Table Store

Purchase Guide

MORE THAN JUST CLOUD | Alibaba Cloud

Purchase Guide

Purchase process

Table Store is a post-paid product. The metering and billing start after the table is created.

For more information, refer to Pricing for Table Store.

Operation procedure

Activate the Table Store service. For details, refer to Table Store activation procedure.

Log on to the Table Store console.

Locate the target instance and click its name or **Manage** in the Action column to go to the **Instance Details** page.

Click **Create Table** and fill in the required table attributes to create a table. For details, refer to Quick Start - Table management.

Billing

Overview

Table Store uses the following four dimensions to meter the resources used by applications for billing purpose:

Data storage

- The reserved read/write throughput
- The additional read/write throughput
- The Internet downstream traffic

The bills will be generated by hour. The fee will be automatically deducted from cash coupon at first by hour.

Data storage

Data storage hourly fees are based on the total volume of instance data. Due to constant changes in the total data volume, Table Store collects the total data volume of all table partitions at regular intervals to calculate the average total data volume per hour. This average value is then multiplied by the unit price to become the fee.

An instance’ s total data volume is the sum of data from all tables in the instance. The table’ s total data volume is the sum of data in all rows of the table. The following example illustrates how to calculate a row’ s and a table’ s data volume.

Calculation of a row’ s data volume

The data in each row of a table occupies the space in Table Store. When the multi-version or TTL feature is enabled, the data of each version includes the version number (eight bits), column name and data value.

The storage space is calculated as follows:

- Data size of a single row = Size of the Primary Key’ s data + Size of all Attribute columns’ data
- Data size of a Primary Key = Total name length of the Primary Key columns + Total size of the values of the Primary Key columns
- When the multi-version and TTL features are not enabled (MaxVersions = 1 and TTL = –1): Data size of a single Attribute column = Total name length of the Attribute column + Total size of the values of the Attribute column
- When the multi-version or TTL feature is enabled (MaxVersions > 1 or TTL = –1), each version number occupies 8 Bytes of the storage space. Data size of a single Attribute column

= (Total name length of the Attribute column + 8) * Number of the valid versions + Total size of the values of all the valid versions in the Attribute column

The data size of the column values is calculated as follows:

String - Bytes of the string in UTF-8 encoding. If the string is null (Table Store supports the null string type), the data size is 0.

Integer - 8 Bytes as reserved.

Double - 8 Bytes as reserved.

Boolean - 1 Byte as reserved.

Binary - Bytes of the Binary data.

An example of how to calculate a row’ s data size is as follows:

“id” (Integer) is the Primary Key column of the table.

id	name	length	comments
1	timestamp = 1466676354000, value = ‘zhangsan’	timestamp = 1466676354000, value = 20	timestamp = 1466676354000, value = String (100 Bytes); timestamp = 1466679954000, value = String (150 Bytes)

In the above table, there are two valid versions for Attribute column “comments” .

When MaxVersion = 2 and TTL = 2592000, the row’ s data size = 10 + 20 + 24 + 282 = 336 Bytes. The detailed calculation is as follows :

Data size of the Primary Key = len (‘id’) + len (1) = 10 Bytes

Data size of the Attribute column “name” = (len (‘name’) + 8) * 1 + len (‘zhangsan’) = 20 Bytes

Data size of the Attribute column “length” = (len (‘length’) + 8) * 1 + len (20) = 22 Bytes

Data size of the Attribute column “comments” = (len (‘comments’) + 8) * 2 + 100 + 150 = 282 Bytes

When MaxVersions = 1 and TTL = -1, the row’ s data size = 10 + 12 + 14 + 158 = 194 Bytes. The detailed calculation is as follows :

Note: Although there are two versions for the column “comments” , as a result of MaxVersions = 1, only the latest version is valid.

Data size of the Primary Key = len (‘id’) + len (1) = 10 Bytes

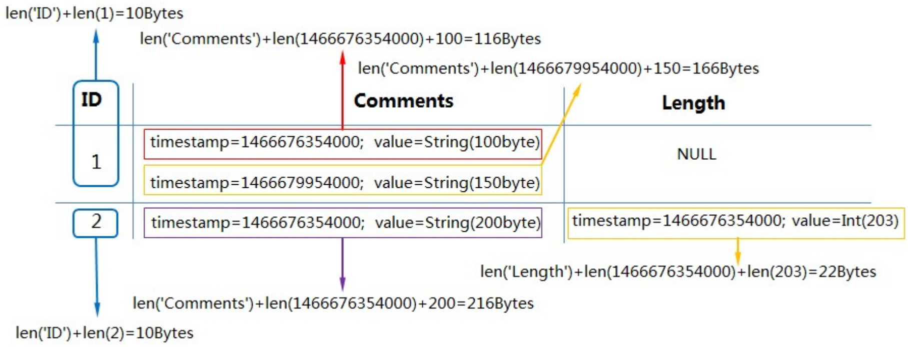
Data size of the Attribute column “name” = len (‘name’) + len (‘zhangsan’) = 12 Bytes

Data size of the Attribute column “length” = len (‘length’) + len (20) = 14 Bytes

Data size of the Attribute column “comments” = len (‘comments’) + 150 (Bytes) = 158 Bytes

Calculation of the table’ s data size

Assume that there is a table whose Primary Key is ID and other columns are Attribute columns. If its MaxVersions = 2 and TTL = –1, the table’ s data size is calculated as follows:



ID	Comments	Length

1	timestamp=1466676354000; value=String(100byte)
	timestamp=1466679954000; value=String(150byte)
		NULL

2	timestamp=1466676354000; value=String(200byte)	timestamp=1466676354000; value=Int(203)

len('ID')+len(1)=10Bytes
len('Comments')+len(1466676354000)+100=116Bytes
len('Comments')+len(1466679954000)+150=166Bytes
len('Length')+len(1466676354000)+len(203)=22Bytes
len('Comments')+len(1466676354000)+200=216Bytes
len('ID')+len(2)=10Bytes

The data size of the row whose ID is 1 = 10 (the Primary Key size) + (116 + 166) (total data size of the two versions in the Attribute column “Comments”) = 292 Bytes

The data size of the row whose ID is 2 = 10 (the Primary Key size) + 216 (data size of a version in the Attribute column “Comments”) + 22 (data size of a version in the Attribute column “Length”) = 248 Bytes

- The table’ s data volume = 292 + 248 = 540 Bytes.

If the table’ s data volume does not change within an hour, the table is billed for 540 bytes. Table Store does not limit the data volume for an individual table. You only need to pay for what you use.

Table Store bills the high-performance instances in real time. For example, data is written at constant speeds to a 1 GB data table during a period of 1 hour. After the data is written, the table size is changed to 5 GB, resulting in an average data size of 3 GB during that hour. Table Store bills the table for 3 GB.

Note:

> With support for the multi-version and TTL features, Table Store asynchronously clears the expired data and the version data exceeding the value of MaxVersions from each partition. Then Table Store calculates the data size of the partition. The clearance duration is related to the total data volume, but is typically finished within 24 hours. The data that is written to a partition after a clearance operation is added to the partition’ s data volume upon completion of the next clearance operation.
>
> With support for the multi-version or TTL feature, Table Store only provides final precise measurement of the stored data volume instead of real-time precise measurement. That is, Table Store measures the data volume after a time if no data is written or the data expires. For the high-performance instances, Table Store still provides the real-time precise measurement of the data volumes before the multi-version or TTL feature is supported.

Reserved read/write throughput

The reserved read/write throughput is a table’ s attribute. You can set a proper reserved read/write throughput for your data tables to reduce the costs of resource usage.

The capacity instances do not support reserved read/write throughput. Table Store charges an hourly fee for the total reserved read/write throughput of all tables in a high-performance instance. The configured reserved read/write throughput may change constantly. Table Store collects the tables’ reserved read/write throughput at regular intervals to calculate the hourly average throughput. The average value is then multiplied by the unit price to become the hourly fee.

Additional throughput

The additional throughput is the portion of the actual consumed read/write throughput that exceeds the reserved read/write throughput per second. The additional read/write throughput is measured every second.

Table Store accumulates the additional read throughputs and write throughputs of all tables in an instance during every billing cycle. The actual consumed additional throughput is multiplied by the corresponding unit price to become the charge.

Table Store charges fees when applications access the Internet downstream traffic of Table Store. Applications’ use of the HTTP method to access the responses returned by Table Store is the main component of the downstream traffic. Even if the operation fails, the operation failure information returned by Table Store will still produce downstream traffic.

Table Store only charges for the Internet downstream traffic, not for the intranet downstream traffic or the Internet upstream traffic. The access among different regions also belongs to the Internet access.

Overdue payment tips

The fee is settled by hour. When the account balance is less than the current bill amount, you will be notified via SMS or email.

For late payment, you will be notified via SMS or email, requesting you to recharge your account within 24 hours. All the services are not affected during that period.

If you still do not successfully credit your account and clear off the overdue payables after 24 hours, Alibaba Cloud will suspend the service and freeze your Table Store, but still store your data and continue billing. You will be notified via SMS or email.

If your account is delinquent for over 15 days, and you still do not successfully credit your account and clear off the overdue payables, Alibaba Cloud will terminate these service terms and conditions and stop providing services for you; and meanwhile, all your data stored on Table Store will be deleted and emptied and be unrecoverable forever. You will be notified via SMS or email one day before the data are emptied.