cooling and other scenarios.
     - Data settlement and elimination of local data while retaining metadata: settle specified data in the GooseFSx directory to a COS storage bucket, eliminate specified data in the GooseFSx directory, and retain metadata. Suitable for scenarios such as building a global namespace with GooseFSx.
5. Click OK.

> ⚠ Note:
> - If the data flow mode of the GooseFSx directory is Lazy-load (auto mode L), the task type does not currently support data settlement or deleting local data.
> - Note: If the data flow mode of the GooseFSx directory is manual mode, the task type does not support data settlement, and local data is eliminated while metadata is reserved.

Managing GooseFSx Data Pre-Warming

Last updated：2025-07-17 17:16:40

Creating a Data Pre-Warming One-Time Task

Create a data pre-warming one-time task to execute data pre-warming between the GooseFSx directory and the COS storage bucket.

Prerequisites

- Creating GooseFSx Instances.
- Creating GooseFSx Data Flow Rules.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select a region and select the target GooseFSx instance. You can quickly find the target instance by AZ, status, type, or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Manage, then select Data Flow > Data Pre-warming.
4. In single task, click Create Single Task. In the pop-up dialog box, configure as follows.
    - Task Name: Input an easy-to-identify name, modifiable after creation.
    - GooseFSx directory: Select from directories that have established data flow rules with COS. Go to the data flow rules page to create.
    - Dataset: Specifies the flow of certain data in a GooseFSx directory. The GooseFSx directory can be specified by directory (a subdirectory or the entire directory) or by manifest (certain files). The data flow maintains a one-to-one correspondence between the data in the GooseFSx directory and the data in the COS storage bucket. The dataset also specifies the corresponding data flow of the COS storage bucket.
        - If you select directory, enter a directory path that does not start with "/" but ends with "/", or provide no input.
            - no input: assign all data in the GooseFSx directory; as shown in the figure below, assign all data in the goosefsx-bj-auto-test-1252246555 directory.
            - bigfile/: assign all data under the bigfile/ subdirectory in the GooseFSx directory.
            - bigfile/pic/: assign all data under the bigfile/pic/ subdirectory in the GooseFSx directory.
        - If you select manifest file, enter the COS default access address of the manifest file as the URL. Custom domains are not supported. Example: https://fsx-bj-test-125xxx.cos.ap-beijing.myqcloud.com/list/test_list1.txt.

- The manifest file must be stored in a COS bucket within the same region as the GooseFSx instance and set to public read permission.
- Each row in the manifest file is a file path under the GooseFSx directory, with different file paths separated by carriage returns.

  For example, to transmit three files in this directory: test_dir1/test_file_1.data, test_dir2/test_file_2.data, test_dir3/test_file_3.data, the list will have three lines. The first line is test_dir1/test_file_1.data, the second line is test_dir2/test_file_2.data, and the third line is test_dir3/test_file_3.data.
- The manifest file format is a txt file.

- Task type:
  - Metadata preheating: Only preheat metadata from a COS storage bucket to the GooseFSx directory. Suitable for initial read performance insensitive scenarios.
  - Data pre-warming: Pre-warm data including metadata from a COS storage bucket to the GooseFSx directory. Suitable for high performance access scenarios.

5. Click Confirm, and the backend will automatically execute data preheating.

> ⚠ Note:
> - If the data flow mode of the GooseFSx directory is Lazy-load (auto mode L), the dataset does not currently support specifying through a manifest, and the task type does not support data pre-warming.
> - Note: If the data flow mode of the GooseFSx directory is manual mode, the task type does not support metadata pre-warming.

Deleting a Data Pre-Warming One-Time Task

> ⚠ Note:
> You can delete completed or pending execution data pre-warming one-time tasks. Note: pay attention to the following points:
> - Delete completed one-time data pre-warming tasks. You cannot query the task's relevant information.
> - Delete pending execution one-time data pre-warming tasks, which cancels the task.
> - Deletion of executing data pre-warming one-time tasks is not supported.

Prerequisites

- Creating GooseFSx Instances.
- Created a one-time data pre-warming task.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select a region and select the target GooseFSx instance. You can quickly find the target instance by AZ, status, type, or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Manage, then select Data Flow > Data Pre-warming.
4. In the one-time task list, select target task, or filter and sort target tasks by condition; or check multiple tasks and delete in batches.

Querying a Data Pre-Warming One-Time Task

Prerequisites

- Creating GooseFSx Instances.
- Created a one-time data pre-warming task.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select a region and select the target GooseFSx instance. You can quickly find the target instance by AZ, status, type, or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Manage, then select Data Flow > Data Pre-warming.
4. In the one-time task list, select target task, or filter and sort target tasks by condition, click Details to view detailed information.

Replicate Data Preheat Single Task

Prerequisites

- Creating GooseFSx Instances.
- Created a one-time data pre-warming task.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select a region and select the target GooseFSx instance. You can quickly find the target instance by AZ, status, type, or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Manage, then select Data Flow > Data Pre-warming.
4. In the one-time task list, select target task, or filter target tasks by condition, then click Copy Task to perform the operation.
5. In the pop-up dialog box, you can modify the copied task.

Creating a Data Pre-Warming Periodic Task

Periodic task, automatically triggered and executed periodically. For example, pre-warm data from COS to GooseFSx every 1-hour interval, suitable for prompt synchronization of COS bucket data changes and other scenarios.
In the periodic task list, you can see created periodic tasks and perform management operations such as adding, deleting, and querying.
The preheating task triggered by a periodic task will appear in the single task list, all named with the periodic task name and distinguished by the task ID.

Prerequisites

- Creating GooseFSx Instances.
- Created a one-time data pre-warming task.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select a region and select the target GooseFSx instance. You can quickly find the target instance by AZ, status, type, or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Manage, then select Data Flow > Data Pre-warming.
4. In Periodic Task page, click Create Periodic Task. In the pop-up dialog box, configure as follows.
   - Task Name: Input an easy-to-identify name, modifiable once created.
   - Trigger cycle:
     - Once per hour: Execute once every 1-hour interval.
     - Daily: Execute once every other day.
     - Weekly: Execute once per interval.
     - Execute once per month at a one-month interval.
   - GooseFSx directory: Select from the directory that has established data flow rules with COS. Go to the instance details page to create.
   - Target path: Specify the directory path to be preheated, such as the GooseFSx directory or a certain subdirectory under it.
   - Task type:
     - Metadata preheating: Only preheat metadata from a COS storage bucket to the GooseFSx directory. Suitable for initial read performance insensitive scenarios.
     - Data pre-warming: Pre-warm data including metadata from a COS storage bucket to the GooseFSx directory. Suitable for high performance access scenarios.
5. Click OK just.

Note:

- If the data flow mode of the GooseFSx directory is on-demand loading (auto mode L), periodic tasks are not currently supported.
- Note: If the data flow mode of the GooseFSx directory is manual mode, the task type does not support metadata pre-warming.

Manage GooseFSx Data Flow Rules

Last updated：2025-07-17 19:20:42

Adjusting Data Flow Bandwidth

Data flow bandwidth. The data accelerator GooseFSx and Cloud Object Storage (COS) perform data flow bandwidth, which can be adjusted at any time to match business requirements. For details, see Product Overview.

Prerequisites

Creating GooseFSx Instances.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select the region and target GooseFSx instance. You can find the target instance by AZ, status, or data type, or search for it by name keyword fuzzy matching.
3. Click the instance ID or Management, select Data flow Data flow policy.
4. Click Data flow bandwidth and then Adjust to enter the Adjust Data Flow Bandwidth dialog box.
5. Use the cursor or input box to select target bandwidth. Bandwidth fees are displayed synchronously with the price of the target bandwidth. For data flow bandwidth pricing, refer to the Purchase Guide. After confirming the information is correct, click Confirm.

Note:

- Adjust the data flow bandwidth, expected to complete in 3–5 minutes. During the adjustment process, the business remains unaffected, and data flow runs normally. After completing the adjustment, data flow smoothly uses the adjusted bandwidth.
- During the adjustment process, the status display of data flow bandwidth shows scaling out or scaling in. At this point, you cannot adjust (scale out or scale in) again to avoid repetition.
- To reduce data flow bandwidth to 0, please delete all data flow rules first.

During the adjustment process, the status display of data flow bandwidth shows scaling out or scaling in. At this point, you cannot adjust (scale out or scale in) again to avoid repetition.
To reduce data flow bandwidth to 0, please delete all data flow rules first.

Adding Data Flow Rules

Add data flow rules refers to creating rules for data flow between a certain directory in the data accelerator GooseFSx and a certain bucket in Cloud Object Storage (COS). For details, see Product

Overview.

Prerequisites

- Creating GooseFSx Instances.
- Data flow bandwidth has been adjusted to above 0.
- Creating a COS bucket is completed.

> Note:
> - If the COS storage bucket is cross-regional, it is recommended to use the COS private network global acceleration domain to flow data. Enable global acceleration for the COS storage bucket enable global acceleration. GooseFSx will access the cross-regional COS storage bucket via the private network global acceleration domain. GooseFSx also supports COS custom access points, requiring users to provide the VIP of the cross-regional COS storage bucket custom access point. For obtaining the COS storage bucket VIP, you may contact us.
> - If you use the private network global acceleration domain of a cross-regional COS bucket, you need to pay the COS private network global acceleration domain fee, which is counted in by the COS product and charged based on actual traffic generated. For more details, see COS Product Pricing.
> - If you use the VIP of a cross-regional COS bucket, you need to pay the cross-region access network traffic fee, which is counted in by the related network product.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select the region and target GooseFSx instance. You can find the target instance by AZ, status, or data type, or search for it by name keyword fuzzy matching.
3. Click the instance ID or Management, then select Data Flow > Data Flow Rule.
4. Click Add Rule, enter the Add Rule dialog box, and configure the following:

> Notes:
> If the data flow bandwidth is 0, unable to add data flow rules. Scale it out first, then rerun the add process.

   - Enter Name, a custom name easy to identify.
   - Select a COS region. The default region is the GooseFSx instance region. You can select another region.

- Select a COS storage bucket. If the current COS storage buckets do not meet your requirements, follow the prompt in ⓘ and click to enter the COS console to create a bucket.
- If you select a cross-region access point, use a global acceleration domain name. If you select a custom access point, enter the COS bucket VIP, such as 192.168.0.10.

> ⚠ Notes:
> Note: If the COS VIP address is not in the subnet range of the VPC associated with GooseFSx, please go to the basic info page, click associate with new network in the network configuration below, and associate with the corresponding VPC subnet. As shown below, associate with the VPC subnet belonging to 192.168.0.10.

- Enter the GooseFSx directory. The bucket name is selected by default and can be customized.
- Select Data Flow Mode, which supports manual mode and auto mode. For details, see Product Overview.

5. Click Confirm. After creation, you can check result in the list.

Deleting Data Flow Rules

Delete data flow rules, delete data flow from GooseFSx directory to COS bucket and related records and data flow rules. Before deletion, confirm that the data in the GooseFSx directory has been sunk to the COS bucket. For details, see Product Overview

> ⚠ Notes:
> - Delete the rule, disable and delete the rule, and remove related data pre-warming and data settlement.
> - Once deleted, you will be unable to flow data with the bucket or check the records of the data flow rule.

Prerequisites

- Creating GooseFSx Instances.
- Creating GooseFSx Data Flow Rules.
- Check current data flow is being executed.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select the region and target GooseFSx instance, which is the instance associated with the bucket. You can find the target instance by AZ, status, or data type, or search for it by name keyword fuzzy

matching.

3. Click instance ID or Management, select Data flow > Data flow rule.
4. Select the data flow rules to be deleted, click Delete, or check multiple and click Delete in Batches.
5. Enter the Delete Rule dialog box, click OK to delete.

Query Data Flow Rules

Enter the data flow details page to query detailed information of a data flow.

Prerequisites

- Creating GooseFSx Instances.
- Creating GooseFSx Data Flow Rules.

Operation Steps

1. Log in to the GooseFS Data Accelerator Console and enter the instance list.
2. Select the region and target GooseFSx instance, which is the instance associated with the bucket. You can find the target instance by AZ, status, or data type, or search for it by name keyword fuzzy matching.
3. Click the instance ID or Management, then select Data Flow > Data Flow Rule.
4. To enter the details page, use the following methods:
   - Click the Instance ID.
   - Click the instance details.
5. On the details page, you can view detailed information such as name, COS storage bucket, GooseFSx directory, data flow mode, and data flow bandwidth.

Managing GooseFSx Tags

Last updated：2025-07-17 17:16:41

Overview

A tag is a marker provided by Tencent Cloud for identifying cloud resources. It is a Key-Value pair. For a detailed introduction, please see Tag Overview.
You can use tags to classify and manage GooseFSx resources from various dimensions (such as business, purpose, responsible person, etc.). Tags can very conveniently help you filter out the resources you need. The Key-Value pair of a tag does not have any semantic meaning for Tencent Cloud. Filtering and permission control based on tags will be parsed and matched strictly according to the string. For more usage restriction statements, please refer to Tag Usage Restrictions.

Prerequisites

The GooseFSx instance has been created. For creation guide, please see Managing GooseFSx Instances.

Directions

1. Log in to the Data Accelerator GooseFS Console, and enter the Instance List.
2. Select the region where the target instance is located, and select the target GooseFSx instance.
3. You can quickly find the target instance by checking the availability zone, status, type, etc., or search for the target instance using keyword fuzzy matching on the name.
4. Click the instance ID or Manage to enter the "Basic Information" page, and view, add, or modify tags. You can also create tags when creating a GooseFSx instance, with similar operation steps.

Managing GooseFSx Clients

Last updated：2025-07-17 20:23:21

Creating a POSIX Client

The GooseFSx POSIX client refers to the host where the GooseFSx POSIX client software is deployed and mounts GooseFSx as a local directory. The POSIX client accesses the mount directory just like accessing a local file system.

> Note:
> Recommended host operating system version. For details, see Client Specifications and Limits.

Prerequisites

- Created GooseFSx instances and POSIX client management nodes. When creating a GooseFSx instance, a POSIX client management node is synchronously created. Please see Managing GooseFSx Instances.
- Added security group rules for the host. Please see Client Specifications and Limits. If you have also set up a security group with a denial policy for the POSIX client, ensure that the POSIX client security group rules are not denied.

Operation Steps

1. Log in to the Data Accelerator GooseFS console, and enter the instance list.
2. Select the region where the target instance is located, select the target GooseFSx instance, which is the GooseFSx instance for creating the POSIX client. You can quickly find the target instance by AZ, status, Type, or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Management, then select POSIX Client > Client List.
4. In the client list section, click Add New Client to create a POSIX client.
5. In the pop-up dialog box for adding a POSIX client, configure as follows.
   - VPC to which the corresponding instance belongs: Select host VPC location.
   - Subnet: Select the subnet where the host resides.
   - Host ID: Quickly find the host you need to add as a POSIX client by fuzzy matching the host name/host IP address, or enter the host IP directly. You can select up to 10 hosts at the same time and add multiple hosts as POSIX clients in one batch operation.

     > Note:
     > Please confirm that security group rules have been added for this host. Please refer to Client Specifications and Limits. If you also set up a security group with a denial policy for

> the POSIX client, ensure that the POSIX client security group rules are not denied.

- Mount directory: Please enter the absolute path. It can be one of the following cases:
  - If the directory already exists and is not mounted, it will be mounted to this directory.
  - If the directory already exists and is mounted, it will not be mounted to this directory.
  - If the directory does not exist, it will be created and mounted to this directory.
  - You can also choose the default mount directory. The default GooseFSx mount directory is `/GooseFSx/x-${model}-${instance id}-proxy`, such as `/GooseFSx/x-c60-089xxxx-proxy`. Among them, `${model}` and `${instance id}` are the model and instance of your specific environment. The GooseFSx file system mounts to this directory on the host.

6. Click Confirm. The creation process is expected to take several minutes. You can check the result in the POSIX client list.

Checking Whether GooseFSx Is Mounted

The default mount directory for GooseFSx is `/GooseFSx/x-${model}-${instance id}-proxy`, such as `/GooseFSx/x-c60-089xxxx-proxy`. Among them, `${model}` and `${instance id}`, please refer to Managing GooseFSx Instances.

The user can run the following command to check whether GooseFSx is mounted:

```
sh goosefsx_mount_status.sh 10
```

Note:

- The input parameter number 10 means the timeout period, which is customizable.
- If `mount success` is returned, mounting is successful.
- If `wait time out` is returned, mounting fails.

The path to execute this command is as follows:

```
/root/goosefsx_client_agent/linux_client_agent/goosefsx_mount_status.sh
```

Mounting GooseFSx to a Specified Directory

Users can mount the specified directory manually using the mount command, such as mounting to `/mnt/test`:

```
mount --bind /goosefsx/x-c60-089xxxx-proxy/ /mnt/test
```

Creating POSIX Clients in Batches

To batch create POSIX clients, you can create a batch of GooseFSx POSIX clients at one time using an image. First, create a GooseFSx POSIX client with the same host OS version. Next, generate an image from this client. Finally, use the image to pull up GooseFSx POSIX clients in bulk.

Prerequisites

- Created GooseFSx instances. Please see Managing GooseFSx Instances. If multiple GooseFSx instances have been created, select the instances for which you want to batch create POSIX clients; if you want to batch create POSIX clients for multiple instances, execute batch creation of POSIX clients for each of these instances, because GooseFSx POSIX clients belong to a certain GooseFSx instance.
- POSIX client management nodes have been created. When creating GooseFSx Instances, POSIX client management nodes are synchronously created.
- Created at least one GooseFSx POSIX client with the same host OS version.

> Note:
> To batch create POSIX clients with different OS versions, first create POSIX clients with different OS versions, generate a mirror, and then pull up POSIX clients with the same OS version using the mirror.
> For example, create a CentOS POSIX client, generate a mirror of the CentOS POSIX client, then batch create CentOS POSIX clients using the mirror; create a TLinux POSIX client, generate a mirror of the TLinux POSIX client, then batch create TLinux POSIX clients using the mirror; and so on.

Operation Steps

1. Log in to the Data Accelerator GooseFS console, and enter the instance list.
2. Select the region where the target instance is located, then select the target GooseFSx instance. You can quickly find the target instance by AZ, status, type, etc., or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Management, then select POSIX Client > Client List.
4. In the POSIX Client List, select the POSIX client to create an image based on. If there is no POSIX client for the target operating system version, you need to Creat POSIX Client first; click Host ID to enter the console of the host instance. If you need to create an image, please refer to Create a Custom Image.

5. Batch create POSIX clients using images. For relevant guidelines, please refer to Instance Creation Using Images.

Querying a POSIX Client

Query POSIX clients, including both automatically and manually added POSIX clients mounted to GooseFSx.

Prerequisites

- Created GooseFSx instances and POSIX client management nodes. When creating GooseFSx instances, POSIX client management nodes are synchronously created. Please see Managing GooseFSx Instances.
- Created a GooseFSx Client.

Operation Steps

1. Log in to the Data Accelerator GooseFS console, and enter the instance list.
2. Select the region where the target instance is located, then select the target GooseFSx instance, which is the GooseFSx instance belonging to the POSIX client. You can quickly find the target instance by AZ, status, type, etc., or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Management, then select POSIX Client > Client List.
4. POSIX Client List displays all POSIX client information.

Deleting a POSIX Client

Delete POSIX client, unbind POSIX client mount from GooseFSx. The client management node no longer manages this POSIX client. Uninstall POSIX client software.
After deleting a POSIX client, you can perform the creation again to mount it to GooseFSx.

Prerequisites

- Created GooseFSx instances and POSIX client management nodes. When creating a GooseFSx instance, a POSIX client management node is synchronously created. See Managing GooseFSx Instances.
- Created a GooseFSx Client.

Operation Steps

1. Log in to the Data Accelerator GooseFS console, and enter the instance list.
2. Select the region where the target instance is located, then select the target GooseFSx instance, which is the POSIX client to be deleted. You can quickly find the target instance by AZ, status, type, etc., or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Management, then select POSIX Client > Client List.

4. Check the POSIX client to be deleted, then click Delete in the Action column and Confirm in the pop-up.

Managing GooseFSx Super Directory and Quota

Last updated：2025-07-17 17:16:41

A super directory (fileset) is a directory with quota and other features. The quota enables setting different storage capacity limits and number limits for different super directories.

Note:
- Super directory and quota details, please see Product Overview.
- Specifications and usage limits of super directories and quotas, please see Specifications and Usage Limits of a Super Directory.

Modify Quota Effective Scope

Prerequisites

GooseFSx instance has been created. For creation guide, please refer to Managing GooseFSx Instances.

Operation Steps

1. Log in to the Data Accelerator GooseFS console, and enter the instance list.
2. Select the region where the target instance is located, then select the target GooseFSx instance. You can quickly find the target instance by availability zone, status, type, or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Management to enter the super directory page.
4. In basic configuration, click Modify to enter the dialog box, as shown below.

   - Applies to all users: The quota applies to all users (including ROOT users). All users cannot exceed the quota limit when writing data. Exceeding the quota will result in an error: "Disk quota exceeded."
   - Only applicable to non-ROOT users: The quota is only applicable to non-ROOT users, meaning non-ROOT users cannot exceed the quota limit when writing data; ROOT users are not subject to configuration limits and can continue writing data even after exceeding the quota.

Note:
- Modify quota effective scope, take effect immediately.
- Modify quota effective scope, take effect for ALL super directories.
- The default value of quota effective scope is effective for all users.

> Example: The capacity quota for super directory A is 10TiB. Non-ROOT users write 2TiB to directory A, and ROOT users write 9TiB to directory A. The quota is only applicable to non-ROOT users. If the quota scope is changed to all users, because all users have already written 11TiB of data, exceeding the quota (10TiB), all users will be restricted from writing data to directory A.

Adding New Super Directory

Prerequisites

GooseFSx instance has been created. For creation guide, please refer to Managing GooseFSx Instances.

Operation Steps

1. Log in to the Data Accelerator GooseFS console, and enter the instance list.
2. Select the region where the target instance is located, then select the target GooseFSx instance. You can quickly find the target instance by availability zone, status, type, or perform a keyword fuzzy matching search by name.
3. Click the instance ID or Management to enter the super directory page.
4. In the super directory list, click Create Super Directory to enter the dialog box, as shown below.

   - Name: The name of the super directory. Please confirm the name first, as it cannot be modified after creation. The naming specification is as follows: 1 to 128 characters, supporting a combination of uppercase and lowercase English letters, digits, "_", "-", and ".".
   - Path: The path of the super directory, which is the absolute path of the super directory in this instance. Please confirm the path first, as it cannot be modified after creation. The naming specification is as follows: the path must start with "/" and end with "/". The format should be `/<parent directory>/.../<parent directory>/<subdirectory>/`, for example: /user/user1/.
   - File deletion record: Turn on or off file deletion record. When enabled, it logs deletion operations under the super directory.
   - Set quota: Turn on or off quota. When enabled, customizable file number quota and capacity quota.
   - File quota: Turn on or off file quota.
   - Capacity quota: Turn on or off capacity quota.

Viewing Super Directory Listing

On the super directory webpage, you can view detailed information such as ID/name, status, directory path, quota, and creation time.

Updating Super Directory

On the super directory webpage, click Update to refresh the configuration message, such as turning on or off deletion records or modifying quota. After the update is installed, it will take effect immediately.

Deleting a Super Directory

On the super directory webpage, select a non-root directory and click Delete. In the pop-up dialog box, click Delete.

> **Note:**
> - The root directory (path `/`) is a super directory created synchronously with instance creation. It is deleted upon instance termination and does not support user deletion.
> - Deleting a super directory will delete all data in this directory and cannot be restored. Operate with caution.

Managing GooseFSx Instances

Last updated：2025-07-17 20:28:34

Creating an instance

Creating a GooseFSx instance will automatically create 3 Tencent Cloud cloud server (Cloud Virtual Machine, CVM) instances in your Tencent Cloud account. The instance specification is not less than 4C16G, deployed as POSIX client management nodes. POSIX client management nodes are created and terminated along with the GooseFSx instance, with no need to manage.

Prerequisites

- Before creating Data Accelerator Goose FileSystem (GooseFSx), you need to understand the region, availability zone, and VPC network of the host using GooseFSx. The system will deploy the GooseFSx instance in the same region and availability zone, and deploy the POSIX client management node in this VPC network.
- Product types for GooseFSx launch. For details, see Product Type.
- Regions where the service is launched. For details, see Available Region.
- If you have other needs or fail to create GooseFSx due to insufficient resources, you can consult online customer service to reserve resources in advance.
- Enable the data accelerator service. For the operation guide, see Quick Start.

Operation Steps

1. Log in to the Data Accelerator GooseFS console, go to GooseFSx > instance list, and click Create.
2. On the redirected GooseFSx purchase page, configure the information related to GooseFSx. The configuration items are described below.
    - Instance Information
        - Billing mode: pay-as-you-go.
        - Region: Select the region to create GooseFSx. Once a region is selected, it cannot be modified.
        - Availability zone (AZ): A physical area within the same region where the power and network are independent from each other. Different availability zones in the same region have private network interconnectivity. It is recommended to deploy GooseFSx in the same availability zone as the host to avoid latency generated by cross-availability zone deployment.
        - Type: Different product types of GooseFSx have different performance and pricing. For details, see Product Type.
        - Capacity: Choose appropriate capacity based on business needs; this capacity is your available capacity.

- Instance name: You can keep the default or set a custom instance name.
- Tag: Optional. It can be configured or modified after creating a GooseFSx instance. For details, see Managing GooseFSx Tags.

- POSIX client
  - VPC to which the corresponding instance belongs: Select the default VPC network; it cannot be changed. GooseFSx supports multiple VPC networks and can dynamically associate/disassociate multiple VPC networks. For details, see Managing GooseFSx Instances.
  - Subnet: Select the default subnet; it cannot be changed. GooseFSx supports multiple subnets and can dynamically associate/disassociate them. For details, see Modify Instance.
  - SSH port: 22 is the default port for SSH mutual trust between POSIX clients and client management nodes. If the current SSH port does not meet your requirements, you can customize the SSH port. For details, see Client Specifications and Limits.
  - Security group: Please refer to Client Specifications and Limits to ensure the security group allows the ports required by GooseFSx. Otherwise, it will cause GooseFSx creation to fail. If the current security groups do not meet your requirements, go to the console to create a security group.

3. Read and check the terms of service, then click Enable Now.
4. In the instance list, you can view the instance being created. The creation process is expected to take about 10 minutes.

Query an Instance

Support viewing instance details, such as type, capacity, billing mode, and creation time.

> Note:
> - Query the POSIX client of the instance. For details, see Managing GooseFSx Client.
> - Query the data flow of the instance and data flow tasks. For details, see Data flow and Data flow tasks.

Prerequisites

GooseFSx instance has been created.

Operation Steps

1. Log in to the Data Accelerator GooseFS console, and enter the instance list.
2. Select the region of the target instance, then select the target GooseFSx instance to be queried.

   You can quickly locate the target instance by Availability Zone, Status, Type, or search for the target instance by name keyword fuzzy matching.

3. Click Instance ID or Manage to view the instance details.

Modifying an Instance

Support modifying the instance name and description.

> Note:
> - Modify the POSIX client of the instance. For details, see Managing GooseFSx Client.
> - Modify data flow and data flow tasks of the instance. For details, see Data Flow and Data Flow Tasks.
> - Modify the tags of an instance. For details, see Managing GooseFSx Tags.

Prerequisites

The GooseFSx instance has been created.

Operation Steps

1. Log in to the Data Accelerator GooseFS console, and enter the instance list.
2. Select the region of the target instance, then select the GooseFSx instance to be modified.

   You can quickly find the target instance by Availability Zone, Status, Type, or search for the target instance by name keyword fuzzy matching.
3. Click the instance ID or Manage to modify the name and description, and dynamically associate/unbind the VPC network and subnet.

> Note:
> - Note: Default VPC and subnet cannot be disassociated. They are assigned when an instance is created and modifications are not allowed.
> - Subnets in the Default VPC network can be associated or disassociated.
> - You can associate or disassociate other VPC networks/subnets. Before associating another VPC network, users are advised to manually interconnect it with the Default VPC network. For VPC network connection, please refer to Private Link and Peering Connection.

Deleting an instance

> Note:
> - Delete the instance with caution. Once deleted, it cannot be restored. Before deletion, confirm the data of the instance has been persisted to COS.

- To view which hosts will delete POSIX client software, unbind associated buckets, and remove data flow tasks, please see Query an Instance.

Delete the instance operation will automatically delete all POSIX clients of the instance and remove all data flow and data flow tasks.

Prerequisites

GooseFSx instance has been created.

Operation Steps

1. Log in to the Data Accelerator GooseFS console, and enter the instance list.
2. Select the region of the target instance, then select the target GooseFSx instance to be deleted.

   You can quickly locate the target instance by Availability Zone, Status, or Type, or search for it using name keyword fuzzy matching.
3. Click Terminate in the operation column on the right side of the target instance.
4. Enter the ID of the instance to be destroyed in the pop-up window. After completing the secondary confirmation, click Confirm.

Tool Guide
GooseFSx-CLI Command Line Tool

Last updated：2025-07-17 19:23:00

Overview

GooseFSx-CLI is the command line tool for Tencent Cloud's data accelerator GooseFSx, suitable for 64-bit Linux systems. You can use command line instructions to automate and script the management of data in GooseFSx.

Download Tool

Download the GooseFSx-CLI tool, copy the GooseFSx-CLI program package to the GooseFSx client, and you can run it directly.

Parameter Configuration

For first-time use, you need to configure the parameters SecretId, SecretKey, and token. For parameter descriptions, please refer to Common Parameters. The guidance for obtaining SecretId and SecretKey is as follows:

- Root account access key management
- Sub-account access key management

The configuration parameters can be set as follows:

Method One: Directly Execute the Configuration Command

1. Execute the following command:

```
./goosefsx-cli
```

2. Follow the instructions, enter the AK, SK, and token in sequence:

```
2023/02/16 19:55:44 Welcome to goosefsx-cli!
When you use goosefsx-cli for the first time, you need to input some
necessary information to generate the default configuration file of
goosefsx-cli.
The path of the configuration file: /root/.goosefsx.yaml
Input Your Secret ID:
xxx
Input Your Secret Key:
```

```
yyy
Input Your Session Token:
test-token123
```

Method Two: Directly Edit the Configuration File `/root/.goosefsx.yaml`

```
# cat /root/.goosefsx.yaml
goosefsx:
  base:
     secretid: xxx
     secretkey: yyy
     sessiontoken: test-token123
     scheme: https
     endpoint: goosefs.internal.tencentcloudapi.com
```

List Objects and Virtual Directories in the Specified Directory

List files (objects) and directories (virtual directories) in the specified directory.

Command Syntax

```
./goosefsx-cli cos ls [flags]
```

Include the following flags:

Flag Name	Flag Abbreviation	Description
No	No	List files and directories in the specified directory, without recursively listing files and subdirectories in subdirectories
No	-r	Carry the -r parameter to recursively list directories; recursively list files and subdirectories in subdirectories under the specified directory.
--limit	No	Carry the --limit parameter to specify the number of files and directories to list; for example, --limit 2, list the first 2 files or directories in lexicographical order.
--inclu	No	Carry the --include parameter to set the included regular filter criteria; for example, --include dir, list files and subdirectories starting with dir.

de		
--exclude	No	Carry the --exclude parameter to set the excluded regular filter criteria; for example, --exclude dir, list files and subdirectories not starting with dir.
No	>	Import the file list to the specified file; otherwise, directly output the file list.
Overlay Superimpose Stack Accumulate Overlaid	Overlay Superimpose Stack Accumulate Overlaid	Carry multiple parameters simultaneously, achieve an overlay effect. For example, `-r --limit 10 --include dir > list` implements recursive listing of files and subdirectories in subdirectories, lists the first 10 files or directories, lists files and subdirectories starting with `dir`, and outputs to the file `list`.

Operation Example

<List Directories in Bucket goosefs-ofs-test1-xxx'S Directory dir/>

```
./goosefsx-cli cos ls --bucket-name goosefs-ofs-test1-xxx dir/  --
bucket-region ap-beijing
```

Return Value

```
[root@VM-10-214-tencentos ~]# ./goosefsx-cli cos ls --bucket-name
goosefs-ofs-test1-xxx dir/  --bucket-region ap-beijing
dir/_sss
dir/dir2/
```

list All Files and Directories in the Directory dir/ of Bucket goosefs-ofs-test1-xxx and Output to File list

```
./goosefsx-cli cos ls --bucket-name goosefs-ofs-test1-xxx dir/  --
bucket-region ap-beijing -r > list
```

Return Value

```
[root@VM-10-214-tencentos ~]# ./goosefsx-cli cos ls --bucket-name
goosefs-ofs-test1-xxx dir/  --bucket-region ap-beijing -r > list
[root@VM-10-214-tencentos ~]# cat list
dir/_sss
dir/dir2/
dir/dir2/_sss (4)
```

Execute Data Flow Task Command

data_flow for data flow tasks.

Prerequisites

Before using the data flow task command, create a data flow/associated bucket first. For details, see Managing GooseFSx Data Flow Rules.

Create a Data Flow Task and Async Return Result

Create a data flow task, return immediately whether the task is issued successfully; the execution time of the data flow task varies with the data volume. You can use the following query command to check the execution results of the data flow task.

Command Syntax

```
./goosefsx-cli data_flow create [flags]
```

The create command includes the following flags:

Flag Name	Flag Abbreviation	Description
--task-type	No	Data flow task type, FS_TO_COS (file system to COS Bucket) or COS_TO_FS (COS Bucket to file system)
--bucket	No	COS bucket name
--region	No	The region where the file system is located.
--filesystem-id	No	The file system ID.

--task-path	No	For FS_TO_COS tasks, --task-path is the relative path of the mapped directory in the Bucket. For COS_TO_FS tasks, --task-path is the path on COS. If left empty, it means all data. Example If you associate the COS bucket beijing-1250000000 with the GooseFSx directory beijing-1250000000 and configure `--task-path=test/` and `--task-path=level2/level3/`, it respectively represents the test/ and level2/level3/ paths under the COS bucket beijing-1250000000, or the test/ and level2/level3/ directories under the GooseFSx directory beijing-1250000000. Among them, level3/ is a subdirectory of level2/. **Note:** --task-path parameter value ends with `/`.
--task-name	No	Task Name

Output Parameter

Parameter Name	Type	Description
Task ID	string	Task ID
Task Request ID	string	unique request ID, which is returned for each request. RequestId is required for locating a problem.

Operation Example

Create data flow task

```
./goosefsx-cli data_flow create --task-type=COS_TO_FS --
bucket=bucket_test-1252246555 --region=ap-shanghai --filesystem-id=xxx -
-task-path=test/ --task-name=test
```

Return Value

```
Create data repository task successfully!
========================
  Task ID: x_task_1676551831746
```

```
  Task Request ID: f90a8741-f59b-465c-b44c-4758e7ed68a0
```

The result of the data flow task can be viewed using the query command.

Retrieve the Execution Result of a Data Flow Task

Query the execution result of the data flow task, in execution, completed, or failed.

Command Syntax

```
./goosefsx-cli data_flow taskstatus [flags]
```

taskstatus command includes the following flags:

Flag Name	Flag Abbreviation	Description
--region	No	The region where the file system is located.
--task-id	No	Task ID
--filesystem-id	No	The file system ID.

Output Parameter

Parameter Name	Type	Description
Query data flow task ID	string	Task ID
Query data flow task status	string	Retrieve the status of data flow tasks, including the numbers 0 (initialization), 2 (completed), and 3 (task failure).
Query data flow task Request ID	string	unique request ID, which is returned for each request. RequestId is required for locating a problem.

Operation Example

Retrieve real-time task status of data flow

```
./goosefsx-cli data_flow taskstatus --region=ap-shanghai --task-id=xxx --filesystem-id=xxx
```

Return Value

```
Query data flow task successfully!
=======================
  Query data flow task ID: x_task_1676551831746
  Query data flow task Request ID: fad14898-b975-423e-9588-7c5be84ad4e9
  Query data flow task status: Data flow task status is 2, finished
```

> **Note:**
> task-id is the returned value for creating a data flow task.

Create a Data Flow Task and Return Result in Real Time

Create a data flow task and return the execution result, completed or failed.

Command Syntax

```
./goosefsx-cli data_flow createAndWaitFinish [flags]
```

The createAndWaitFinish command includes the following flags:

Flag Name	Flag Abbreviation	Description
--task-type	No	Data flow task type, FS_TO_COS (file system to COS Bucket) or COS_TO_FS (COS Bucket to file system)
--bucket	No	COS bucket name
--region	No	The region where the file system is located.
--filesystem-id	No	The file system ID.
--task-path	No	For FS_TO_COS, TaskPath is the relative path of the mapped directory in the Bucket. For COS_TO_FS, it is the path on COS. If left empty, it means all data.
--task-name	No	Task Name

Output Parameter

Parameter Name	Type	Description
Task ID	string	Create data flow task ID
Task Request ID	string	Create data flow task unique request ID, which is returned for each request. RequestId is required for locating a problem.
Query data flow task ID	string	Retrieve data flow task ID
Query data flow task Request ID	string	Retrieve data flow task unique request ID, which is returned for each request. RequestId is required for locating a problem.
Query data flow task status	string	Retrieve the status of data flow tasks, including the numbers 0 (initialization), 2 (completed), and 3 (task failure).

Operation Example

Creating a Data Flow Task

```
./goosefsx-cli data_flow createAndWaitFinish --task-type=COS_TO_FS --
bucket=bucket_test-xxxx --region=ap-shanghai --filesystem-id=xxx --task-
path=test/ --task-name=test
```

Return Value

```
Create data repository task successfully!
=========================
  Task ID: x_task_1676553288519
  Task Request ID: 2902f31e-4bfc-4709-828f-999072c92478
Query data flow task successfully!
=========================
  Query data flow task ID: x_task_1676553288519
  Query data flow task Request ID: ee9d0646-9c9b-4589-b661-f492631147af
  Query data flow task status: Data flow task status is 2, completed
```

Loading Data Best Practices From Fusion Bucket

Execute two commands to load data from the fusion bucket. First, retrieve the file list of the fusion bucket, then load data from the fusion bucket according to the file list.

Prerequisites

Before use, create a data flow/associated bucket first. For details, see Managing GooseFSx Data Flow Rules.

Step 1: Retrieve Fusion Bucket File List

Get the file list, synchronously return the execution result, and save the file list to the specified file; the execution time varies with the data volume. For example, generating a file list for 4 million files takes over 6 minutes.

Command Syntax

```
./goosefsx-cli cos ls [flags]
```

Operation Example

```
[root@VM-10-117-tencentos ~]# time ./goosefsx-cli cos ls -r --bucket-name goosefs-ofs-test1-125 --bucket-region ap-beijing > file.list

real    6m29.254s
user    1m27.966s
sys     0m14.220s
```

Save file.list in the current directory.

Step 2: Load Data From Fusion Bucket According to File List

Load data from the COS fusion bucket based on the file list.

Command Syntax

```
./goosefsx-cli data_flow create [flags]
```

Operation Example

```
[root@VM-10-117-tencentos ~]# ./goosefsx-cli data_flow create --file-list file.list --bucket goosefs-ofs-test1-125 --region ap-beijing --filesystem-id x-c60-8zbg8z78 --tas
k-name list_test --task-type COS_TO_FS --file-list-bucket goosefsx-bj-test-125

Create data repository task successfully!
=========================
  Task ID: x_task_1723617
  Task Request ID: 41d059e4-a00b-4364-9344-f5eb518
```

Check task status:

```
[root@VM-10-117-tencentos ~]# ./goosefsx-cli data_flow taskstatus --filesystem-id x-c60-8zbg8z78 --region ap-beijing --task-id x_task_1723617437978
Query data flow task successfully!
=========================
Data flow task info: [taskId: x_task_1723617437978], [requestId: fe8e2783-6087-4f80-bada-082abfc1cb07], [status: 数据流通任务状态为 1, 进行中], [finishedFileNumber: 1290063], [f
nishedFileCapacity: 396301855940]
```

Log in to the console to check task status:

Practical Tutorial
Authorizing Sub-Account Access to GooseFSx

Last updated：2025-07-17 19:32:06

Overview

For Tencent Cloud GooseFSx resources, between different enterprises or between multiple teams in the same enterprise, you need to configure different access permissions for different teams or individuals. You can use Cloud Access Management (CAM) to set different operation permissions for GooseFSx, enabling different teams or individuals to collaborate with each other.
For terms related to CAM such as root account, sub-account (user), and user group, as well as detailed configuration descriptions, please see the Glossary in access management.

Root Account

Root account is also known as the developer. When a user applies for a Tencent Cloud account, the system creates a root account identity for logging into Tencent Cloud services. The root account is the basic entity for metering and billing Tencent Cloud resource usage.
The root account has by default full access to its owned resources, including accessing billing information, changing user password, creating users and user groups, and accessing other cloud service resources. By default, resources can only be accessed by the root account, and any other user access requires authorization from the root account.

Sub-accounts (users) and user groups

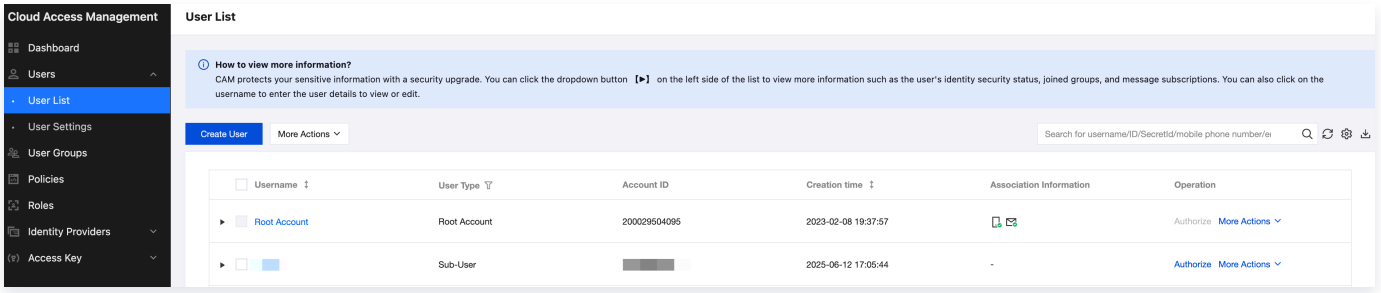
- A sub-account is an entity created by a root account, with a definite identity ID and credentials, and has the permission to log in to the Tencent Cloud console.
- Sub-accounts do not own resources by default and must be authorized by the root account to which they belong.
  - One root account can create multiple sub-accounts (users).
  - A sub-account can belong to multiple root accounts, and assist multiple root accounts in managing their respective cloud resources. However, a sub-account can only log in to one root account at a time to manage the cloud resources of the root account.
- Sub-accounts can switch developers (root accounts) through the console, and switch from one root account to another.
  - When a sub-account logs in to the console, it will automatically switch to the default root account and have the access permissions granted by the default root account.

- After switching developers, the sub-account will have the access permissions authorized by the root account it switches to, and the access permissions granted by the root account before the switch will become invalid immediately.

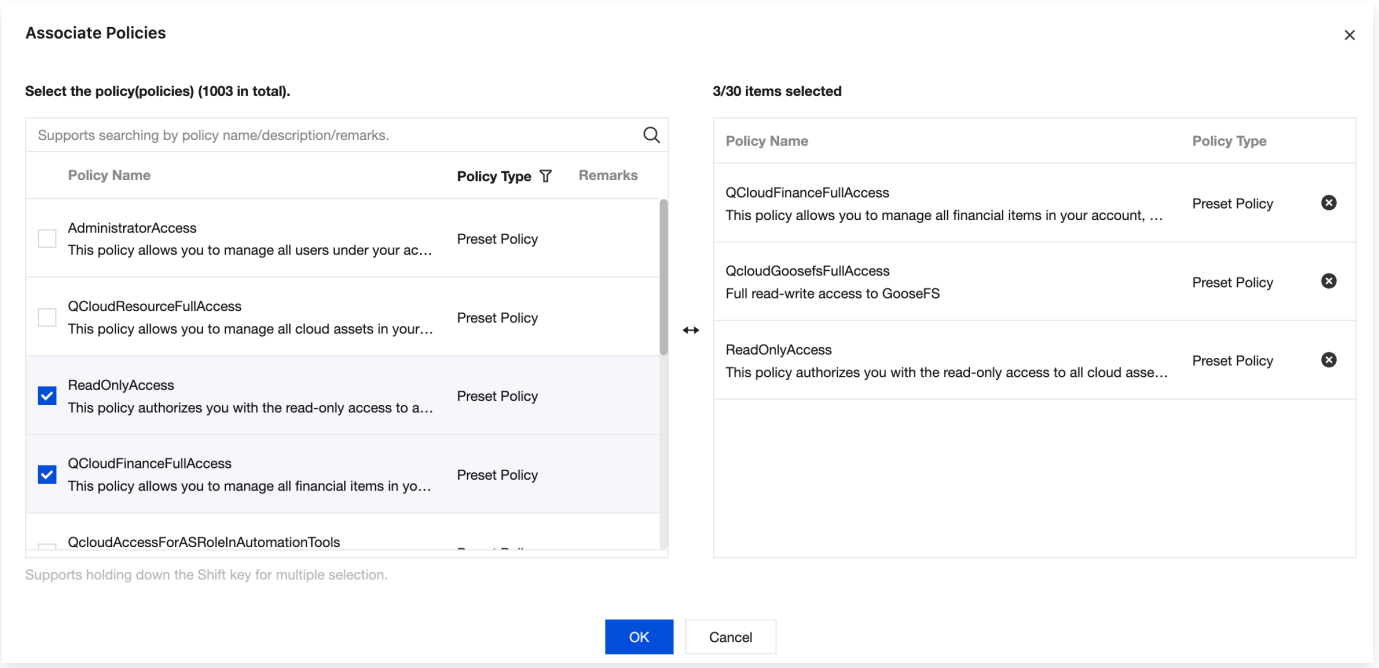
- A user group is a collection of multiple users (sub-accounts) with the same function. You can create different user groups based on business needs and associate appropriate policies with user groups to assign different permissions.

Authorize Sub-Accounts to Read-Write GooseFSx Permissions

1. Log in to the CAM console using the Tencent Cloud Main Account.
2. Select User > User List to enter the user list page.
3. Select the sub-user that requires authorization, and click the Authorize operation on the right. If there is no sub-user, please refer to the Create Sub-user document to create one.



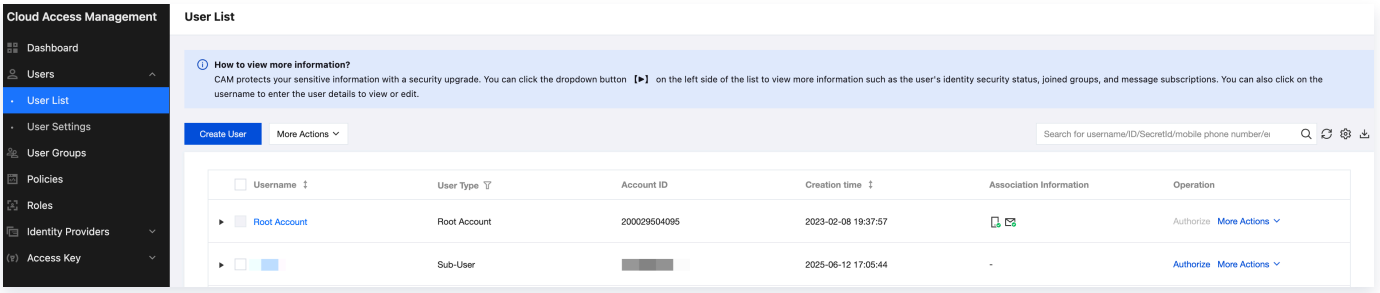
4. In the Associate Policy window, search for and select the following preset policy, then click OK.
   - QcloudCamReadOnlyAccess
   - QcloudVPCReadOnlyAccess
   - QcloudCVMReadOnlyAccess
   - QcloudTATReadOnlyAccess
   - QcloudAPIReadOnlyAccess
   - QcloudGoosefsFullAccess
   - QCloudFinanceFullAccess



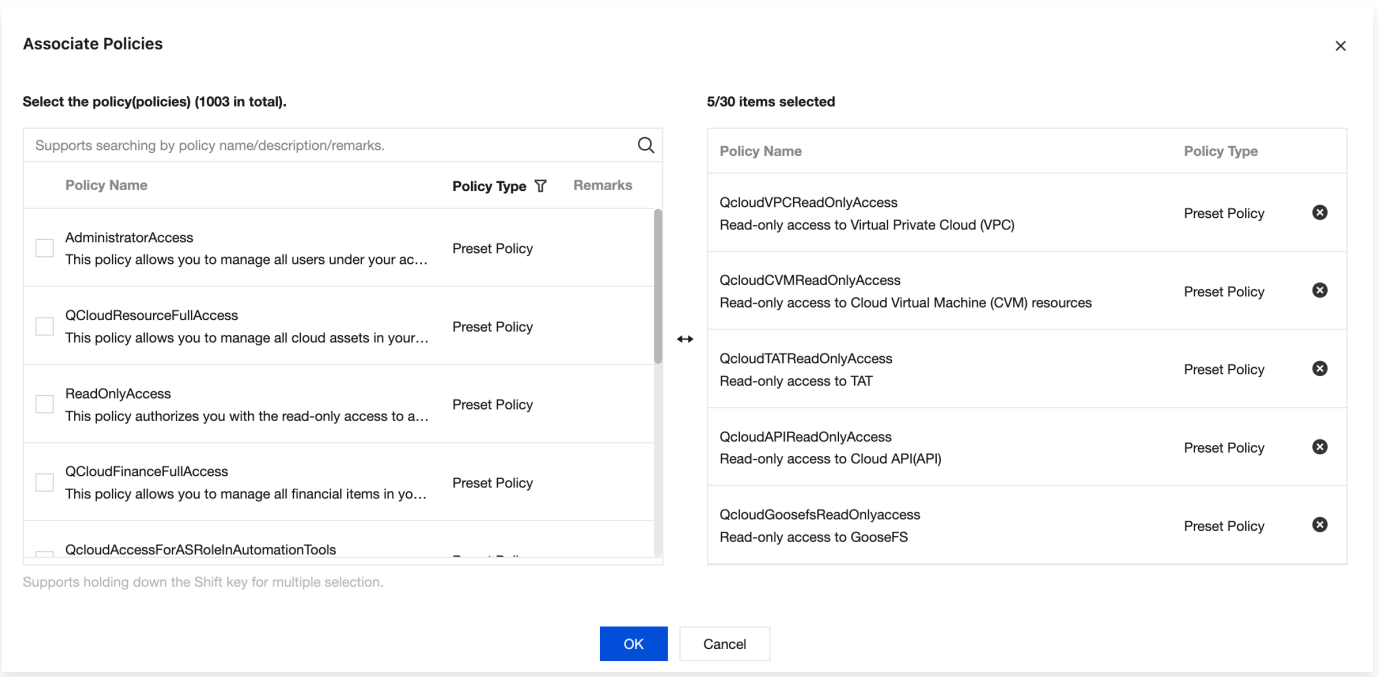
5. After authorization is completed, log in using a sub-account.

Authorize Sub-Accounts to Read GooseFSx Permissions

1. Log in to the CAM console using the Tencent Cloud Main Account.
2. Select User > User List to enter the user list page.
3. Select the sub-user that requires authorization, and click the Authorize operation on the right. If there is no sub-user, please refer to the Create Sub-user document to create one.



4. In the Association Policy window, search and select the following preset policies, then click Confirm to complete.
   - QcloudVPCReadOnlyAccess
   - QcloudCVMReadOnlyAccess
   - QcloudTATReadOnlyAccess
   - QcloudAPIReadOnlyAccess
   - QcloudGoosefsReadOnlyaccess



5. Authorization completed, just log in with a sub-account.

Service Level Agreement

Last updated：2025-07-17 17:16:41

For details, see Tencent Cloud Data Accelerator GooseFSx Service Level Agreement.

Glossary

Last updated：2025-07-17 17:16:42

Data Flow

Data flow refers to the on-demand movement of data between the data accelerator GooseFSx and Cloud Object Storage (COS). GooseFSx directly loads data from COS so that hosts can access cached data in GooseFSx at high speed. GooseFSx sinks the generated computation results into COS for persistent, low-cost storage.

Data Flow Task

A data flow task refers to executing a one-time data transfer between a GooseFSx associated directory and a COS associated bucket. It automatically, completely, and efficiently transfers your specified data (for example: data from a GooseFSx associated directory or a certain subdirectory, or data from a COS associated bucket or a certain prefix) to the destination. It then outputs a data flow task report and displays the details of the data flow task.
Before initiating a data flow task, please first associate a bucket by linking a COS bucket to a GooseFSx directory, and specify the data flow from a certain COS bucket to a certain GooseFSx directory.

POSIX Client

GooseFSx POSIX client refers to a host where GooseFSx POSIX client software is deployed and GooseFSx is mounted as a local directory. The POSIX client accesses the mounted directory, just like accessing a local file system.