

ALIBABA CLOUD

Alibaba Cloud

大数据计算服务
产品简介

文档版本：20201124

 阿里云

法律声明

阿里云提醒您在使用或阅读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 |
|----------------------------------------------------------------------------------------|------------------------------------|-----------------------------------------------------------------------------------------------------------------|
|  危险 | 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。 |  危险 重置操作将丢失用户配置数据。 |
|  警告 | 该类警示信息可能会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，恢复业务时间约十分钟。 |
|  注意 | 用于警示信息、补充说明等，是用户必须了解的内容。 |  注意 权重设置为0，该服务器不会再接受新请求。 |
|  说明 | 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。 | 单击设置>网络>设置网络类型。 |
| 粗体 | 表示按键、菜单、页面名称等UI元素。 | 在结果确认页面，单击确定。 |
| Courier字体 | 命令或代码。 | 执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。 |
| 斜体 | 表示参数、变量。 | <code>bae log list --instanceid</code> <i>Instance_ID</i> |
| [] 或者 [a b] | 表示可选项，至多选择一个。 | <code>ipconfig [-all -t]</code> |
| { } 或者 {a b} | 表示必选项，至多选择一个。 | <code>switch {active stand}</code> |

目录

| | |
|--------------------|----|
| 1.什么是MaxCompute | 05 |
| 2.使用须知 | 08 |
| 3.基本概念 | 11 |
| 3.1. MaxCompute术语表 | 11 |
| 3.2. 项目 | 13 |
| 3.3. 表 | 13 |
| 3.4. 分区 | 14 |
| 3.5. 生命周期 | 16 |
| 3.6. 资源 | 17 |
| 3.7. 函数 | 18 |
| 3.8. 任务 | 18 |
| 3.9. 任务实例 | 18 |
| 3.10. ACID语义 | 19 |
| 4.使用限制 | 21 |
| 5.功能与可用区 | 22 |
| 6.存储性能 | 23 |

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。

MaxCompute 大数据计算服务 数据仓库解决方案

随着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（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的计算模型。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、生物医药、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[什么是DataWorks](#)。

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。
- MaxCompute通过了独立的第三方审计师针对阿里云对AICPA可信服务标准中关于安全性、可用性和机密性原则符合性描述的审计。审计报告请参见[SOC 3报告](#)。

● 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

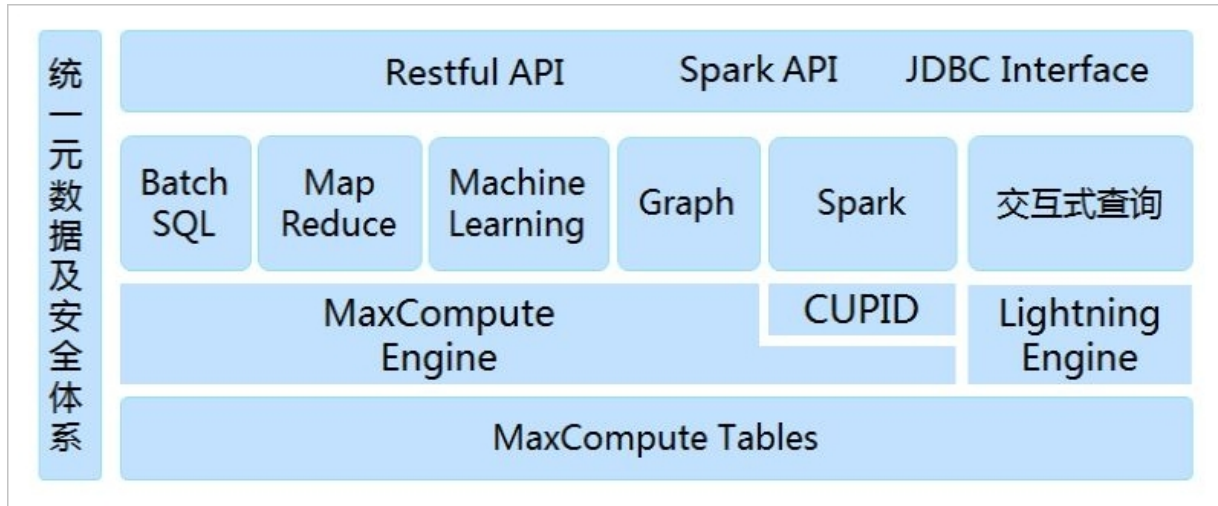
基于MaxCompute的Serverless无服务器的设计思路，用户只需关心作业和数据，而无需关心底层分布式架构及运维。

● 极致弹性扩展

MaxCompute提供按量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，支持多种数据类型版本说明，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。
- 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。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。
- **Spark on MaxCompute**：Spark on MaxCompute是阿里云开发的大数据分析引擎，为您提供大数据处理能力。详情请参见**MaxCompute Spark概述**。

- **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**。

2.使用须知

本文根据您的角色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，建议先熟悉如下模块，然后再有针对性地对深入学习其他模块。

| 模块 | 说明 |
|---------------|--------------------------------------------------------------------------------------|
| 产品简介 | 介绍MaxCompute产品的概况及主要功能。通过阅读该章节，您会对MaxCompute有一个总体的认知。 |
| 快速开始 | 通过示例，指导您如何申请账号、安装客户端、创建表、授权、导入导出数据及运行SQL、UDF和MapReduce作业。 |
| MaxCompute术语表 | 介绍MaxCompute的基本概念。 |
| 常用命令列表 | 介绍MaxCompute的常用命令。您可以进一步熟悉如何操作MaxCompute。 |
| 工具 | 您需要在分析数据前掌握MaxCompute常用工具的下载、配置及使用方法。 |
| 客户端工具 | 您可以通过此工具对MaxCompute进行操作。 |
| 配置Endpoint | 介绍MaxCompute各区域的开通情况和连接方式，并对您在与其他云产品（ECS、Tablestore或OSS）互访场景中遇到的网络连通性和下载数据收费等问题进行说明。 |

如果您是数据分析师

如果您是数据分析师，建议熟读SQL模块的内容。您可以查询并分析存储在MaxCompute上的大规模数据。MaxCompute SQL支持如下主要功能：


- DDL语句。
- 使用CREATE、DROP和ALTER语句管理表和分区。
- 使用SELECT语句选择表中的记录，使用WHERE语句查看满足条件的记录，实现过滤功能。
- 使用等值连接JOIN关联两张表。
- 使用GROUP BY实现列聚合操作。
- 使用INSERT OVERWRITE或INSERT INTO把结果记录插入到另一张表中。
- 使用内建函数和自定义函数（UDF）实现一系列的计算。
- 收集表的统计信息和设置表生命周期。
- 正则表达式。

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，且希望解决某些无法用SQL实现的数据分析问题，推荐您学习MaxCompute更高级的功能模块。

| 模块 | 说明 |
|----|----|
|----|----|

| 模块 | 说明 |
|------------|----------------------------------------------------------------------------------------------|
| 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。 |
| Tunnel | 您可以使用Tunnel服务向MaxCompute批量上传离线数据或从MaxCompute下载离线数据。 |
| Java SDK | 向开发者提供的Java接口。 |
| Python SDK | 向开发者提供的Python接口。 |

 **说明** MapReduce和Graph功能仍处于公测中，若您想使用这部分功能，可以[提工单](#)申请。申请时请指明您的项目名称，系统会在7个工作日内处理。

如果您是项目Owner或管理员

如果您是一个项目的Owner（创建和使用项目）或管理员（项目、安全和费用管理）需要熟知如下模块。

| 模块 | 子模块 | 说明 |
|----|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | 创建项目前期工作 | <p>项目（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的权限，通过安全授权，可以在一个项目中访问另一个项目中的对象，例如表（Table）、资源（Resource）、函数（Function）和实例（Instance）。使用MaxCompute，实际是操作项目中的各种对象。前期准备工作如下：</p> <ul style="list-style-type: none"> 资源预算 <p>MaxCompute收费资源主要包含存储、计算和公网下载流量。</p> <ul style="list-style-type: none"> 存储资源：按量阶梯计费。您可以按照数据量套用公式预估费用。由于数据不是当天全部存储在MaxCompute，且每时每刻都会存在数据导入导出，所以预算结果不是绝对值。 计算资源：计算资源（SQL、MapReduce、Spark和Lightning计算作业）分为按量计费 and 包年包月模式。由于使用初期不容易评估计算资源使用量，建议您先使用按量计费模式，测试一段时间后根据使用量再决定是否使用包年包月模式。 公网下载流量：按量计费，只有通过公网下载才会收费。 <p>详细计费说明请参见存储费用（按量计费）、计算费用和下载费用（按量计费）。</p> 准备账号并开通服务 <p>创建MaxCompute项目前，必须先开通MaxCompute服务，且只能将阿里云账号作为主账号，同时该账号为计费主体。确定账号后，在开通MaxCompute服务时，您需要根据资源预算结论选择按量计费或包年包月模式。</p> |

| 模块 | 子模块 | 说明 |
|------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 项目管理 | 创建项目 | 创建项目具体操作请参见 创建项目空间 。 |
| | 项目成员管理 | 成员管理主要考虑成员的职责和安全问题，如果通过DataWorks使用MaxCompute，您需要考虑两个产品之间的关联权限。 |
| | 子账号管理 | <p>MaxCompute项目支持主账号（云账号）和RAM子账号两种账号体系。您可以将主账号的任意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。RAM子账号详情请参见准备RAM用户。</p> <p>通过DataWorks使用MaxCompute和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。</p> <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 10px; margin-top: 10px;"> <p>说明</p> <ul style="list-style-type: none"> 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，需要及时清理对应子账号。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。 </div> |
| | 调度资源管理 | <p>即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的调度资源分为如下两种模式：</p> <ul style="list-style-type: none"> 默认调度资源。指DataWorks的公共资源池。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。 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或其他任务。自定义资源组可以有效避免默认调度资源组的限制，当前新建自定义资源组功能需要您提交工单申请，已有的自定义资源组不受影响。 |
| | 项目设置 | 在项目开发过程中，部分项目的设置操作需要项目Owner来执行。例如，设置项目是否允许全表扫描、设置项目默认打开2.0新类型等。详情请参见 项目操作 。 |
| 费用管理 | 无 | <p>资源预算是在使用之前进行成本预估。基于MaxCompute的计费方式，很多业务无法更准确地预估成本，因此在整个业务开发过程中需要进行费用管理，主要需要关注：</p> <ul style="list-style-type: none"> 产品的计费定价，详情请参见计费方式。 产品支持按量计费和包年包月转换，详情请参见计费方式转换。 |

3. 基本概念

3.1. MaxCompute术语表

本文列举了MaxCompute常见的概念和术语。

术语

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[AccessKey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、项目的用户与授权管理、跨项目的资源分享以及项目的数据保护。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可以提交命令完成项目管理、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[数据类型版本说明](#)。

- DDL

数据定义语言（Data Definition Language）。例如创建表、创建视图等操作，详情请参见[DDL语句](#)。

- DML

数据操作语言（Data Manipulation Language）。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底层任务调度模块为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

MapReduce是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（一个或多个字段的组合）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（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。详情请参见[项目](#)。

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 Kit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

项目管理员或者项目 Owner授予您对MaxCompute中的Object（或称之为对象，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（Sandboxie）

沙箱是一种按照安全策略限制程序行为的执行环境。沙箱机制是一种安全机制，将Java代码限定在特定的运行范围中，并且严格限制代码对本地系统资源访问，通过这样的措施来保证对代码的有效隔离，防止对本地系统造成破坏。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批量上传数据或者向本地进行批量数据下载。相关命令请参见[Tunnel命令参考](#)或[批量数据通道SDK](#)。

U

- UDF

广义的UDF（User Defined Function），代表了自定义标量函数、自定义聚合函数及自定义表函数三种类型的自定义函数的集合。MaxCompute提供的Java编程接口开发自定义函数，详情请参见[概述](#)。

狭义的UDF指用户自定义标量值函数（User Defined Scalar Function），它的输入与输出是一一对应的关系，即读入一行数据，写出一条输出值。

- UDAF

自定义聚合函数（User Defined Aggregation Function），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[UDAF](#)。

- UDTF

自定义表值函数（User Defined Table Valued Function），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[UDTF](#)。

3.2. 项目

项目（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。项目中包含多个对象，例如表（Table）、资源（Resource）、函数（Function）和实例（Instance）等。


项目空间 用户隔离 访问控制

一个用户可以同时拥有多个项目的权限。通过安全授权，可以在一个项目中访问另一个项目中的对象，详情请参见[基于Package的跨项目空间资源访问](#)。

您可以通过`use project;`命令进入一个项目，如下所示。

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

运行此命令后，您会进入一个名为my_project的项目，可以直接操作该项目下的对象，例如表、资源、函数和实例等。`Use Project;`是MaxCompute客户端提供的命令，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。详情请参见[简单模式和标准模式的区别](#)。

3.3. 表

表是MaxCompute的数据存储单元。它在逻辑上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，表的结构由各个列的名称和类型构成。

表结构 数据存储单元 表

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[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，表中列的数据类型可以是MaxCompute支持的任何一种[数据类型版本说明](#)。
- 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表的使用：

- MaxCompute与Oracle等数据库不同，系统并不会自动创建DUAL表。
- 如果您习惯使用DUAL表作为最原始的测试表，您可以使用如下命令创建DUAL表进行测试。

```
---打开新数据类型开关。
set odps.sql.type.system.odps2=true;
---创建一个名为DUAL且只有一个字段的空表。
CREATE TABLE IF NOT EXISTS DUAL (DUMMY VARCHAR(1));
```

- DUAL的使用方式等同于Oracle，如 `SELECT getdate() FROM DUAL;`

3.4. 分区

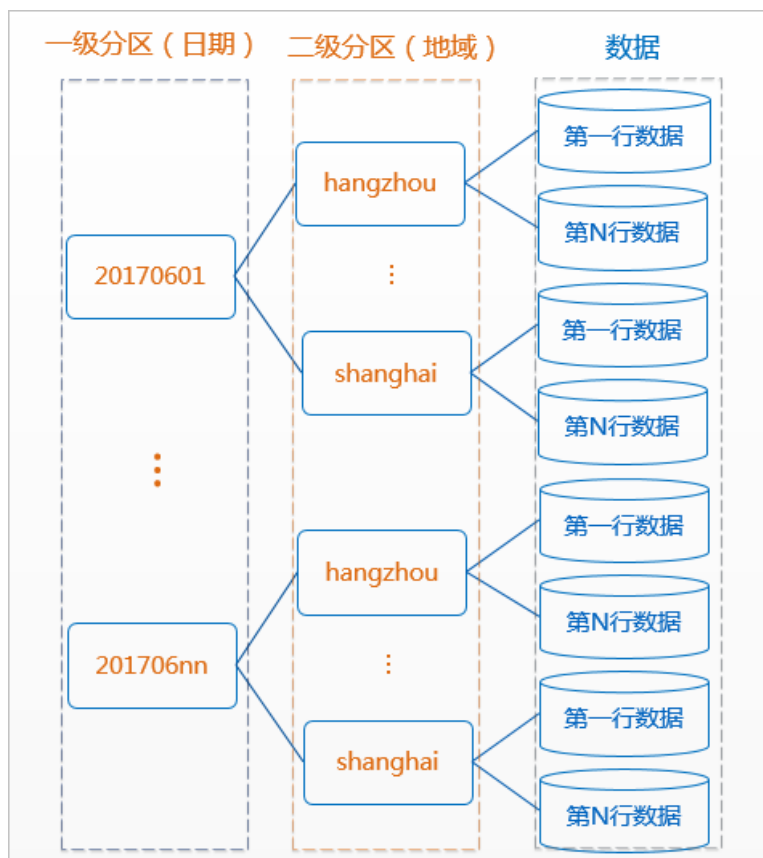
分区表是指拥有分区空间的表，即在创建表时指定表内的一个或者某几个字段作为分区列。分区表实际就是对应分布式文件系统上的独立的文件夹，一个分区对应一个文件夹，文件夹下是对应分区所有的数据文件。

分区概述

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。

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，分区之间类似多级目录的关系。

分区表的意义在于优化查询。查询表时通过WHERE子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。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。



部分对分区操作的SQL的运行效率较低，会给您带来较高的费用，例如[输出到动态分区 \(DYNAMIC PARTITION\)](#)。

对于部分操作MaxCompute的命令，处理分区表和非分区表时语法有差别，详情请参见[表操作](#)和[INSERT操作](#)。

使用限制

- 单表分区层级最多为6级。
- 单表分区数最大值为60000个。
- 单次查询允许查询最多的分区个数为10000个。
- STRING分区类型的分区值不支持使用中文。

分区列的数据类型

MaxCompute 2.0数据类型版本支持的分区字段为TINYINT、SMALLINT、INT、BIGINT、VARCHAR、STRING。

MaxCompute 1.0数据类型版本支持的分区字段仅有STRING。虽然可以指定分区列的类型为BIGINT，但是除了表的字段显示为BIGINT类型，任何其他情况（例如，字段的计算和比较）下都当作STRING类型处理，示例如下。执行下述语句后，返回的结果只有一行。因为10被当作字符串和2比较，因此该行不返回。

```
---创建表parttest。
CREATE TABLE parttest (a bigint) PARTITIONED BY (pt bigint);
---向表中插入数据。
INSERT INTO parttest partition(pt) SELECT 1, 2;
INSERT INTO parttest partition(pt) SELECT 1, 10;
---查询表中字段pt大于等于2的行。
SELECT * FROM parttest WHERE pt >= '2';
```

示例

- 创建分区。

```
--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 (shop_name string, customer_id bigint) PARTITIONED BY (pt string, region string);
```

- 使用分区列作为过滤条件查询数据。

```
--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'and region='hangzhou';
--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，此时有可能会精度损失。
select * from src where pt = 20170601;
```

3.5. 生命周期

本文为您介绍MaxCompute表的生命周期概念。

生命周期 回收表

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。

- 生命周期单位为天，取值为正整数。
- 对于非分区表，如果表数据在生命周期内没有被修改，经过指定天数后此表将会被MaxCompute自动回收（类似DROP TABLE操作）。生命周期从最后一次表数据被修改的时间（LastDataModifiedTime）起开始计算。
- 对于分区表，每个分区可以分别被回收。在生命周期内未被修改数据的分区，经过指定的天数后此分区将会被回收，否则会被保留。每个分区的生命周期是从最后一次分区数据被修改的时间（LastDataModifiedTime）起开始计算。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

- 生命周期回收为每天定时启动，扫描全量分区。LastDataModifiedTime需要超过生命周期指定的时间才回收。

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2020年2月17日15时。如果在2020年2月18日15时之前扫描此表（不到一天），则不会回收表分区。如果2020年2月19日回收扫描时发现表分区LastDataModifiedTime超过生命周期指定的时间，则上述分区会被回收。

- 生命周期主要提供定期回收表或分区的功能，每天根据服务的繁忙程度，不定时回收。不能确保表或分区的生命周期到期后，立刻被回收。

- 只能在表级别设置生命周期，不能在分区级设置生命周期。为分区表指定的生命周期，适用于该表所有的分区。创建表时即可指定生命周期。
- 如果您没有为表指定生命周期，则表（分区）不会根据生命周期规则被MaxCompute自动回收。

关于建表时如何指定、修改表生命周期、修改表 LastDataModifiedTime 等操作，请参见[表操作](#)。

3.6. 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。

资源 resource MapReduce Archive

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](#)。

3.7. 函数


本文为您介绍MaxCompute提供的函数功能，包括内建函数和UDF。

内建函数 UDF

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](#)。

3.8. 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。

任务 Task

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，得到任务的执行计划。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，您可以对这个实例（Instance）进行操作，例如[获取实例状态（Status Instance）](#)、[终止实例运行（Kill Instance）](#)等。

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[项目空间（Project）](#)、[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。

3.9. 任务实例

本文向您介绍MaxCompute任务实例及实例状态。

任务实例 实例状态

在MaxCompute中，SQL、Spark和Mapreduce任务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。

运行阶段的实例状态为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。

```
--查看某实例的状态。
status instance_id;
--停止某实例，将其状态设置为Canceled。
kill instance_id;
--查看某实例的运行日志。
wait instance_id;
```

其中，instance_id为需要查询的实例ID号，请您根据实际的ID号进行替换。

3.10. ACID语义

本文为您介绍MaxCompute在作业并发情况下ACID的语义。

ACID 原子性 一致性 隔离性 持久性

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：指INSERT INTO、DYNAMIC INSERT INTO等包含关键字INTO的SQL作业。
- OVERWRITE类作业：指INSERT OVERWRITE、DYNAMIC INSERT OVERWRITE等包含关键字OVERWRITE的SQL作业。
- 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特性

- 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。
 - 对于单个表或分区的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。

4.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

使用限制汇总：

- [SQL开发限制](#)
- [使用限制](#)
- [操作命令限制](#)
- [使用限制](#)
- [安全配置限制](#)
- [图模型使用限制](#)
- [Lightning使用限制](#)

5.功能与可用区

MaxCompute以数据为中心，内建多种计算模型，满足广泛的数据分析需求。本文为您介绍每种计算模型在当前可用区的开通情况。

| 可用区 | SQL | MapReduce | Spark | Mars |
|------------|-----|-----------|-------|------|
| 华北2（北京） | 已开通 | 已开通 | 已开通 | 已开通 |
| 华东1（杭州） | 已开通 | 已开通 | 已开通 | 已开通 |
| 华东2（上海） | 已开通 | 已开通 | 已开通 | 已开通 |
| 华南1（深圳） | 已开通 | 已开通 | 已开通 | 已开通 |
| 西南1（成都） | 已开通 | 已开通 | 已开通 | 已开通 |
| 华北3（张家口） | 已开通 | 已开通 | 已开通 | 已开通 |
| 中国（香港） | 已开通 | 已开通 | 已开通 | 已开通 |
| 新加坡（新加坡） | 已开通 | 已开通 | 已开通 | 未开通 |
| 马来西亚（吉隆坡） | 已开通 | 已开通 | 已开通 | 未开通 |
| 印度尼西亚（雅加达） | 已开通 | 已开通 | 已开通 | 未开通 |
| 澳大利亚（悉尼） | 已开通 | 已开通 | 已开通 | 未开通 |
| 日本（东京） | 已开通 | 已开通 | 已开通 | 未开通 |
| 美国（硅谷） | 已开通 | 已开通 | 已开通 | 未开通 |
| 美国（弗吉尼亚） | 已开通 | 已开通 | 已开通 | |
| 德国（法兰克福） | 已开通 | 已开通 | 已开通 | 未开通 |
| 印度（孟买） | 已开通 | 已开通 | 已开通 | 未开通 |
| 英国（伦敦） | 已开通 | 已开通 | 已开通 | 未开通 |

6. 存储性能

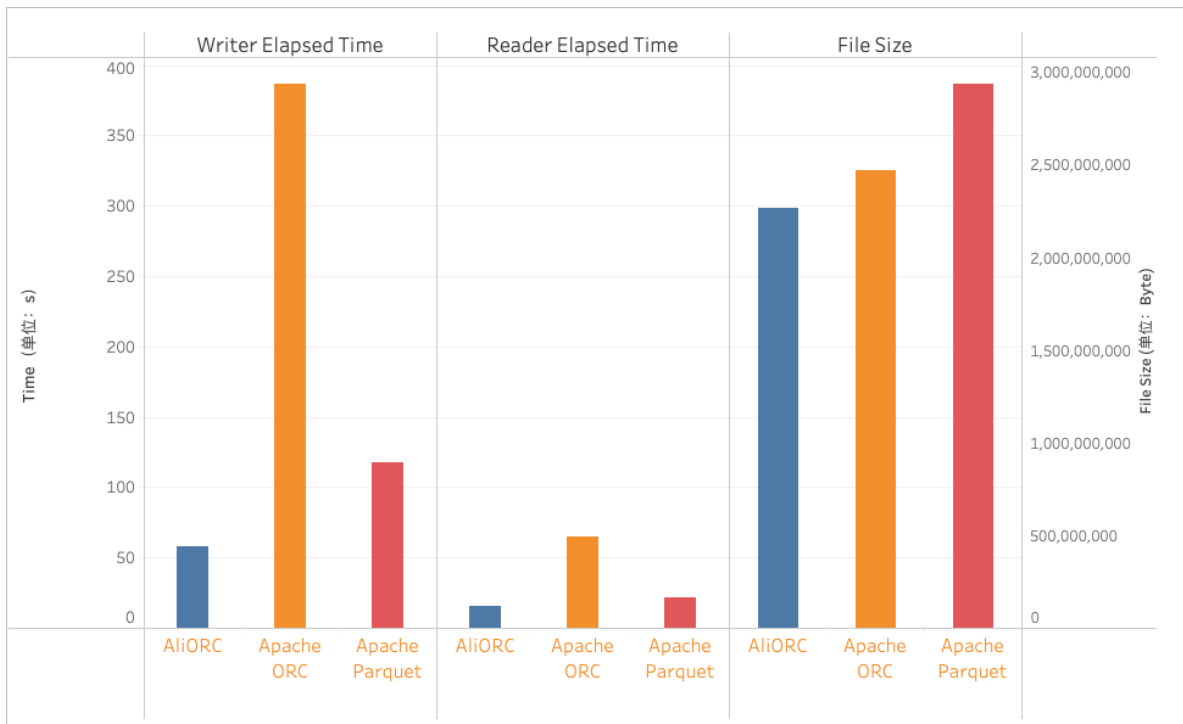
2020年2月，MaxCompute数据存储格式全面升级为AliORC。本文通过TPC-DS测试数据对AliORC、Apache ORC和Apache Parquet进行测试对比，为您提供MaxCompute数据存储性能参照。

测试结果

- 数据集（即24张测试表）测试结果对比数据如下。

| 比较类型 | 写入数据时长 (Writer Elapsed Time) | 读取数据时长 (Reader Elapsed Time) | 数据存储量 (File Size) |
|-------------------------|------------------------------|------------------------------|-------------------|
| AliORC与Apache ORC比较 | 下降超过85% | 下降超过76% | 下降超过8% |
| AliORC与Apache Parquet比较 | 下降超过50% | 下降超过28% | 下降超过22% |

数据集测试结果如下。

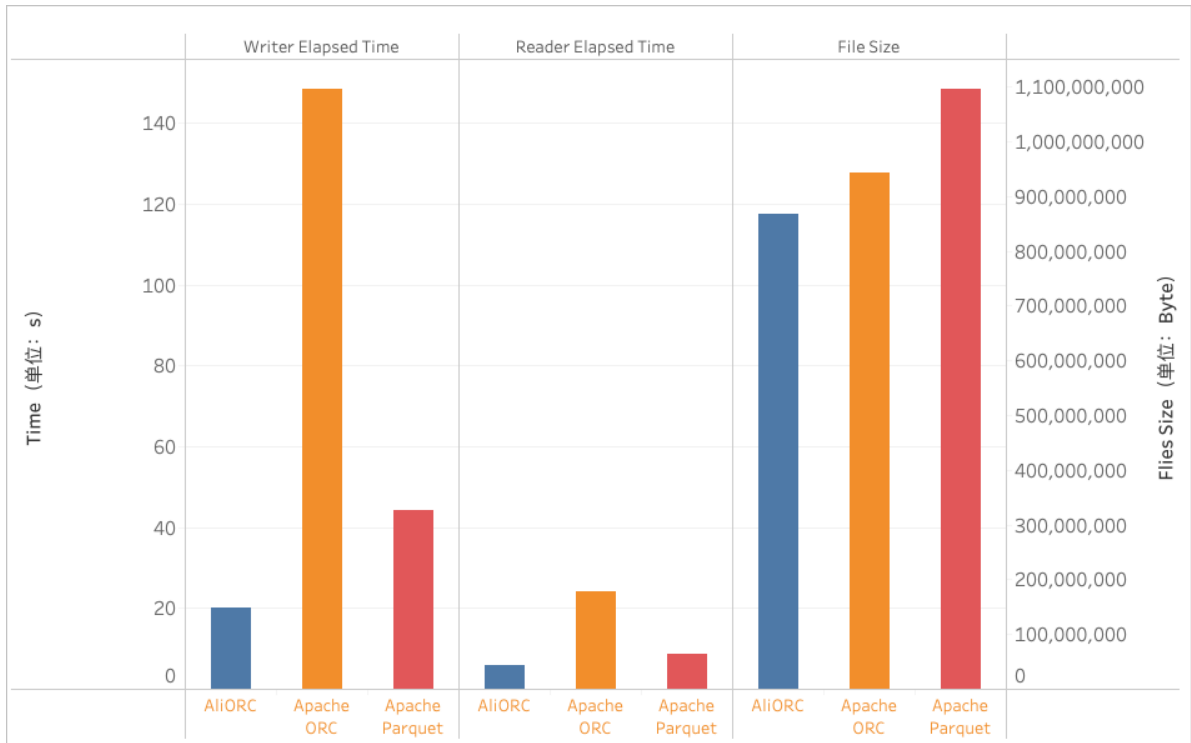


参数说明如下：

- Writer Elapsed Time：写入数据时长，即导入TPC-DS的CSV格式数据至ORC或Parquet的时间。单位为秒。
 - Reader Elapsed Time：读取数据时长，即ORC或Parquet执行完整数据扫描的时间。单位为秒。
 - File Size：数据存储大小，即所有表文件大小的总和。单位为字节（Byte）。
- 最大单表store_sales测试结果对比数据如下。

| 比较类型 | 写入数据时长 (Writer Elapsed Time) | 读取数据时长 (Reader Elapsed Time) | 数据存储量 (File Size) |
|-------------------------|------------------------------|------------------------------|-------------------|
| AliORC与Apache ORC比较 | 下降超过86% | 下降超过74% | 下降超过7% |
| AliORC与Apache Parquet比较 | 下降超过54% | 下降超过30% | 下降超过20% |

最大单表store_sales的测试结果如下。



测试环境

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

数据集列表

TPC-DS是一个面向决策支持的测评标准，采用星型、雪花型等多维数据模式。TPC-DS包含7张事实表，17张维度表，平均每张表有18列数据。数据和值存在倾斜的，与真实场景相似。TPC-DS是客观衡量多个不同Hadoop版本以及SQL on Hadoop技术的最佳测试集。

本次测试数据集中包含以下24张TPC-DS表。

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