TaurusDB

Service Overview

Issue 02

Date 2025-07-10

HUAWEI

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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

Notice

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

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

Huawei Cloud Computing Technologies Co., Ltd.

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

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

Contents

1 What Is TaurusDB?

TaurusDB is an enterprise-grade cloud-native database fully compatible with MySQL. It decouples compute from storage and supports up to 128 TB of storage per instance. With TaurusDB, a failover can be completed within seconds. It provides the high availability and superior performance of a commercial database at the price of an open-source database.

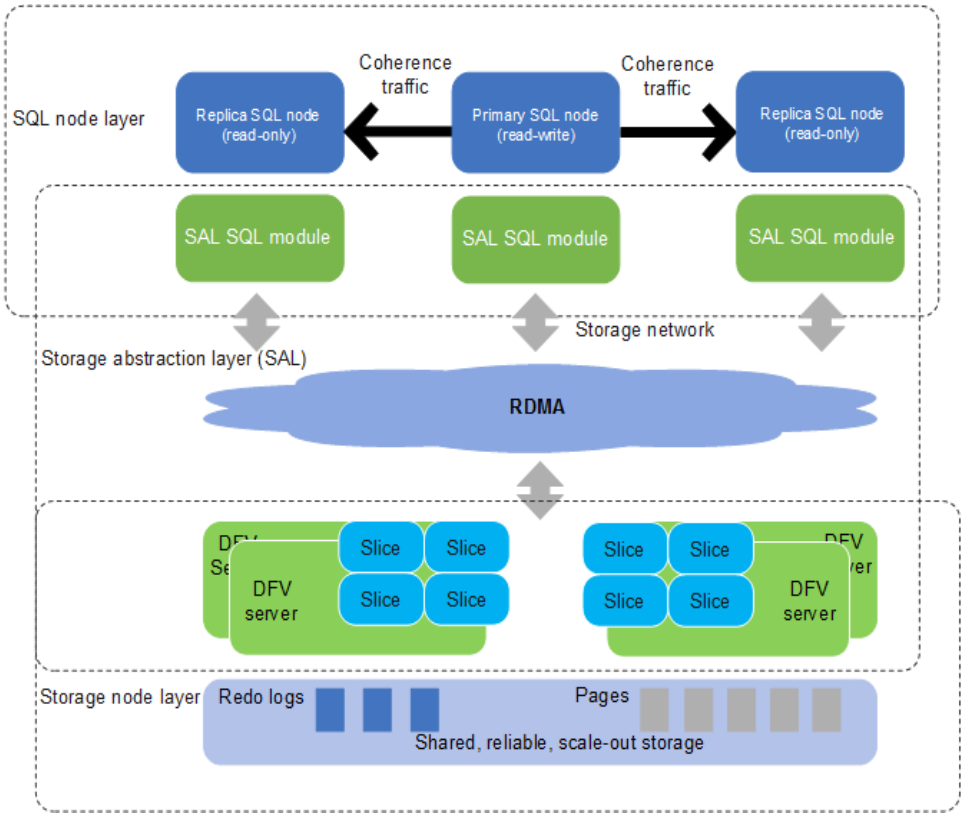
For details about the DB engines and versions supported by TaurusDB, see **DB Engines and Versions**.

2 Product Architecture

The TaurusDB architecture consists of three layers. From bottom to top, they are:

1. Storage node layer. Data Function Virtualization (DFV) storage provides distributed, strong-consistency, and high-performance storage, featuring high data reliability and excellent horizontal scalability. The data reliability is no less than 99.999999999% (11 nines). DFV is a solution for a data-centric full-stack service architecture with decoupled storage and compute.
2. Storage abstraction layer. This layer is key to ensuring database performance. It connects to the DFV storage pool below it and provides semantics upward for ensuring efficient storage scheduling. Table file operations are abstracted into distributed storage.
3. SQL parsing layer: 100% compatible with open-source MySQL 8.0. You can use MySQL-native syntax and tools to migrate your workloads from MySQL to TaurusDB, saving your time and efforts. In addition to full compatibility with MySQL, TaurusDB comes with an optimized kernel and a hardened system.

Figure 2-1 Product architecture



3 Basic Concepts

Cluster Instances

TaurusDB uses a decoupled compute and storage architecture that auto-scales up to 128 TB per DB instance. A cluster DB instance contains one primary node and up to 15 read replicas which can be created in minutes.

Single-Node Instances

A single-node instance contains only one primary node and there are no read replicas. Single-node instances do not involve data synchronization between nodes and can easily ensure atomicity, consistency, isolation, and durability of transactions.

Instance Specifications

Each instance is configured with compute and memory resources, for example, 16 vCPUs and 64 GB.

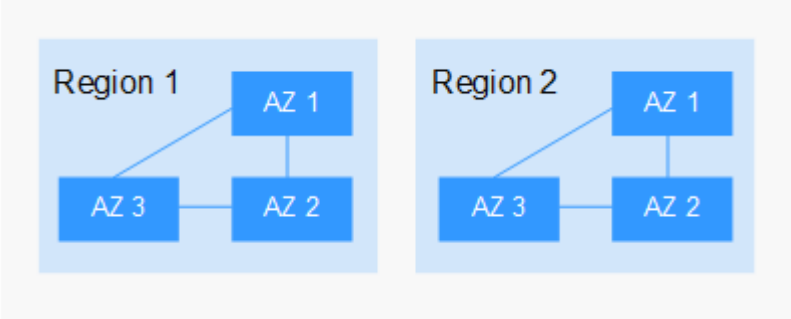
Regions and AZs

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

- A region is a geographic area where physical data centers are located. Each region is isolated from the other regions, improving fault tolerance and stability. After a resource is created, its region cannot be changed.
- An AZ is a physical location where resources use independent power supplies and networks. A region can contain multiple AZs, which are physically isolated but interconnected through internal networks. This ensures the independence of AZs and provides low-cost and low-latency network connections.

Figure 3-1 shows the relationship between regions and AZs.

Figure 3-1 Regions and AZs



4 Product Benefits

TaurusDB is an enterprise-grade cloud database with decoupled compute and storage.

Performance

TaurusDB can deliver seven times the performance of open-source MySQL for certain service loads and achieve millions of queries per second (QPS).

Scalability

- Horizontal scaling: You can add up to 15 read replicas within minutes as required.
- Vertical scaling: You can change the vCPUs and memory of instances to process uncertain workload growth.
- Storage scaling: The storage automatically grows as the amount of data in your database increases. An instance supports up to 128 TB of storage.

Reliability

TaurusDB supports cross-AZ and remote disaster recovery for financial-grade reliability.

There are three data copies to ensure data security.

Compatibility

TaurusDB is fully compatible with MySQL, so there is no need to reconstruct applications.

Middleware-free architecture

When the service performance is normal, Distributed Database Middleware (DDM) is not required.

5 Instance Description

5.1 Instance Specifications

Table 5-1 x86 instance specifications

Instance Specifications	vCPUs	Memory (GB)	Maximum Connections
Dedicated	4	16	5,000
	4	32	10,000
	8	32	10,000
	8	64	10,000
	16	64	18,000
	16	128	18,000
	32	128	30,000
	32	256	30,000
	60	256	60,000
	64	512	60,000

The DB instance specifications vary according to site requirements.

5.2 Instance Statuses

Instance Statuses

The status of an instance reflects the health of the instance. You can view the instance status on the console.

Table 5-2 Instance statuses

Status	Description
Available	The instance is available.
Abnormal	The instance is abnormal.
Creating	The instance is being created.
Creation failed	The instance failed to be created.
Rebooting	The instance is being rebooted.
Changing a DB instance name	The instance name is being changed.
Changing port	The instance port is being changed.
Changing instance specifications	The CPU and memory specifications of the instance are being changed.
Adding read replicas	Read replicas are being added to the instance.
Deleting a read replica	The read replica is being deleted from the DB instance.
Promoting to primary	A read replica is being promoted to primary.
Creating	The backup is being created.
Changing certificate settings	The certificate settings of the instance are being changed.
Deleted	The instance has been deleted and will not be displayed in the instance list.

5.3 DB Engines and Versions

The DB engine version supported by TaurusDB is TaurusDB V2.0.

Table 5-3 DB engines and versions

DB Engine	Compatible With
TaurusDB	MySQL 8.0

5.4 Storage Types

TaurusDB provides two types of storage: Cloud Database Engine Level 6 (DL6) and Cloud Database Engine Level 5 (DL5). Standard Edition instances use DL5 by default, while Enterprise Edition instances support both DL6 and DL5.

This section describes the differences between the two storage types, helping you choose the one that best suits your needs.

Storage Type Description

Table 5-4 Storage type description

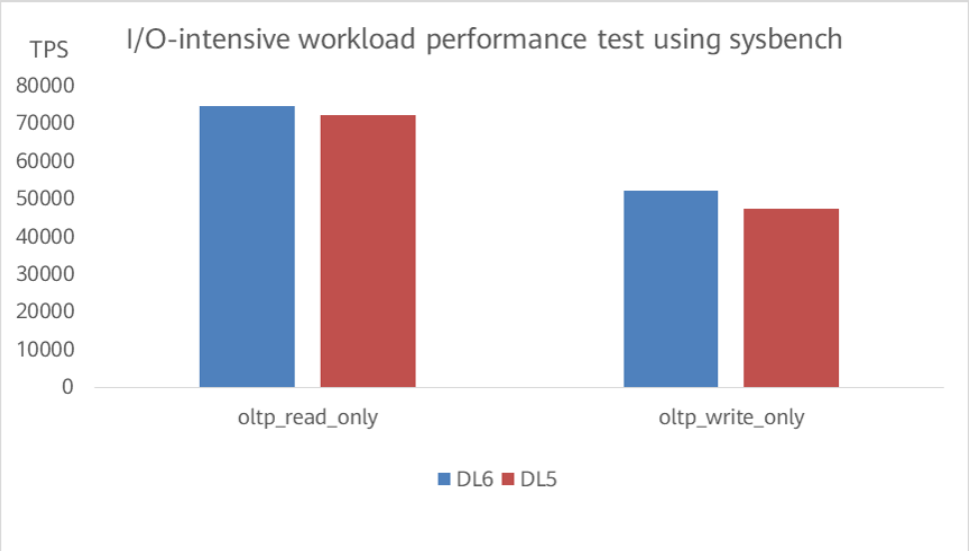
Storage Type	Description	Applicable Scenario
DL6	The shared storage is the default storage type for TaurusDB instances created before July 2024. DL6-based instances achieve zero RPO with a 3-AZ deployment and deliver better performance and higher peak throughput.	Core application systems that are sensitive to performance and have demanding requirements on storage I/O during peak hours, such as those in finance, e-commerce, government, and gaming
DL5	The new storage type uses hardware and network infrastructure technologies, ensuring that DL5-based instances maintain the same high availability (zero RPO in the 3-AZ deployment) as DL6-based instances. Although the peak performance of DL5-based instances may be a bit less than what you get with DL6-based instances, the cost per unit of capacity is a lot less.	CPU-intensive sub-core business systems, or application modules that need to minimize costs

> **NOTE**
>
> As the two storage types rely on different physical media, you cannot change the storage type for an existing instance. To change the storage type, you are advised to purchase a new TaurusDB instance, select the desired storage type, and migrate data from the original instance to the new instance using DRS.

Performance Comparison

When DL6- and DL5-based instances with the same compute specifications and I/O-intensive workloads were compared using sysbench, there was only an about 3% difference in read performance and less than 10% difference in write performance.

Figure 5-1 Performance comparison results



6 Permissions

If you need to assign different permissions to personnel in your enterprise to access your TaurusDB resources, Identity and Access Management (IAM) is a good choice for fine-grained permissions management. IAM provides identity authentication, permissions management, and access control, helping you securely manage access to your resources.

With IAM, you can create IAM users and assign permissions to control their access to specific resources. For example, if you want some software developers in your enterprise to use TaurusDB resources but do not want them to delete TaurusDB resources or perform any other high-risk operations, you can create IAM users for the software developers and grant them only the permissions required for using TaurusDB resources.

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

TaurusDB Permissions

New IAM users do not have any permissions assigned by default. You need to first add them to one or more groups and attach policies or roles to these groups. The users then inherit permissions from the groups and can perform specified operations on cloud services based on the permissions they have been assigned.

TaurusDB is a project-level service deployed in specific physical regions. To assign TaurusDB permissions to a user group, specify the scope as region-specific projects and select projects for the permissions to take effect. If **All projects** is selected, the permissions will be granted to the user group in all region-specific projects. When accessing TaurusDB, the users need to switch to a region where they have been authorized to use this service.

You can grant permissions by using roles and policies.

- Roles: A coarse-grained authorization strategy provided by IAM to assign permissions based on users' job responsibilities. Only a limited number of service-level roles are available for authorization. Cloud services depend on each other. When you grant permissions using roles, you also need to attach any existing role dependencies. Roles are not ideal for fine-grained authorization and least privilege access.
- Policies: A fine-grained authorization strategy that defines permissions required to perform operations on specific cloud resources under certain

conditions. This type of authorization is more flexible and is ideal for least permission access.

Table 6-1 lists all the system-defined permissions for TaurusDB.

Table 6-1 System-defined roles and policies supported by TaurusDB

Policy	Description	Type
GaussDB FullAccess	Full permissions for TaurusDB	System-defined policy
GaussDB ReadOnlyAccess	Read-only permissions for TaurusDB	System-defined policy

Table 6-2 lists the common operations supported by each system-defined policy or role of TaurusDB. Choose appropriate system policies based on this table.

Table 6-2 Common operations supported by each system-defined policy or role of TaurusDB

Operation	GaussDB FullAccess	GaussDB ReadOnlyAccess
Creating an instance	Supported	Not supported
Deleting an instance	Supported	Not supported
Querying an instance list	Supported	Supported

Table 6-3 Common operations and supported actions

Operation	Action	Description
Modifying parameters in a parameter template	taurusdb:param:modify	-
Changing DB instance specifications	taurusdb:instance:modifySpec	-

Operation	Action	Description
Creating a DB instance	taurusdb:instance:create	To select a VPC, subnet, and security group, configure the following actions: vpc:vpcs:list vpc:vpcs:get vpc:subnets:get vpc:securityGroups:get To create an encrypted instance, configure the KMS Administrator permission for the project.
Creating a manual backup	taurusdb:backup:create	-
Querying backups	taurusdb:backup:list	-
Querying error logs	taurusdb:log:list	-
Rebooting a DB instance	taurusdb:instance:restart	-
Querying DB instances	taurusdb:instance:list	-
Creating a parameter template	taurusdb:param:create	-
Deleting a parameter template	taurusdb:param:delete	-
Modifying a backup policy	taurusdb:instance:modifyBackupPolicy	-
Viewing parameter templates	taurusdb:param:list	-
Deleting a DB instance	taurusdb:instance:delete	-
Deleting a manual backup	taurusdb:backup:delete	-
Querying project tags	taurusdb:tag:list	-
Applying a parameter template	taurusdb:param:apply	-
Adding or deleting project tags in batches	taurusdb:instance:dealTag	-
Changing quotas	taurusdb:quota:modify	-
Upgrading a DB instance version	taurusdb:instance:upgrade	-

Operation	Action	Description
Promoting a read replica to the primary node	taurusdb:instance:switchover	-
Changing a database port	taurusdb:instance:modifyPort	-
Changing a security group	taurusdb:instance:modifySecurityGroup	-
Changing the private IP address	taurusdb:instance:modifyIp	To select an IP address, configure the following actions: vpc:vpcs:list vpc:vpcs:get
Enabling or disabling SSL	taurusdb:instance:modifySSL	-
Changing an instance name	taurusdb:instance:rename	-
Adding read replicas	taurusdb:instance:addNodes	-
Deleting read replicas	taurusdb:instance:deleteNodes	-
Scaling storage space	taurusdb:instance:modifyStorageSize	-
Changing a DB instance password	taurusdb:instance:modifyPassword	-
Binding an EIP to a DB instance	taurusdb:instance:bindPublicIp	To display EIPs on the console, configure: vpc:publicIps:get vpc:publicIps:list
Unbinding an EIP from a DB instance	taurusdb:instance:unbindPublicIp	-
Modifying a monitoring policy	taurusdb:instance:modifyMonitorPolicy	-
Changing a failover priority	taurusdb:instance:modifySwitchoverPriority	-
Changing the maintenance window	taurusdb:instance:modifyMaintenanceWindow	-
Isolating nodes	taurusdb:instance:isolateNodes	-
Creating database users	taurusdb:user:create	-

Operation	Action	Description
Deleting database users	taurusdb:user:delete	-
Changing the password of a database user	taurusdb:user:modify	-
Querying database users	taurusdb:user:list	-
Authorizing database permissions to users	taurusdb:user:grantPrivilege	-
Revoking database permissions from users	taurusdb:user:revokePrivilege	-
Creating databases	taurusdb:database:create	-
Deleting databases	taurusdb:database:delete	-
Querying databases	taurusdb:database:list	-
Querying predefined tags	-	To query predefined tags, configure the following action: tms:resourceTags:list

7 Constraints

To ensure the stability and security of TaurusDB, certain constraints are put in place for access or permission control. For details, see **Table 7-1** and **Table 7-2**.

Table 7-1 Function constraints

Function	Constraint
TaurusDB access	• If TaurusDB instances do not have EIPs bound, the instances must be in the same VPC subnet as the ECSs associated with these instances. • Security group rules must be added to allow ECSs to access TaurusDB instances. By default, a TaurusDB instance cannot be accessed by an ECS in a different security group. To enable access, you must add an inbound rule to the TaurusDB security group. When adding the rule, set the protocol and port, respectively, to TCP and to the default database port of the instance. • Default database port of TaurusDB: For cluster instances, the default port is **3306**. You can change it if you want to access TaurusDB through another port over a private or public network.
Database root permissions	Only the **root** permissions for cluster instances are available on the instance creation page.
Database parameter modification	Most parameters can be modified on the TaurusDB console.
Data migration	The mysqldump tool can be used to migrate data to TaurusDB.
TaurusDB instance reboot	TaurusDB instances can only be rebooted on the TaurusDB console.
TaurusDB backup files	TaurusDB backup files are stored in OBS buckets and are not visible to you.

Function	Constraint
Small-scale instances	For TaurusDB instances with 2 vCPUs and 8 GB memory, there are a maximum of 300,000 tables in a single instance and a maximum of 5,000 tables in a single database.

Table 7-2 Naming constraints

Parameter	Constraint
DB instance name	The name can contain 4 to 64 characters and must start with a letter. Only letters, digits, hyphens (-), and underscores (_) are allowed.
Database name	The name can contain 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. The total number of hyphens (-) cannot exceed 10.
Non-administrator account	The account can contain 1 to 32 characters. Only letters, digits, and underscores (_) are allowed.
Password	The password can contain 8 to 32 characters and must consist of at least three types of the following characters: uppercase letters, lowercase letters, digits, and special characters (~!@#$%^*-_=+?,()&).
Parameter template name	The template name can contain 1 to 64 characters. Only letters (case-sensitive), digits, hyphens (-), underscores (_), and periods (.) are allowed.
Backup name	The backup name can consist of 4 to 64 characters and must start with a letter. Only letters (case-sensitive), digits, hyphens (-), and underscores (_) are allowed.
Reserved keywords	In addition to the reserved keywords of MySQL 8.0 Community Edition, some other reserved keywords are added to TaurusDB. Do not use these keywords when naming objects to prevent errors.

8 Related Services

Table 8-1 shows the relationships between TaurusDB and other services.

Table 8-1 Related services

Service	Description
Elastic Cloud Service (ECS)	Enables you to access TaurusDB through an internal network. You can then access applications faster and you do not need to pay for public network traffic.
Virtual Private Cloud (VPC)	Isolates your networks and controls access to your TaurusDB instances.
Object Storage Service (OBS)	Stores automated and manual backups of your TaurusDB instances.
Cloud Trace Service (CTS)	Records operations on cloud service resources.
Data Replication Service (DRS)	Smoothly migrates databases to the cloud.
Tag Management Service (TMS)	Makes it simple for users to implement, manage, and maintain tags on cloud resources across regions and services.
Distributed Database Middleware (DDM)	Connects to multiple TaurusDB instances and allows you to access distributed databases.

9 Differences Between TaurusDB and RDS for MySQL

TaurusDB has good performance, scalability, and usability. For details, see **Table 9-1**.

Table 9-1 Differences between TaurusDB and RDS for MySQL

Item	RDS for MySQL	TaurusDB
Architecture	Traditional primary/standby architecture. Data is synchronized between the primary and standby nodes using binlog.	Decoupled storage and compute architecture. Compute nodes share the same data and data does not need to be synchronized using binlog.
Performance	Hundreds of thousands of QPS, delivering three times the performance of the open-source MySQL in high concurrency.	Millions of QPS, seven times the performance of open-source MySQL for certain service loads. In complex queries, operations, such as column extraction, conditional filtering, and aggregation calculation, can be pushed down to the storage layer, improving the performance by dozens of times compared with traditional databases.

Item	RDS for MySQL	TaurusDB
Scalability	• Up to five read replicas can be added for an instance. The time required for adding read replicas depends on the data volume. Adding read replicas require additional storage. • The storage can grow as needed, with up to 4 TB for an instance.	• Up to 15 read replicas can be added for an instance. Thanks to the shared storage, the time required for adding read replicas is not affected by the data volume. In addition, no additional storage is needed for read replica creation. • The storage grows as needed with up to 128 TB for an instance.
Availability	If the primary instance fails, the standby instance can be automatically promoted to the primary, with an RTO of less than 30s.	If the primary node is faulty, a read replica can be automatically promoted to the primary, with an RTO of less than 10s. It has lower latency because no data synchronization is required between the primary node and read replicas using binlog.
Backup restoration	Data can be restored to a specific point in time using full backups and binlog playback.	Data can be restored to a specific point in time using full backup (snapshots) and redo playback. Its restoration speed is faster.
DB engine version	MySQL 5.6, 5.7, and 8.0.	MySQL 8.0

