TDSQL-C for MySQL

Serverless Service

Product Documentation

Contents

Serverless Service

Serverless Introduction

Overview

Last updated：2025-05-28 14:15:24

TDSQL-C for MySQL Serverless adopts Tencent Cloud's proprietary serverless architecture for next-gen cloud-native relational database services. It is billed based on the actual computing and storage resource usage, so you only need to pay for what you use while benefiting from Tencent Cloud native technologies.

Background

In modern enterprises, databases play a critical role in IT systems. When a database is created, it is essential to carefully configure database cluster resources, including CPU, memory, storage, and various parameters, to ensure smooth business operations during both peak and off-peak periods. However, traditional resource allocation methods often come with inherent drawbacks: During off-peak periods, cluster resources remain idle, leading to wasted resources, while during peak periods, resource shortages may occur, affecting system performance.

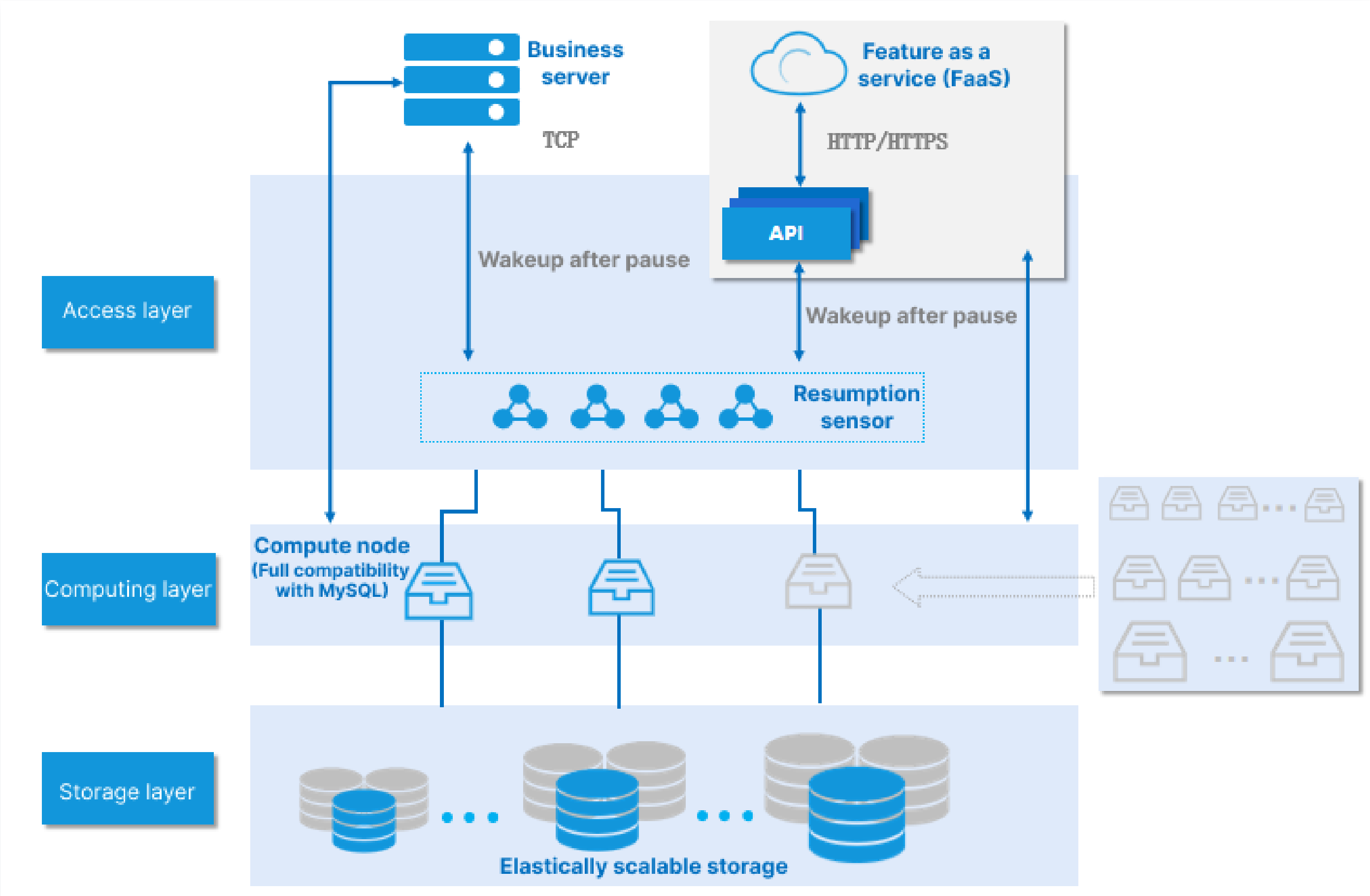
The Serverless service effectively addresses this issue by enabling database cluster resources to scale dynamically based on business loads. This eliminates the need for Ops personnel to perform complex resource evaluation and management, significantly reducing their workload.

TDSQL-C for MySQL is divided into **provisioned resource clusters** and **Serverless clusters** based on instance mode. In scenarios with significant business fluctuations, a Serverless cluster can efficiently handle low-load periods and traffic spikes while also reducing overall business costs.

- **Provisioned resource cluster**: Due to fixed specifications, if timely configuration adjustments are not made, it is easy to waste resources during service downtime and face resource shortages during peak periods.
- **Serverless cluster**: Since it allows precise control over resource scaling limits, continuously monitors workload metrics such as CPU and memory usage, and triggers automatic scaling policies based on predefined rules. Additionally, it supports automatic start and stop and elastic anti-jitter capabilities. As a result, using a Serverless cluster improves resource utilization and reduces costs.

> **Note:**
> The provisioned resource cluster now supports mounting Serverless read-only instances. You can reasonably deploy instance modes in the cluster based on your business needs. For details, see Enabling or Disabling Serverless for a Provisioned Resource Cluster.

Serverless service architecture



- Startup and shutdown on demand.
- Automatic scaling.
- Application-independent scaling.

Strengths of Serverless Service

- Autopilot: The database can automatically start/stop according to the business load and scale in an imperceptible manner without causing disconnections.
- Utility pricing: The database is billed based on the actual computing and storage resource usage. Computing fees are charged based on the number of CCUs, and storage fees are charged based on the amount of storage used (GB). The billing system is calculated per second and settled hourly.

Use cases

- Low-frequency database usage scenarios such as development and test environments.
- Scenarios where the load is uncertain, such as IoT and edge computing.
- SaaS application scenarios such as Mini Program Cloud Base and SME website development.
- Education scenarios such as experiment and teaching environment.
- Fully managed and Ops-free scenarios.
- Business scenarios with uncertain and intermittent fluctuations.

Documentation

Service Features

Last updated：2025-01-09 15:50:07

TDSQL-C for MySQL provides the serverless service to meet your database requirements in specific business scenarios, helping you reduce costs and increase efficiency. This document describes the major features of the serverless service.

Feature	Description
Resource scaling range (CCU)	You can adjust the range of elastic CCU scaling. The serverless cluster will automatically increase or decrease the number of CCUs within this range based on the actual business load.
Elastic policy	The serverless cluster will continuously monitor your CPU, memory, and other workloads and trigger automatic scaling policies according to certain rules.
Automatic start/stop	The serverless service allows you to customize the automatic pause time of the instance when there is no connection. When a task connection is established, the instance will be automatically started in seconds.

Resource Scaling Range (CCU)

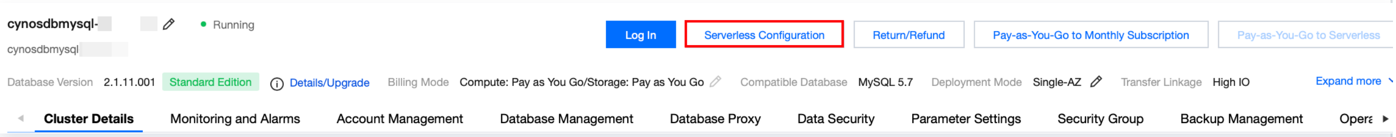
TDSQL-C Compute Unit (CCU) is the computing and billing unit for the Serverless Edition. A CCU is approximately equal to 1 CPU core and 2 GB memory. The number of CCUs used in each billing cycle is the greater of the number of CPU cores used by the database and 1/2 of the memory size.
You need to set the scaling range for the serverless service. For more information, see Compute Unit.
The minimum capacity configuration of the Serverless service supports setting to 0.25 CCUs. We recommend that when setting the elastic range for the first time, set the minimum capacity configuration to 1 CCUs and select a higher value for the maximum capacity configuration. A smaller capacity configuration allows your cluster to be reduced to the maximum extent when it is completely idle to avoid incurring additional costs. A larger capacity configuration can maximize expansion when your cluster load is too heavy, helping you to stably survive business peak periods.

Note:

- If your business requires fast scale-out to a very high capacity, consider increasing the value of the minimum capacity.
- To adjust the resource scaling range, log in to the console and make the change based on the actual view mode.

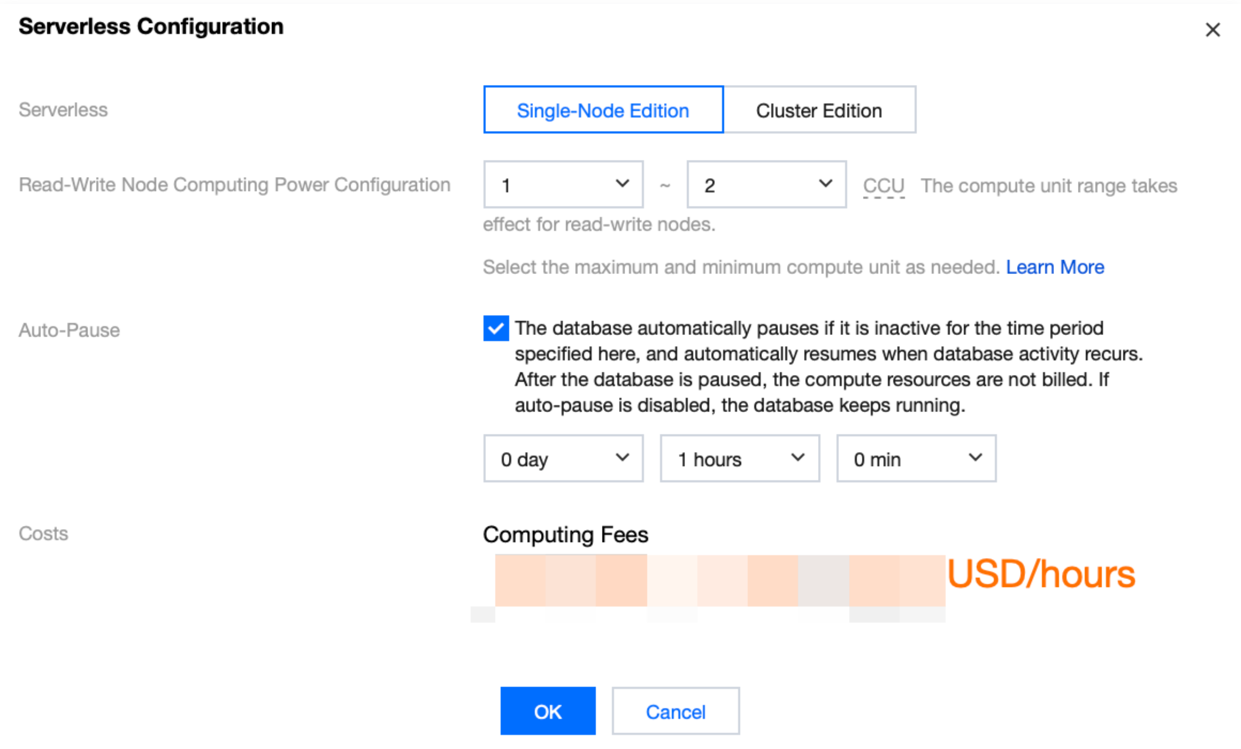
Tab view

Within the **Cluster Management Page**, click on **Serverless Configuration** located at the top right corner, and proceed to modify the computing power settings in the pop-up window. Adjustments will take effect immediately, seamlessly integrating without impacting operations.



List view

In the **Operation** column under **Cluster Management** > **Instance List**, click **More** > **Adjust Configurations**. Adjustments will take effect immediately, seamlessly without impacting operations.



Elastic Policy

The elastic policy of the serverless service is implemented by monitoring the computing layer. By monitoring the business load, the system automatically scales computing resources and bills the resources consumed at that moment. When there is no database request, the monitoring service will repossess computing resources and notify the access layer. When you access the cluster again, the access layer will wake up the cluster to make it accessible.

Initially, the elastic policy of the serverless service will limit the CPU and memory resources to the maximum specifications based on the capacity range you selected during purchase, greatly reducing the time impact and usage restrictions caused by CPU and memory scaling. When the cluster triggers the automatic scaling load threshold, the buffer pool will be adjusted within seconds in advance based on the monitoring data. Under this scheme, the CPU can be scaled in an imperceptible manner when you use the database, and no instance OOM events will occur due to the connection surge.

Note:

- When Serverless is used, due to its inherent flexible elasticity, cross-server scale-out may occur. To avoid the impact of cross-server scale-out on your business, it is recommended to use database proxy to prevent disconnection issues caused by cross-server momentary disconnection. It is also recommended that your business has a reconnection mechanism to reduce the impact of cross-server scale-out.
- Currently, only vertical scaling is allowed for read-only nodes, with no support for horizontal scaling.

Automatic Pause/Start

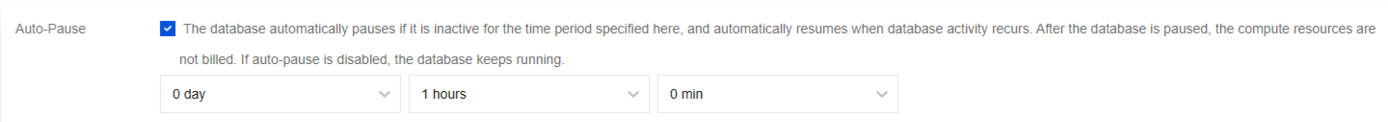
You can enable/disable the auto-pause feature in the console based on your business needs.

Note:
The serverless service will be automatically paused when there is no user connection. If your business needs to use `event_scheduler` to trigger SQL regularly, we recommend you not enable the auto-pause feature.

To enable or disable the auto-pause feature, you may perform the corresponding actions based on the actual view mode.

Tab view

Within the **Cluster Management Page**, click on **Serverless Configuration** located at the upper right corner. In the pop-up window, either check or uncheck the option for automatic suspension.



List view

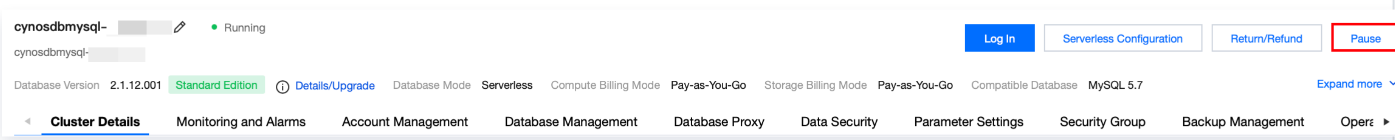
Click **More** > **Adjust Configurations** in the **Operation** column in **Instance List** > **Read-Write Instance** on the target cluster management page.

- After this feature is enabled, you need to set the auto-pause time, which is one hour by default. The database will be automatically paused if it has no active connections and CPU usage during this time. After the pause, the computing resources will not be billed, and the storage resources will be billed by the actual usage.
- If this feature is disabled, the database will keep running. When there are no active connections and CPU usage, the database will be billed based on the minimum CCU you configure. This is suitable for scenarios where your business has a heartbeat connection.

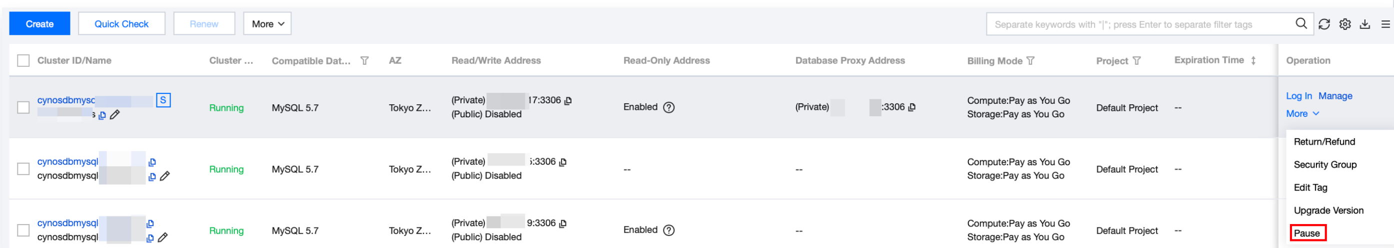
Manual Pause

You can also manually pause specified databases in the console based on the actual view mode.

Tab view



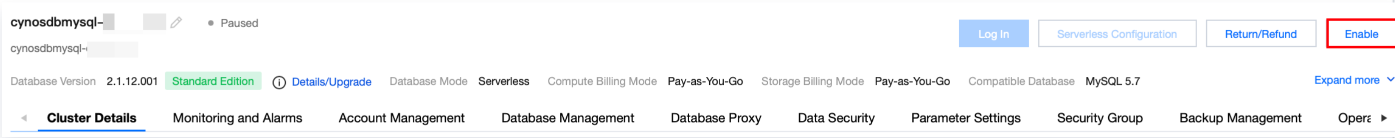
List view



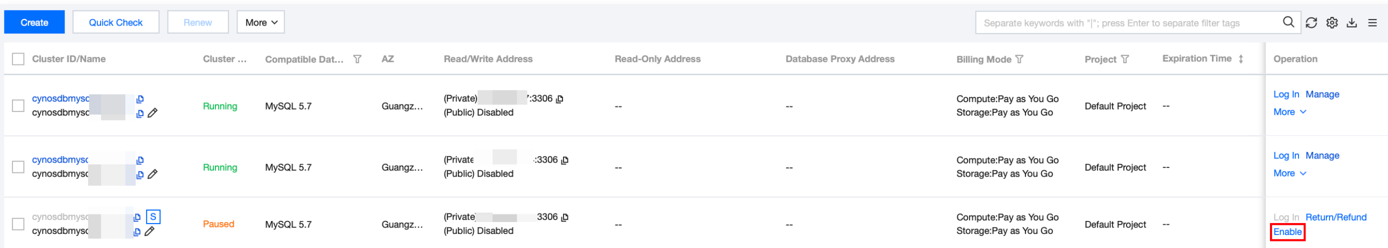
Manual Restart After Suspension

You cannot use the features in the console for a paused serverless database until it is automatically started or its serverless data is manually started based on the actual view mode.

Tab view



List view

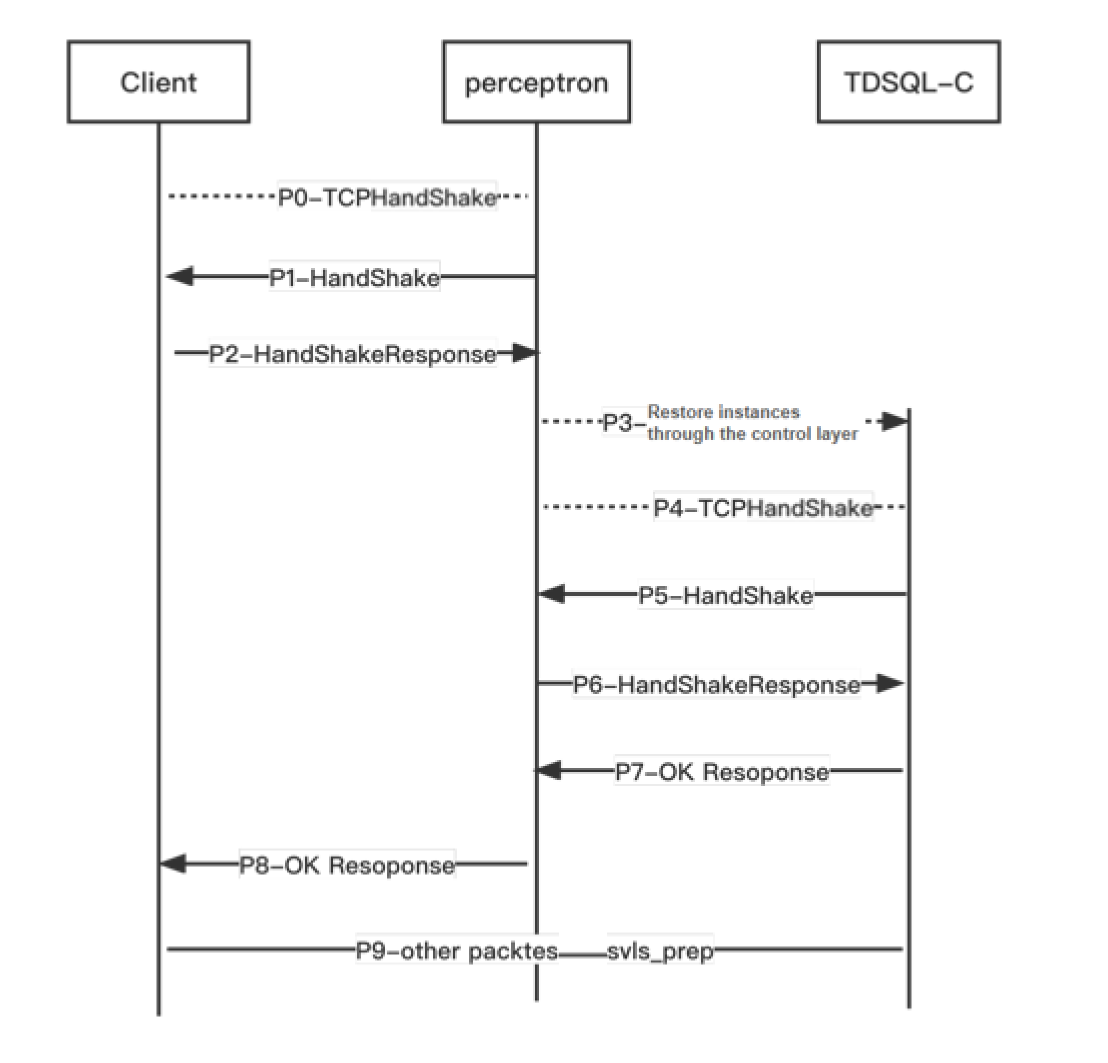


Forwarding requests without interrupting the connection

When a paused database is accessed, the system will automatically start it in seconds, so you don't need to configure a reconnection mechanism.

The access layer of TDSQL-C for MySQL has a resumption perceptron module to implement request forwarding. After the perceptron shakes hands with the client, the TDSQL-C for MySQL cluster will be resumed, without interrupting the user connection. Then, it will shake hands with the cluster and forward layer-4 packets.

The overall process uses two random challenge numbers for authentication, so that the perceptron can verify the username and password without storing them. This ensures the security of the user password and eliminates the inconsistency issue of stored passwords.



When the instance is paused, if a connection is initiated to it, the MySQL client will first perform a TCP handshake (P0) with the perceptron. After the TCP handshake is completed, the perceptron will send "random number A" to the client for challenge (P1). The MySQL client will use its own account and password and "random number A" to calculate and reply with its own "login response A" (P2). As the perceptron does not store the user's account and password, it cannot verify whether "login response A" is correct, but it can tell whether the client is a MySQL client or not (it is a classifier in the machine learning field, and distinguishing between different types of clients is one of the reasons why it is named perceptron). The verification of "login response A" will be completed by the computing layer of TDSQL–C for MySQL. After the perceptron wakes up TDSQL–C through control (P3), the next step of the login verification process will begin.

After handshaking with perceptron TCP (P4), TDSQL–C will regard the perceptron as a general MySQL client, so it will send "random number B" (P5) to the perceptron for challenge. The perceptron's reply is a special MySQL packet (P6). First, it uses "random number B" and its own authentication mechanism to calculate "login response B" and puts it into the packet; then, it adds "random number A" and "login response A" to the packet. TDSQL–C will perform two checks after receiving the special response packet: it will first check the correctness of "random number B" and "login response B" and the authenticity of the perceptron, and if the check is passed, it will check the correctness of "random number A" and "login

response A", and if the check is also passed, it will log in as the user and reply to the perceptron that the login is successful (P7). Then, the perceptron will reply to the user that the login is successful (P8).
When the cluster is paused, only the route of the perceptron will be retained. After the cluster is resumed, the system will retain the routes of both the perceptron and TDSQL-C and set the route weight of the perceptron to 0, so that new connections can be directly made to TDSQL-C, while existing connections to the perceptron can still communicate.

Elastic Anti-Jitter Capabilities

Last updated：2024-10-25 10:04:52

The Serverless service has elastic anti-jitter capabilities, which can help users better adapt to different workloads and performance requirements. This document shows you the elastic anti-jitter capabilities of TDSQL-C for MySQL instances in a Serverless cluster.

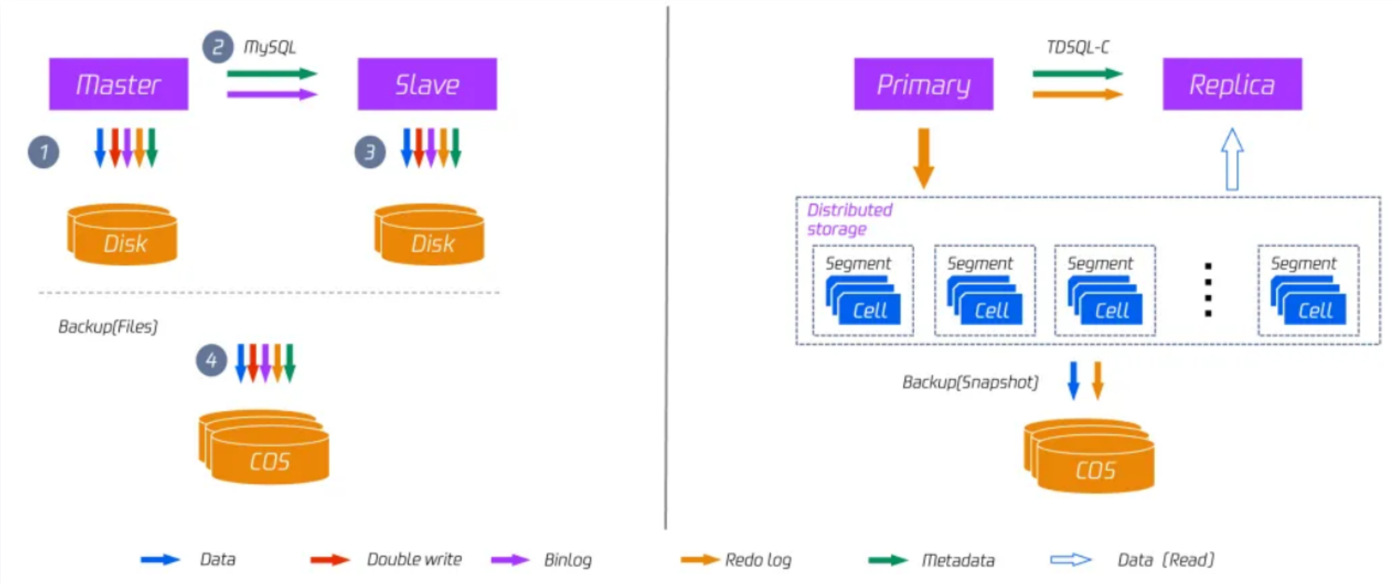
Background

In the InnoDB storage engine of TDSQL-C for MySQL, the Buffer Pool is a critical memory area for caching data and indexes. When querying data, it will first be searched in the Buffer Pool. If the data is in the Buffer Pool (that is, cached), InnoDB can immediately return the result, thus avoiding the overhead of the disk I/O. Therefore, correctly configuring the Buffer Pool size is particularly important for improving database performance. The Serverless architecture decouples the purchased instance specifications and can be automatically enabled, disabled and scaled based on the actual load of the user database, achieving extreme elasticity in computing resources. The computing resources mainly include CCU (CPU + memory). CPU can be limited by technologies such as cgroup or docker, while memory is allocated to the database process, with the majority used by the Buffer Pool module to cache user data. The allocation and release process of the Buffer Pool memory involves the distribution and relocation of user data and the mutual exclusion of global resources in the kernel. Correctly configuring the Buffer Pool size can enable the TDSQL-C for MySQL Serverless service to provide more stable elastic service capabilities.
MySQL officially supports the dynamic configuration of the Buffer Pool size, which is achieved by directly adjusting the parameter innodb_buffer_pool_size. The task will be completed in the background. If the parameter is changed again before the task is completed, it will be ignored. The scaling-out logic is relatively simple, while the scaling-in logic is relatively complex and is also where bottlenecks are prone to occur, such as I/O bottlenecks, free/lru list mutex bottlenecks, and global lock bottlenecks. Therefore, a series of optimizations at the kernel level have been made for the TDSQL-C for MySQL Serverless service, enabling the database to be "elastic" more stably.

Bottleneck Analysis and Optimization Scheme

I/O Bottleneck

During the test with the official MySQL 8.0, the kernel team has found that the main bottleneck of scaling-in is flushing the lru list, because in most scenarios, the first scan of the free list cannot meet the reclaiming requirements, and the scan depth is determined based on the number of blocks to be reclaimed, which may be a relatively large value. For buf_flush_do_batch, page cleaner and page persistence are needed, during which the lru mutex will also frequently be acquired and released, competition with user threads will occur, and glitches will be generated. Page persistence involves I/O operations, which is the main bottleneck. This issue can be completely avoided in the TDSQL-C for MySQL architecture, because the pages in the distributed storage are asynchronously generated by applying redo logs at the storage layer. For computing nodes, page cleaner is not needed, and the pages that need to be eliminated can be directly discarded. The product architecture is shown in the figure below.



Free/lru List Mutex Bottleneck

In the main process of scaling-in, the free list and the lru list are traversed in each loop, and the corresponding mutex is held during the traversal. At this time, user threads for both reading and writing operations are inaccessible and it may be necessary to acquire the free/lru list mutex. All blocks in the Buffer Pool are maintained on these two linked lists, so the traversal process has an O(N) (N represents the number of blocks) time complexity. This value may be very large, and the mutex will be continuously held for a long time, causing glitches for users.

Optimization Scheme

The optimization policy is to traverse the blocks in the chunk that need to be reclaimed by address. The number of blocks traversed in this way is related to the size of the scaling-in and does not depend on the size of the entire Buffer Pool. The locking range changes from the entire lru linked list to a single block. This will reduce the lock holding range and the lock holding time.

Global Lock Bottleneck

Whether performing scaling-out or scaling-in operations, there is a logic that requires acquiring the global lock of the Buffer Pool. During this period, the Buffer Pool is almost unavailable to users. If this step takes too long, it can also lead to users experiencing a temporary performance drop, known as a "glitch". Through analysis, we have found that there are three main stages in this process that are time-consuming:

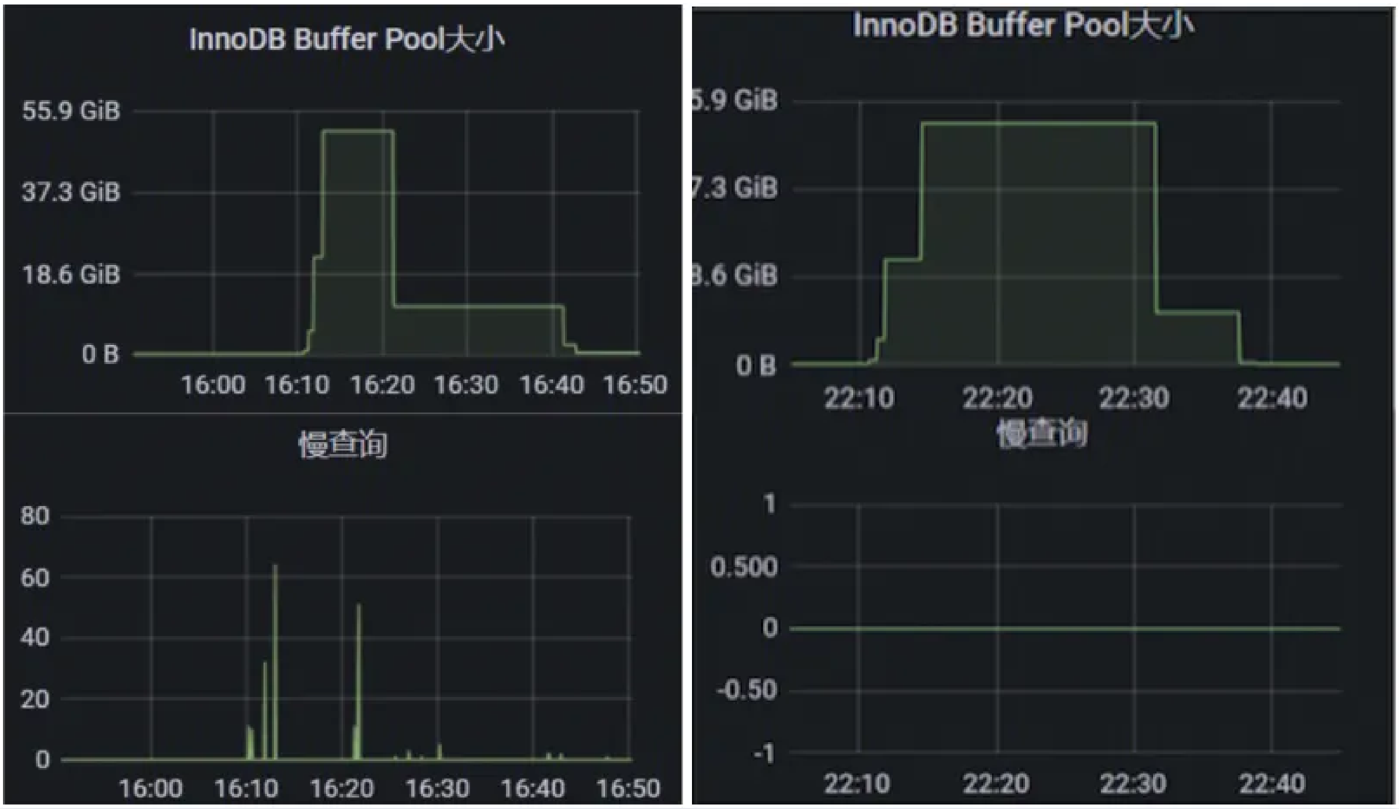
- Reclaim chunks memory and free blocks mutex.
- Allocate chunks memory and initialize blocks.
- Resize Hash.

Optimization Scheme

For the first two stages, the kernel team adopts the policy of delaying the release of chunks and pre-allocating chunks in advance, so that the main work can be done outside the buffer pool mutex. The original complexity is O(N), and the initialization of blocks and free blocks result in N being the number of blocks. After optimization, N in O(N) is the number of chunks, and the complexity is relatively controllable. For the Resize Hash stage, this is essentially a Rehash issue. The fundamental solution is to optimize the algorithm, such as lock-free Hash or consistent Hash allocation. The Hash Table is a fundamental element in InnoDB. If this complexity is changed, the risk will be high and the cycle will be long. If the Hash Table is too large, space will be wasted and many cells will not be used. If it is too small, there will be many Hash collisions, impacting performance. The kernel team adopts the policy of exchanging space for time, making the trigger frequency configurable, and reducing or not triggering Resize Hash within a certain scaling range.

Optimization Effect

In the test, use sysbench oltp_read_only, set long_query_time = 0.1, and check the number of slow queries. The comparison before and after optimization is as follows.



Use Instructions

For the above bottlenecks and optimizations, in addition to the optimizations directly performed by the kernel team, we provide relevant parameter settings in the Resize Hash stage, allowing users to choose whether to scale the size of the Hash Table to prevent glitches. The main cause of glitches is the scaling of the size of the Hash Table, so it is possible to prevent glitches without updating the size of the Hash Table. The detailed introduction of the parameters is as follows.

Parameter Name	Re-enablement Required	Global Parameter	Default Value	Parameter Value Range	Description
innodb_ncdb_decrease_buffer_pool_hash_factor	Yes	No	2	0 | 2	Adjusting the frequency of the Buffer Pool size thread will reduce the Hash Table size. • A value of 2 indicates the

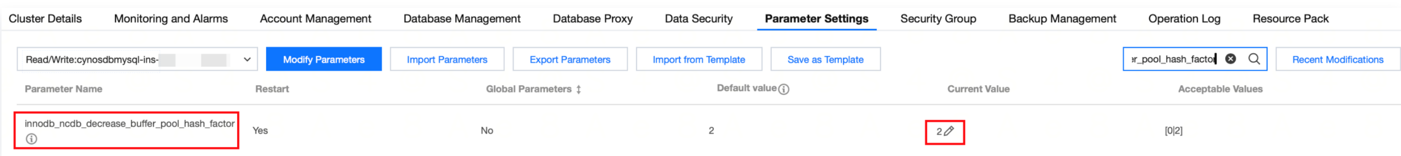
					disabled state, meaning that the Hash Table size will be updated when the Buffer Pool decreases by more than half. • A value of 0 indicates the enabled state, meaning that the Hash Table size will not be updated. **Note:** When this parameter is in the enabled state, changing the upper limit of node computing power setting will re-enable the instance.

Kernel Versions Supported by Parameters

- Kernel version TXSQL 5.7 2.1.12 and later.
- Kernel version TXSQL 8.0 3.1.14 and later.

Parameter Setting Method

For the setting method, see Setting Instance Parameters.



Billing Overview

Last updated：2025-10-21 18:21:46

This document describes the pricing of TDSQL-C for MySQL Serverless.

Billing Mode

In the serverless mode, computing and storage are billed separately: computing is billed by the number of CCUs, while storage is billed by the usage in GB. The billing system calculates the usage by second and settles fees by hour.

Billing Formula

Total serverless fees = compute node fees + storage space fees = serverless computing power price * number of CCUs + storage space price * storage space

Serverless Computing Power Pricing

Billing Unit	CCU Pay-per-Use Pricing (USD/unit/second)	
	Supported regions: Guangzhou, Shanghai, Beijing, and Nanjing Regions to be supported: Chongqing	Supported regions: Hong Kong (China), Singapore, Silicon Valley, Virginia, Seoul, Tokyo, and Jakarta Regions to be supported: Frankfurt
Serverless Instances	0.00001397	0.00001529

Note:

- TDSQL-C Compute Unit (CCU) is the computing and billing unit for the Serverless Edition. A CCU is approximately equal to 1 CPU core and 2 GB memory. The number of CCUs used in each billing cycle is the greater of the number of CPU cores used by the database and 1/2 of the memory size.
- You can refer to Compute Unit to select the corresponding maximum and minimum CCU values. The upper limits of storage space vary for different compute units. For details, refer to Compute Unit.

Storage Space Pricing

Serverless service edition cluster storage space is billed on a pay-as-you-go basis. In Guangzhou, Shanghai, Beijing, Nanjing, Chengdu, and Chongqing, the price is USD0.00072/GB/hour, while in Hong Kong

(China), Singapore, Silicon Valley, Virginia, Seoul, Tokyo, and Frankfurt, the price is USD0.000792/GB/hour.

Serverless Resource Package Overview

TDSQL-C for MySQL has launched the resource package service, allowing for the advance purchase of compute and Storage resources at lower unit prices, achieving cost savings. The following will introduce the definition and billing sequence of the resource package.

A resource package is a type of prepaid resource, divided into compute resource packages and storage resource packages. They can be used to offset the compute or Storage resources generated by a Serverless service edition cluster. After a user purchases and binds a resource package to a cluster, it will first consume the compute or storage resources from the resource package. When the resources are fully consumed within the validity period, you can purchase a new resource package and continue to bind it for use. If no new resource package is bound for deduction, the cluster will switch to a pay-as-you-go billing method. At that time, the fees incurred by using the cluster will be deducted from your Tencent Cloud account balance.

> **Note:**
> For easier understanding, you can compare the resource package to a mobile data plan package. Suppose the normal price for mobile data is USD1/GB, and a user spends USD10 to buy a 50 GB mobile data plan for one month at USD0.2/GB. Thus, the user can enjoy the discounted unit price of USD0.2/GB during the one-month validity period. Once the data plan is used up within the validity period, the user can continue to purchase new mobile data plan packages to enjoy the discounted price. If the user does not purchase and use new mobile data plan packages, the fees of data will revert to USD1/GB.

Notes on Arrears

Last updated：2025-05-28 10:24:55

This document introduces the overdue policy for Serverless cluster instances.

> **Note:**
> Your actual resource consumption may vary from time to time, so there might be some deviation in the balance alert.

Alerts

Charges for Serverless cluster resources are incurred on the hour. When your account balance falls below 0, the system will send a notification to the creator of the Tencent Cloud account and to the global resource collaborators and financial collaborators via email and Short Message Service (SMS), according to the subscription configuration in the Message Center.

Arrears Processing

1. **From the moment your account balance is less than 0:**
    - You can continue to use your TDSQL-C for MySQL cluster in 24 hours. The billing will continue for this period.
    - Your TDSQL-C for MySQL cluster will be automatically isolated and moved into the recycle bin after 24 hours, and the billing will stop.
2. **After it is isolated:**
    - If you top up your account within 3 days after the isolation to a positive balance, the billing will continue, and the cluster will be automatically recovered for normal use.
    - After it is isolated for 3 days, if your account balance is still below 0, the isolated cluster will be taken offline and placed into the recycle queue. Upon recycling, all data will be cleared and cannot be retrieved.
      When a cluster is being recycled, the system will send a notification to the creator of the Tencent Cloud account and to the global resource collaborators and financial collaborators via email and SMS, according to the subscription configuration in the Message Center.

Creating and Managing a Serverless Cluster
Creating Serverless Cluster

Last updated：2025-10-23 20:28:00

This document describes how to create a serverless cluster in the TDSQL-C for MySQL console.

Prerequisite

To make a purchase, you need to complete identity verification first. For more information, see Identity Verification Guide.

Directions

1. Log in to the purchase page to complete the **Database Configuration**, **Instance Selection**, and **Storage** settings, and then click **Next**.

Parameter	Description
Instance Mode	Select **Serverless**. Serverless cost estimation is supported to help you better understand whether the cost of using the Serverless service meets your budget requirements. For more details, see Serverless Cost Estimator.
Instance Engine	Select a transactional cluster. The instance engine for Serverless clusters can only be set to transactional clusters. TDSQL-C for MySQL supports two instance engines: transactional cluster and analysis cluster. The transactional cluster can provide HTAP capability by enabling a read-only analysis engine in conjunction with TXSQL read-write instances, while the analysis cluster only contains read-only analysis engine instances and synchronizes the configured data source data to the analysis engine via Zero-ETL capability.
Region	Select a region for database deployment. Currently, the serverless billing mode is supported only in Guangzhou, Shanghai, Beijing, Chengdu, Nanjing, Hong Kong (China), Silicon Valley, Virginia, Singapore,Tokyo and Seoul regions. If you need to use it in other regions, submit a ticket for assistance.
Source AZ	Select an AZ for deployment. Specific AZs in the selected region are shown on the actual purchase page.
Multi-AZ	Select whether to enable multi-AZ deployment. If Yes is chosen, the option for a

Deployment	secondary availability zone will appear.
Secondary Availability Zone	By default, this feature is disabled. Upon enabling multi-AZ deployment, you may select your standby availability zone. For regions and availability zones that support multi-AZ deployment, please refer to the Multi-AZ Deployment section. For the most up-to-date supported regions and availability zones, see the purchase page for accurate information.
Transfer Linkage	It is "High IO" by default.
Network	Select the network for a database instance. Currently, only Virtual Private Cloud (VPC) is supported. • For performance and security reasons, only devices within the selected VPC can access this database instance through the private network. Use the private network for business access. • For non-business access, such as testing or auxiliary management of a database, you can also access the database instance through the public network. For detailed instructions and notes, see Enable or Disable Public Network Connection.
Database Version	MySQL 5.7 and 8.0 are supported.
Compute Billing Mode	Resource pack (monthly subscription) and pay-as-you-go billing are supported. The compute resource pack will be used preferably for the deduction of actual usage of pay-as-you-go products. When the pack is used up, the resource usage will be pay-as-you-go. Compute resource packs are deducted based on the actual CCU used per second. The resource pack mode is more cost-effective and flexible than the pay-as-you-go option.
Compute resource pack (displayed when selecting **Resource Pack** in the compute billing mode)	Binding a resource pack allows you to associate one or more (up to a maximum of ten) compute resource packs with the cluster selected in the region under the current account, provided they are within their validity period. If there are no available resource packs, you may proceed to purchase a resource package first.
Serverles	Supports the selection of either a Single-Node or Cluster architecture.

s Architecture	• Single-node architecture: Within the cluster, there is only one read-write instance. Upon purchase, it supports switching to a clustered architecture via the console. • Cluster architecture: The cluster comprises one read-write instance and 1 to 15 optional read-only instances. Upon purchase, it supports switching to a single-node architecture via the console.
Compute Unit	Select the upper and lower limits of the TDSQL-C compute unit (CCU), and the instance will be automatically and elastically scaled within the selected resource range. CCU is the computing and billing unit for the serverless mode. A CCU is approximately equal to 1 CPU core and 2 GB memory. The number of CCUs used in each billing cycle is the greater of the number of CPU cores used by the database and 1/2 of the memory size. For more information, see Compute Unit. • Single-node architecture: It is only necessary to configure the computational power range for the read-write instance. • Cluster architecture: It is necessary to configure the computational power range for both read and write instances, followed by selecting the number of read-only instances within the read-only group (ranging from 1 to 15). Additionally, set the computational power range for nodes within the read-only group, which will be effective for all nodes in the group.
Auto-Pause	Configure the automatic pause time of the instance. If there is no connection to access the database within the set time, the instance will be automatically paused, with billing stopped.
Storage Billing Mode	Resource pack (monthly subscription) and pay-as-you-go billing are supported. The storage resource pack will be used preferably for the deduction of actual usage of pay-as-you-go products. When the pack is used up, the resource usage will be pay-as-you-go. Storage resource packs are deducted based on the actual storage used per hour. The resource pack mode is more cost-effective and flexible than the pay-as-you-go option.
Storage resource pack (displayed when selecting **Resource Pack** in the storage billing mode)	Binding a resource pack allows you to associate one or more (up to a maximum of ten) storage resource packs with the cluster selected in the region under the current account, provided they are within their validity period. If there are no available resource packs, you may proceed to purchase a resource package first.

2. Complete the **Basic Info** settings and **Advanced Configuration**, select the purchase **Quantity** (each user can purchase up to 50 TDSQL-C for MySQL clusters in each AZ), and then click **Buy Now** after the fees are confirmed.

- **Basic Info**

Parameter	Description
Cluster Name	Set a name for the cluster now or later. It must contain less than 60 letters, digits, or symbols (-_.).
Admin Username	It is `root` by default.
Password	The password can contain 8—64 characters in at least three of the following character types: uppercase letters, lowercase letters, digits, and symbols ~!@#$%^&*_-+=|\(){}[]:;',.?/.
Default Character Set	UTF8, GBK, LATIN1, and UTF8MB4 are supported.
Default collation rule	Select the default collation rule for the character set. Specific options are subject to the actual purchase page.
Custom Port	The default port is 3306, and custom input is supported. Port value range: 1024—65534.

- Advanced Configuration

Parameter	Description
Security Group	Select or create a security group.
Parameter Template	Select or create a parameter template.
Table Name Case Sensitivity	Select **Case-Insensitive** or **Case-Sensitive**.
Project	Specify a project for the cluster to be created.
Alarm Policy	Select or create an alarm policy.
Tag	Add a tag to facilitate resource categorization and management.
Terms and Conditions	Read and indicate your consent to the terms and conditions.

3. After the purchase is completed, you will be redirected to the cluster list. After the status of the cluster

becomes **Running**, it can be used normally.

Creating and Managing a Read-Only Instance

Last updated: 2025-05-28 14:47:58

Serverless clusters support mounting read-only instances to enhance read performance and concurrency while reducing the load on the read-write instance. This document describes how to create a read-only instance and modify its configuration for a serverless cluster.

Billing Description

The read-only and read-write instances have the same billing logic. For more information, see Billing Overview.

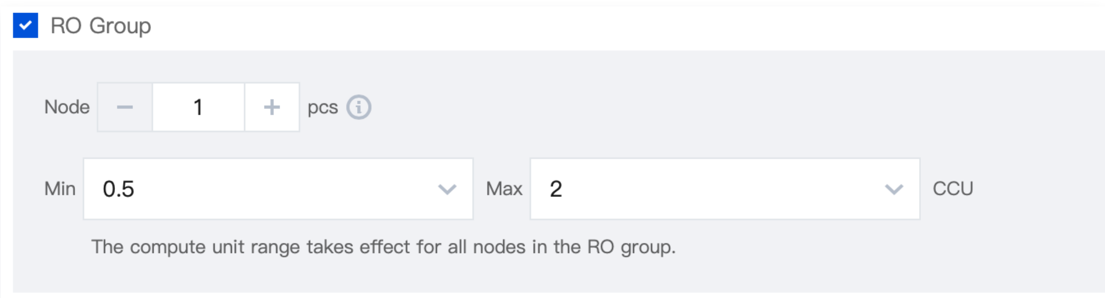
Creating Read-Only Instance

Note:

- Read-only instance on the current version only supports vertical scaling but not horizontal scaling.
- After purchasing a read-only instance group, you can adjust its compute unit and quantity.
- The computing power of read-write instances or read-only instances has no impact on each other.

Creating a read-only instance when purchasing a cluster

1. Log in to the TDSQL-C for MySQL console
2. Behind the **Serverless Architecture**, select the **Cluster Edition**.
3. Beneath the **Read-Only Group**, select the number of read-only nodes within the group. You can add a maximum of fifteen read-only nodes.



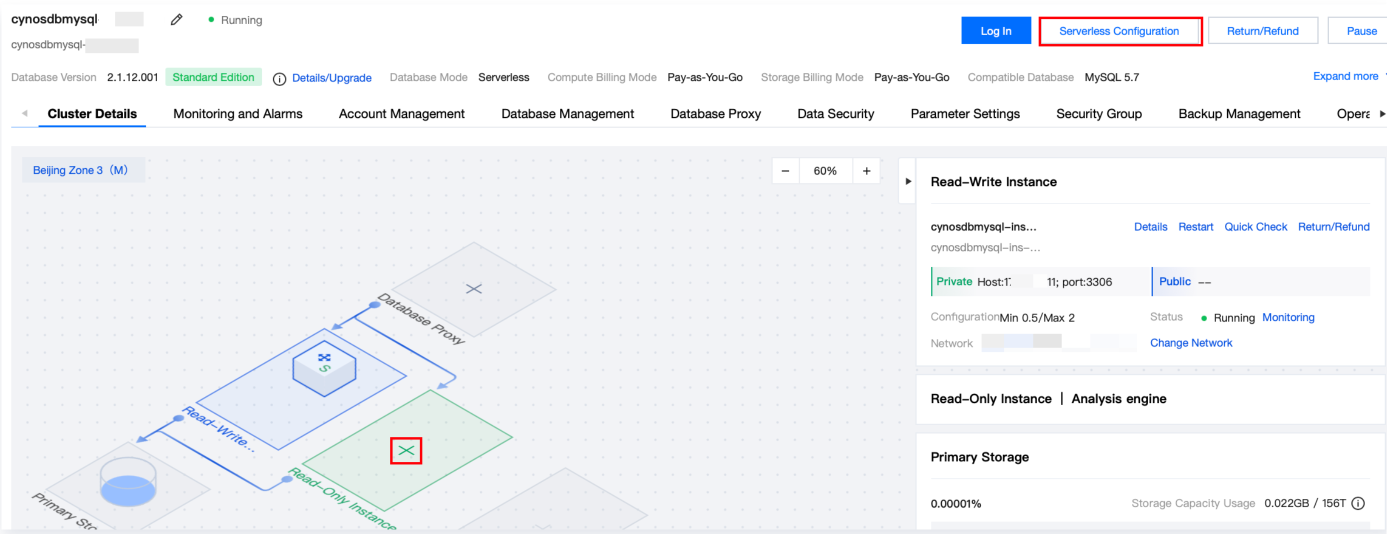
4. The configured compute unit range only takes effect for all the read-only nodes in the read-only group.
5. After configuring other configuration items for the serverless clusters, click **Buy Now**. For more information on the other configuration description of the serverless clusters, see Creating Serverless

Cluster.

Adding Read-Only Instances to Existing Serverless Clusters

> **Note:**
> You can only perform the operations on the cluster in the running status.

1. Log in to the TDSQL-C for MySQL console, and select a region at the top.
2. Click **Target Cluster** in the cluster list on the left to enter the cluster management page.
3. Within the **Cluster Management** page, click on the "+" icon located in the read-only instance area of the topology diagram, or click on **Serverless Configuration** at the top right corner.



4. Behind the **Serverless Architecture**, select the **Cluster Edition**.
5. Depending on the requirements, the number of read-only nodes within a read-only group can be increased, with a maximum of 15 read-only nodes allowable.
6. Configure the computational power range for the read-only group, which will be effective for all read-only nodes in the group.
7. Click **Buy Now** to add a read-only instance.

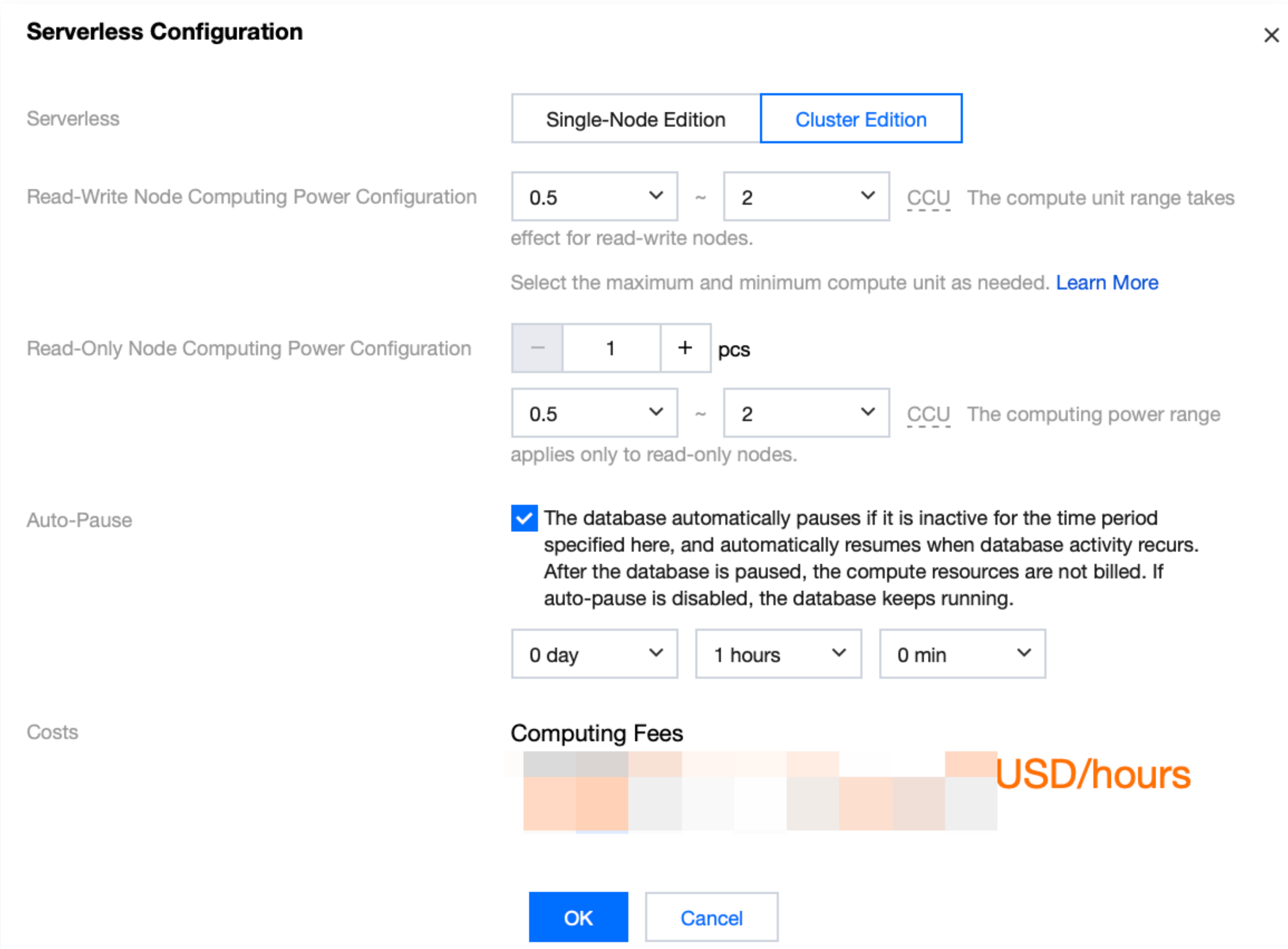
Modifying the configuration of the read-only instance

You can adjust the number of the read-only instances and the compute unit range of the read-only group.

> **Note:**
> - The cluster must be in a running state; configuration changes for read-only instances cannot be

made while it is in a paused state.
- After adjusting the number of read-only instances, you can manually trigger a load balancing to balance traffic.
- You can adjust the number of the read-only instances on the configuration modification page. But you can't add or delete a read-only instance manually.
- After a downgrade, the removed read-only instances will not be isolated in the Recycle Bin but will be eliminated directly. If you need to add read-only instances, please do so by adjusting the configuration to create read-only instances.
- When the database proxy is enabled, the business will not be interrupted during the configuration modification.

1. Log in to the TDSQL-C for MySQL console, and select a region at the top.
2. Click the target cluster in the cluster list on the left to enter the cluster management page.
3. Within the **Cluster Management** page, click on the "+" icon located in the read-only instance area of the topology diagram, or click on **Serverless Configuration** at the top right corner.
4. In the pop-up window, simply select the desired number of read-only nodes and the computational power range to complete the configuration changes for the read-only instance.



Setting Automatic or Manual Start/Stop

Last updated: 2025-05-28 14:27:23

This document introduces how to set automatic or manual start/stop for a Serverless Cluster via the console.

Prerequisites

You have Creating Serverless Cluster.

Setting Automatic Start/Stop

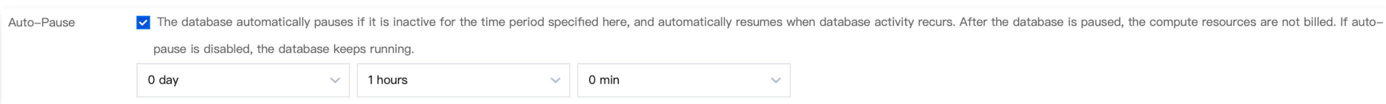
The Serverless service supports an automatic start/stop feature. When there are no connections, the instance automatically pauses. When a task connection is received, the instance resumes instantly without interruption. You can enable or disable this feature through the console.

> **Note:**
> - If auto-pause is enabled, you should set an auto-pause duration, which is 1 hour by default. When there are no active connections or CPU usage within this period, the database will automatically pause. After pausing, computing fees will not be charged, while storage fees will continue to be charged based on actual usage.
> - If auto-pause is disabled, the database remains running continuously. If there are no active connections or CPU usage, billing is based on the minimum CCU compute unit configured by the user. This is suitable for use cases that require persistent heartbeat connections.

Method 1: Enabling Automatic Start/Stop When a Cluster Is Created

You can enable automatic start/stop while creating a Serverless cluster by following these steps.

1. Log in to the purchase page.
2. Select **Serverless** as the instance mode.
3. In **Instance Selection** > **Auto-Pause**, check **The database automatically pauses if it is inactive for the time period specified here, and automatically resumes when database activity recurs. After the database is paused, the compute resources are not billed. If auto-pause is disabled, the database keeps running.**

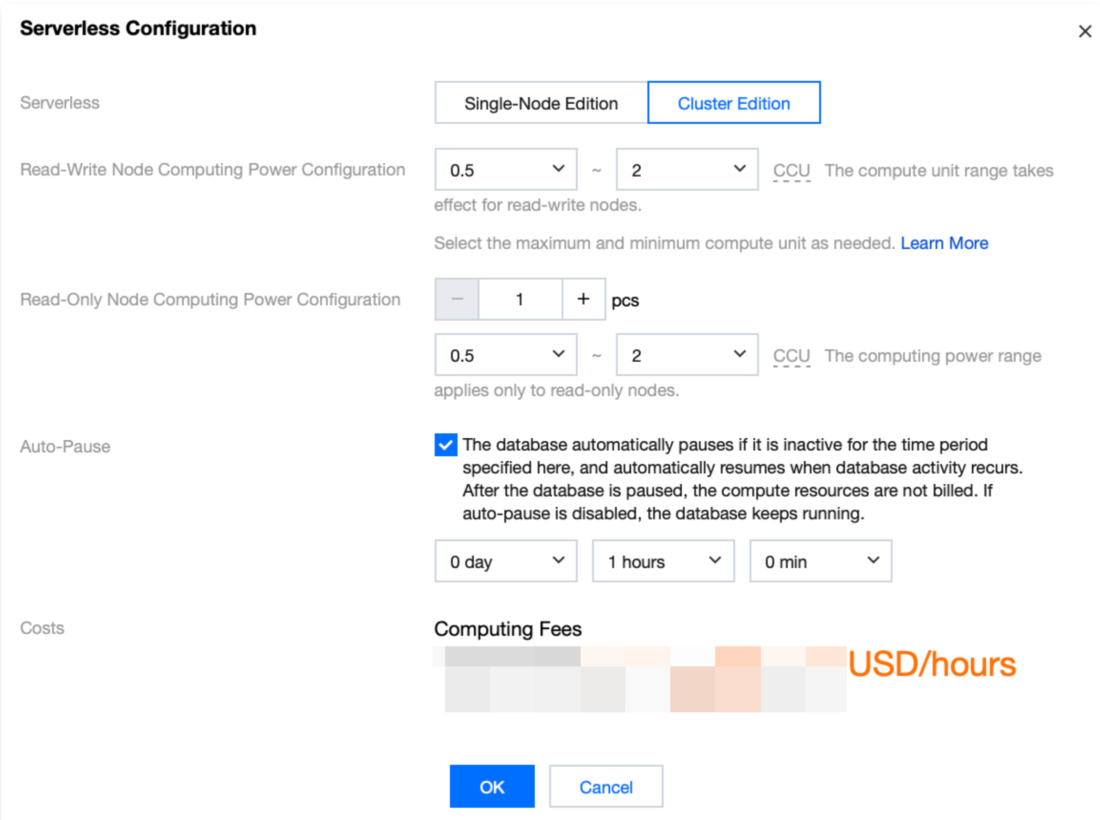


4. Set the auto-pause time (range: 0—7 days; 0—23 hours; 0, 10, 20, 30, 40, or 50 minutes).
5. Complete the remaining configuration and purchase the cluster.

Method 2: Enabling Automatic Start/Stop via the Console

If you did not check **The database automatically pauses if it is inactive for the time period specified here, and automatically resumes when database activity recurs. After the database is paused, the compute resources are not billed. If auto-pause is disabled, the database keeps running** when creating a cluster, it means auto start/stop is not enabled. You can refer to the following method to operate.

1. Log in to the TDSQL-C for MySQL console.
2. In the Cluster list on the left, click the **target cluster** to enter the cluster management page.
3. In the topology diagram, click the + icon in the read-only instance area, or click **Serverless Configuration** in the upper right corner.
4. In the pop-up, check "**The database automatically pauses if it is inactive for the time period specified here, and automatically resumes when database activity recurs. After the database is paused, the compute resources are not billed. If auto-pause is disabled, the database keeps running**".
5. Set the auto-pause time (range: 0—7 days; 0—23 hours; 0, 10, 20, 30, 40, or 50 minutes).
6. Click **OK**.



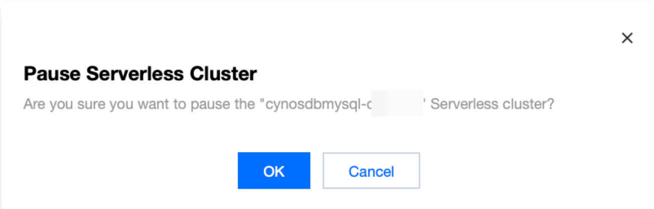
Setting Manual Start/Stop

If automatic start/stop is disabled, you can manually pause or start the Serverless cluster.
If automatic start/stop is enabled and the Serverless cluster is currently paused due to no active connections, you can manually start the cluster.

Manually Pausing a Cluster via the Console

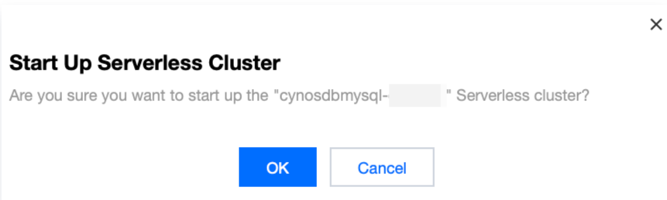
> **Note:**
> If you pause a cluster while it has active connections, it will disrupt connecting businesses. Operate with caution. It is recommended to set automatic start/stop instead.

1. Log in to the TDSQL-C for MySQL console.
2. In the Cluster list on the left, click the **target cluster** to enter the cluster management page.
3. Click **Pause** in the upper right corner.
4. In the pop-up window, click **OK**.



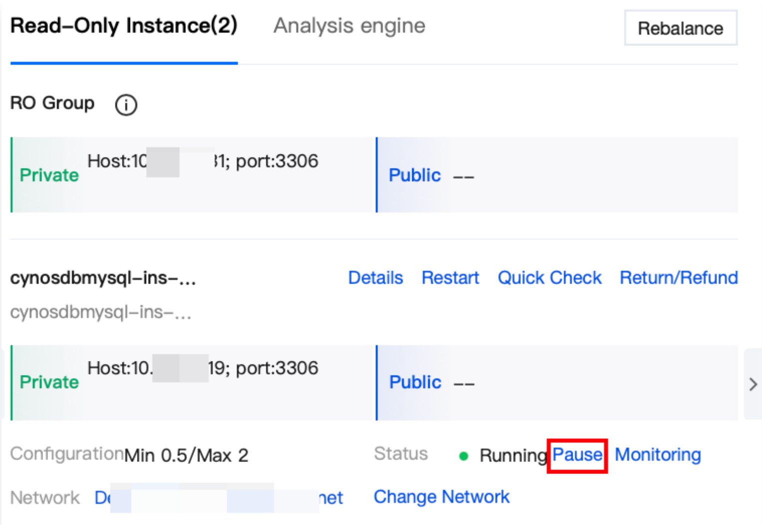
Manually Starting a Cluster via the Console

1. Log in to the TDSQL-C for MySQL console.
2. In the Cluster list on the left, click the **target cluster** to enter the cluster management page.
3. Click **Start** in the upper right corner.
4. In the pop-up window, click **OK**.



Manually Pausing or Starting a Read-Only Instance in the Cluster via the Console

1. Log in to the TDSQL-C for MySQL console.
2. In the Cluster list on the left, click the **target cluster** to enter the cluster management page.
3. Under Instance Information on the right, find the target read-only instance, and click **Pause** or **Start** after the status.



4. In the pop-up window, click **OK**.

Serverless Resource Pack
Resource Pack Overview

Last updated：2024-08-27 10:11:18

In TDSQL-C for MySQL, there are two different types of prepaid resource packs: compute resource packs and storage resource packs, which can be used to deduct the storage and compute resources used by the serverless cluster. By purchasing resource packs, you can reserve resources in advance. Compared to the pay-as-you-go option, the resource packs can help you save more costs. It is more cost-effective to purchase larger capacity and longer validity period.

Resource Pack Description

Prepaid Resource Pack	Compute Resource Pack	Storage Resource Pack
Deduction object	The compute resource pack will be used preferably for the deduction of actual compute resources used by its bound severless cluster. When the pack is used up, the resource usage will be pay-as-you-go.	The storage resource pack will be used preferably for the deduction of actual storage resources used by its bound severless cluster. When the pack is used up, the resource usage will be pay-as-you-go.
Deduction rules	The compute resource pack will be deducted based on the actual CCU usage per second, which is more cost-effective and flexible than the pay-as-you-go option. Before the deduction of the serverless cluster, you must first bind a resource pack. The cluster will not be terminated when the resource pack is unbound, used up, or expires. It will instead be billed on a pay-as-you-go basis.	The storage resource pack will be deducted based on the actual storage used per hour, which is more cost-effective and flexible than the pay-as-you-go option. Before the deduction of the serverless cluster, you must first bind a resource pack. The cluster will not be terminated when the resource pack is unbound, used up, or expires. It will instead be billed on a pay-as-you-go basis.
Renewal	Not supported	
Batch Binding	Up to 10 compute resource packs can be bound to a single cluster at a time.	Up to 10 storage resource packs can be bound to a single cluster at a time.

Resource Packs		
Consumption priority	When a cluster is bound to multiple resource packs, the system automatically sets the consumption priority sequence based on the order of binding. It also supports manually adjustment of consumption priority order. The resources are consumed according to the consumption priority level (1 to 10), from the lowest to the highest. **Note:** • The consumption priorities of compute resource packs and storage resource packs are set separately, and the set priorities are only effective for the same type of resource pack. For example, if a cluster is bound with two compute resource packs, and their consumption priorities are set as 1 and 2 respectively, then the compute resource pack with a priority level of 1 will be consumed first. • The consumption priority of the resource pack that is bound by default is set to 1, with subsequent priorities assigned in ascending order. • Upon the depletion or expiration of a resource pack, the priority ranking for the consumption of the corresponding resource pack will be annulled.	
Deduction period	The compute resource pack will be deducted based on the actual accumulated usage per second.	The storage resource pack will be deducted based on the actual average usage per hour.
Validity period	It is valid for 6 months (180 days) or 1 year (365 days) and can't be used any longer once expired.	It is valid for 6 months (180 days) or 1 year (365 days) and can't be used any longer once expired.
Region	It is available in or out of the Chinese mainland.	It is available in or out of the Chinese mainland.
Use Instructions	• Purchasing Resource Pack • Binding and Unbinding Resource Pack	• Purchasing Resource Pack • Binding and Unbinding Resource Pack

Available Region

In Chinese Mainland	Outside Chinese Mainland
Guangzhou, Nanjing, Shanghai, Beijing, Chengdu, Chongqing, Beijing Finance, Shanghai Finance	Hong Kong (China), Singapore, Seoul, Tokyo, Silicon Valley, Virginia, Frankfurt

Billing Mode

Serverless clusters offer the following four billing modes for you to choose flexibly. If you choose the fourth mode of **resource pack for both compute and storage**, you can enjoy a more favorable price than the monthly subscription option.

- Pay-as-you-go for compute + resource pack for storage
- Resource pack for compute + pay-as-you-go for storage
- Pay-as-you-go for both compute and storage
- Resource pack for both compute and storage

For more information, see Compute Resource Pack and Storage Resource Pack.

Purchasing Resource Pack

Last updated：2024-12-18 11:10:07

This document describes how to purchase a resource pack.

Prerequisites

You have registered a Tencent Cloud account and completed identity verification.

- Register a Tencent Cloud account.
- Complete identity verification.

Purchasing Resource Pack

1. Log in to the TDSQL-C for MySQL console.
2. On the left sidebar, select **Resource Pack** and click **Purchase Resource Pack**.

> **Note:**
> You can also directly purchase one on the resource pack purchase page.

3. Select various configuration items based on your actual needs, confirm that everything is correct, and click **Buy Now**.

Parameter	Description
Resource Pack Type	Select a resource pack type as needed, which can be compute resource pack or storage resource pack.
Region	Set the pack deduction region same as the region where your cluster is deployed. The region can be in or out of the Chinese mainland.
Resource Pack Specifications	Select one of the three specifications: Basic Edition, General Edition, and Enterprise Edition. • The larger the capacity purchased, the greater the discount. For detailed pricing information, refer to Compute Resource Package and Storage Resource Package. • Using the Serverless model can help you reduce database usage costs. Feel free to use the Cost Estimator to estimate Serverless usage fees.
Name Resource Pack	Select one of the options: **Name It Later** (default) or **Name It Now**. If the latter is selected, you can customize the a name, which can contain up to 60 letters, digits, or symbols (-_.).

Terms and Conditions	Read and indicate your consent to the terms and conditions.
Quantity	Select the quantity of resource packs. You can purchase multiple resource packs of the same specifications in batches.

Relevant Documents

- Resource Pack Overview
- Compute Resource Pack
- Storage Resource Pack
- Binding and Unbinding Resource Pack

Compute Resource Pack

Last updated：2025-01-13 10:06:03

The compute resource pack is a prepaid resource type in TDSQL-C for MySQL, which can be used to deduct the compute resources used by the serverless instance. This document describes the specification types and prices of compute resource packs.

Specification Types and Prices

Specification Type	Applicable Scope of CCU	Unit Price in Chinese Mainland (USD/Thousand CCU)	Unit Price Outside Chinese Mainland (USD/Thousand CCU)
Basic Edition	400—10,000 thousand CCU (10,000 exclusive)	6-month validity period: 0.0119 1-year validity period: 0.0126	6-month validity period: 0.013 1-year validity period: 0.0138
General Edition	10,000—40,000 thousand CCU (40,000 exclusive)	6-month validity period: 0.0112 1-year validity period: 0.0119	6-month validity period: 0.0122 1-year validity period: 0.013
Enterprise Edition	40,000—200,000 thousand CCU (200,000 exclusive)	6-month validity period: 0.0105 1-year validity period: 0.0112	6-month validity period: 0.0115 1-year validity period: 0.0122

Note:
TDSQL-C Compute Unit (CCU) is the computing and billing unit for the serverless instance. A CCU is approximately equal to 1 CPU core and 2 GB memory. The number of CCUs used in each billing cycle is the greater of the number of CPU cores used by the database and 1/2 of the memory size.

Case

We will compare the compute resource fees of two instances (one without a compute resource package and the other with a compute resource package) to more intuitively understand the cost difference between using and not using a compute resource package.

Case background

- Region: Beijing.
- Instance Mode: Single-Node Edition.

- The first instance: Unused compute resource package will be billed on a pay-as-you-go basis, denoted as Instance A.
- The second instance: Used the compute resource package, the basic edition of the compute resource package is purchased and bound, with a CCU amount of 400,000 and a validity period of 6 months. Denote it as Instance B.
- Operation scenario: Instance A and Instance B each continued to run at 1 CCU per second (equal to 1 core 2GB full load per second) for 10 hours. The compute resource fees of these 10 hours are calculated and compared below.

Cost comparison

Comparative item	Instance A (**Unused compute resource package**)	Instance B (**Used compute resource package**)
Billing formula	compute resource fees = serverless compute unit price × CCU amount	Compute resource fees = Serverless compute unit price × CCU amount
Billing model	pay as you go	compute resource pack
Serverless compute unit price	0.00001397 USD per unit	0.0119 USD per 10,000 CCU, equal to 0.00000119 USD per unit
CCU amount	36000 calculated based on the running scenario, instance A continues to run at 1 CCU per second for 10 hours, then the total CCU amount for 10 hours is: 1 CCU x 3600 seconds x 10 hours = 36,000	36000 calculated based on the running scenario, instance B continues to run at 1 CCU per second for 10 hours, then the total CCU amount for 10 hours is: 1 CCU x 3600 seconds x 10 hours = 36,000
Compute resource fees	**0.00001397 x 36000 = 0.50 USD**	**0.00000119 x 36000 = 0.04 USD**
Subsequent billing	subsequent operations of instance A will continue to use pay-as-you-go billing, resulting in new fees	Instance B purchased a total compute resource package of 400,000 CCU. In the past 10 hours, 36,000 CCU were consumed, and **the remaining amount was 364,000 CCU**. This compute resource package will continue to be consumed in the future until the computing resource package is consumed or expires. Instance B will return to pay-as-you-go billing. You can also continue to purchase and bind other compute resource

		packages before the compute resource package is consumed.

Compare results

It can be seen from the above comparison that **since the unit price of Serverless compute in a compute resource package is lower than that of pay-as-you-go, the overall compute resource fees is reduced, and it is more favorable to use a compute resource package**. The unit prices of compute resource packages for Basic Edition, General Edition, and Enterprise Edition are different. You can choose the appropriate specification type based on the actual amount of CCU required.

Note

The compute resource pack will be deducted based on the actual CCU usage per second, which is more cost-effective and flexible than the pay-as-you-go option. Before the deduction of the serverless cluster, you must first bind a compute resource pack. The cluster will not be terminated when the resource pack is unbound, used up, or expires. It will instead be billed on a pay-as-you-go basis. It is important to note the following points:

- Compute resource packs are classified into two types: those for the Chinese mainland and those for regions outside the Chinese mainland. Each type is specified to be shared by all serverless clusters in the region, that is, one compute resource pack can be shared by multiple serverless clusters.
- A serverless cluster can be bound to a maximum of ten compute resource packs, with support for adjusting the consumption priority order of multiple compute resource packs. The cluster will consume resources in ascending order based on the priority level numbers.
- A compute resource pack is deducted based on the accumulated usage per second.
- A compute resource pack doesn't support downgrade once purchased.

Management

- Binding and Unbinding Resource Pack
- Viewing Resource Pack Usage
- Modifying Resource Pack Name

Refund

For resource packages that remain unused within their validity periods, each Tencent Cloud account (root account) is eligible for a maximum of 20 refund instances per calendar year. Refunds are not available for resource packages that have either expired or been fully used. For detailed refund instructions, please see the Requesting Refund for Resource Package.

Note:

> Regarding the limitations on the number of refunds, the system calculates this on a calendar year basis. The first calendar year cycle is from June 30, 2023, at 23:59:59 to July 1, 2024, at 00:00:00. The count resets annually on July 1 at 00:00:00. Therefore, the second calendar year cycle is from July 1, 2024, at 00:00:00 to June 30, 2025, at 23:59:59, and so on.

Relevant Documents

- Storage Resource Pack
- Purchasing Resource Pack

Storage Resource Pack

Last updated：2025-01-13 10:00:02

The storage resource pack is a prepaid resource type in TDSQL-C for MySQL, which can be used to deduct the storage resources used by the serverless instance. This document describes the specification types and prices of storage resource packs.

Specification Types and Prices

Specification Type	Applicable Scope of Storage Capacity	Unit Price in Chinese Mainland (USD/TB)	Unit Price Outside Chinese Mainland (USD/TB)
Basic Edition	2—200TB (200 exclusive)	6-month validity period: 0.2568 1-year validity period: 0.271	6-month validity period: 0.2806 1-year validity period: 0.2962
General Edition	200—2000 TB (2000 exclusive)	6-month validity period: 0.2425 1-year validity period: 0.2568	6-month validity period: 0.265 1-year validity period: 0.2806
Enterprise Edition	2000—20,000 TB (20,000 exclusive)	6-month validity period: 0.2282 1-year validity period: 0.2425	6-month validity period: 0.2494 1-year validity period: 0.265

Case

We will compare the storage resource fees of two instances (one without a storage resource package and the other with a storage resource package) to more intuitively understand the cost difference between using and not using a storage resource package.

Case background

- Region: Beijing.
- Instance Mode: Single-Node Edition.
- The first instance: Unused storage resource package will be billed on a pay-as-you-go basis, denoted as Instance A.
- The second instance: Utilizing the storage resource package, a Basic Edition storage resource package was purchased and bound, offering a storage capacity of 10TB with a validity period of 6 months. Denoted as Instance B.
- Operation scenario: Instance A and instance B have been running continuously for 3 hours respectively. The storage space used in the first hour is 0.5TB, the storage space used in the second hour is 1TB, and

the storage space used in the third hour is 2TB. The storage resource costs of these 3 hours are calculated and compared below.

Cost comparison

Comparative item	Instance A (**Unused storage resource package**)	Instance B (**Used storage resource package**)
Billing formula	storage space fees = storage space price × storage space total storage resource fees of running for 3 hours = fees for the first hour + fees for the second hour + fees for the third hour	storage space fees = storage space price × storage space total storage resource fees of running for 3 hours = fees for the first hour + fees for the second hour + fees for the third hour
Billing model	pay as you go	storage resource pack
Storage space fees (unit price)	0.00072 USD per GB Approximately equal to 0.72 USD per TB	0.2568 USD per TB
Storage space	the storage space used in the first hour: 0.5TB the storage space used in the second hour: 1TB the storage space used in the third hour: 2TB	the storage space used in the first hour: 0.5TB the storage space used in the second hour: 1TB the storage space used in the third hour: 2TB
Storage resource fees	**0.72 x 0.5 + 0.72 x 1 + 0.72 x 2 = 2.52 USD**	**0.2568 x 0.5 + 0.2568 x 1 + 0.2568 x 2 = 0.8988 USD**
Subsequent billing	subsequent operations of instance A will continue to use pay-as-you-go billing, resulting in new fees	the total storage resource package purchased by instance B is 10TB. In the past 3 hours, 3.5TB has been consumed, and **the remaining amount is 6.5TB**. This storage resource package will continue to be consumed in the future. Instance B will not be restored until the storage resource package is consumed or expires. For pay-as-you-go billing, you can also continue to purchase and bind other storage resource packages before the storage resource package is consumed.

Compare results

It can be seen from the above comparison that **since the unit price of storage space in a storage resource package is lower than that of pay-as-you-go, the overall storage resource fees is reduced, and it is more favorable to use a storage resource package**. The unit prices of storage resource packages for Basic Edition, General Edition, and Enterprise Edition are different. You can choose the appropriate specification type based on the actual storage required.

Note

Storage resource packages will accumulate the actual storage usage on an hourly basis for deduction, which is more cost-effective and flexible than the pay-as-you-go option. Before the deduction of the serverless cluster, you must first bind a storage resource pack. The cluster will not be terminated when the resource pack is unbound, used up, or expires. It will instead be billed on a pay-as-you-go basis. It is important to note the following points:

- Storage resource packs are classified into two types: those for the Chinese mainland and those for regions outside the Chinese mainland. Each type is specified to be shared by all serverless clusters in the region, that is, one storage resource pack can be shared by multiple serverless clusters.
- A serverless cluster can be bound to a maximum of ten storage resource packages, with support for adjusting the consumption priority order of multiple storage resource packages. The cluster will consume resources in ascending order based on the priority level numbers.
- Storage resource packages will accumulate the consumption on an hourly basis for deduction.
- Storage resource packages do not support downgrade once purchased.

Management

- Binding and Unbinding Resource Pack
- Viewing Resource Pack Usage
- Modifying Resource Pack Name

Refund

For resource packages that remain unused within their validity period, each Tencent Cloud account (primary account) is calculated on a calendar year basis, with a maximum of 20 refundable resource packages per year. As for resource packages that have expired or been fully utilized, refunds are currently not supported. For detailed refund instructions, see Requesting Refund for Resource Pack.

> **Note:**
> Regarding the limitation on the number of refunds, the system calculates this on a calendar year basis. The first calendar year cycle is from June 30, 2023, at 23:59:59 to July 1, 2024, at 00:00:00. The count refreshes on July 1 at 00:00:00 every year, thus the second calendar year cycle is from July 1, 2024, at 00:00:00 to June 30, 2025, at 23:59:59, and so on.

Relevant Documents

- Storage Resource Pack
- Purchasing Resource Pack

Binding and Unbinding Resource Pack

Last updated：2025-06-03 15:30:07

If you want to use a resource pack, you need to bind the resource pack after purchasing it. You can also unbind a resource pack that has been bound to clusters. This document describes how to bind or unbind a resource pack.

Note:
- A resource package can be bound to multiple clusters.
- Each cluster can be bound to up to 10 computing resource packages and 10 storage resource packages.
- Storage resource packages can only be bound to Serverless clusters.
- Computing resource packages can be bound to Serverless clusters or provisioned resource clusters with Serverless enabled (that is, the cluster contains Serverless read-only instances).

Prerequisites

- You have created a serverless cluster. For more information, see Creating Serverless Cluster.
- You have purchased a resource pack. For more information, see Purchasing Resource Pack.

Binding a resource pack

Scenario 1: Binding a resource pack when creating a serverless cluster

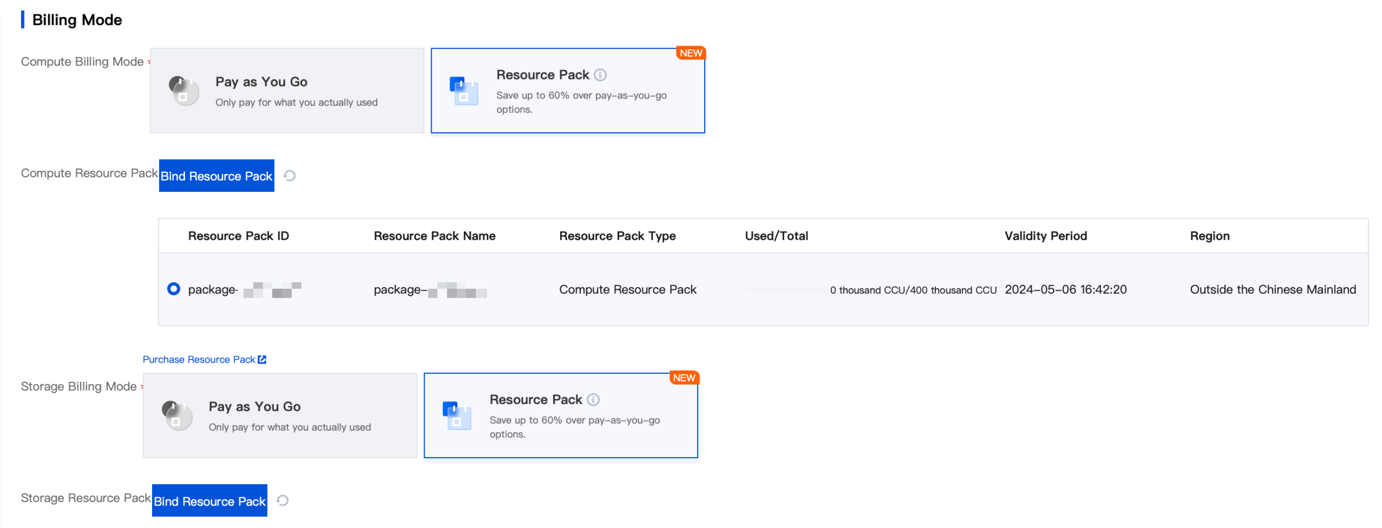
1. Go to the purchase page and complete the **Database Configuration** settings.

Parameter	Description
Instance Mode	Select **Serverless**.
Database Engine	Select **MySQL**.
Region	Select a region for database deployment. Currently, Serverless is only supported in the following regions: Guangzhou, Shanghai, Beijing, Chengdu, Nanjing, Hong Kong (China), Silicon Valley, Singapore, Seoul, Virginia, and Tokyo. If you need to use it in other regions, submit a ticket for assistance.
Source	Select an AZ for deployment. Specific AZs in the selected region are shown on the

AZ	actual purchase page.
Multi-AZ Deployment	Currently, serverless instances don't support multi-AZ deployment.
Transfer Linkage	It is "High IO" by default.
Network	For performance and security considerations, only VPC network is supported currently. CVM instances can communicate with TDSQL-C for MySQL instances only in the same VPC.
Database Version	MySQL 5.7 and 8.0 are supported.
Compute Unit	Select the upper and lower limits of the TDSQL-C compute unit (CCU), and the instance will be automatically and elastically scaled within the selected resource range. CCU is the computing and billing unit for the serverless mode. A CCU is approximately equal to 1 CPU core and 2 GB memory. The number of CCUs used in each billing cycle is the greater of the number of CPU cores used by the database and 1/2 of the memory size. For more information, see Compute Unit.
Auto-Pause	Configure the automatic pause time of the instance. If there is no connection to access the database within the set time, the instance will be automatically paused, with billing stopped.

2. Complete the **Specification Billing** configuration and click **Next**.

> **Note:**
> **Total serverless cluster fees = compute node fees + storage space fees = serverless computing power price number of CCUs + storage space price storage space**



Parameter	Description
Compute Billing Mode	Select **Resource Pack**. The compute resource pack will be used preferably for the deduction of actual usage of pay-as-you-go products. When the pack is used up, the resource usage will be pay-as-you-go. Compute resource packs are deducted based on the actual CCU used per second. The resource pack mode is more cost-effective and flexible than the pay-as-you-go option.
Compute Resource Pack	Binding a resource pack allows you to associate one or more (up to a maximum of ten) compute resource packs with the cluster selected in the region under the current account, provided they are within their validity period. If there are no available resource packs, you may proceed to purchase a resource package first.
Storage Billing Mode	Select **Resource Pack**. The storage resource pack will be used preferably for the deduction of actual usage of pay-as-you-go products. When the pack is used up, the resource usage will be pay-as-you-go. Storage resource packs are deducted based on the actual storage used per hour. The resource pack mode is more cost-effective and flexible than the pay-as-you-go option.
Storage Resource Pack	Binding a resource pack allows you to associate one or multiple (up to a maximum of ten) storage resource packs with the cluster selected in the region under the current account, provided they are within their validity period. If there are no available resource packs, you may proceed to purchase a resource package first.

3. Select the number of clusters. You can batch purchase multiple clusters of the same specification. Then,

click **Next**.

4. Complete the **Basic Info** and **Advanced Configuration** settings, confirm the fees, and click **Buy Now**.

- **Basic Info**

Parameter	Description
Cluster Name	Name the instance now or later. It must contain less than 60 letters, digits, or symbols (-_.).
Admin Username	It is `root` by default.
Password	The password can contain 8—64 characters in at least three of the following character types: uppercase letters, lowercase letters, digits, and symbols ~!@#$%^&*_-+=|\(){}[]:;'<>,.?/.
Default Character Set	UTF8, GBK, LATIN1, and UTF8MB4 are supported.
Custom Port	It is 3306 by default and can be customized.

- Advanced Configuration

Parameter	Description
Security Group	Select or create a security group.
Parameter Template	Select or create a parameter template.
Table Name Case Sensitivity	Select **Case-Insensitive** or **Case-Sensitive**.
Project	Specify a project for the cluster to be created.
Alarm Policy	Select or create an alarm policy.
Tag	Add a tag to facilitate resource categorization and management.
Terms and Conditions	Read and indicate your consent to the terms and conditions.

5. After the purchase is completed, you will be redirected to the cluster list. After the status of the cluster becomes **Running**, it can be used normally.

> **Note:**
> - If only the compute resource pack is bound, the compute nodes of the serverless cluster will

be deducted by using the bound compute resource pack, while the storage nodes will be billed on a pay-as-you-go basis.

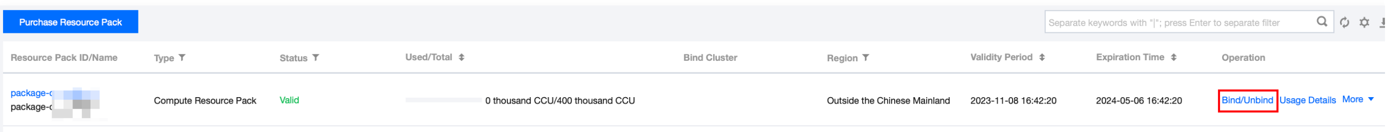
- If only the storage resource pack is bound, the storage nodes of the serverless cluster will be deducted by using the bound storage resource pack, while the compute nodes will be billed on a pay-as-you-go basis.
- If both the compute and storage resource packs are bound, both the compute and storage nodes of the serverless cluster will be deducted by using the bound compute and storage resource packs.

Scenario 2: Binding Resource Packages to Existing Clusters

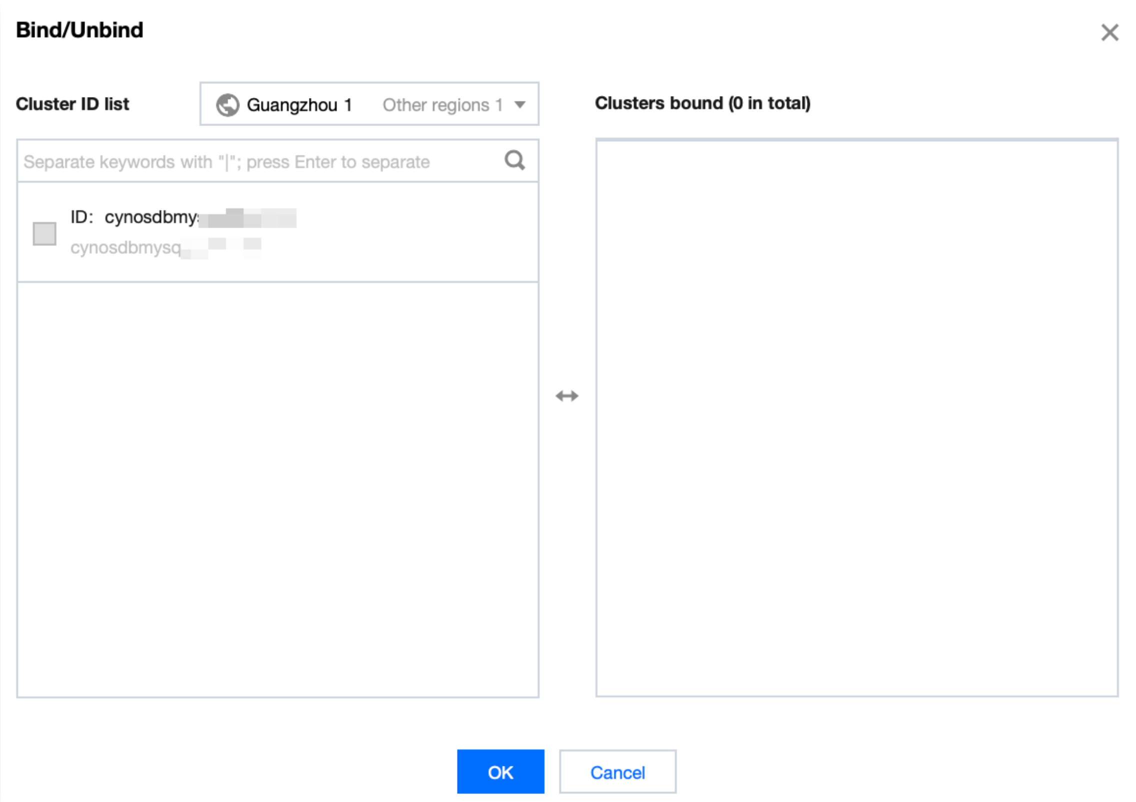
Binding Existing Clusters from the Resource Package List

After purchasing a resource package, you can bind it to an existing cluster from the resource package list.

1. Log in to the TDSQL-C for MySQL console.
2. On the left sidebar, click **Resource Pack** to enter the resource pack management page.
3. Find the target resource pack directly on the page or quickly filter it out in the search box on the right, and click **Bind/Unbind** in the **Operation** column.

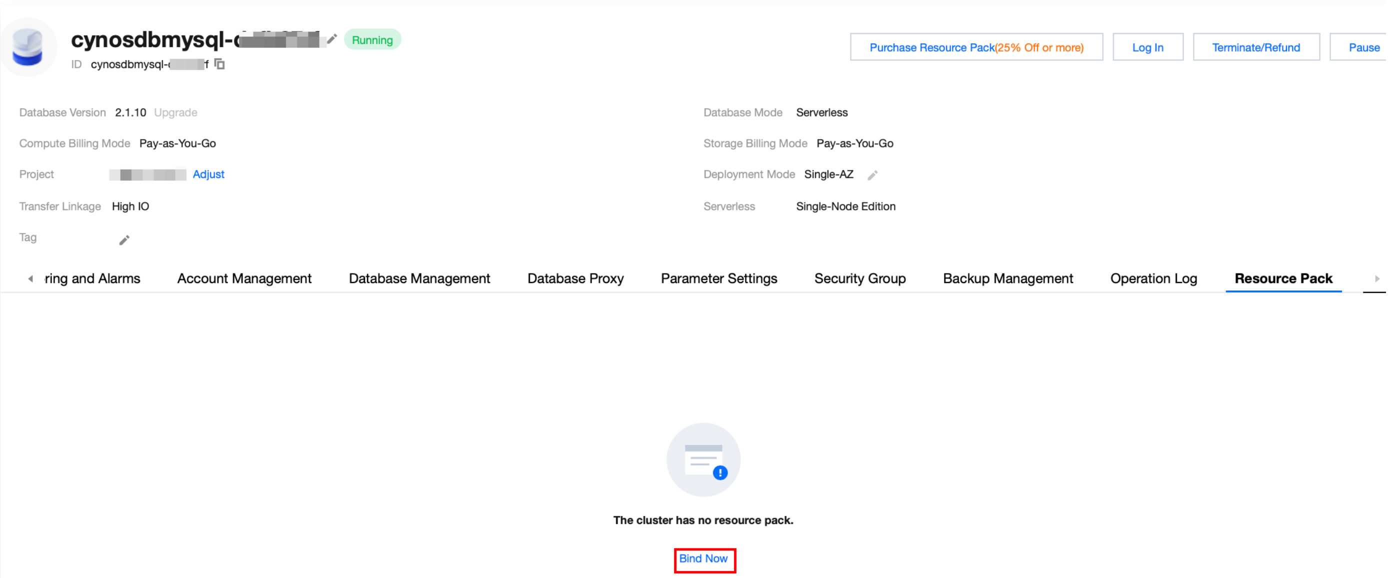


4. In the pop-up window, select a region, and then select clusters to bind the resource packages to (multiple selections are supported). Click **OK**.

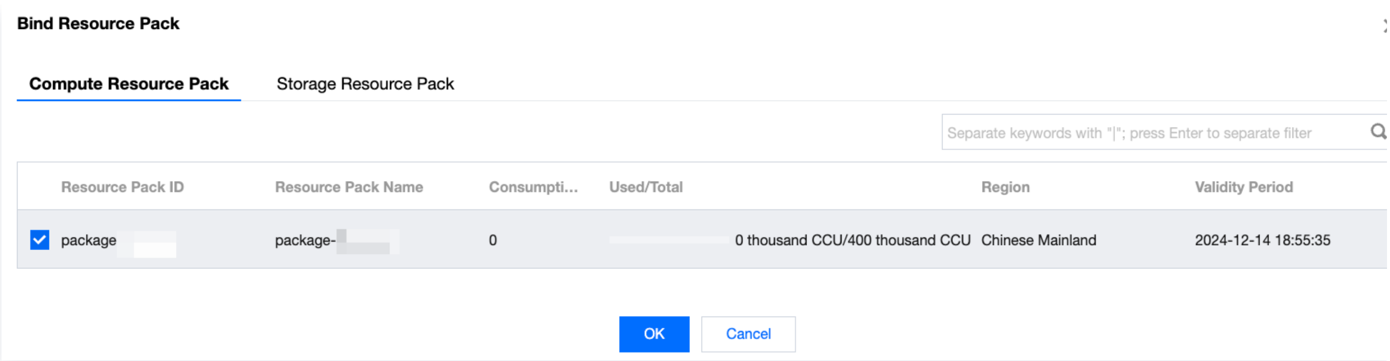


Binding Resource Packages to Existing Clusters from the Cluster Management Page

1. Log in to the TDSQL-C for MySQL console.
2. Click the target cluster in the cluster list on the left to enter the cluster management page.
3. On the cluster management page, select **Resource Pack** and click **Bind Now**.



4. In the pop-up window, select the required resource packs. Multiple selections are supported. Click **OK**.



Unbinding a resource pack

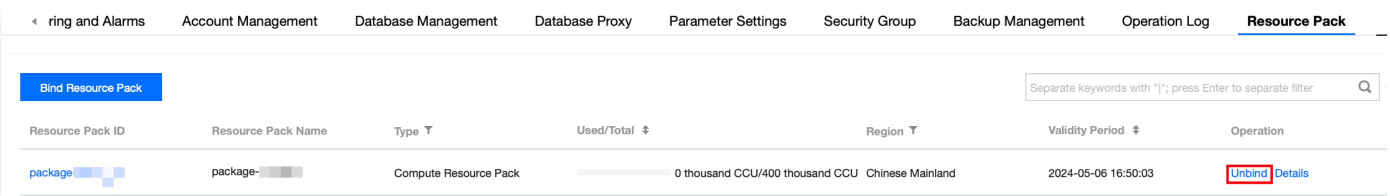
When a resource pack expires or is used up, the resource pack will be automatically unbound. You can also unbind it manually.

Unbinding Existing Clusters from the Resource Package List

1. Log in to the TDSQL-C for MySQL console.
2. On the left sidebar, click **Resource Pack** to enter the resource pack management page.
3. Find the target resource pack directly on the page or quickly filter it out in the search box on the right, and click **Bind/Unbind** in the **Operation** column.
4. In the pop-up window, select a region, and then delete the bound clusters on the right. Click **OK**.

Unbinding Resource Packages from Existing Clusters from the Cluster Management Page

1. Log in to the TDSQL-C for MySQL console.
2. Click the target cluster in the cluster list on the left to enter the cluster management page.
3. On the cluster management page, select **Resource Pack**.
4. Find the target resource pack, and click **Unbind** in the **Operation** column.



5. In the pop-up window, click **OK**.

Adjusting Consumption Priority Order

Last updated：2024-06-07 14:18:59

A serverless cluster supports binding multiple resource packs. When multiple resource packs of the same type (compute or storage) are bound, the system automatically sets the consumption order of the resource packs according to the order of binding. You can also manually adjust this order, that is, adjust the consumption priority sorting of the resource packs. Resources are consumed from the lowest to the highest consumption priority level (1 to 10).

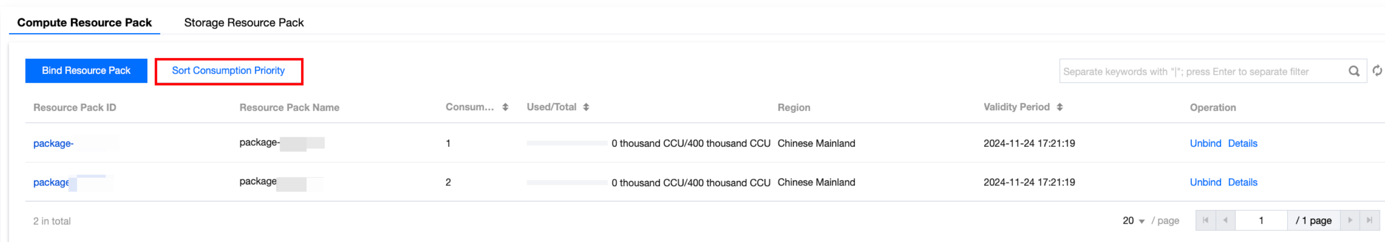
This topic introduces how to adjust the consumption priority order of resource packs through the console.

Prerequisites

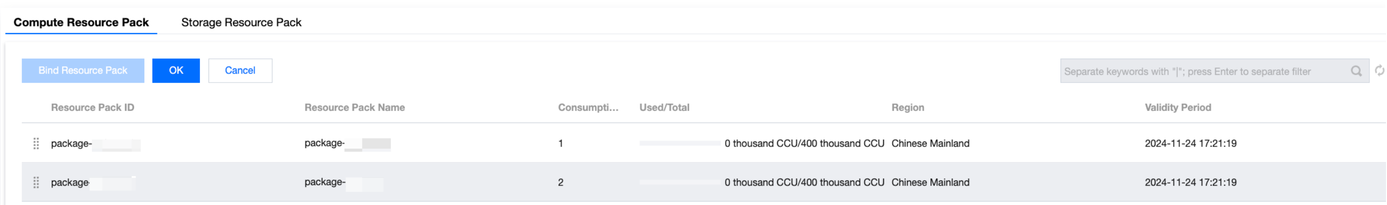
- A serverless cluster has been created.
- You have purchased a resource pack.
- You have bound multiple resource packs.

Adjusting Consumption Priority Order

1. Log in to the TDSQL-C for MySQL console.
2. On the left, click a cluster in the **Cluster List**, and select the **target cluster** to enter the **cluster management page**.
3. On the cluster management page, select **Resource Pack**, choose either **Compute Resource Pack** or **Storage Resource Pack** according to the actual situation, and then click **Sort Consumption Priority**.



4. Adjust the needed consumption priority order by dragging with the mouse, and click **OK**.



Viewing Resource Pack Usage

Last updated：2023-06-14 17:07:59

This document describes how to view the usage of a resource pack.

Prerequisites

- You have purchased a resource pack. For more information, see Purchasing Resource Pack.
- You have bound the resource pack. For more information, see Binding and Unbinding Resource Pack.

Directions

Option 1:

1. Log in to the TDSQL-C for MySQL console.
2. On the left sidebar, click **Resource Pack** to enter the resource pack management page.
3. Find the target resource pack directly on the page or quickly filter it out in the search box on the right, and click **Usage Details** in the **Operation** column.
4. On the **Usage Details** page, select the target cluster to be queried in the upper right corner, or select all clusters to view the resource pack usage.

> **Note:**
> - You can view the resource pack usage in the last 24 hours, last 48 hours, last 7 days, last 30 days, a custom time range, or all time ranges.
> - You can check the following usage details fields: Deduction Period (Hour), Resource Pack ID, Used/Total, and Cluster ID.

Option 2:

1. Log in to the TDSQL-C for MySQL console.
2. Click **Target Cluster** in the cluster list on the left to enter the cluster management page.
3. On the cluster management page, select **Resource Pack**.
4. Find the target resource pack, and click **Resource Pack ID** or **Details** in the **Operation** column.
5. On the **Usage Details** page, select the target cluster to be queried in the upper right corner, or select all clusters to view the resource pack usage.

Resource Pack Reminder Policy

Last updated：2024-03-25 15:36:20

This document describes the reminder policies regarding the usage and validity of a resource pack.

Reminder Policy for Resource Pack Usage

Scenario: Resource pack is about to run out

- Reminder messages will be sent when the remaining resource pack usage reaches 30%, 20%, or 10%.
  - Channels: Message Center, email, and SMS.
  - Sample message:

    Dear user,
    You have already used 280,000 CCUs of your TDSQL-C: cynosdbmysql resource pack (ID: package-*, name: to--1) under your account (ID: 100028*, nickname: ce***), and the remaining 120,000 CCUs is less than 30% of the total capacity. Please purchase the resource pack promptly to avoid interruptions.
- Reminder messages will be sent when the resource pack runs out.
  - Channels: Message Center, email, and SMS.
  - Sample message:

    Dear user,
    You have used up 400,000 CCUs of your TDSQL-C: cynosdbmysql resource pack (ID: package-*, name: to--1) under your account (ID: 100028*, nickname: ce***) at 11:43:43 on 2023-05-31. Please purchase the resource pack promptly to avoid interruptions.

Reminder Policy for Resource Pack Validity

Scenario: Resource pack is about to expire

- Reminder messages will be sent when the resource pack is 30 days, 7 days, 5 days, 3 days, or 1 day from expiration.
  - Channels: Message Center, email, and SMS.
  - Sample message:

    Dear user,
    Your TDSQL-C: cynosdbmysql resource package (ID: package-*, name: to--1) under your account (ID: 100028*, nickname: ce***) will expire at 11:43:43 on November 10, 2023. Once the resource pack has expired, it cannot be refunded. Please purchase the resource pack promptly to avoid interruptions.
- Reminder messages will be sent when the resource pack expires.

- Channels: Message Center, email, and SMS.
- Sample message:

  Dear user,

  The TDSQL-C resource pack cynosdbmysql- ***(resource pack ID: package-, resource package name: to--1) you purchased for your account (account ID: 100028***, nickname: ce***) has expired at 11:43:43 on November 10, 2023. Please purchase the resource pack promptly to avoid interruptions.

Modifying Resource Pack Name

Last updated：2024-12-04 21:02:31

This document describes how to modify the name of a resource pack.

Prerequisites

You have purchased a resource pack. For more information, see Purchasing Resource Pack.

Directions

1. Log in to the TDSQL-C for MySQL console.
2. On the left sidebar, click **Resource Pack** to enter the resource pack management page.
3. Find the target resource pack directly on the page or quickly filter it out in the search box on the right, and click the edit icon under the **Resource Pack ID/Name**.
4. In the pop-up window for modifying the resource pack name, enter the new name of the resource pack and click **OK**.

> **Note:**
> The resource pack name can contain up to 60 letters, digits, or symbols (-_.).

Requesting Refund for Resource Pack

Last updated: 2024-04-25 12:45:31

This document describes how to request a refund for a resource pack.

Prerequisites

You have purchased a resource pack. For more information, see Purchasing Resource Pack.

Refund Rule

You can request a refund for a valid resource pack in a self-service manner in the console.
Refund policy: For resource packages that remain unused within their validity period, each Tencent Cloud account (primary account) is eligible for a maximum of 20 refund instances per calendar year. Refunds are not available for resource packages that have either expired or been fully used.

> **Note:**
> Regarding the limitation on the number of refunds, the system calculates this on a calendar year basis. The first calendar year cycle is from June 30, 2023, at 23:59:59 to July 1, 2024, at 00:00:00. The count resets annually on July 1 at 00:00:00. Therefore, the second calendar year cycle is from July 1, 2024, at 00:00:00 to June 30, 2025, at 23:59:59, and so on.

Refund Amount

Refund amount = paid amount - (consumed usage x unit price x discount applicable)

Directions

1. Log in to the TDSQL-C for MySQL console.
2. On the left sidebar, click **Resource Pack** to enter the resource pack management page.
3. Find the target resource pack directly on the page or quickly filter it out in the search box on the right, and click **More** > **Refund** in the **Operation** column.
4. In the pop-up window, confirm the requested refund and click **OK**.

Multi-AZ Deployment

Overview of Multi-AZ Deployment

Last updated：2025-10-23 20:20:27

TDSQL-C for MySQL offers both preset resources and Serverless cluster types, all supporting multi-availability zone deployment. Compared to single availability zone deployment, multi-availability zone deployment offers higher disaster recovery capabilities, ensuring database protection against instance failures or availability zone interrupts, and providing resilience against data center level failures. Multi-availability zone deployment enhances database instances with high availability and failover support. A multi-availability zone configuration involves combining multiple single availability zones within the same region into a physical zone. This article introduces multi-availability zone deployment for Serverless clusters. For information on multi-availability zone deployment with preset resources, please see Overview of Multi-availability zone Deployment.

Prerequisites

- The cluster region has at least two AZs.
- The target AZ has sufficient computing resources.
- Database version requirements:
  - Database version 5.7 with kernel minor version 2.0.15 or later.
  - Database version 8.0 with kernel minor version 3.0.1 or later.

Supported Regions and AZs

- Currently, this feature is in beta test and only supports the following regions and AZs.
- This feature will gradually support more regions and AZs.
- If your business requires it, you can Submit a Ticket to request deployment in other Regions and AZs.

Supported Region	Supported Primary AZ	Supported Secondary AZ
Beijing	Beijing Zone 3	Beijing Zone 5
	Beijing Zone 5	Beijing Zone 7
	Beijing Zone 6	Beijing Zone 7
		Beijing Zone 8
	Beijing Zone 7	Beijing Zone 5
Shanghai	Shanghai Zone 2	Shanghai Zone 4

	Shanghai Zone 4	Shanghai Zone 2
	Shanghai Zone 5	Shanghai Zone 4
Guangzhou	Guangzhou Zone 4	Guangzhou Zone 6
	Guangzhou Zone 6	Guangzhou Zone 4
		Guangzhou Zone 7
	Guangzhou Zone 7	Guangzhou Zone 4
Hong Kong (China)	Hong Kong (China) Zone 2	Hong Kong (China) Zone 3
Singapore	Singapore Zone 3	Singapore Zone 4
	Singapore Zone 4	Singapore Zone 3
Silicon Valley	Silicon Valley Zone 2	Silicon Valley Zone 1
Virginia	Virginia Zone 1	Virginia Zone 2
	Virginia Zone 2	Virginia Zone 1
Tokyo	Tokyo Zone 1	Tokyo Zone 2
	Tokyo Zone 2	Tokyo Zone 1
Nanjing	Nanjing Zone 1	Nanjing Zone 2
Jakarta	Jakarta Zone 2	Jakarta Zone 1
Chengdu	Chengdu Zone 1	Multi-AZ deployment is not supported
Seoul	Seoul Zone 2	Multi-AZ deployment is not supported

Multi-AZ Deployment Fee Description

There are no additional fees for the Multi-AZ Deployment feature for the time being.

Operations Related to Multi-AZ Deployment

Last updated：2025-05-28 14:34:37

The operations related to multi-AZ deployment for Serverless clusters are the same as those for provisioned resource clusters. You can see the table below for details.

Supported Features and Operation Guide

Feature Item	Operation Guide
Setting multi-AZ deployment	You can set multi-AZ deployment during cluster creation and after cluster creation via the console. For detailed operations, see Setting Multi-AZ Deployment.
Switching AZs	Clusters deployed in multiple AZs support the primary-secondary switch. For detailed operations, see Switching AZs.
Modifying AZ deployment	You can upgrade the cluster deployment method from single AZ deployment to multi-AZ deployment or downgrade from multi-AZ deployment to single-AZ deployment. For detailed operations, see Modifying AZ Deployment.

Configuration Change

Compute Unit

Last updated: 2025-10-23 20:22:03

This document describes the computing power specification of TDSQL-C for MySQL Serverless.

Note:

- The current list of compute unit configurations may include some specifications that have been discontinued. Refer to the actual purchase page for the available specifications.
- The compute unit configurations supported vary by region. Refer to the table below for the support status in each region.
- For the purchase of compute unit configurations of 32 CCU and above, submit a ticket to contact our staff for assistance.
- Starting from October 12, 2024, TDSQL-C for MySQL will optimize and adjust the maximum storage capacity, maximum IOPS, and I/O bandwidth corresponding to each instance specification through technical upgrades. For Serverless-related adjustments, see Serverless Limitation Changes.

Serverless compute unit

Guangzhou, Nanjing

Min Compute Unit (CCU)	Max Compute Unit (CCU)	Supported Max Storage Space (TB)
0.25	0.5	100
0.25	1	100
0.25	2	156
0.25	4	156
0.25	8	156
0.25	16	192
0.25	32	256

0.5	0.5	100
0.5	1	100
0.5	2	156
0.5	4	156
0.5	8	156
0.5	16	192
0.5	32	256
1	1	100
1	2	156
1	4	156
1	8	156
1	16	192
1	32	256
2	2	156
2	4	156
2	8	156
2	16	192
2	32	256
4	4	156
4	8	156
4	16	192
4	32	256
8	8	156
8	16	192
8	32	256

8	64	384
16	16	192
16	32	256
16	64	384
32	32	256
32	64	384
64	64	384

Shanghai, Hong Kong (China), Beijing, Silicon Valley, Chengdu, Tokyo

Min Compute Unit (CCU)	Max Compute Unit (CCU)	Supported Max Storage Space (TB)
0.25	0.5	100
0.25	1	100
0.25	2	156
0.25	4	156
0.25	8	156
0.25	16	192
0.25	32	256
0.5	0.5	100
0.5	1	100
0.5	2	156
0.5	4	156
0.5	8	156
0.5	16	192
0.5	32	256

1	1	100
1	2	156
1	4	156
1	8	156
1	16	192
1	32	256
2	2	156
2	4	156
2	8	156
2	16	192
2	32	256
4	4	156
4	8	156
4	16	192
4	32	256
8	8	156
8	16	192
8	32	256
16	16	192
16	32	256
32	32	256

Seoul Zone 2, Singapore Zone 3, Singapore Zone 4

Min Compute Unit	Max Compute Unit	Supported Max Storage Space (TB)

(CCU)	(CCU)	
0.25	0.5	100
0.25	1	100
0.25	2	156
0.25	4	156
0.25	8	156
0.25	16	192
0.25	32	256
0.5	0.5	100
0.5	1	100
0.5	2	156
0.5	4	156
0.5	8	156
0.5	16	192
0.5	32	256
1	1	100
1	2	156
1	4	156
1	8	156
1	16	192
1	32	256
2	2	156
2	4	156
2	8	156
2	16	192

2	32	256
4	4	156
4	8	156
4	16	192
4	32	256
8	8	156
8	16	192
8	32	256
16	16	192
16	32	256
32	32	256

Virginia Zone 1

Min Compute Unit (CCU)	Max Compute Unit (CCU)	Supported Max Storage Space (TB)
0.25	0.5	100
0.25	1	100
0.25	2	156
0.25	4	156
0.25	8	156
0.25	16	192
0.5	0.5	100
0.5	1	100
0.5	2	156
0.5	4	156

0.5	8	156
0.5	16	192
1	1	100
1	2	156
1	4	156
1	8	156
1	16	192
2	2	156
2	4	156
2	8	156
2	16	192
4	4	156
4	8	156
4	16	192
8	8	156
8	16	192
16	16	192

Virginia Zone 2

Min Compute Unit (CCU)	Max Compute Unit (CCU)	Supported Max Storage Space (TB)
0.25	0.5	100
0.25	1	100
0.25	2	156

0.25	4	156
0.25	8	156
0.25	16	192
0.5	0.5	100
0.5	1	100
0.5	2	156
0.5	4	156
0.5	8	156
0.5	16	192
1	1	100
1	2	156
1	4	156
1	8	156
1	16	192
2	2	156
2	4	156
2	8	156
2	16	192
4	4	156
4	8	156
4	16	192
8	8	156
8	16	192
16	16	192
32	32	256

Serverless Limitation Changes

1. Maximum Storage Capacity

Maximum Configured Computing Power (CCU)	Maximum Storage Capacity (TB)	
	Before Adjustment and Optimization	**After Adjustment and Optimization**
0.5	1	**100**
1	1	**100**
2	5	**156**
4	10	**156**
8	10	**156**
16	20	**192**
32	30	**256**
64	40	**384**

2. Maximum IOPS

Maximum Configured Computing Power (CCU)	Maximum IOPS	
	Before Adjustment and Optimization	**After Adjustment and Optimization**
0.5	24000	**24000**
1	24000	**24000**
2	24000	**24000**
4	96000	**96000**
8	216000	**320000**
16	384000	**400000**
32	576000	**600000**
64	720000	**720000**

3. I/O Bandwidth

Maximum Configured Computing Power (CCU)	I/O Bandwidth	
	Before Adjustment and Optimization	**After Adjustment and Optimization**
0.5	3 Gbps	**3 Gbps**
1	3 Gbps	**3 Gbps**
2	3 Gbps	**3 Gbps**
4	12 Gbps	**12 Gbps**
8	27 Gbps	**40 Gbps**
16	48 Gbps	**50 Gbps**
32	72 Gbps	**75 Gbps**
64	90 Gbps	**90 Gbps**

Adjusting Configurations

Last updated: 2025-05-28 14:37:57

After creating a Serverless cluster instance, you can modify the cluster's architecture and computing power configuration, and enable/disable automatic suspension by **Serverless configuration** in the console.

Prerequisites

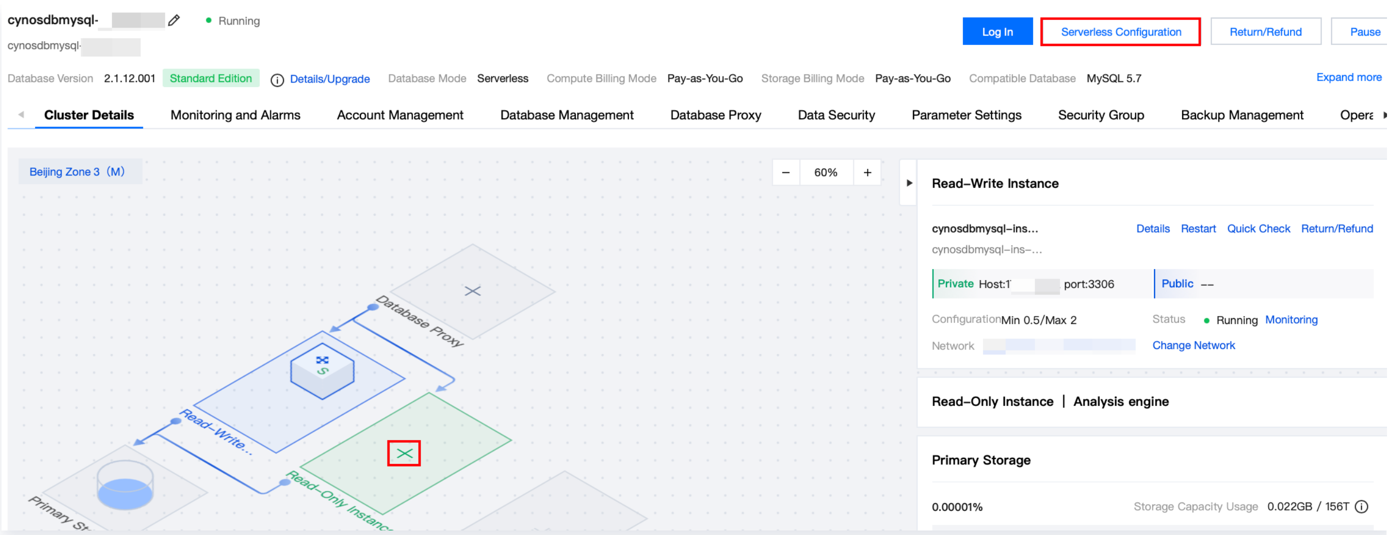
- A Serverless cluster has been created.
- The cluster status is started.

Modify the cluster's architecture

Clusters with a Serverless architecture support two configurations: the single-node edition and the cluster edition. The single-node edition comprises only one read-write instance. After purchase, it can be switched to the cluster version architecture through Serverless configuration in the console. The cluster edition includes one read-write instance and 1 to 15 optional read-only instances. Post-purchase, it can be transitioned back to the single-node edition architecture via Serverless configuration in the console. The corresponding operations are as follows:

Switching from Single-Node to Cluster Architecture

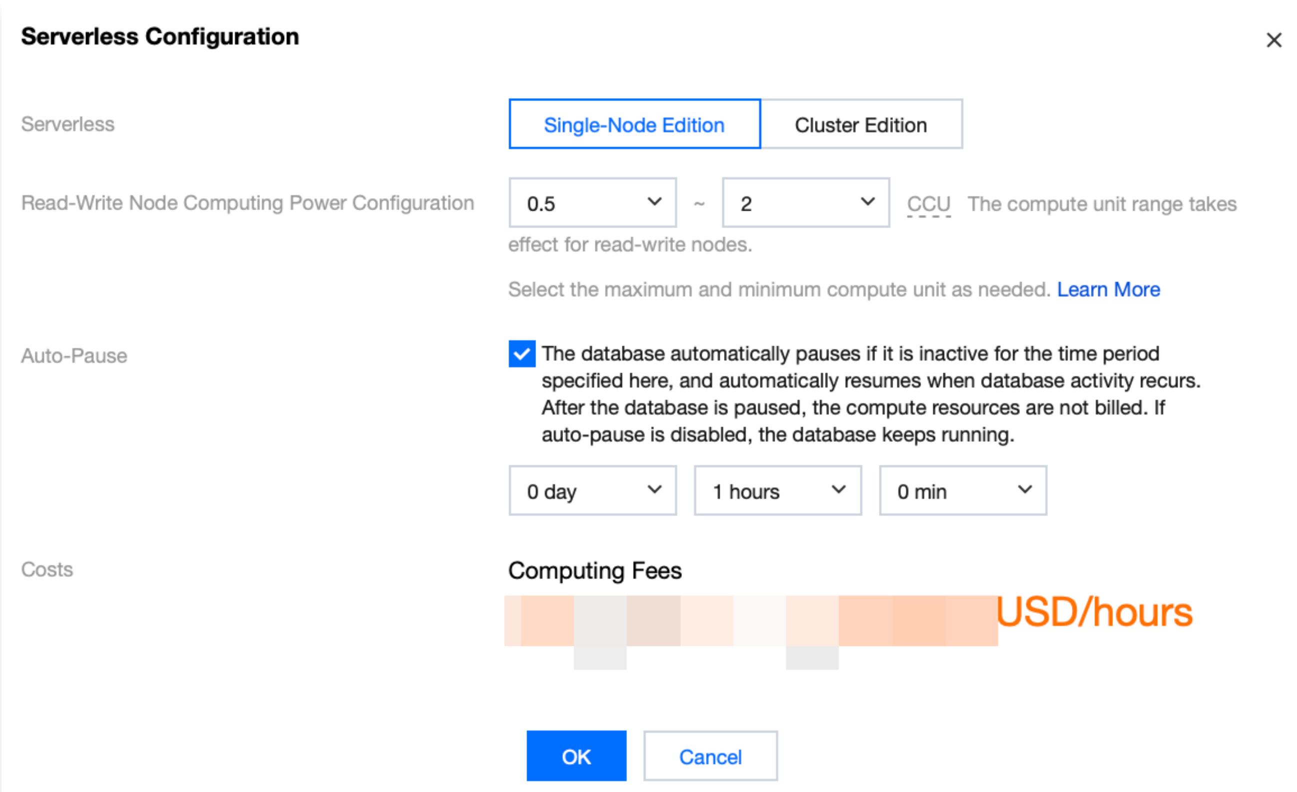
1. Log in to TencentDB for CynosDB console, and select a region at the top.
2. Click **Target Cluster** in the cluster list on the left to enter the cluster management page.
3. In the topology diagram, click on the "+" icon within the read-only instance area, or click on **Serverless Configuration** in the upper right corner.



4. On the **Adjust Configuration** page, select the Serverless architecture as **Cluster Edition**.
5. Select the number of read-only nodes as required.
6. Configure the computing power range for read-only nodes, which will be applicable to all read-only nodes.
7. Click **OK**.

Switching from Cluster to Single-Node Architecture

1. Log in to TencentDB for CynosDB console, and select a region at the top.
2. Click **Target Cluster** in the cluster list on the left to enter the cluster management page.
3. In the topology diagram, click on the "+" icon within the read-only instance area, or click on **Serverless Configuration** in the upper right corner.
4. In the pop-up window, select the Serverless architecture as **Single-node Edition** and click **OK**.



Modifying the Computing Power Configuration of the Cluster

1. Log in to TencentDB for CynosDB console, and select a region at the top.
2. Click **Target Cluster** in the cluster list on the left to enter the cluster management page.
3. In the topology diagram, click on the "+" icon within the read-only instance area, or click on **Serverless Configuration** in the upper right corner.
4. In the pop-up window, adjust the computing power settings for both the read-write and read-only nodes (applicable only to cluster edition), then click **OK**.

Enabling/Disabling Automatic Suspension

On the **Serverless Configuration** page, you can enable or disable automatic suspension. For service features and operation methods related to cluster automatic suspension, see Automatic Start and Stop.

FAQs

Serverless FAQs

Last updated：2025-05-28 14:41:25

This document introduces FAQs related to Serverless.

What Is the Serverless Service?

TDSQL-C for MySQL supports two modes of database instances: provisioned resource and Serverless. The Serverless service is a serverless architecture version of TDSQL-C for MySQL, which charges based on actual computing and storage resource usage and can automatically scale in/out according to business loads. This maximizes resource utilization and optimizes cost control. It is suitable for low-frequency database usage, such as development and testing environments, or scenarios with fluctuating business peaks.

How Is the Serverless Service Billed?

Computing and storage resources in Serverless are billed separately: Computing fees are charged based on the number of CCUs, and storage fees are charged based on the amount of storage used (GB). Billing is calculated per second and settled on an hourly basis. The billing formula is as follows:
Total Serverless fees = Compute node fees + Storage space fees
= Serverless compute unit price x CCU usage + Storage space price x storage capacity
For specific compute unit prices and storage space prices, see Billing Overview.

What Is CCU?

TDSQL-C Compute Unit (CCU) is the billing unit for computing resources in Serverless. One CCU is approximately equal to one CPU and 2 GB of memory.
For each billing cycle, the number of used CCUs is determined by taking the maximum value between the number of CPU cores used by the database and half of the allocated memory (in GB).

What Are the Key Features of the Serverless Service?

The Serverless service provides the following key features:

- The adjustable elastic scaling range of CCUs allows the Serverless cluster to automatically increase or decrease the number of CCUs within this range based on actual business pressure, making it more adaptable to fluctuating computing demands.
- The Serverless service continuously monitors the user's CPU, memory, and other workload conditions and triggers the automatic scaling policy based on predefined rules.
- The Serverless service supports custom auto-pause settings. If no connection is detected, the instance automatically pauses and instantly resumes without interruption when a task connection is received.

- The kernel team has implemented relevant optimizations, enabling the Serverless service to have elastic anti-jitter capabilities, which make the cluster's elastic scaling more stable.

How Do I Create a Serverless Cluster?

You can create a Serverless cluster through the Purchase page by selecting Serverless as the instance mode. For details, see Creating a Serverless Cluster.

In Which Regions Is the Serverless Cluster Available?

Currently, Serverless is supported only in Guangzhou, Shanghai, Beijing, Chengdu, Nanjing, Hong Kong (China), Silicon Valley, Virginia, Singapore, Tokyo, and Seoul. If your desired region is not listed, you can submit a ticket to provide feedback.

Which Database Versions Are Supported for Serverless Clusters?

Serverless clusters support MySQL 5.7 and MySQL 8.0, the same as those supported by provisioned resource clusters.

What Deployment Architectures Are Available for Serverless Clusters?

Both single-node edition and cluster edition architectures are supported.

- Single-node edition architecture: A cluster with one read-write instance. It can be switched to cluster edition architecture in the console after purchase.
- Cluster edition architecture: A cluster with one read-write instance and 1-15 read-only instances (optional). It can be switched to single-node edition architecture in the console after purchase.

What Is the Compute Unit Elasticity Range for Serverless Clusters?

The minimum compute unit configuration for a single Serverless instance can be set to 0.25 CCU. The maximum compute unit configuration for a single Serverless instance can be set to 64 CCUs. The elastic range of compute unit for the same Serverless instance can be set from 0.25 to 32 CCUs (approximately 32 vCPU and 64 GB of RAM). Different availability zones (AZs) may have different resource limits. For details, see Service Compute Unit Configuration.

> **Note:**
> It is recommended to set the minimum capacity to 1 CCU and select a higher value for the maximum capacity when configuring the elastic range for the first time. A lower capacity setting allows the cluster to scale in to the maximum extent when idle, avoiding unnecessary fees. A higher capacity setting enables the cluster to scale out as needed during peak workloads, ensuring stable performance during business traffic surges.

What Is the Maximum Storage Space for Serverless Clusters?

The maximum storage space varies based on the compute unit configuration. The maximum supported storage space is 384 TB. For details, see Service Compute Unit Configuration.

How Much Cost Savings Can a Serverless Cluster Provide Compared to a Provisioned Resource Cluster?

To estimate the cost savings of switching from a non-Serverless cluster (provisioned resource cluster) to a Serverless cluster, you can use the Serverless Cost Estimator for comparison.

Can a Serverless Cluster Mount a Read-Only Instance from a Provisioned Resource Cluster?

It is not supported.

Can a Provisioned Resource Cluster Mount a Serverless Read-Only Instance?

Yes, this can be achieved by enabling the Serverless feature. For operations and instructions, see Enabling or Disabling Serverless for a Provisioned Resource Cluster.

Can the Configuration of a Serverless Cluster Be Adjusted?

Yes, you can modify the configuration of a Serverless cluster through the console. For details, see Adjusting Configuration.

Does a Serverless Cluster Support Multi-AZ Deployment?

Yes. A Serverless cluster can be deployed across multiple AZs if the following conditions are met: The selected region has at least two AZs. The target AZ has sufficient computing resources. For database version MySQL 5.7, the kernel minor version should be at least 2.0.15. For database version MySQL 8.0, the kernel minor version should be at least 3.0.1.

Which Regions and AZs Support Multi-AZ Deployment for Serverless Clusters?

For details, see Multi-AZ Deployment. If your desired region or AZ is not available, you can submit a ticket to apply to deploy Serverless clusters in the region and AZ.

How Are Refunds Handled for Serverless Clusters?

Since Serverless clusters are billed on a pay-as-you-go basis, fees are calculated based on actual resource usage at the time of settlement. If you have not purchased a resource package, refunds are not involved. Ensure that your account remains funded, as overdue payments may result in cluster isolation and deletion, leading to data loss. For details, see Notes on Arrears.

Can Individual Read-Only Instances Be Manually Started or Stopped?

Yes. Regardless of whether automatic start/stop is enabled for the Serverless cluster, individual read-only instances within the cluster can be manually started or stopped.

Resource Package FAQs

Last updated: 2025-05-28 14:43:09

This document introduces FAQs related to resource packages.

What Is a Resource Package?

A resource package is a prepaid resource type introduced by TDSQL-C for MySQL. It is divided into computing resource packages and storage resource packages, which can be used to offset the computing and storage resources consumed by Serverless clusters. The unit price of a resource package is more affordable compared to pay-as-you-go billing. It is recommended to purchase and bind resource packages to Serverless instances first.

Can a Provisioned Resource Cluster Use Resource Packages?

Provisioned resource clusters support mounting read-only instances with provisioned resources and Serverless read-only instances. However, only Serverless read-only instances support using resource packages. A Serverless cluster can use resource packages.

What Types of Resource Packages Are Available?

Resource packages are divided into computing resource packages and storage resource packages. The computing resource package is a fixed quota resource package for computing resources, used to offset the computing resources of Serverless instances bound to the resource package. Once the resource package quota is used up, the cluster will continue to charge based on pay-as-you-go billing. The storage resource package is a fixed quota resource package for storage resources, used to offset the storage resources of Serverless instances bound to the resource package. Once the resource package quota is used up, the cluster will continue to charge based on pay-as-you-go billing.

How Many Resource Packages Can Be Bound to a Cluster?

A cluster can be bound to up to 10 computing resource packages and 10 storage resource packages simultaneously.

How to Use a Resource Package?

- If you have purchased a Serverless cluster, you can directly purchase a Serverless resource package to offset the computing and storage resource fees within the cluster.
- If you have purchased a provisioned resource cluster and mounted a Serverless read-only instance by enabling Serverless, you can directly purchase a Serverless computing resource package to offset the computing resource fees of the Serverless read-only instance within the cluster.
- If you have not purchased any Serverless clusters, you can quickly use the resource package by following these three steps.

  - Purchasing Resource Packages
  - Creating Serverless Clusters
  - Binding Resource Packages
- If your current provisioned resource cluster has not enabled Serverless, you only need the following three steps to quickly use a Serverless computing resource package.
  - Purchasing Compute Resource Packages
  - Enabling Serverless for Pre-configured Resource Clusters
  - Binding Resource Packages

How to Purchase a Resource Package?

You can purchase a resource package through the resource package purchase page or the console. For detailed operations, see Purchasing Resource Packages.

What Is the Consumption Order of Multiple Resource Packages?

When multiple resource packages of the same type (computing or storage) are bound, the backend will automatically set the consumption order based on the order of binding. You can manually adjust this order by modifying the consumption priority ranking. The consumption rule follows the priority level (1–10) in ascending order, with lower numbers being consumed first. For details, see Adjusting Consumption Priority Order.

Will Unused Resources in a Resource Package Expire?

Resource packages have a **validity period**. You can choose a validity period of 6 months or 1 year when purchasing the package. If the resources are consumed before the validity period expires, or if the validity period ends, the resource package can no longer be used. You will need to purchase and bind a new resource package. You can check the resource package usage details and expiration time via Viewing Resource Package Usage.

Is It Necessary to Purchase a Serverless Resource Package After Purchasing a Serverless Cluster?

No, after you purchase a Serverless cluster, the billing mode is pay-as-you-go. You can choose whether to purchase a Serverless computing or storage resource package. After purchasing and binding, the resource package will automatically offset the computing or storage resource fees in the Serverless cluster. If the resource package is used up and no new resource package is bound, the Serverless cluster will continue to charge based on the pay-as-you-go mode.

Do Serverless Clusters Created Before a Serverless Resource Package Is Purchased Support Using Resource Packages?

It is supported.

Do Provisioned Resource Clusters that Enable Serverless Before a Serverless Resource Package Is Purchased Support Using Resource Packages?

Yes, but only computing resource packages can be bound, and they are used to offset the elastic computing resources of Serverless read-only instances under the provisioned resource cluster.

Can Resource Packages Be Upgraded?

It is not supported.

Can Resource Packages Be Renewed?

It is not supported.

Can the Validity Period of a Resource Package Be Modified?

No, since the unit prices of 6-month and 1-year resource packages differ, modifying the validity period after purchase is not supported.

How to Request a Refund for a Resource Package?

- Resource packages that have been used up or have expired do not support refunds.
- For resource packages that have remaining resources within the validity period, self-service refunds can be requested via the console. Each Tencent Cloud account (root account) can request up to 20 resource package refunds per calendar year. For refund instructions, see Requesting Refund for Resource Package.

> **Note:**
> Regarding the limit of refunds, the system calculates it based on the calendar year. The first calendar cycle is from June 30, 2023, 23:59:59 to July 1, 2024, 00:00:00. The number of refunds is reset at 00:00:00 on July 1 each year. The second cycle runs from July 1, 2024, 00:00:00 to June 30, 2025, 23:59:59, and so on.

Serverless Cost Estimator

Last updated：2024-12-18 11:07:51

TDSQL-C for MySQL provides Serverless service to meet the database service requirements of enterprises in specific business scenarios (such as development testing and volatile business). It also supports Serverless cost estimation to help you better understand whether the cost of using Serverless service meets your budget requirements. This document introduces the Serverless Cost Estimator and how to use it.

Prerequisites

- You already have a Tencent Cloud account. For information on how to create an account, see Account Registration.
- You have completed identity verification for your Tencent Cloud account.

Application Scenario

- If you already have fixed-specification TDSQL-C for MySQL instances under your Tencent Cloud account and need to migrate the business hosted on these existing instances to Serverless service, you can use the Serverless Cost Estimator to compare costs before and after the migration.
- If you do not have any fixed-specification TDSQL-C for MySQL or TencentDB for MySQL instances under your Tencent Cloud account and want to purchase Serverless service, you can use the Serverless Cost Estimator to check whether the cost of using Serverless service is suitable based on the usage of the database instances in your daily operations.

Precautions

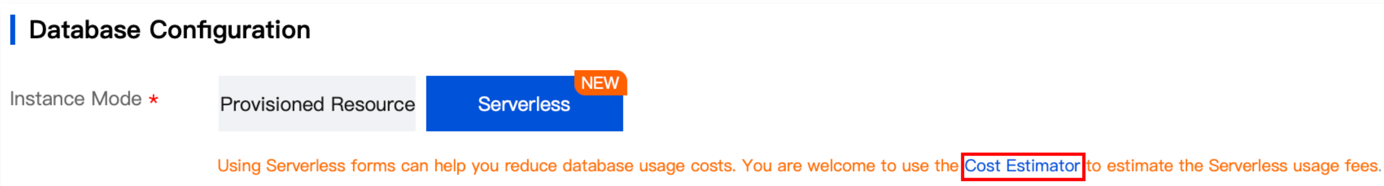
- The Serverless Cost Estimator currently supports estimated cost comparisons between existing non-Serverless instances and Serverless instances. The non-Serverless instance types currently supported are: TDSQL-C for MySQL instances with provisioned resources.
- When you are comparing the estimated costs between existing non-Serverless instances and Serverless instances, the billing mode for the compute and storage resource prices displayed for the existing instances is consistent with that of the source instance. That is, if the source instance is billed on a monthly subscription basis, the cost comparison will be based on monthly subscription. If the source instance is billed on a pay-as-you-go basis, the cost comparison will be based on pay-as-you-go.
- When you are comparing the estimated costs between new instances and Serverless instances, the billing mode for the compute resource prices displayed for the new instances is monthly subscription, while the billing mode for the storage resource prices is pay-as-you-go.

Using Cost Estimator

Note:

- The system evaluation results are based on monitoring data within the evaluation time range, simulating a month's traffic load for evaluation. Therefore, there may be discrepancies between the estimated costs and the actual costs. This estimation is for reference only. You can refer to your bill for the actual usage costs.
- Monthly compute resource consumption: Based on historical monitoring data, the compute resource consumption over a one-month period is estimated as a reference for purchasing compute resource packages.
- Monthly storage resource consumption: Based on historical monitoring data, the storage resource consumption over a one-month period is estimated as a reference for purchasing storage resource packages.
- All prices in the Cost Estimator are estimated based on list prices and do not include any discounts, vouchers, or other promotional offers associated with an account.

1. Log in to the purchase page.
2. In **Database Configuration** > **Instance Mode**, select Serverless.
3. Click **Cost Estimator** in the prompt.



4. In the sidebar that appears on the right, complete the following steps based on your needs to obtain a system evaluation of the costs.

Welcome to Use the Serverless Cost Estimator ×

The Serverless Cost Estimator helps you estimate the cost savings from migrating from non-Serverless to Serverless databases, and recommends suitable resource package purchase methods based on your resource usage.

step1: Select Source Instance

◉ Comparison of Existing Instances ○ Evaluation of New Instances

Region	Instance ID	Evaluation Time (At Least 1 Day and up to 7 Days)
Beijing ▾	cynosdbmysql-ins cynosdbmysql-i) ▾	2024-12-10 ~ 2024-12-11

step2: Evaluation Results

After migration to Serverless instances, the estimated monthly consumption of computing resources is 648.00 thousand CCU and that of storage resources is 23.04 GB. The cost is compared based on the list price as follows:

Cost Compari...	Source Instance 54.16 USD/month	Serverless (pay-as-you-go) 9.08 USD/month Save 45.08 USD/month (Cost-efficient)	Serverless (Resource Pack) 6.82 USD/month Save 47.34 USD/month (Cost-efficient) Enterprise Edition ▾ 6 months ▾
Cost De...			
Compute	48.00 USD (2 cores/4GB Dedicated/Monthly Subscription)	9.07 USD	6.82 USD
Storage	6.16 USD (Monthly Subscription)	0.02 USD	0.01 USD
Trends			

step3: Evaluation Recommendations

According to the system evaluation, after migrating the source instance to a Serverless instance, you will save 45.08USD per month. If you purchase a resource package, you will save 47.34USD per month. It is recommended to purchase Serverless instances or resource packages.

Purchase Resource Pack

4.1 **Select Source Instance**. Select the source instance for cost comparison. You can select an existing instance or a newly purchased instance for evaluation.

- Existing instance comparison: If you choose an existing instance for comparison, select the region, instance ID, and evaluation time (at least 1 day, up to 7 days) for your existing instance.
- New Instance Purchase Evaluation: If you do not have a source instance in your Tencent Cloud account for comparison, you can choose New Instance Purchase Evaluation and then select the region, instance specifications, average daily CPU utilization (minimum value can only be set to 0.01%), average daily memory usage, and average daily storage usage.

> **Note:**
> - If you choose an existing instance for comparison, after you set the region, instance ID, and evaluation time, the system will automatically conduct an evaluation. If you choose to evaluate a newly purchased instance, after setting the region, instance specifications, and other information, you need to click **Start Evaluation** to perform the system evaluation.
> - Each day within the selected evaluation time period must have complete 24-hour monitoring data. Therefore, the purchase day of the instance and the evaluation day are not supported for inclusion in the evaluation.

4.2. **Evaluation Results**. In this step, you can view the cost comparison between the source instance, Serverless (pay-as-you-go), and Serverless (resource package).
4.3. **Evaluation Recommendations**. In this step, you can view system evaluation suggestions. If the system evaluation shows that migrating to Serverless instances can save costs every month, it is recommended to purchase Serverless instances and resource packages to provide services for your business. If the system evaluation shows that migrating to Serverless instances results in higher costs compared to the source instance, it is recommended to continue using the source instance or purchase TDSQL-C for MySQL instances with provisioned resources.

5. After you finish using the Serverless Cost Estimator, click the close button in the upper right corner to return to the purchase page and continue with the subsequent purchase operations.