

**ALIBABA CLOUD**

# **Alibaba Cloud**

## **MaxCompute Product Introduction**

**Document Version: 20200927**









 **Alibaba Cloud**

## Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.
2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.
3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.
4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).
5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.
6. Please directly contact Alibaba Cloud for any errors of this document.

# Document conventions

Style	Description	Example
 <b>Danger</b>	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	 <b>Danger:</b> Resetting will result in the loss of user configuration data.
 <b>Warning</b>	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	 <b>Warning:</b> Restarting will cause business interruption. About 10 minutes are required to restart an instance.
 <b>Notice</b>	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	 <b>Notice:</b> If the weight is set to 0, the server no longer receives new requests.
 <b>Note</b>	A note indicates supplemental instructions, best practices, tips, and other content.	 <b>Note:</b> You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click <b>Settings&gt; Network&gt; Set network type</b> .
<b>Bold</b>	<b>Bold</b> formatting is used for buttons, menus, page names, and other UI elements.	Click <b>OK</b> .
<b>Courier font</b>	Courier font is used for commands	Run the <code>cd /d C:/window</code> command to enter the Windows system folder.
<i>Italic</i>	Italic formatting is used for parameters and variables.	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[ ] or [a b]	This format is used for an optional value, where only one item can be selected.	<code>ipconfig [-all -t]</code>
{ } or {a b}	This format is used for a required value, where only one item can be selected.	<code>switch {active stand}</code>

---

# Table of Contents

1.What is MaxCompute? .....	05
2.Usage notes .....	08
3.Definitions .....	13
3.1. MaxCompute glossary .....	13
3.2. Project .....	15
3.3. Table .....	16
3.4. Partition .....	17
3.5. Lifecycle .....	19
3.6. Resource .....	20
3.7. Function .....	21
3.8. Task .....	21
3.9. Instance .....	22
3.10. ACID semantics .....	22
4.Limits .....	24
5.MaxCompute features in different regions .....	25
6.Storage performance .....	27

# 1. What is MaxCompute?


The big data computing service (MaxCompute, formerly called ODPS) is a fast and fully hosted GB/TB/PB level data warehouse solution.

MaxCompute supports a variety of classic distributed computing models that enable you to solve massive data calculation problems while reducing business costs, and maintaining data security.

MaxCompute seamlessly integrates with DataWorks, which provides one-stop data synchronization, task development, data workflow development, data operation and maintenance, and data management for MaxCompute. For more information, see [What is DataWorks?](#).

MaxCompute is mainly used to store and compute batches of structured data. It provides a massive range of data warehouse solutions as well as big data analysis and modeling services. As data collection techniques are becoming increasingly diverse and comprehensive, industries are amassing larger and larger volumes of data. The scale of data has increased to the level of massive data (100 GB, TB and even PB) that traditional software industry can not carry.

Given these massive data volumes, the limited processing capacity of a single server has prompted analysts to move towards distributed computing. However, distributed computing models are not easy to maintain and demand highly-qualified data analysts. When using a distributed model, data analysts not only need to understand their business needs, but also must be familiar with the underlying computing model. The purpose of MaxCompute is to provide you with a convenient way of analyzing and processing mass data, and you can achieve the purpose of analyzing large data without having to care about the details of distributed computing.

 **Note** MaxCompute is widely used by Alibaba Group in scenarios such as data warehouse and BI analysis, web log analysis, transaction analysis of e-commerce sites, and customer behavior analysis.

## MaxCompute learning path

You can quickly learn about MaxCompute's related concepts, basic operations and advanced operations through [MaxCompute learning path](#).

## Product advantage

- **Large-scale computing and storage**

MaxCompute is suitable for the storage and processing of large volumes of data (up to PB-level).

- **Multiple computational models**

MaxCompute supports data processing methods based on SQL, MapReduce, Graph, MPI iteration algorithm, and other programming models.

- **Strong data security**

MaxCompute has stabilized alloffline analysis for all Alibaba Group's business for more than seven years, providing multilayer sandbox protection and monitoring.

- **Cost-effective**

MaxCompute can help reduce procurement costs by 20%-30% compared with on-premises private cloud models.

## Function

- **Data tunnel**

- Supports large volumes of historical data channels

**TUNNEL** provides high concurrency data upload and download services. This service supports the import and export of terabytes or petabytes of data on a daily basis, which is particularly useful for the batch import of full or historical data. Tunnel Provides you with a Java programming interface, and in the MaxCompute client tool, there are corresponding commands for local file and service data interchange.

- Real-time and incremental data channels

For real-time data upload scenarios, MaxCompute provides DataHub services with low latency and convenient usage. It is especially suitable for incremental data imports. DataHub also supports a variety of data transmission plug-ins, such as Logstash, Flume, Fluentd, and Sqoop, it supports Log. Service's delivery log to MaxCompute, and then use DataWorks to do log analysis and mining.

- **Computing and analysis tasks**

MaxCompute provides multiple computing models.

- **SQL:** In MaxCompute, data is stored in tables. MaxCompute provides an SQL query function for the external interface. You can operate MaxCompute similarly to a traditional database software but with the ability to process PB-level data.

 **Note**

- MaxCompute SQL does not support transactions, index, or Update/Delete operations.
- MaxCompute SQL syntax differs from Oracle and MySQL, notably, you cannot seamlessly migrate SQL statements of other databases into MaxCompute.
- In terms of usage, MaxCompute SQL can complete queries at the second- to millisecond-level, and can not return results at milliseconds.
- The advantage of MaxCompute SQL is low learning cost. You don't need to understand the concept of complex distributed computing. If you have experience in database operations, you can familiarize yourself with MaxCompute SQL quickly.

- **UDF:** A user-defined function.

MaxCompute provides numerous **built-in functions** to meet your computing needs, while also supporting the creation of custom functions.

- **MapReduce:** MapReduce is a Java MapReduce programming model provided by MaxCompute. It uses the Java programming interface and is designed to simplify the development process. However, users are recommended to have a basic understanding of the concept of distribution, and relevant programming experience before using MapReduce. MaxCompute MapReduce provides you with Java programming interface.

- **Graph**: Graph in MaxCompute is a processing framework designed for iterative graph computing. Graph computing jobs use graphs to build models. Graphs are composed of vertices and edges. Vertices and edges contain values. After performing iterative graph editing and evolution, you can get the final result. Typical applications include PageRank, SSSP algorithm, and K-Means algorithm. The graph is edited and evolved through an iteration, and the results are finally solved. Typical applications: [PageRank](#), [single source shortest distance algorithm](#), [K-means clustering algorithm](#), and so on.
- **SDK**  
A convenient toolkit provided for developers. For more information, see [MaxCompute SDK](#).
- **Secure**  
Maxcompute offers powerful security services to protect your data, for more information, see the [security guide](#).

## 2.Usage notes

This topic provides reading recommendations based on your roles.

learning path

### MaxCompute beginners

If you are a beginner in MaxCompute, we recommend that you first familiarize yourself with the modules described in the following table.

Module	Description
<a href="#">Product Introduction</a>	Provides the MaxCompute overview and describes its features. This topic helps you obtain a general knowledge of MaxCompute.
<a href="#">Quick Start</a>	Describes how to apply for an account, install the client, create a table, grant permissions, import and export data. It also describes how to execute SQL jobs, user-defined functions (UDFs), and MapReduce jobs.
<a href="#">MaxCompute glossary</a>	Introduces the basic terms of MaxCompute.
<a href="#">List of common commands</a>	Describes the commonly used commands in MaxCompute. This topic helps you familiarize yourself with operations on MaxCompute.
<a href="#">Tools and Downloads</a>	Before you analyze data, you must understand how to download, configure, and use the commonly used tools in MaxCompute.
<a href="#">Client</a>	You can use the client to perform operations on MaxCompute.
<a href="#">Configure endpoints</a>	Describes the regions in which MaxCompute is available, MaxCompute connection methods, and issues that arise from its use with other Alibaba Cloud services, such as Elastic Compute Service (ECS), Tablestore, and Object Storage Service (OSS). These issues include network connectivity issues and issues related to data download charges.

### Data analysts

If you are a data analyst, we recommend that you familiarize yourself with the [SQL](#) topic. You can query and analyze large volumes of data stored in MaxCompute. MaxCompute SQL provides the following features:

- Supports data description language (DDL) statements.
- Uses CREATE, DROP, and ALTER statements to manage both tables and partitions.
- Uses the SELECT statement to select data records in a table and the WHERE clause to view data records that meet specific conditions. These statements help filter data records.
- Joins two tables by using equi-joins.
- Uses the GROUP BY clause to aggregate columns.
- Uses the INSERT OVERWRITE or INSERT INTO statement to insert data records into another table.




- Uses built-in functions and UDFs to complete a variety of computations.
- Collects table statistics and configures table lifecycles.
- Supports regular expressions.

## Users with development experience

If you have development experience, understand the distributed architecture, and want to obtain data analytics capabilities that SQL cannot deliver, we recommend that you read advanced functional modules of MaxCompute.

Module	Description
<b>MapReduce</b>	MaxCompute provides the MapReduce programming model for Java. You can use the Java API provided by MapReduce to write MapReduce programs and process MaxCompute data.
<b>Graph</b>	Graph is a processing framework for iterative graph computing. A graph consists of vertices and edges, both of which contain values. MaxCompute Graph iteratively edits and evolves graphs to obtain analysis results.
<b>Tunnel</b>	MaxCompute Tunnel enables you to upload or download large amounts of data to or from MaxCompute at a time.
<b>Java SDK</b>	A Java API is provided for developers.
<b>Python SDK</b>	A Python API is provided for developers.


 **Note** **MapReduce** and **Graph** features are in public preview. If you want to use these features, [submit a ticket](#). In the ticket, you must specify the name of your project. The system processes the ticket within seven business days.

## Project owners or administrators

If you are a project owner or administrator, we recommend that you familiarize yourself with the modules described in the following table. A project owner can create and use projects while a project administrator can manage projects, security operations, and costs.

Module	Feature	Description
--------	---------	-------------

Module	Feature	Description
	Prepare for project creation	<p>A project is a basic organizational unit of MaxCompute. Similar to a database or schema in a traditional database system, a project is used to isolate users and control access requests. A user can have permissions on multiple projects. After a user is authorized to access multiple objects, the user can access objects across the projects, such as tables, resources, functions, and instances. MaxCompute is used to manage the various objects in projects. Preparations before project creation:</p> <ul style="list-style-type: none"> <li>• Budget for resources                     <p>You are charged for storage resources, computing resources, and resources for Internet-based data downloads.</p> <ul style="list-style-type: none"> <li>◦ Storage resources: You are charged for these resources based on the pay-as-you-go billing method and tiered unit prices. You can estimate their costs based on the volume of data stored. Data stored in MaxCompute changes all the time. As a result, the costs also change.</li> <li>◦ Computing resources: You are charged for these resources based on both the pay-as-you-go and subscription billing methods. These resources are consumed to execute computing jobs, such as SQL statements, MapReduce jobs, Spark jobs, and Lightning jobs. It is difficult to estimate the number of required computing resources at the beginning of your project. We recommend that you use the pay-as-you-go billing method and then decide whether to switch to the subscription billing method based on the number of computing resources used.</li> <li>◦ Resources for Internet-based data downloads: You are charged for these resources based on the pay-as-you-go billing method.</li> </ul> <p>For more information, see <a href="#">Storage pricing (pay-as-you-go)</a>, <a href="#">Computing pricing</a>, and <a href="#">Download pricing(Pay-As-You-Go)</a>.</p> </li> <li>• Prepare an account and activate the service                     <p>Before you create a MaxCompute project, you must create an Alibaba Cloud account and then activate MaxCompute. Bills are issued to the Alibaba Cloud account. After the account is created, you must choose the pay-as-you-go or subscription billing method based on your budget for the resources you require.</p> </li> </ul>
	Create a project	For more information, see <a href="#">Create a project</a> .
	Manage project members	Members are managed from the perspectives of responsibilities and data security. If you use MaxCompute in the DataWorks console, you must understand the relationship between the permissions for the two services..

Module	Feature	Description
Project management	Manage RAM users	<p>You can manage MaxCompute projects by using your Alibaba Cloud account or a RAM user. You can add RAM users under your Alibaba Cloud account to a MaxCompute project. However, MaxCompute does not authenticate these RAM users based on the permissions that are granted to the RAM users in Resource Access Management (RAM). For more information about RAM users, see <a href="#">Prepare a RAM user</a>.</p> <p>If you manage MaxCompute projects and DataWorks workspaces in the DataWorks console, you can add only RAM users under your Alibaba Cloud account as members. Therefore, you must use your Alibaba Cloud account to create RAM users and manage these RAM users.</p> <div style="background-color: #e6f2ff; padding: 10px; border: 1px solid #d9e1f2;"> <p> <b>Note</b></p> <ul style="list-style-type: none"> <li>• We recommend that you use one RAM user as one project member. Do not allow multiple members to share the same RAM user.</li> <li>• You must promptly delete the RAM users that correspond to members who are transferred to new positions or resign. If a RAM user is added as a project member in the DataWorks console, delete the project member in DataWorks and then delete the RAM user in RAM.</li> </ul> </div>
	Manage scheduling resources	<p>Scheduling resources of DataWorks. These resources are used to execute or distribute the tasks that are delivered by the scheduling system. Scheduling resources of DataWorks are categorized into the following types. For more information, see <a href="#">View resource groups</a>.</p> <ul style="list-style-type: none"> <li>• Default scheduling resources. Default scheduling resources are the public resource pool of DataWorks. If the concurrency of DataWorks nodes is high and scheduling resources are insufficient, the nodes wait for resources. After resources are allocated to the nodes, the nodes execute the delivered tasks.</li> <li>• Custom scheduling resources. You can configure your ECS instance as a scheduling server to execute or distribute delivered tasks. You can use your Alibaba Cloud account to create such resources. Scheduling resources include several physical machines or ECS instances that are used to execute tasks, such as data synchronization. You can submit a ticket to create a scheduling resource group. Custom scheduling resource groups that exist are not affected. This eliminates the limits of the default scheduling resource group.</li> </ul>

Module	Feature	Description
	Configure projects	<p>Only the owner of a project has the permissions to configure the project. For example, the project owner can specify whether to enable full table scan and whether to use the MaxCompute V2.0 data type edition for a project by default. For more information, see <a href="#">Project operations</a>.</p>
Cost management	None	<p>Budgets for resources help you estimate costs before you use the resources. It is difficult to estimate the precise costs due to the billing methods of MaxCompute. You must manage costs during the entire business development process.</p> <ul style="list-style-type: none"><li>• For more information about pricing, see <a href="#">Billing</a>.</li><li>• You can switch between the pay-as-you-go and subscription billing methods. For more information, see <a href="#">Switch billing methods</a>.</li></ul>

## 3. Definitions

### 3.1. MaxCompute glossary

This article lists the common concepts and terminologies of MaxCompute. For detailed description, please refer to the link in this article.

#### A

- AccessKey

Access Key (AK for short, including Access Key Id and Access Key Secret) is the key to access the Aliyun API. After registering cloud account on Ali cloud official website, it can be generated on the Accesskeys management page to identify users and do signature verification for accessing MaxComputer or other cloud products. Access Key Secret must be kept secret.

- Security

MaxCompute multi-tenant data security system mainly includes user authentication, user and authorization management in project space, resource sharing across project space and data protection in project space. For more details on MaxCompute security operation, see [Safety Guide](#).

#### C

- Console

MaxCompute Console is a client tool running under Windows/Linux. It can submit commands to complete project management, DDL, DML and other operations through Console. For tool installation and common parameters, See the [Client](#) for tool installations and common parameters.

#### D

- Data Type

The data type corresponding to all columns in the MaxCompute table. For data types currently supported, see [Basic Concepts>Data types](#).

- DDL

Data Definition Language. Like creating tables, creating views, and so on, MaxCompute Div syntax see [User's Guide>DDL](#).

- DML

Data Manipulation Language. For example, INSERT operations, MaxCompute DML syntax, please see [Insert Operation](#).

#### F

- Fuxi

Fuxi is the module responsible for resource management and task scheduling in the core of Flying Platform. It also provides a basic programming framework for application development. The bottom task scheduling module of MaxCompute is the scheduling module of Fuxi.

#### I

- **Instance (Instance)**

A specific instance of a job that represents a job that actually runs, similar to the concept of a job in hadoop. See [Basic Concepts>Task Instance](#) for details.

## M

- **MapReduce**

MaxCompute a programming model for processing data, usually used for parallel operation of large data sets. You can use the interface provided by MapReduce (Java API) to write MapReduce programs to process data in MaxCompute. The idea of programming is to divide data processing methods into Map (mapping) and Reduce (Protocol).

Before formally executing Map, a partition is required. A slice is a cut of input data into equal-size blocks, each of which acts as a single map. The input of the worker is processed so that multiple Map Worker can work together. Each Map Worker reads in its own data, calculates and processes them, and finally integrates the intermediate results through the Reduce function to get the final results. For details, refer to the [User's guide>MapReduce](#).

## O

- **ODPS**

ODPS is the original name of MaxCompute.

## P

- **Partition (partition)**

Partition partition refers to a table, based on the partition field (one or more combinations) divide the data store. That is, if the table does not have a partition, the data is placed directly under the directory where the table is located. If a table has a partition, each partition corresponds to one of the directories in the table, the data is stored separately in a different partition directory. For more information about partitions, see [Basic Concepts>Partitions](#).

- **Project (Project)**

Project is the basic organizational unit of MaxCompute. It is similar to the concept of database or Scheme in traditional database, and is the main boundary of multi-user isolation and access control. For details, please see [Basic Concepts>Project](#) .

## R

- **Role (role)**

Roles are concepts used in MaxCompute security functions and can be seen as a collection of users with the same privileges. Multiple users can appear at one role at the same time, and a user can also belong to multiple roles. After all roles are authorized, all users under the role have the same permissions. For more information about role management, please see [User's guide>Role management](#) .

- **Resource (resources)**

Resource (Resource) is a unique concept in MaxCompute. If you want to use MaxCompute's custom function (UDF) or MapReduce function, you need to rely on resources to complete it. For details, please see [Basic Concepts>Resource](#).

## S

- SDK

Software Development Kits. Generally, it is a collection of development tools used by software engineers to build application software for specific software packages, software instances, software frameworks, hardware platforms, operating systems, document packages, etc. MaxCompute currently supports [Java SDK](#) and Python SDK.

- Authorization

The project space administrator or project owner gives you permission to perform certain operations on Objects (or objects, such as tables, tasks, resources, etc.) in MaxCompute, including reading, writing, viewing, etc. For the specific operation of authorization, see [User management](#).

- Sandbox

Security restrictions: MaxCompute MapReduce and UDF programs are restricted by Java Sandbox while running in a distributed environment.

## T

- Table (table)

The table is the data storage unit of MaxCompute. See [Basic Concepts>Table](#).

- Tunnel

MaxCompute's data channels provide high concurrency offline data upload and download service. You can use the tunnel service to bulk upload or download data to MaxCompute. Please refer to the tunnel command operation or bulk data channel SDK for relevant commands.

## U

- UDF

Generalized UDF, user defined Function, the Java programming interface provided by MaxCompute develops custom functions, for more information, refer to [User 's guide>UDF](#).

In a narrow sense, UDF refers to user-defined scalar function, whose input and output are one-to-one, that is, reading in a row of data and writing out an output value.

- UDAF

User Defined Aggregation Function, Custom aggregation function, whose input and output are many-to-one, aggregates multiple input records into one output value. It can be combined with the Group By statement in SQL. For details, please see [Java UDF>UDAF](#).

- UDTF

User Defined Table Valued Function, custom table valued function, the userDefinedTablevaluedFunction, the custom table valued function, the function of the function called to the function, the set to the the function to the function, the function to the set to the value to the returned the function, the the function of the function of the the function of the function call to the value of the [Java UDF>UDAF](#).

## 3.2. Project

A project is a basic organizational unit of MaxCompute. Similar to a database or schema in a traditional database system, a project is used to isolate users and control access requests. A project contains multiple objects, such as tables, resources, functions, and instances.

project user isolation access control

A user can have permissions on multiple projects. After being authorized, a user in one project can access objects that belong to another project, such as [tables](#), [resources](#), [functions](#), and [instances](#).

You can run the `use project;` command to enter a project. An example is as follows:

```
-- Enter a project named my_project.  
use my_project;
```


Then, you can manage objects in this project, such as tables, resources, functions, and instances. The `use project;` command is provided by the MaxCompute client. For more information about other commands, see [Common MaxCompute commands](#).

 **Note** A project in MaxCompute is called a workspace in DataWorks.

## 3.3. Table

A table is the data storage unit in MaxCompute. A table is a two-dimensional data structure composed of rows and columns. Each row represents a record, and each column represents a field with the same data type. One record can contain one or more columns. The column name and data type comprise the schema of a table.


The operating objects (input, output) of various computing tasks in MaxCompute are tables. You can create a table, delete a table, and import data into a table. For more information, see [Table-level operations](#).

 **Note** The data management module of DataWorks allows you to create, organize, and modify data lifecycles for MaxCompute tables and grant management permissions.

MaxCompute 2.0 supports internal tables and external tables.

- Data of internal tables is stored in MaxCompute. The columns in external tables can be of any [data types](#) supported by MaxCompute.
- Data of external tables is not stored in MaxCompute. Instead, this data can be stored in [OSS](#) or [OTS](#). MaxCompute only records metadata of the external tables. You can use MaxCompute to process unstructured data of external tables, such as video, audio, or meteorological data.



 **Note** Use of DUAL tables:

- Unlike databases such as Oracle, MaxCompute does not automatically create DUAL tables.
- If you are using DUAL tables for testing, you can run the `CREATE TABLE IF NOT EXISTS DUAL (DUMMY VARCHAR(1));` command to create a table named DUAL with only one field for testing.
- DUAL tables are used in the same way as Oracle. For example, you can run the `select getdate() from dual;` to use a DUAL table.

## 3.4. Partition

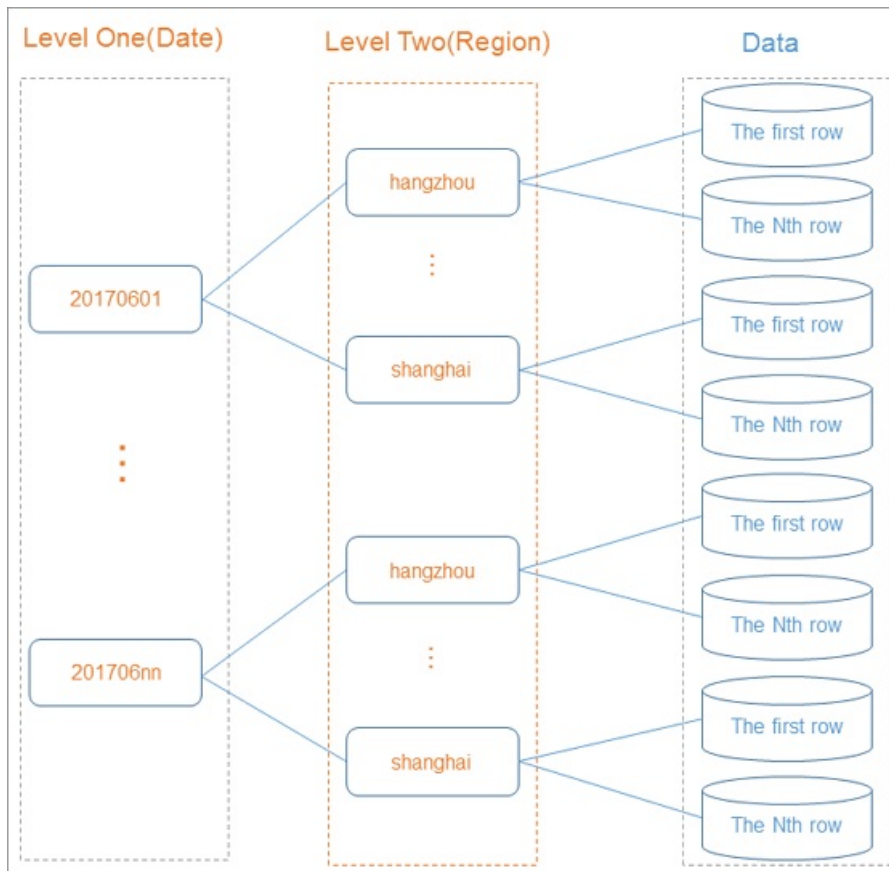
A partitioned table is a table that is divided into multiple segments (partitions) based on one or more partitioning columns in the table. A partition corresponds to an independent folder in a distributed file system. Partitioning groups data of the same category into the same folder.

partitioned table partitioning column query optimization

Partitioning improves query performance. You can specify the name of the partition that you want to access by using the `WHERE` clause. This way, MaxCompute only scans the specified partition. This improves processing efficiency and reduces cost.

Each value in a partitioning column is called a partition or directory in MaxCompute. You can group multiple fields of a table to a single partition to create multi-level partitions. Multi-level partitions are similar to multi-level directories.

If you specify the name of the partition that you want to access, MaxCompute only scans the specified partition. This improves processing efficiency and reduces cost.



## Data types

MaxCompute V2.0 supports the following data types for partitioning columns: TINYINT, SMALLINT, INT, BIGINT, VARCHAR, and STRING.

MaxCompute V1.0 supports only the STRING type. You can specify the data type for a partitioning column as BIGINT. However, only the partition field is of the BIGINT type, and other data in this column is actually processed as a string. After the following statements are executed, the returned result contains only one row because 10 is considered as a string.

```

--- Create a table named parttest.
CREATE TABLE parttest (a bigint) PARTITIONED BY (pt bigint);
--- Insert data into parttest.
INSERT INTO parttest partition(pt) SELECT 1, 2 from dual;
INSERT INTO parttest partition(pt) SELECT 1, 10 from dual;
--- Query the rows whose pt field is greater than or equal to 2.
SELECT * FROM parttest WHERE pt >= '2';
    
```

## Limits

The limits of partitions are as follows:

- A table can contain a maximum of six levels of partitions.

- A table can contain a maximum of 60,000 partitions.
- A maximum of 10,000 partitions can be queried at a time.
- The partitioning column value of a partition whose data type is STRING cannot be Chinese characters.

#### Examples

- Create a partition.

```
-- Create a partitioned table that contains two levels of partitions in which date is the level-1 partition and region is the level-2 partition.
```

```
CREATE TABLE src (key string, value bigint) PARTITIONED BY (pt string,region string);
```

- During the query, use a partitioning column as a filter condition in the WHERE clause.

```
-- Correct usage. When MaxCompute generates a query plan, only the data of the '20170601' partition whose region is 'hangzhou' is used as input data.
```

```
select * from src where pt='20170601'and region='hangzhou';
```

```
-- Incorrect usage. In this example, the effectiveness of the partition filtering cannot be guaranteed. The pt value is of the STRING type. When the value of the STRING type is compared with that of the BIGINT type (20170601), MaxCompute converts both data types to DOUBLE, which causes a loss in precision.
```

```
select * from src where pt = 20170601;
```

Some SQL operations on partitions are less efficient and may result in higher costs, for example, DYNAMIC PARTITION. For more information, see [Insert data in dynamic partition mode \(DYNAMIC PARTITION\)](#).

The syntax of some operations that are used to process partitioned and non-partitioned tables is different in MaxCompute. For more information, see [Table operations](#) and [INSERT OVERWRITE and INSERT INTO](#).

## 3.5. Lifecycle

This article introduces the concept of life cycle of MaxCompute tables in detail.

The lifecycle of a MaxCompute table or partition is measured from the last update time. If the table or partition remains unchanged after a specified time, MaxCompute automatically recycles it. The specified time indicates the lifecycle.

- Lifecycle units: days, positive integers only.
- When a lifecycle is specified for a non-partition table, the lifecycle is counted from the last time the table data was modified (LastDataModifiedTime). If table data has not been changed, MaxCompute recycles the table automatically without manual operation (similar to the drop table operation).
- When a lifecycle is specified for a partition table, we will decide whether a partition should be recycled according to each partition's LastDataModifiedTime. Unlike non-partition tables, a partition table will not be deleted even if its last partition has been recycled.

**Note**

Lifecycle scanning is started at a scheduled time every day, and entire partitions are scanned. If the partition remains unchanged after its lifecycle, MaxCompute automatically recycles it.

When a lifecycle is specified for a partition table, MaxCompute determines whether to recycle the partition based on its LastDataModifiedTime. Unlike non-partition tables, a partition table cannot be deleted even when all its partitions have been recycled.

- You can set the lifecycle of tables, but not of partitions. The lifecycle of a table can be specified during table creation.
- If no lifecycle is specified, the table, or partition cannot be automatically recycled by MaxCompute.

For more information on specifying or modifying lifecycles during table creation, and modifying a table's LastDataModifiedTime, see [Table operations](#) .

## 3.6. Resource

This paper introduces the concept of MaxCompute Resource, which can provide resource dependency for MaxCompute specific operations.

### The concept of resources

Resources is a concept that is unique to MaxCompute. To accomplish tasks using user-defined functions (for more information, see [Overview](#)), or [MapReduce](#), you must use resources.

- **SQL UDF:** After writing a UDF, you must compile it as a Jar package and upload the package to MaxCompute as a resource. Then, when you run this UDF, MaxCompute automatically downloads its corresponding JAR package to obtain the written code. The JAR package is one type of MaxCompute resource.
- **MapReduce:** After writing a MapReduce program, you must compile it as a Jar package and upload the package to MaxCompute as a resource. Then, when running a MapReduce job, the MapReduce framework automatically downloads the corresponding JAR package and obtain the written code. You can upload text files and MaxCompute tables to MaxCompute as different types of resources. Then, you can read or use these resources when running UDF or MapReduce.


### Resource type

MaxCompute provides interfaces for you to read and use resources. For more information, see [Resource samples](#) and [MapReduce](#) .

**Note** For more information about the resource reading capabilities of user-defined functions ([Overview](#)) or [MapReduce](#) , see [Limits](#).

The max size that MaxCompute support for single resource is 500MB. Types of MaxCompute resources include:

- File
- Table: tables in MaxCompute

 **Note** Currently, only BIGINT, DOUBLE, STRING, DATETIME, and BOOLEAN fields are supported in tables referenced by MapReduce.

- Jar type, which is compiled Java JAR packages
- Archive type, which is the compression type, and is determined by the resource name suffix. Supported compression types include: .zip/.tgz/.tar.gz/.tar/jar

For more information about resources, see [Resource operations](#), [Resource operations](#), [Resource operations](#) and [Resource operations](#).


## 3.7. Function

This article gives you an overview of the functions that will be used in MaxCompute to provide you with computational functions.

MaxCompute provides SQL computing capabilities. In MaxCompute SQL, you can use the [Mathematical functions](#) to perform common computing and counting tasks. If the built-in functions do not meet your requirements, you can use the Java programming interface provided by MaxCompute to develop user-defined functions (UDFs).

[Overview](#) can be divided into scalar valued functions, user-defined aggregate functions (UDAFs), and user-defined tables functions (UDTFs).

After writing the code for a UDF, you must compile the code into a JAR package and upload this package to MaxCompute. Then, you can register the UDF in MaxCompute.

 **Note** UDFs are used in the same way as built-in functions, in that you specify the UDF name and input relevant parameters in SQL.

For more information, see [Functions operations](#).


## 3.8. Task

A task is the basic computing unit of MaxCompute. Computing tasks such as those involving SQL, DML and MapReduce functions are completed using tasks.

For most user-submitted tasks, such as [INSERT OVERWRITE](#) and [INSERT INTO](#), [MapReduce](#), etc. MaxCompute first analyzes them and then generates a task execution plan. The execution plan is composed of multiple execution stages that are dependent on each other. An execution plan consists of multiple stages with dependency links.

Currently, an execution plan can be logically viewed as a directed graph whose vertices represent the stages and whose edges represent the dependency links of the stages. MaxCompute executes each stage according to the dependencies in the graph (execution plan). A single stage comprises multiple threads, also known as workers. These workers complete the computing in this stage. Different workers in the same stage have exactly the same execution logic, but they process different data. Computational tasks are executed directly in MaxCompute instances, for example, [Instance operations](#) and [Instance operations](#).

For MaxCompute tasks that are not computational tasks, such as DDL statement in SQL, these tasks can only read and modify the metadata information in MaxCompute. This means that no execution plan can be analyzed and generated from the task.

 **Note** Not all the requests are converted into tasks in MaxCompute, for example, the operations of **Project**, **Resource**, **Function** and **Instance** can be completed without MaxCompute tasks.

## 3.9. Instance

This article introduces you to the MaxCompute task instance and its status.

In MaxCompute, most **Task** are initiated in MaxCompute instances. MaxCompute instances can be in one of two phases: Running and Terminated.

The status of the running phase is 'Running', while the status of the Terminated phase can be 'Success', 'Failed' or 'Canceled'. You can query or change the status using the instance ID assigned by MaxCompute. For example:

```
status <instance_id>; --View the status of a certain instance.
kill <instance_id>; --Stop an instance and set its status as 'Canceled'.
wait <instance_id>; --View the running logs of a certain instance.
```

## 3.10. ACID semantics

This topic describes the ACID semantics of MaxCompute for concurrent jobs.

ACID atomicity consistency isolation durability

### Terms

- **Operation:** a single job submitted in MaxCompute
- **Data object:** an object that contains data, such as a partitioned or non-partitioned table
- **INTO job:** an SQL job that contains the INTO keyword, such as `INSERT INTO` or `DYNAMIC INSERT INTO`
- **OVERWRITE job:** an SQL job that contains the OVERWRITE keyword, such as `INSERT OVERWRITE` or `DYNAMIC INSERT OVERWRITE`
- **Data upload by using Tunnel:** an INTO or OVERWRITE job

### Descriptions of ACID semantics

- **Atomicity:** An operation is either fully completed or not executed at all.
- **Consistency:** The integrity of data objects is maintained while operations are executed.
- **Isolation:** An operation is completed independent of other concurrent operations.
- **Durability:** After an operation is completed, modified data is permanently valid and not lost even if a system failure occurs.

### Scenarios of ACID semantics for MaxCompute

- **Atomicity**
  - When multiple jobs conflict with each other, MaxCompute always ensures that only one job succeeds.

- The atomicity of the CREATE, OVERWRITE, and DROP operations on a single table or partition is guaranteed.
- Atomicity cannot be guaranteed for cross-table operations such as MULTI-INSERT.
- In extreme cases, the following operations may not be atomic:
  - The `DYNAMIC INSERT OVERWRITE` operation is performed on more than 10,000 partitions.
  - An INTO operation fails because data cleansing fails during a transaction rollback. However, this does not cause raw data loss.
- Consistency
  - OVERWRITE jobs ensure consistency.
  - If an INTO job fails due to a conflict, data from the failed job may remain.
- Isolation
  - Non-INTO operations ensure that read operations are committed.
  - In INTO operations, some read operations may not be committed.
- Durability
  - MaxCompute ensures data durability.

## 4.Limits

This topic provides limits of MaxCompute product modules.

The limits include:

- [Limits of SQL](#)
- [Limits of data upload and download](#)
- [Limits of using common commands](#)
- [Limits](#)
- [Limits of MaxCompute Graph](#)
- [Limits of security configuration](#)
- [Limits of MaxCompute Lightning](#)



## 5. MaxCompute features in different regions

MaxCompute is a big data computing service that provides multiple built-in computing models to meet a wide range of data analytics requirements. This topic lists the enabling status of these computing models in different regions.

Region	SQL	MapReduce	Spark	Mars
China (Beijing)	Enabled	Enabled	Enabled	Enabled
China (Hangzhou)	Enabled	Enabled	Enabled	Enabled
China (Shanghai)	Enabled	Enabled	Enabled	Enabled
China (Shenzhen)	Enabled	Enabled	Enabled	Enabled
China (Chengdu)	Enabled	Enabled	Enabled	Enabled
China (Zhangjiakou-Beijing Winter Olympics)	Enabled	Enabled	Enabled	Enabled
China (Hong Kong)	Enabled	Enabled	Enabled	Enabled
Singapore (Singapore)	Enabled	Enabled	Enabled	Disabled
Malaysia (Kuala Lumpur)	Enabled	Enabled	Enabled	Disabled
Indonesia (Jakarta)	Enabled	Enabled	Enabled	Disabled
Australia (Sydney)	Enabled	Enabled	Enabled	Disabled
Japan (Tokyo)	Enabled	Enabled	Enabled	Disabled
US (Silicon Valley)	Enabled	Enabled	Enabled	Disabled
US (Virginia)	Enabled	Enabled	Enabled	
Germany (Frankfurt)	Enabled	Enabled	Enabled	Disabled
India (Mumbai)	Enabled	Enabled	Enabled	Disabled

---

Region	SQL	MapReduce	Spark	Mars
UK (London)	Enabled	Enabled	Enabled	Disabled

# 6.Storage performance

The data storage format of MaxCompute has been updated to Alibaba Optimized Row Columnar (AliORC) since February 2020. To help you better understand the data performance of MaxCompute, this topic compares AliORC with Apache Optimized Row Columnar (ORC) and Apache Parquet based on TPC Benchmark DS (TPC-DS) tests.

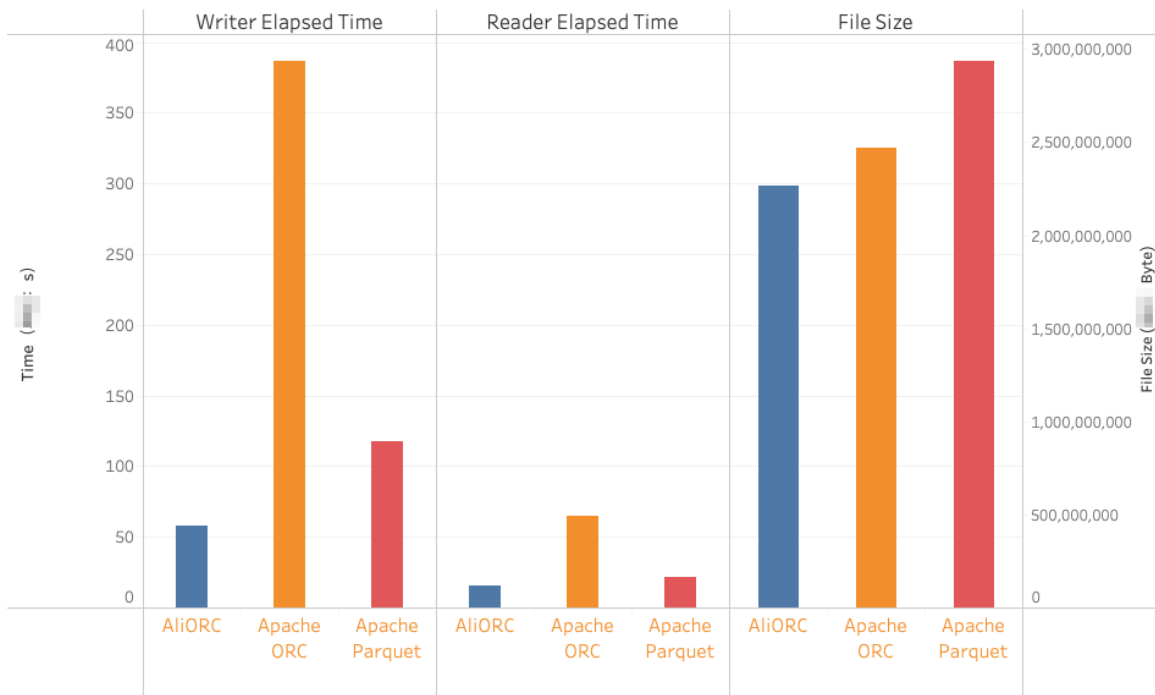
AliORC MaxCompute data storage performance

## Test results

- The following table describes the comparison between the test results of AliORC and Apache ORC, and between the test results of AliORC and Apache Parquet. The test results are compared based on the dataset that contains 24 TPC-DS test tables.

Item	File size	Writer elapsed time	Reader elapsed time
AliORC compared with Apache ORC	Drops by more than 8%.	Drops by more than 85%.	Drops by more than 76%.
AliORC compared with Apache Parquet	Drops by more than 22%.	Drops by more than 50%.	Drops by more than 28%.

The following figure shows the test results of the dataset.



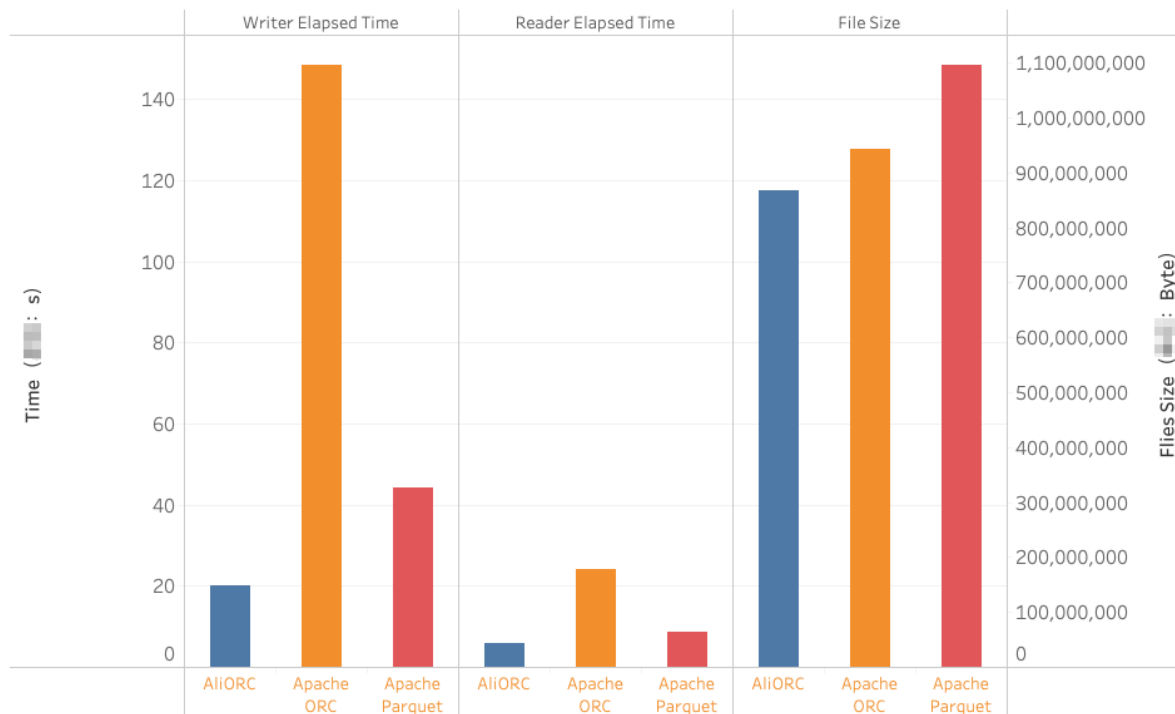
### Parameter description:

- File size: the data storage size of all tables combined. Unit: bytes.
- Writer elapsed time: the time it takes to import CSV data of TPC-DS to AliORC, Apache ORC, or Apache Parquet. Unit: seconds.

- Reader elapsed time: the time it takes for AliORC, Apache ORC, or Apache Parquet to complete a data scan. Unit: seconds.
- The following table describes the comparison results based on the store\_sales table, which is the largest table among the test tables.

Item	File size	Writer elapsed time	Reader elapsed time
AliORC compared with Apache ORC	Drops by more than 7%.	Drops by more than 86%.	Drops by more than 74%.
AliORC compared with Apache Parquet	Drops by more than 20%.	Drops by more than 54%.	Drops by more than 30%.

The following figure shows the test results of the store\_sales table.



### Test environments

- Apache Parquet version: Apache Arrow C++ V0.16.0
- Apache ORC version: C++ V1.6.2
- Dataset: TPC-DS 10G (SF=10)

### Dataset

TPC-DS is a decision support benchmark that uses multi-dimensional data models, such as star and snowflake data models. The benchmark contains 7 fact tables and 17 dimension tables, with an average of 18 columns per table. The tables contain skewed data and values to simulate a real scenario. TPC-DS provides the best test set to measure different versions of Hadoop and SQL on Hadoop.

The following list shows the 24 tables of the TPC-DS dataset used in this test:

store\_sales  
catalog\_sales  
inventory  
web\_sales  
store\_returns  
catalog\_returns  
web\_returns  
customer\_demographics  
customer  
item  
customer\_address  
date\_dim  
time\_dim  
catalog\_page  
household\_demographics  
promotion  
store  
web\_page  
web\_site  
call\_center  
reason  
warehouse  
ship\_mode  
income\_band