阿里云
MaxCompute

产品简介

文档版本：20190617

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的”现状“、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含”阿里云”、Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **禁止:** 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **警告:** 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 **说明:** 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 **确定**。
`courier` 字体	命令。	执行 `cd /d C:/windows` 命令，进入Windows系统文件夹。
##	表示参数、变量。	`bae log list --instanceid` *Instance_ID*
[]或者[a|b]	表示可选项，至多选择一个。	`ipconfig` *[-all|-t]*
{}或者{a|b}	表示必选项，至多选择一个。	`swich` *{stand | slave}*

目录

1 什么是MaxCompute

大数据计算服务（MaxCompute，原名ODPS）是一种快速、完全托管的EB级数据仓库解决方案。

当今社会数据收集手段不断丰富，行业数据大量积累，数据规模已增长到了传统软件行业无法承载的海量数据（百TB、PB、EB）级别。MaxCompute致力于批量结构化数据的存储和计算，提供海量数据仓库的解决方案及分析建模服务。

由于单台服务器的处理能力有限，海量数据的分析需要分布式计算模型。分布式的计算模型对数据分析人员要求较高且不易维护：数据分析人员不仅需要了解业务需求，同时还需要熟悉底层分布式计算模型。MaxCompute为您提供完善的数据导入方案以及多种经典的分布式计算模型，您可以不必关心分布式计算和维护细节，便可轻松完成大数据分析。

目前，MaxCompute服务已覆盖全球16个国家和地区，客户遍及金融、互联网、生物医疗、能源、交通，传媒等行业，为全球用户提供海量数据存储和计算服务。MaxCompute的多个客户案例荣获“2017大数据优秀产品和应用解决方案案例”奖。此外，MaxCompute，DataWorks以及AnalyticDB代表阿里云入选了Forrester Wave™ Q4 2018云数据仓库报告。

> 说明:
> MaxCompute已经在阿里巴巴集团内部得到大规模应用，例如大型互联网企业的数据仓库和BI分析、网站的日志分析、电子商务网站的交易分析、用户特征和兴趣挖掘等。

DataWorks和MaxCompute关系紧密：DataWorks为MaxCompute提供一站式的数据同步、业务流程设计、数据开发、管理和运维功能，详情请参见DataWorks产品概述。

MaxCompute视频简介

MaxCompute学习路径

您可以通过*MaxCompute*学习路径快速了解MaxCompute的相关概念、基础操作、进阶操作等。

产品优势

- 大规模计算存储

  MaxCompute适用于100GB以上规模的存储及计算需求，最大可达EB级别。

- 多种计算模型

  MaxCompute支持SQL、MapReduce、UDF（Java/Python）、Graph，基于DAG的处理，交互式，内存计算，机器学习等计算类型及MPI迭代类算法。简化了企业大数据平台的应用架构。

- 强数据安全

  MaxCompute已稳定支撑阿里全部数据仓库业务9年以上，提供多层沙箱防护、细粒度权限管理及监控。

- 低成本

  与企业自建专有云相比，MaxCompute的计算存储更高效，可以降低30%~50%的采购成本。

- 免运维

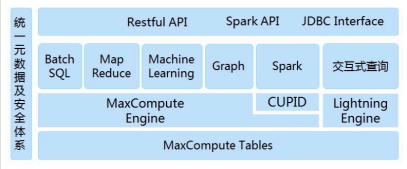
  基于MaxCompute的Serverless无服务器的设计思路，用户只需关心作业和数据，而无需关心底层分布式架构及运维。

- 极致弹性扩展

  MaxCompute提供后付费模式下的作业级别的资源管理。用户无需受困于资源扩展难题，系统会自动扩展计算、存储、网络等资源，最大程度地节省成本。

系统架构

MaxCompute以数据为中心，内建多种计算模型和服务接口，满足广泛的数据分析需求。一切服务“开通”即用，更好地赋能数据业务。



功能概述

- 数据通道

  - 批量历史数据通道

    *TUNNEL*是MaxCompute为您提供的数据传输服务，提供高并发的离线数据上传下载服务。支持每天TB/PB级别的数据导入导出，特别适合于全量数据或历史数据的批量导

入。Tunnel为您提供Java编程接口，并且在MaxCompute的客户端工具中，提供对应的命令实现本地文件与服务数据的互通。

- 实时增量数据通道

  针对实时数据上传的场景，MaxCompute提供了延迟低、使用方便的DataHub服务，特别适用于增量数据的导入。DataHub还支持多种数据传输插件，例如Logstash、Flume、Fluentd、Sqoop等，同时支持日志服务Log Service中的投递日志到MaxCompute，进而使用DataWorks进行日志分析和挖掘。

· 计算及分析任务

MaxCompute支持多种计算模型，详情如下：

- SQL：MaxCompute以表的形式存储数据，支持多种数据类型，并对外提供SQL查询功能。您可以将MaxCompute作为传统的数据库软件操作，但其却能处理TB、PB级别的海量数据。

  说明：

  ■ MaxCompute SQL不支持事务、索引，也不支持Update或Delete操作。

  ■ MaxCompute的SQL语法与Oracle、MySQL有一定差别，您无法将其他数据库中的SQL语句无缝迁移至MaxCompute中。详情请参见与其他SQL语法的差异。

  ■ MaxCompute主要用于100GB以上规模的数据计算，因此MaxCompute SQL最快支持在分钟或秒钟级别完成查询返回结果，但无法在毫秒级别返回结果。

  ■ MaxCompute SQL的优点是学习成本低，您不需要了解复杂的分布式计算概念。如果您具备数据库操作经验，便可快速熟悉MaxCompute SQL的使用。

- UDF：即用户自定义函数。

  MaxCompute提供了很多内建函数来满足您的计算需求，同时您还可以通过创建自定义函数来满足不同的计算需求。

- MapReduce：MaxCompute MapReduce是MaxCompute提供的Java MapReduce编程模型，它可以简化开发流程，更为高效。您若使用MaxCompute MapReduce，需要对分布式计算概念有基本了解，并有相对应的编程经验。MaxCompute MapReduce为您提供Java编程接口。

- Graph：MaxCompute提供的Graph功能是一套面向迭代的图计算处理框架。图计算作业使用图进行建模，图由点（Vertex）和边（Edge）组成，点和边包含权值（Value）。通过迭代对图进行编辑、演化，最终求解出结果，典型应用：PageRank、单源最短距离算法、K-均值聚类算法等。

- SDK

  SDK是MaxCompute提供给开发者的工具包，当前支持*Java SDK*及*Python SDK*。

- 安全

  MaxCompute提供了功能强大的安全服务，为您的数据安全提供保护，详情请参见**安全指南**。

后续步骤

现在，您已经学习了MaxCompute的产品优势、功能特性。您可以继续了解MaxCompute的相关收费情况，详情请参见**产品定价**。

2 与其它阿里云服务的集成使用

MaxCompute（原ODPS）是一种大数据计算服务，能提供快速、完全托管免运维的GB到EB级云数据仓库解决方案，已经与阿里云部分产品集成，快速实现多种业务场景。

MaxCompute与DataWorks

*DataWorks*是基于MaxCompute计算和存储，提供工作流可视化开发、调度运维托管的一站式海量数据离线加工分析平台。在数加（一站式大数据平台）中，DataWorks控制台即为MaxCompute控制台。

MaxCompute和DataWorks一起向用户提供完善的ETL和数仓管理能力，以及SQL、MR、Graph等多种经典的分布式计算模型，能够更快速地解决用户海量数据计算问题，有效降低企业成本，保障数据安全。更多使用说明请参见DataWorks产品概述。

说明:

您可以将DataWorks理解成MaxCompute的一种Web客户端。

MaxCompute与数据集成

MaxCompute可以通过数据集成加载不同数据源数据，同样也可以通过数据集成把MaxCompute的数据导出到各种业务数据库。

数据集成已经集成到DataWorks作为数据同步任务进行配置、运行。您可直接在DataWorks上配置*MaxCompute*数据源，再配置读取*MaxCompute*表或者写入*MaxCompute*表任务，整个过程只需在一个平台上进行操作。

MaxCompute与机器学习PAI

机器学习*PAI*是基于MaxCompute的一款机器学习算法平台。它实现了数据无需搬迁，便可进行从数据处理、模型训练、服务部署到预测的一站式机器学习。创建好MaxCompute项目，开通好机器学习，即可通过机器学习平台的算法组件对MaxCompute数据进行模型训练等操作。详情请参见机器学习*PAI*操作文档。

MaxCompute与QuickBI

数据在MaxCompute进行加工处理后，将Project添加为*QuickBI*数据源，即可在QuickBI页面对MaxCompute表数据进行报表制作，实现数据可视化分析。

MaxCompute与AnalyticDB

AnalyticDB是海量数据实时高并发在线分析（Realtime OLAP）的云计算服务，与MaxCompute结合实现大数据驱动业务系统的场景。通过MaxCompute离线计算挖掘，产出高质量数据后，导入分析型数据库，供业务系统调用分析。

将MaxCompute数据导入到AnalyticDB，有以下两种方式：

- 通过DMS for AnalyticDB的导入导出功能进行配置。
- 通过DataWorks配置数据同步任务，读MaxCompute和写AnalyticDB。

MaxCompute与推荐引擎

推荐引擎是在阿里云计算环境下建立的一套推荐服务框架，推荐服务通常由三部分组成：日志采集、推荐计算和产品对接，而推荐计算的离线计算输入和输出都是MaxCompute（原ODPS）表。

在推荐引擎控制台的资源管理页面，通过添加云计算资源的方式，将MaxCompute项目添加为推荐引擎的计算资源。

MaxCompute与表格存储

表格存储#Table Store#是构建在阿里云飞天分布式系统之上的分布式NoSQL数据存储服务，MaxCompute2.0支持直接通过外部表方式访问表格存储中的表数据并进行处理，详情请参见访问OTS非结构化数据。

MaxCompute与OSS

对象存储OSS 是海量、安全、低成本、高可靠的云存储服务，MaxCompute2.0支持直接通过外部表方式访问表格存储中的表数据并进行处理，详情请参见访问OSS非结构化数据。

MaxCompute与OpenSearch

阿里云开放搜索OpenSearch是一款自主研发的大规模分布式搜索引擎平台。数据通过MaxCompute进行计算处理后，可以在OpenSearch平台上通过添加数据源的方式将MaxCompute数据接入。

MaxCompute与移动数据分析

移动数据分析#Mobile Analytics#是阿里云推出的一款移动APP数据统计分析产品，为开发者提供一站式数据化运营服务。当移动数据分析自带的基础的分析报表不能满足APP开发者的个性化需求时，APP开发者可以将数据一键同步至MaxCompute，结合自己的业务需求来进一步加工、分析自己的数据。

MaxCompute与日志服务

日志服务能快速完成数据采集、消费、投递以及查询分析等功能。日志数据采集后，需要更多的个性化分析、挖掘，您可以在日志服务上投递日志到MaxCompute，通过MaxCompute对日志数据进行个性化、深层次的数据分析、挖掘。

MaxCompute与RAM

RAM是阿里云为客户提供的用户身份管理与资源访问控制服务。MaxCompute与RAM的集成使用主要有两个场景：

- 通过DataWorks使用MaxCompute时，子账户的身份管理

  主帐号开通并创建项目后，若需要通过DataWorks使用MaxCompute且多个账户协同开发，必须由主帐号到RAM服务中创建子账户，将RAM子账户添加为项目成员从而进行协同开发，详情请参见准备RAM子账号、添加工作空间成员和角色。

  > 说明:
  > 此时RAM只起到用户身份管理功能，相关的权限管理不在RAM上控制。

- MaxCompute处理非结构化数据时，通过RAM对非结构化数据进行授权

  目前MaxCompute支持直接处理非结构化数据（包含OSS和Table Store），前提条件之一就是需要在RAM中授予MaxCompute访问OSS或Table Store的权限，详情请参见访问OSS非结构化数据、访问OTS非结构化数据。

阿里云其它产品支持字符集的情况

产品名称	支持的字符集
表格存储	UTF-8
机器学习PAI	UTF-8
OSS	UTF-8
QuickBI	UTF-8
DataWorks	在DataStudio中进行数据上传，支持UTF-8、GBK、CP936、ISO-8859，但到DataWorks中会统一为UTF-8。数据下载支持UTF-8、GBK。

3 使用须知

本文根据您的角色为您推荐不同的文档阅读顺序。

如果您是MaxCompute初学者

如果您是初学者，建议您从如下模块开始读起：

- 产品简介： MaxCompute产品的总体介绍以及包含的主要功能。通过阅读该章节，您会对MaxCompute有一个总体的认识。
- 快速开始：通过示例，指导您如何进行申请账号、安装客户端、创建表、授权、导入导出数据、运行SQL任务、运行UDF/MapReduce程序等操作。
- 基本介绍：MaxCompute的基本概念及常用命令介绍。您可以进一步熟悉如何操作MaxCompute。
- 工具：在分析数据之前，您需要掌握MaxCompute常用工具的下载、配置以及使用方法。

  我们为您提供工具，您可以通过此工具对MaxCompute进行操作。
- 配置*Endpoint*：MaxCompute Region的开通情况和连接方式，解答您在与其他云产品（ECS、TableStore、OSS）互访场景中遇到的网络连通性和下载数据收费等问题。

建议您熟悉以上的模块后，再有针对性地对其他模块进行深入学习。

如果您是数据分析师

如果您是数据分析师，建议您熟读SQL模块的内容。

- *SQL*：您可以查询并分析存储在MaxCompute上的大规模数据。包含的主要功能如下：
  - 支持DDL语句，您可以通过Create、Drop和Alter对表和分区进行管理。
  - 您可以通过Select选择表中的某几条记录，通过Where语句查看满足条件的记录，实现过滤功能。
  - 您可以通过等值连接Join实现两张表的关联。
  - 您可以通过对某些列Group By，实现聚合操作。
  - 您可以通过Insert Overwrite/Into把结果记录插入到另一张表中。
  - 您可以通过内置函数和自定义函数（UDF）来实现一系列的计算。

如果您拥有一定开发经验

如果您拥有一定的开发经验，了解分布式概念，并且某些数据分析可能无法用SQL来实现，此时推荐您学习MaxCompute更高级的功能模块。如下所示：

- *MapReduce*：MaxCompute提供的Java MapReduce编程模型。您可以使用MapReduce提供的接口（Java API）编写MapReduce程序，处理MaxCompute中的数据。
- *Graph*：一套面向迭代的图计算处理框架。使用图进行建模，图由点（Vertex）和边（Edge）组成，点和边包含权值（Value）。通过迭代对图进行编辑、演化，最终得出结果。
- *Eclipse Plugin*：方便您使用MapReduce、UDF以及Graph的Java SDK进行开发工作。
- *Tunnel*：您可以使用Tunnel服务向MaxCompute批量上传离线数据或者从MaxCompute下载离线数据。
- SDK：
  - *Java SDK*：向开发者提供Java接口。
  - *Python SDK*：向开发者提供Python接口。

> **说明：**
> 目前*MapReduce*以及*Graph*功能仍处于公测中，若您想使用这部分功能，可以通过工单系统提交申请。申请时请指明您的项目空间名称，我们会在7个工作日内处理。

如果您是项目Owner或者管理员

如果您是一个项目空间的Owner，需要创建和使用项目。如果您是管理员，需要在项目管理、安全管理（人员以及权限管理）、费用管理等方面应用本产品。您需要熟知以下模块：

- 项目管理

  项目空间（Project）是MaxCompute的基本组织单元，它类似于传统数据库的Database或Schema的概念，是进行多用户隔离和访问控制的主要边界。一个用户可以同时拥有多个项目空间的权限，通过安全授权，可以在一个项目空间中访问另一个项目空间中的对象，例

如表（Table）、资源（Resource）、函数（Function）和实例（Instance）。使用MaxCompute，实际是操作Project的各种对象。

- 创建项目前期工作

  ■ 资源预算

  MaxCompute收费资源主要是3大模块资源：存储、计算、公网下载流量。

  1. 存储资源：按量付费，阶梯计费，可以根据数据量套公式进行预算，当然数据不是一天就全部放在MaxCompute上。同时每天不同时刻都有可能有进有出，所以这个预算结果不是绝对值。
  2. 计算资源：计算资源（目前包括SQL、MR、Spark、Lightning任务计算）分按量付费和预付费。开始使用会不容易评估计算资源使用量，建议先用后付费形式，测试一段时间后看使用量再决定是否使用预付费。
  3. 公网下载流量：按量付费，只有通过公网下载才会收费。

  详细的计量计费说明请参见计费方式。

  ■ 账号准备并开通服务

  创建MaxCompute项目前，必须先开通MaxCompute服务。而目前只能由阿里云账号作为主账号开通，同时这个账号将作为扣费的主体。确定好账号后，开通MaxCompute服务时，需要您根据前面的预算结论选择后付费/预付费资源进行开通。

- 创建Project

  通过MaxCompute控制台（也是DataWorks控制台）创建项目空间进行Project创建。
  创建项目空间时，从业务角度考虑是需要标准模式还是简单模式项目；从安全角度考虑MaxCompute访问身份是用个人账号还是计算引擎指定账号，可以参考MaxCompute数据安全管理指南进行规划。

  创建Project具体操作请参考文档创建项目。

- 项目成员管理

  成员管理主要考虑职责和安全问题，如果通过DataWorks使用MaxCompute，那么还要考虑两个之间相关的权限关联，同样参考MaxCompute数据安全管理指南。

- 子账号管理

  MaxCompute Project支持云账号和RAM账号两种账号体系。对于RAM账号，仅识别账号体系不识别RAM权限体系，即可将主账号的任意RAM子账号加入MaxCompute的某一个项

目中，但MaxCompute在对该RAM子账号做权限验证时，并不会考虑RAM中的权限定义。关于RAM子账号详细介绍请参见准备RAM子账号。

通过DataWorks使用MaxCompute，DataWorks的工作空间，仅支持添加主账号（云账号）下的RAM子账号为成员。因此，需要主账号通过RAM系统创建子账号，并对子账号进行维护管理。

 说明:

- MaxCompute不考虑RAM中的权限定义，因此子账号的权限还需通过MaxCompute命令行或者DataWorks的相关功能实现，具体可参考MaxCompute数据安全管理指南。
- 建议一个子账号对应一个项目成员，禁止多个成员共用一个子账号。
- 离职或转岗的成员，需及时清理对应子账号。

 说明:

若子账号在DataWorks中被加为项目成员，请先清除项目成员再到RAM系统中删除子账号。

- 调度资源管理

■ 关于调度资源

这里介绍的调度资源是指DataWorks上的调度资源，调度资源用于执行或分发调度系统下发的任务，DataWorks的调度资源分为以下两种模式。具体可参考文档管理控制台 > 调度资源列表。

1. 默认调度资源。指DataWorks默认的调度资源，公共的资源池。当DataWorks节点并发量很高，调度资源紧张时会进入等待调度状态。直到占用到资源，节点才开始执行下发任务。
2. 自定义调度资源。指将您自助购买的ECS配置为可以执行分发任务的调度服务器。主账号可以新建自定义调度资源，调度资源包括若干台物理机或ECS，主要用于执行数据同步任务。主账号管理员可在调度资源页面中新建、配置调度资源以完成专用调度资源的创建，也可对已有的调度资源进行编辑。

■ 自定义调度资源

租户管理员可以新建专用的调度资源，以便后续可将专用的调度资源分配给某些项目空间来执行特定的任务。指定的任务可以是数据同步任务、SHELL脚本、MaxCompute SQL

脚本，也可以是MaxCompute MR和算法实验室等任务。而在未指定专用调度资源的情况下，项目空间内的所有任务都将默认使用系统托管集群中的资源。

- 其他项目管理操作

  项目空间的一些设置。在项目开发过程中，有些项目空间的设置操作需要Owner来进行，如设置项目空间是否允许全表扫描、设置项目空间默认打开2.0新类型等。可参见文档常用命令 > 项目空间操作。

· 安全管理

安全管理包括人员管理、角色管理、权限管理等。通过DataWorks使用MaxCompute时，由于DataWorks和MaxCompute有各种权限模型，因此需要理清楚两个产品之间的权限关系，再从业务需求出发进行权限管理。安全管理过程中，您可以通过阅读安全模型章节，了解如何进行给用户授权、跨项目空间的资源共享、设置项目空间的数据保护功能、policy授权等操作：

1. 权限模型：*MaxCompute*和*DataWorks*权限关系、用户与权限管理以及相关安全功能的启用等安全管理基础综合介绍。
2. 安全功能详解：详细的安全相关功能命令等介绍。
3. 安全管理案例：真实客户业务需求转化的安全案例。

· 费用管理

前面提到的创建项目前期工作中的预算是在为使用之前进行成本预估，但是限于目前MaxCompute的计费方式，很多业务无法更准确的进行成本预估。因此在业务开发整个过程中避免不了需要进行费用管理，主要需要关注：

- 产品的计费定价，可参考文档产品定价 > 计量计费说明。
- 产品欠费预警与停机策略，可参考文档产品定价 > 欠费预警与停机策略。
- 预付费升级降配操作说明可参考文档产品定价 > 升级和降配。
- 预付费续费说明产品定价 > 续费管理。
- 当使用后付费一段时间后想转预付费，或者预付费转后付费，产品都支持。MaxCompute的收费模式具体操作可参考文档产品定价 > 计费方式转换。
- 查看MaxCompute账单信息请参考文档账单查看与*MaxCompute*账单分析最佳实践。

4 基本概念

4.1 MaxCompute术语表

本文列举了MaxCompute常见的概念和术语。详细的描述请参见文中的链接。

A

- AccessKey

  AccessKey（简称AK，包括AccessKey ID和AccessKey Secret），是访问阿里云API的密钥，在阿里云官网注册云账号后，可在accesskeys管理页面生成，用于标识用户，为访问MaxCompute或者其他云产品做签名验证。AccessKey Secret必须保密。

- 安全

  MaxCompute多租户数据安全体系，主要包括用户认证、项目空间的用户与授权管理、跨项目空间的资源分享以及项目空间的数据保护。关于MaxCompute安全操作的更多详情请参见 安全指南。

C

- Console

  MaxCompute Console是运行在Window/Linux下的客户端工具，通过Console可以提交命令完成Project管理、DDL、DML等操作。对应的工具安装和常用参数请参见客户端。

D

- Data Type

  MaxCompute表中所有列对应的数据类型。目前支持的数据类型详情请参见#unique_63。

- DDL

  Data Definition Language（数据定义语言）。例如创建表、创建视图等操作，MaxCompute DDL语法请参见DDL语句。

- DML

  Data Manipulation Language（数据操作语言）。例如INSERT操作，MaxCompute DML语法请参见INSERT语句。

F

- fuxi

  伏羲（fuxi）是飞天平台内核中负责资源管理和任务调度的模块，同时也为应用开发提供了一套编程基础框架。MaxCompute底层任务调度模块即fuxi的调度模块。

I

- Instance（实例）

  作业的一个具体实例，表示实际运行的Job，类同Hadoop中Job的概念。详情请参见任务实例。

M

- MapReduce

  MaxCompute处理数据的一种编程模型，通常用于大规模数据集的并行运算。您可以使用MapReduce提供的接口（Java API）编写MapReduce程序，来处理MaxCompute中的数据。编程思想是将数据的处理方式分为Map（映射）和Reduce（规约）。

  在正式执行Map前，需要将输入的数据进行分片。所谓分片，就是将输入数据切分为大小相等的数据块，每一块作为单个Map Worker的输入被处理，以便于多个Map Worker同时工作。每个Map Worker在读入各自的数据后，进行计算处理，最终通过Reduce函数整合中间结果，从而得到最终计算结果。详情请参见*MapReduce*。

O

- ODPS

  ODPS是MaxCompute的原名。

P

- Partition（分区）

  分区Partition是指一张表下，根据分区字段（一个或多个组合）对数据存储进行划分。也就是说，如果表没有分区，数据是直接放在表所在的目录下。如果表有分区，每个分区对应表下的一个目录，数据是分别存储在不同的分区目录下。关于分区的更多介绍请参见分区。

- Project（项目/项目空间）

  项目空间（Project）是MaxCompute的基本组织单元，它类似于传统数据库的Database或Scheme的概念，是进行多用户隔离和访问控制的主要边界。详情请参见项目空间。

R

- Role（角色）

  角色是MaxCompute安全功能里使用的概念，可以看成是拥有相同权限的用户的集合。多个用户可以同时存在于一个角色下，一个用户也可以隶属于多个角色。给角色授权后，该角色下的所有用户拥有相同的权限。关于角色管理的更多介绍请参见角色管理。

- Resource（资源）

  资源（Resource）是MaxCompute中特有的概念。如果您想使用MaxCompute的自定义函数（UDF）或MapReduce功能，则需要依赖资源来完成。详情请参见资源。

S

- SDK

  Software Development Kits软件开发工具包。一般都是一些被软件工程师用于为特定的软件包、软件实例、软件框架、硬件平台、操作系统、文档包等建立应用软件的开发工具的集合。MaxCompute目前支持*Java SDK*介绍和*Python SDK*。

- 授权

  项目空间管理员或者Project Owner授予您对MaxCompute中的Object（或称之为客体，例如表、任务、资源等）进行某种操作的权限，包括读、写、查看等。授权的具体操作请参见用户管理。

- 沙箱

  MaxCompute MapReduce及UDF程序在分布式环境中运行时受到*Java*沙箱的限制。

T

- Table（表）

  表是MaxCompute的数据存储单元，详情请参见表。

- Tunnel

  MaxCompute的数据通道，提供高并发的离线数据上传下载服务。您可以使用Tunnel服务向MaxCompute批量上传数据或者将数据下载。相关命令请参见*#unique_74*或批量数据通道*SDK*。

U

- UDF

  广义的UDF，即User Defined Function，MaxCompute提供的Java编程接口开发自定义函数，详情请参见#unique_75。

  狭义的UDF指用户自定义标量值函数（User Defined Scalar Function），它的输入与输出是一对一的关系，即读入一行数据，写出一条输出值。

- UDAF

  User Defined Aggregation Function，自定义聚合函数，它的输入与输出是多对一的关系，即将多条输入记录聚合成一条输出值。可以与SQL中的GROUP BY语句联用。详情请参见#unique_76/unique_76_Connect_42_section_vdy_4kf_vdb。

- UDTF

  User Defined Table Valued Function，自定义表值函数，用来解决一次函数调用输出多行数据的场景。它是唯一能返回多个字段的自定义函数，而UDF只能一次计算输出一条返回值。详情请参见#unique_76/unique_76_Connect_42_section_a4t_34f_vdb。

4.2 项目空间

项目空间（Project）是MaxCompute的基本组织单元，它类似于传统数据库的Database或Schema的概念，是进行多用户隔离和访问控制的主要边界。项目空间中包含多个对象，例如表（Table）、资源（Resource）、函数（Function）和实例（Instance）等。

一个用户可以同时拥有多个项目空间的权限。通过安全授权，可以在一个项目空间中访问另一个项目空间中的对象，例如表#Table#、资源#Resource#、函数#Function#和实例#Instance#。

您可以通过use project命令进入一个项目空间，如下所示：

```
use my_project -- 进入一个名为my_project的项目空间。
```

运行此命令后，您会进入一个名为my_project的项目空间。您可以直接操作该项目空间下的对象，例如表、资源、函数和实例等。Use Project是MaxCompute客户端提供的命令，关于其它常用的命令介绍请参见MaxCompute常用命令。

 说明：

MaxCompute项目（Project）即DataWorks的工作空间。

4.3 表

表是MaxCompute的数据存储单元，它在逻辑上也是由行和列组成的二维结构，每行代表一条记录，每列表示相同数据类型的一个字段，一条记录可以包含一个或多个列，各个列的名称和类型构成这张表的Schema。

MaxCompute中不同类型计算任务的操作对象（输入、输出）都是表。您可以创建表、删除表以及向表中导入数据。

说明:

DataWorks的数据管理模块可以对MaxCompute表进行新建、收藏、修改数据生命周期管理、修改表结构和数据表/资源/函数权限管理审批等操作，详情请参见数据管理概述。

MaxCompute的表格有两种类型：内部表和外部表（MaxCompute2.0版本开始支持外部表）。

- 对于内部表，所有的数据都被存储在MaxCompute中，表中的列可以是MaxCompute支持的任意一种数据类型。
- 对于外部表，MaxCompute并不真正持有数据，表格的数据可以存放在*OSS*或*OTS*中。MaxCompute仅会记录表格的Meta信息，您可以通过MaxCompute的外部表机制处理OSS或OTS上的非结构化数据，例如视频、音频、基因、气象、地理信息等。

说明:

Dual表使用

- MaxCompute与Oracle等数据库不同，并不会系统自动创建DUAL表。
- 如果您习惯使用Dual表作为最原始的测试表，您可以手动使用命令`CREATE TABLE IF NOT EXISTS DUAL (DUMMY VARCHAR(1));`创建一个名为DUAL且只有一个字段的空表进行测试。需通过`set odps.sql.type.system.odps2=true;`打开新数据类型开关。
- DUAL的使用方式等同于Oracle，如`select getdate() from dual;`

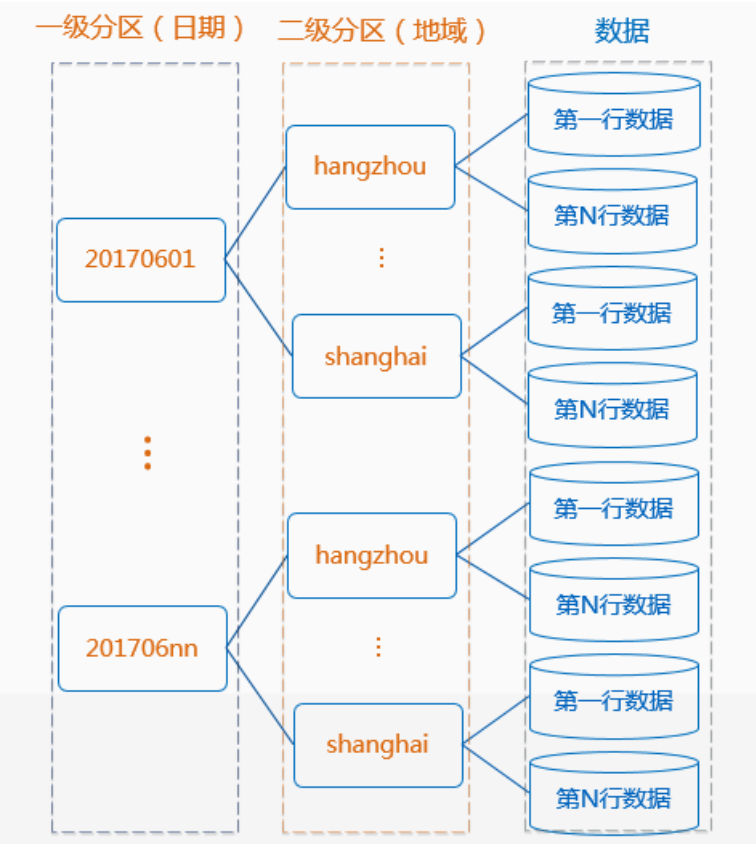
4.4 分区

分区表是指在创建表时指定分区空间，即指定表内的某几个字段作为分区列。分区表实际就是对应分布式文件系统上的的独立的文件夹，该文件夹下是该分区所有数据文件。而分区可以理解为分类，通过分类把不同类型的数据放到不同的目录下。分类的标准就是分区字段，可以是一个，也可以是多个。

分区表的意义在于优化查询。查询表时通过where字句查询指定所需查询的分区，避免全表扫描，提高处理效率，降低计算费用。

MaxCompute将分区列的每个值作为一个分区（目录），您可以指定多级分区，即将表的多个字段作为表的分区，分区之间如多级目录的关系。

使用数据时，如果指定需要访问的分区名称，则只会读取相应的分区，可避免全表扫描，提高处理效率，降低费用。



分区类型

MaxCompute2.0对分区类型的支持进行了扩充，目前MaxCompute支持Tinyint、Smallint、Int、Bigint、Varchar和String分区类型。

说明:

在旧版MaxCompute中，仅支持String类型分区。因为历史原因，虽然可以指定分区类型为Bigint，但是除了表的schema表示其为Bigint外，任何其他情况实际都被处理为String。示例如下：

```
create table parttest (a bigint) partitioned by (pt bigint);
insert into parttest partition(pt) select 1, 2 from dual;
insert into parttest partition(pt) select 1, 10 from dual;
select * from parttest where pt >= '2';
```

执行上述语句后，返回的结果只有一行，因为10被当作字符串和2比较，所以没能返回。

分区使用限制

分区有以下使用限制。

- 单表分区层级最多6级。
- 单表分区数最多允许60000个分区。
- 一次查询最多查询分区数为10000个分区。
- String分区类型的分区值不支持使用中文。

示例如下：

```
--创建一个二级分区表，以日期为一级分区，地域为二级分区
create table src (key string, value bigint) partitioned by (pt string,
region string);
```

查询时，Where条件过滤中使用分区列作为过滤条件。

```
select * from src where pt='20170601'and region='hangzhou'; --正确使用方式。MaxCompute在生成查询计划时只会将'20170601'分区下region为'hangzhou'二级分区的数据纳入输入中。
select * from src where pt = 20170601; -- 错误的使用方式。在这样的使用方式下，MaxCompute并不能保障分区过滤机制的有效性。pt是String类型，当String类型与Bigint (20170601) 比较时，MaxCompute会将二者转换为Double类型，此时有可能会有精度损失。
```

部分对分区操作的SQL的运行效率则较低，会给您带来较高的计费，例如输出到动态分区*#DYNAMIC PARTITION#*。

对于部分MaxCompute的操作命令，处理分区表和非分区表时的语法有差别，详情请参见表操作和 *INSERT*操作。

4.5 生命周期

本文详细介绍了MaxCompute表的生命周期概念。

MaxCompute表的生命周期（LIFECYCLE），指表（分区）数据从最后一次更新的时间算起，在经过指定的时间后没有变动，则此表（分区）将被MaxCompute自动回收。这个指定的时间就是生命周期。

- 生命授权单位：days（天），只接受正整数。
- 非分区表若指定生命周期，自最后一次数据被修改的时间（LastDataModifiedTime）开始计算，经过days天后数据仍未被改动，则此表无需您干预，将会被MaxCompute自动回收（类似drop table操作）。

- 分区表若指定生命周期，则根据各个分区的LastDataModifiedTime判断该分区是否该被回收。不同于非分区表，分区表的最后一个分区被回收后，该表不会被删除。

  说明:

  生命周期回收都是每天定时启动，扫描全量分区，扫到的时刻，Last modify time超过lifecycle指定的时间才回收。

  假设某个分区表生命周期为1天，该分区数据最后一次被修改的时间是17号15点零分，如果18号的回收扫描在15点前扫到这个表（不到一天），就不会回收上述分区。19号回收扫描时才发现这个表的这个分区Last modify time超过lifecycle指定的时间，这时上述分区会被回收。

- 生命周期只能设定到表级别，不能再分区级设置生命周期。创建表时即可指定生命周期。
- 表若不指定生命周期，则表（分区）不会根据生命周期规则被MaxCompute自动回收。

关于建表时怎么指定/修改表生命周期、修改表LastDataModifiedTime等操作请参见表操作。

4.6 资源

本文介绍了MaxCompute的资源（Resource）概念，可为MaxCompute特定操作提供资源依赖。

概念

资源（Resource）是MaxCompute的特有概念，如果您想使用MaxCompute的自定义函数#UDF#或MapReduce功能需要依赖资源来完成，如下所示：

- SQL UDF：您编写UDF后，需要将编译好的Jar包以资源的形式上传到MaxCompute。运行此UDF时，MaxCompute会自动下载这个Jar包，获取您的代码来运行UDF，无需您干预。上传Jar包的过程就是在MaxCompute上创建资源的过程，这个Jar包是MaxCompute资源的一种。
- MapReduce：您编写MapReduce程序后，将编译好的Jar包作为一种资源上传到MaxCompute。运行MapReduce作业时，MapReduce框架会自动下载这个Jar资源，获取您的代码。您同样可以将文本文件以及MaxCompute中的表作为不同类型的资源上传到MaxCompute，您可以在UDF及MapReduce的运行过程中读取、使用这些资源。

MaxCompute提供了读取、使用资源的接口。详情请参见资源使用示例及UDTF使用说明 。

说明:

MaxCompute的自定义函数#UDF#或MapReduce对资源的读取有一定的限制，详情请参见MR限制汇总。

资源类型

MaxCompute支持上传的单个资源大小上限为500MB，资源包括以下几种类型：

- File类型。
- Table类型：MaxCompute中的表。

  
  说明：
  MapReduce引用的table类型资源中，table表字段类型目前只支持BIGINT、DOUBLE、STRING、DATETIME、BOOLEAN，其他类型暂未支持。
- Jar类型：编译好的Java Jar包。
- Archive类型：通过资源名称中的后缀识别压缩类型，支持的压缩文件类型包括.zip/.tgz/.tar.gz/.tar/jar。

资源的相关操作请参见创建资源、删除资源、查看资源清单和查看资源信息。

4.7 函数

本文为您介绍MaxCompute提供的函数功能。

MaxCompute为您提供了SQL计算功能，您可以在MaxCompute SQL中使用系统的内建函数完成一定的计算和计数功能。但当内建函数无法满足要求时，您可以使用MaxCompute提供的Java编程接口开发自定义函数（User Defined Function，以下简称UDF）。

自定义函数#UDF#可以进一步分为标量值函数（UDF），自定义聚合函数（UDAF）和自定义表值函数（UDTF）三种类型。

您在开发完成UDF代码后，需要将代码编译成Jar包，并将此Jar包以Jar资源的形式上传到MaxCompute，最后在MaxCompute中注册此UDF。


说明：
使用UDF时，只需在SQL中指明UDF的函数名及输入参数即可，使用方式与MaxCompute提供的内建函数相同。

函数的相关操作请参见创建函数， 删除函数及查看函数清单。

4.8 任务

任务（Task）是MaxCompute的基本计算单元，SQL及MapReduce功能都是通过任务完成的。

对于您提交的大多数任务，特别是计算型任务，例如SQL DML语句、MapReduce，MaxCompute会对其进行解析，得出任务的执行计划。执行计划由具有依赖关系的多个执行阶段（Stage）构成。

目前，执行计划逻辑上可以被看做一个有向图，图中的点是执行阶段，各个执行阶段的依赖关系是图的边。MaxCompute会依照图（执行计划）中的依赖关系执行各个阶段。在同一个执行阶段内，会有多个进程，也称之为Worker，共同完成该执行阶段的计算工作。同一个执行阶段的不同Worker只是处理的数据不同，执行逻辑完全相同。计算型任务在执行时，会被实例化，您可以操作这个实例（Instance）的信息，例如获取实例状态#Status Instance#、终止实例运行#Kill Instance#等。

另一方面，部分MaxCompute任务并不是计算型的任务，例如SQL中的DDL语句，这些任务本质上仅需要读取、修改MaxCompute中的元数据信息。因此，这些任务无法被解析出执行计划。

说明:

在MaxCompute中，并不是所有的请求都会被转化为任务（Task），例如工作空间#Workspace#、资源#Resource#、自定义函数#UDF#及实例(Instance)的操作均不需要通过MaxCompute的任务来完成。

4.9 任务实例

本文向您介绍MaxCompute任务实例及实例状态。

在MaxCompute中，部分任务#Task#在执行时会被实例化，以MaxCompute实例（下文简称为实例或Instance）的形式存在。实例会经历运行（Running）和结束（Terminated）两个阶段。

运行阶段的状态为Running（运行中），而结束阶段则会有Success（成功）、Failed（失败）和Canceled（被取消）三种状态。您可以根据运行任务时MaxCompute给出的实例ID进行查询、改变任务的状态等操作，示例如下：

```
status; --查看某实例的状态。
kill; --停止某实例，将其状态设置为Canceled。
```

```
wait; --查看某实例的运行日志。
```

4.10 ACID语义

本文为您介绍MaxCompute在作业并发情况下ACID的语义。

相关术语

- 操作：指在MaxCompute上提交的单个作业。
- 数据对象：指持有实际数据的对象，例如非分区表、分区。
- INTO类作业：`INSERT INTO`、`DYNAMIC INSERT INTO`等。
- OVERWRITE类作业：`INSERT OVERWRITE`、`DYNAMIC INSERT OVERWRITE`等。
- Tunnel数据上传：可以归结为INTO类或OVERWRITE类作业。

ACID语义描述

- 原子性（Atomicity）：一个操作或是全部完成，或是全部不完成，不会结束在中间某个环节。
- 一致性（Consistency）：从操作开始至结束的期间，数据对象的完整性没有被破坏。
- 隔离性（Isolation）：操作独立于其它并发操作完成。
- 持久性（Durability）：操作处理结束后，对数据的修改将永久有效，即使出现系统故障，该修改也不会丢失。

MaxCompute的ACID具体情形

- 原子性（Atomicity）
  - 任何时候MaxCompute会保证在冲突时只会一个作业成功，其它冲突作业失败。
  - 对于单个表/分区的CREATE/OVERWRITE/DROP操作，可以保证其原子性。
  - 跨表操作时不支持原子性（例如MULTI-INSERT）。
  - 在极端情况下，以下操作可能不保证原子性：
    - `DYNAMIC INSERT OVERWRITE`多于一万个分区，不支持原子性。
    - INTO类操作：这类操作的失败的原因是事务回滚时数据清理失败，但不会造成原始数据丢失。
- 一致性（Consistency）
  - OVERWRITE类作业可保证一致性。
  - INTO类作业在冲突失败后可能存在失败作业的数据残留。
- 隔离性（Isolation）
  - 非INTO类操作保证读已提交。
  - INTO类操作存在读未提交的场景。

- **持久性（Durability）**

  MaxCompute保证数据的持久性。

5 使用限制

本文为您介绍MaxCompute产品各模块的使用限制。

使用限制汇总：

- SQL开发限制
- 数据上传下载限制
- 操作命令限制
- MapReduce限制
- 图模型限制
- 安全配置限制
- Lightning限制
- PyODPS限制