

ALIBABA CLOUD

# 阿里云

## 表格存储Tablestore 产品简介

文档版本：20220524

 阿里云

## 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
 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。
<b>粗体</b>	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击 <b>确定</b> 。
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

1.什么是表格存储	05
2.功能特性	09
3.产品架构	12
4.产品优势	17
5.应用场景	19
6.常见问题	23
7.基础概念	26
7.1. 实例	26
7.2. 服务地址	28
7.3. 读写吞吐量	29
7.4. 地域	30
8.快速玩转Tablestore入门与实战	32
9.产品技术服务	36

# 1.什么是表格存储

表格存储（Tablestore）面向海量结构化数据提供Serverless表存储服务，同时针对物联网场景深度优化提供一站式的IoTstore解决方案。适用于海量账单、IM消息、物联网、车联网、风控、推荐等场景中的结构化数据存储，提供海量数据低成本存储、毫秒级的在线数据查询和检索以及灵活的数据分析能力。

## 表格存储学习路径

您可以通过[表格存储学习路径](#)快速了解表格存储产品，学习表格存储的基础数据读写和进阶使用，以及高阶解决方案实操案例。

## 基本概念

在使用表格存储前，您需要了解以下基本概念。

术语	说明
地域	地域（Region）物理的数据中心，表格存储服务会部署在多个阿里云地域中，您可以根据自身的业务需求选择不同地域的表格存储服务。更多信息，请参见 <a href="#">表格存储已经开通的Region</a> 。
读写吞吐量	读吞吐量和写吞吐量的单位为读服务能力单元和写服务能力单元，服务能力单元（Capacity Unit，简称CU）是数据读写操作的最小计费单位。更多信息，请参见 <a href="#">读写吞吐量</a> 。
实例	实例（Instance）是使用和管理表格存储服务的实体，每个实例相当于一个数据库。表格存储对应用程序的访问控制和资源计量都在实例级别完成。更多信息，请参见 <a href="#">实例</a> 。
服务地址	每个实例对应一个服务地址（EndPoint），应用程序在进行表和数据操作时需要指定服务地址。更多信息，请参见 <a href="#">服务地址</a> 。
数据生命周期	数据生命周期（Time To Live，简称TTL）是数据表的一个属性，即数据的存活时间，单位为秒。表格存储会在后台对超过存活时间的数据进行清理，以减少您的数据存储空间，降低存储成本。更多信息，请参见 <a href="#">数据版本和生命周期</a> 。

## 数据存储模型

表格存储提供了宽表（WideColumn）模型、时序（TimeSeries）模型和消息（Timeline）模型三种数据存储模型，请根据使用场景选择合适的模型。不同数据存储模型支持的功能特性请参见[功能特性](#)。

模型	描述
宽表模型	类Bigtable/HBase模型，可应用于元数据、大数据等多种场景，支持数据版本、生命周期、主键列自增、条件更新、局部事务、原子计数器、过滤器等功能。更多信息，请参见 <a href="#">宽表模型</a> 。
时序模型	针对时间序列数据的特点进行设计的模型，可应用于物联网设备监控、设备采集数据、机器监控数据等场景，支持自动构建时序元数据索引、丰富的时序查询能力等功能。更多信息，请参见 <a href="#">时序模型</a> 。
消息模型	针对消息数据场景设计的模型，可应用于IM、Feed流等消息场景。能满足消息场景对消息保序、海量消息存储、实时同步的需求，同时支持全文检索与多维度组合查询。更多信息，请参见 <a href="#">消息模型</a> 。

## 产品计费

表格存储支持预留模式（预付费）和按量模式（后付费）两种计费模式，详细说明请参见下表。

计费模式	描述
预留模式	<p>根据业务提前规划硬件资源消耗后，预先购买计算能力等资源以供业务使用。此模式适用于业务峰值较为平稳，可预测的场景。</p> <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 5px; margin: 5px 0;"> <p> <b>说明</b> 有关评估选型的更多信息，请参见<a href="#">使用预留模式时如何进行预估选型</a>。</p> </div> <p>计费项包括计算能力、数据存储量和外网下行流量，其中数据存储量包括高性能存储、容量型存储和多元索引存储。更多信息，请参见<a href="#">计量计费项</a>。</p>
按量模式	<p>根据业务的实时读写吞吐量、存储空间等资源计费，无需提前规划硬件资源消耗。此模式适用于业务峰谷变化较大，不可预测的场景。</p> <p>计费项包括读吞吐量、写吞吐量、数据存储量和外网下行流量。更多信息，请参见<a href="#">计量计费项</a>。</p>

## 使用方式

您可以通过如下方式使用表格存储产品。

使用方式	描述
控制台	阿里云提供的Web服务页面，方便您便捷的操作表格存储。更多信息，请参见 <a href="#">表格存储控制台</a> 。
SDK	支持主流开发语言Java、Go、Python、Node.js、.Net和PHP。更多信息，请参见 <a href="#">SDK概览</a> 。
命令行工具	支持通过简单的命令操作表格存储。更多信息，请参见 <a href="#">命令行工具</a> 。

## 快速入门

通过控制台或者命令行工具，您可以快速体验宽表模型中数据表或者时序模型中时序表的操作。更多信息，请参见[快速入门](#)。

## 计算与分析

表格存储支持通过MaxCompute、Spark、Data Lake Analytics（简称DLA）、Hive或者HadoopMR以及函数计算进行计算与分析，请根据实际场景选择相应分析工具。

分析工具	操作	描述
MaxCompute	<a href="#">使用MaxCompute</a>	通过MaxCompute客户端为表格存储的数据表创建外部表，即可访问表格存储中的数据。
Spark	<a href="#">使用Spark计算引擎</a>	使用Spark计算引擎时，支持通过E-MapReduce SQL或者DataFrame编程方式访问表格存储。

分析工具	操作	描述
DLA	<a href="#">使用DLA服务</a>	通过控制台、MySQL Client或者JDBC方式接入DLA服务并使用SQL查询表格存储中的数据。
Hive或者Hadoop MR	<a href="#">使用Hive或者HadoopMR</a>	使用Hive或者HadoopMR访问表格存储中的数据。
函数计算	<a href="#">使用函数计算</a>	通过函数计算访问表格存储，对表格存储增量数据进行实时计算。
Flink	<a href="#">使用Flink</a>	通过实时计算Flink访问表格存储中的维表或者结果表，实现大数据实时计算与分析。
表格存储SQL查询	<a href="#">使用SQL查询</a>	SQL查询为多数据引擎提供统一的访问接口。通过SQL查询功能，您可以对表格存储中数据进行复杂的查询和高效的分析。

## 迁移同步

您可以将异构数据平滑迁移同步到表格存储，还可以将数据同步到对象存储OSS（Object Storage Service）等服务中。

分类	数据同步	猫叔
数据导入	<a href="#">同步MySQL数据</a>	使用DataX、DTS或者canal工具将MySQL数据库中的数据同步迁移到表格存储中。
	<a href="#">同步Kafka数据</a>	基于Tablestore Sink Connector将Apache Kafka中的数据批量导入到表格存储的数据表或者时序表中。
	<a href="#">同步HBase数据</a>	使用DataX将HBase数据库中的全量数据同步到表格存储中。
	<a href="#">同步MaxCompute数据</a>	使用DataWorks将MaxCompute中的全量数据同步到表格存储中。
	<a href="#">同步表格存储数据</a>	使用通道服务、DataWorks或者DataX将表格存储数据表中数据同步到另一个数据表中。
数据导出	<a href="#">同步到MaxCompute</a>	使用DataWorks将表格存储中的全量数据或者增量数据导出到MaxCompute。
	<a href="#">同步到OSS</a>	使用DataWorks将表格存储中的全量数据或者增量数据导出到OSS。

## 技术支持

表格存储为您提供专业的免费的技术咨询服务，欢迎通过钉钉扫描二维码加入相应交流群。

- 为互联网应用、大数据、社交应用等开发者提供的技术交流群有11789671（表格存储技术交流群）和23307953（表格存储技术交流群-2）。



表格存储技术交流群



表格存储技术交流群-2

- 为物联网和时序模型开发者提供的技术交流群有44327024（物联网存储IoTstore开发者交流群）。



物联网IoTstore开发者交流群

## 2.功能特性

表格存储（Tablestore）提供了宽表模型、时序模型和消息模型三种数据存储模型。本文介绍了不同模型支持的功能特性。

### 宽表模型

宽表模型支持的功能特性请参见下表。

功能特性	描述	相关文档
表操作	支持列出实例中的全部数据表、创建一张数据表、查询数据表的配置信息、更新数据表的配置信息以及删除一张数据表。	<a href="#">表操作</a>
基础数据操作	表格存储提供了PutRow、GetRow、UpdateRow和DeleteRow的单行数据操作接口以及BatchWriteRow、BatchGetRow和GetRange的多行数据操作接口。您可以通过单行数据操作接口或者多行数据操作接口读写表中数据。	<ul style="list-style-type: none"> <li><a href="#">单行数据操作</a></li> <li><a href="#">多行数据操作</a></li> </ul>
数据版本和生命周期	使用数据版本以及数据生命周期（TTL）功能，您可以有效的管理数据，减少数据存储空间，降低存储成本。	<a href="#">数据版本和生命周期</a>
主键列自增	设置非分区键的主键列为自增列后，在写入数据时，无需为自增列设置具体值，表格存储会自动生成自增列的值。该值在分区键级别唯一且严格递增。	<a href="#">主键列自增</a>
条件更新	只有满足条件时，才能对数据表中的数据进行更新；当不满足条件时，更新失败。	<a href="#">条件更新</a>
局部事务	创建数据范围在一个分区键值内的局部事务。对局部事务中的数据进行读写操作后，可以根据实际提交或者丢弃局部事务。	<a href="#">局部事务</a>
原子计数器	将列当成一个原子计数器使用，对该列进行原子计数操作，可用于为某些在线应用提供实时统计功能，例如统计帖子的PV（实时浏览量）等。	<a href="#">原子计数器</a>
过滤器	在服务端对读取的结果再进行一次过滤，根据过滤器中的条件决定返回哪些行。由于只返回符合条件的数据行，所以在大部分场景下，可以有效降低网络传输的数据量，减少响应时间。	<a href="#">过滤器</a>
二级索引	<p>通过创建一张或多张索引表，使用索引表的主键列查询，相当于把数据表的主键查询能力扩展到了不同的列。二级索引包括全局二级索引和本地二级索引。</p> <ul style="list-style-type: none"> <li>全局二级索引：以异步方式将数据表中被索引的列和主键列的数据自动同步到索引表中，正常情况下同步延迟达到毫秒级别。</li> <li>本地二级索引：以同步方式将数据表中被索引的列和主键列的数据自动同步到索引表中，当数据写入数据表后，即可从索引表中查询到数据。</li> </ul>	<ul style="list-style-type: none"> <li><a href="#">二级索引</a></li> <li><a href="#">全局二级索引</a></li> <li><a href="#">本地二级索引</a></li> </ul>

功能特性	描述	相关文档
多元索引	多元索引基于倒排索引和列式存储，可以解决大数据的复杂查询难题，包括非主键列查询、全文检索、前缀查询、模糊查询、多条件组合查询、嵌套查询、地理位置查询、统计聚合（max、min、count、sum、avg、distinct_count、group_by）、并发导出数据等功能。	<ul style="list-style-type: none"> <li>多元索引</li> <li>使用控制台</li> <li>使用命令行工具</li> <li>使用SDK</li> </ul>
SQL查询	SQL查询功能为大数据引擎提供统一的访问接口。通过SQL查询功能，您可以对表格存储中数据进行复杂的查询和高效的分析。使用SQL查询数据时，您还可以配合索引来优化查询。	<ul style="list-style-type: none"> <li>SQL查询</li> <li>使用控制台</li> <li>使用SDK</li> <li>使用JDBC <ul style="list-style-type: none"> <li>JDBC连接表格存储</li> <li>通过Hibernate使用</li> <li>通过MyBatis使用</li> </ul> </li> <li>使用Go语言驱动</li> </ul>
通道服务	表格存储提供了增量、全量、增量加全量三种类型的分布式数据实时消费通道，可以实现对表中历史存量 and 新增数据的消费处理。	<ul style="list-style-type: none"> <li>通道服务</li> <li>快速入门</li> <li>使用SDK</li> </ul>
数据湖投递	表格存储数据湖投递可以全量备份或实时投递数据到数据湖OSS中存储，以满足更低成本的历史数据存储，以及更大规模的离线和准实时数据分析需求。	<ul style="list-style-type: none"> <li>数据湖投递</li> <li>快速入门</li> <li>使用SDK</li> </ul>
数据可视化	支持对接数据可视化工具DataV或者Grafana。通过对接数据可视化工具可以实现可视化展示表格存储中的数据。	<ul style="list-style-type: none"> <li>对接Grafana</li> <li>对接DataV</li> </ul>
监控与报警	通过查看表格存储资源的监控信息，您可以了解资源的使用情况。通过为资源的重要监控指标设置报警规则，您还可以在第一时间得知指标异常并快速处理异常。	<ul style="list-style-type: none"> <li>查看监控数据</li> <li>配置监控指标报警</li> </ul>
备份与恢复	通过混合云备份HBR（Hybrid Backup Recovery）定期备份表格存储实例中的数据，并在数据丢失或受损时及时恢复。HBR支持全量与增量数据备份，同时支持数据冗余机制，可以提高存储库的数据可靠性。	<ul style="list-style-type: none"> <li>备份Tablestore数据</li> <li>恢复Tablestore数据</li> </ul>
HBase支持	开源HBase API的Java应用可以通过Tablestore HBase Client直接访问表格存储服务。	<ul style="list-style-type: none"> <li>HBase支持</li> <li>快速入门</li> </ul>

## 时序模型

时序模型支持的功能特性请参见下表。

功能特性	描述	相关文档
表操作	支持列出实例中的全部时序表、创建一张时序表、查询时序表的配置信息、更新时序表的配置信息以及删除一张时序表。	
读写时序数据	将时序数据批量写入一张时序表中。数据写入后，您可以通过指定时间线标识来查询一条时间线在某段时间范围内的数据。	<ul style="list-style-type: none"> <li>使用控制台</li> <li>使用命令行工具</li> <li>使用SDK</li> </ul>
时间线检索	检索一张时序表中的时间线，检索条件支持多种条件组合。检索到时间线后，您可以通过调用接口进一步查询该时间线中的数据。	
SQL查询分析	<p>时序表支持通过SQL进行查询，SQL中支持通过指定时间线的元数据条件筛选时间线以及通过统计聚合操作按照不同维度对数据进行聚合操作。</p> <p>此外，SQL还支持仅对时间线的元数据进行查询，方便通过SQL进行时间线的元数据管理。</p>	使用SQL查询时序数据
对接Grafana	表格存储的表数据接入Grafana后，Grafana可以根据表数据生成大盘面板，将数据实时展示给需要的用户。	对接Grafana

## 消息模型

消息模型支持的功能特性请参见下表。

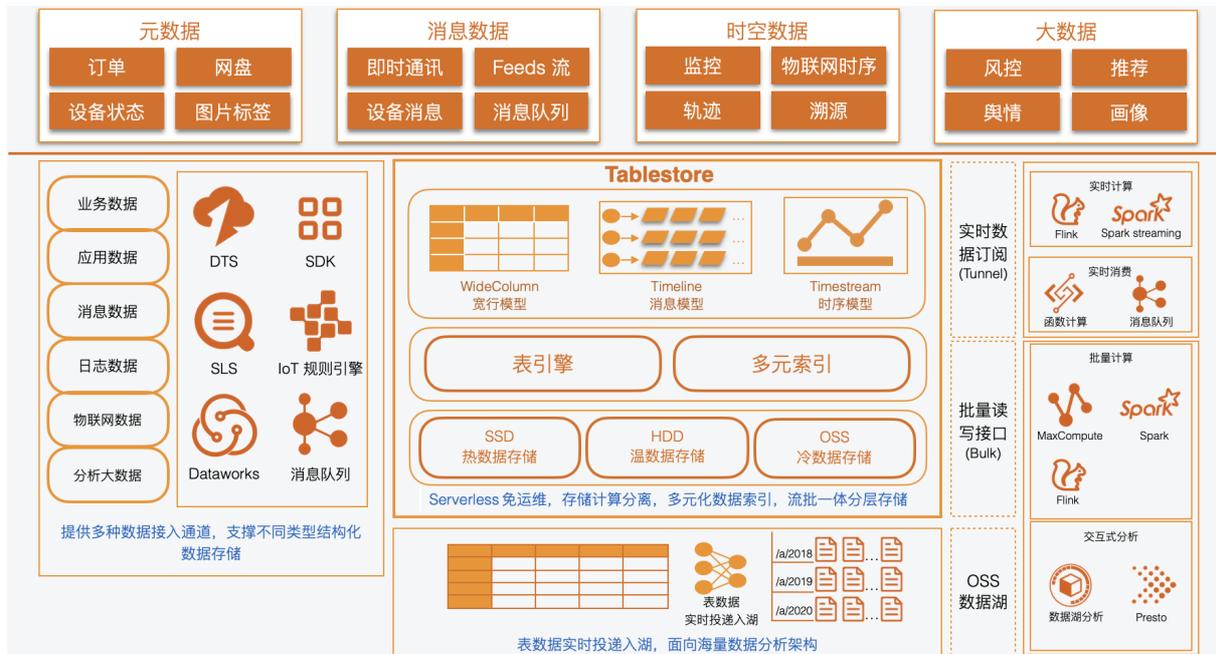
功能特性	描述	相关文档
表操作	<ul style="list-style-type: none"> <li>创建或者删除Meta表及其索引。</li> <li>创建或者删除Timeline表及其索引。</li> </ul>	表操作
Meta管理	Meta管理提供了增、删、改、单行读、多条件组合查询等接口。	Meta管理
Timeline管理	Timeline管理提供了消息模糊查询、多条件组合查询接口。	Timeline管理
Queue管理	Queue是单存储库下单Identifier对应的消息队列的管理实例，主要有同步写、异步写、批量写、删、同步改、异步改、单行读、范围读等接口。	Queue管理

# 3. 产品架构

本文介绍了表格存储（Tablestore）的系统架构和典型应用架构。

## 系统架构

表格存储的架构如下图所示。



## 业务场景

表格存储适用于元数据、消息数据、时空数据、大数据等场景下的系统搭建。

## 数据接入

表格存储提供SDK、DataWorks、IoT 规则引擎等多种数据接入方式，支撑应用数据、消息数据、物联网数据等不同类型结构化数据的存储。

## Tablestore

- 多模型数据存储

表格存储针对不同类型的数据提供了宽表（Wide column）模型、时序（TimeSeries）模型和消息（Timeline）模型三种数据存储模型。

模型	描述
宽表模型	类Bigtable/HBase模型，可应用于元数据、大数据等多种场景，支持数据版本、生命周期、主键列自增、条件更新、局部事务、原子计数器、过滤器等功能。更多信息，请参见 <b>宽表模型</b> 。
时序模型	针对时间序列数据的特点进行设计的模型，可应用于物联网设备监控、设备采集数据、机器监控数据等场景，支持自动构建时序元数据索引、丰富的时序查询能力等功能。更多信息，请参见 <b>时序模型</b> 。

模型	描述
消息模型	针对消息数据场景设计的模型，可应用于IM、Feed流等消息场景。能满足消息场景对消息保序、海量消息存储、实时同步的需求，同时支持全文检索与多维度组合查询。更多信息，请参见 <a href="#">消息模型</a> 。

● 多元化数据索引

除了支持主键查询，表格存储还支持二级索引和多元索引的索引方式，提供强大的数据查询能力。

索引类型	描述
数据表主键	数据表类似于一个巨大的Map，它的查询能力也就类似于Map，只能通过主键查询。
二级索引	通过创建一张或多张索引表，使用索引表的主键列查询，相当于把数据表的主键查询能力扩展到了不同的列。
多元索引	使用了倒排索引、BKD树、列存等结构，具备丰富的查询能力，例如非主键列的条件查询、多条件组合查询、地理位置查询、全文检索、模糊查询、嵌套结构查询、统计聚合等。

● 冷热分层存储

数据存储支持自动冷热分层，同时表格存储支持高性能实例和容量型实例两种实例规格来满足不同业务的数据存储需求。

实例规格	描述
高性能实例	适用于对读写性能和并发都要求非常高的场景，例如游戏、金融风控、社交应用、推荐系统等。
容量型实例	适用于对读性能不敏感，但对成本较为敏感的业务，例如日志监控数据、车联网数据、设备数据、时序数据、物流数据、舆情监控等。

● 数据湖投递

将表数据全量备份或实时投递数据到数据湖OSS中存储。投递的数据兼容开源生态标准，按照Parquet列存储格式存储，兼容Hive命名规范。您可以使用数据湖分析DLA和E-MapReduce直接对投递到OSS的数据进行外表分析。

## 计算生态对接

- 支持对接主流开源流批计算引擎，包括Flink、Spark、Presto等。
- 与阿里大数据平台生态组件有较完善的对接，包括DataWorks、DataHub、MaxCompute等。

## 典型应用架构

根据使用场景不同，表格存储有互联网应用架构、数据湖架构和物联网架构三种典型应用架构。

### 互联网应用架构

互联网应用架构包括数据库分层架构和分布式结构化数据存储架构，主要用于电商订单、直播弹幕、网盘中文件元数据、社交网络中即时通讯等场景。

- 数据库分层架构

在数据库分层架构中，使用Tablestore配合MySQL来完成应用系统的业务需求，利用MySQL的事务能力来处理对事务强需求的写操作与部分读操作，利用Tablestore的数据检索能力和大数据存储来实现数据存储、查询与分析。



- 分布式结构化数据存储架构

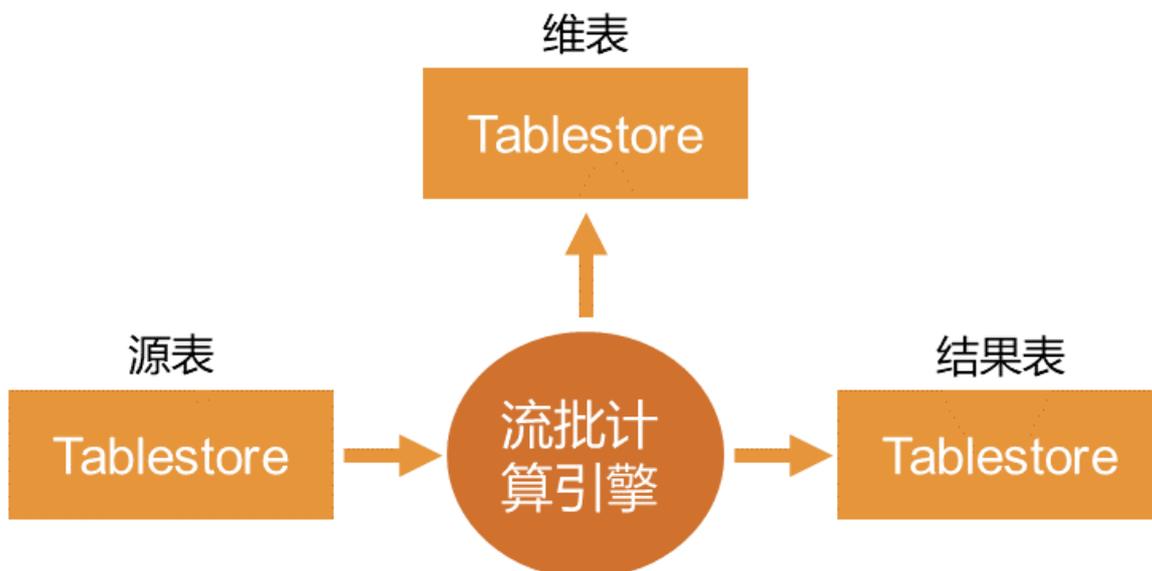
在分布式结构化数据存储架构中，Tablestore直连应用系统实现简单的事务处理和高并发数据读写。



### 数据湖架构

数据湖架构主要用于数据中台、推荐系统、风控系统等场景。

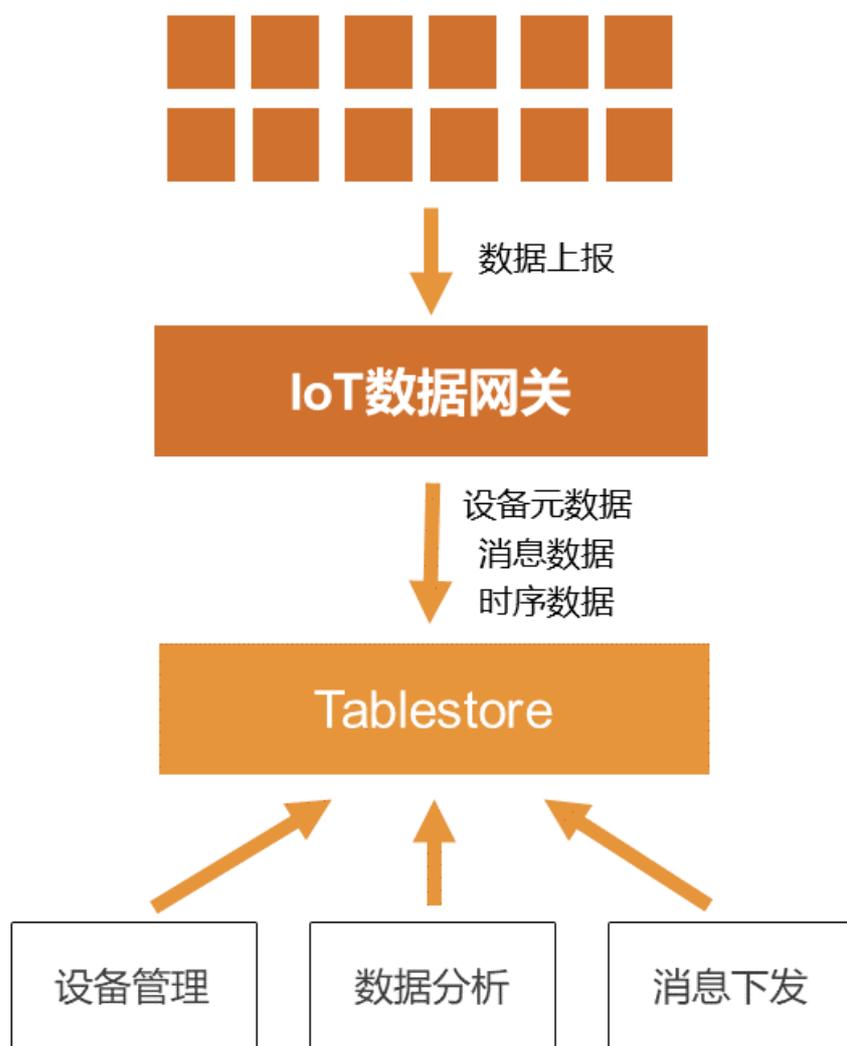
在数据湖架构中，表格存储作为源表、结果表或者维表对接流批计算引擎实现大数据计算与分析。



## 物联网架构

物联网架构主要用于车联网、智能家电、工业物联网、物流等场景。

在物联网架构中，表格存储作为IoT基础设施中的统一数据存储平台来存储物联网平台相关的时序数据、元数据、消息数据等，并提供丰富的数据分析处理能力。



## 4. 产品优势

本文介绍了表格存储的优势，包括多模型数据存储、多元化数据索引、多计算生态接入、访问安全性等。

### 多模型数据存储

表格存储支持宽表（WideColumn）模型、时序（TimeSeries）模型、消息（Timeline）等多种数据存储模型，能实现多种类型数据的一体化存储。

- **宽表模型**：经典模型，目前绝大部分半结构化、结构化数据均使用宽表模型进行存储。
- **时序模型**：适用于时序数据、时空数据等核心数据场景。
- **消息模型**：表格存储自研模型，主要用于消息数据，适用于IM、Feed和物联网设备消息下推等消息系统中消息的存储和同步。

### 多元化数据索引

除了支持主键查询，表格存储还支持二级索引和多元索引的索引方式，提供强大的数据查询能力。

- **二级索引**：相当于给数据表提供了另外一种排序方式，即对查询条件预先设计了一种数据分布，可加快数据查询的效率。
- **多元索引**：基于倒排索引和列式存储，支持多字段自由组合查询、模糊查询、地理位置查询、全文检索等，可解决大数据的复杂查询难题。

### 多计算生态接入

- 表格存储支持接入开源生态体系与阿里自研生态体系。
- 表格存储支持对接MaxCompute、Spark等批计算以及通过实时数据通道对接Flink流计算。

### 访问安全性

- 表格存储提供多种权限管理机制，并对每一次请求都进行身份认证和鉴权，以防止未授权的数据访问，确保数据访问的安全性。
- 支持数据访问权限管理，包括登录权限、创建表权限、读写权限、白名单控制权限等。

### 无缝扩展

表格存储通过数据分片和负载均衡技术，实现了存储无缝扩展。随着表数据量的不断增大，表格存储会进行数据分区的调整从而为该表配置更多的存储。表格存储可支持不少于10 PB数据存储量，单表可支持不少于1 PB数据存储量或1万亿条记录。

### 高可靠

表格存储将数据的多个备份存储在不同机架的不同机器上，并会在备份失效时进行快速恢复，提供99.99999999%（10个9）的可靠性。

### 数据一致性强

表格存储保证数据写入强一致，并保证数据3副本均写入磁盘，且所有数据保持一致。写操作一旦返回成功，应用程序就能立即读到最新的数据。

### 高并发读写

表格存储支持千万级并发读写能力。

### 运维便捷

使用表格存储，您只需专注于业务研发，无需担心软硬件预置、配置、故障、集群扩展、安全等问题，在保证高服务可用性的同时，极大地减少了管理及运维成本。

## 5. 应用场景

表格存储有互联网应用架构（包括数据库分层架构和分布式结构化数据存储架构）、数据湖架构和物联网架构三种典型应用架构。本文结合表格存储的典型应用架构介绍了不同应用架构下的应用场景。

### 互联网应用

目前互联网已在日常生活中广泛应用，人们可以在互联网上购物、社交、游戏等。表格存储互联网应用架构能够满足互联网中大部分应用的业务需求，典型应用场景包括历史订单数据场景、IM场景和Feed流场景。

#### 历史订单数据场景

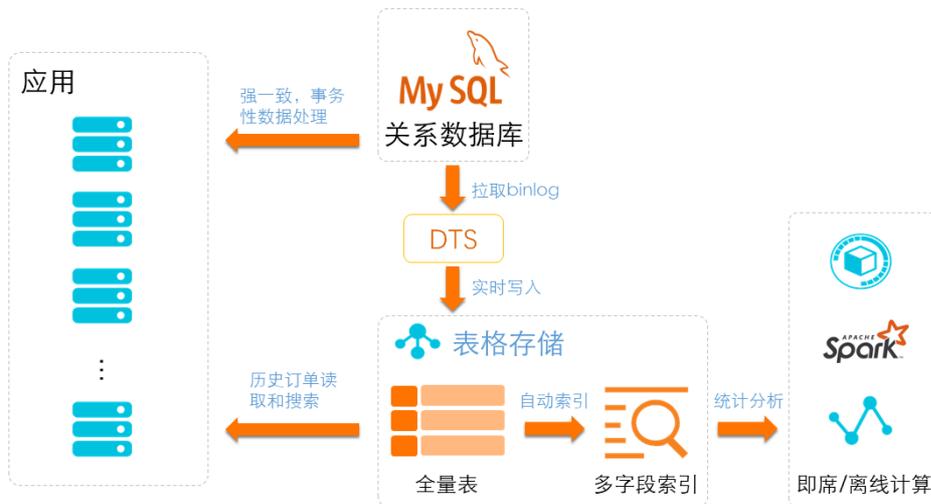
订单系统是一个非常通用的系统，存在于各行各业，例如电商订单、银行流水、运营商话费账单等。随着互联网的发展以及各企业对数据的重视，需要存储和持久化的订单量越来越大。传统关系型数据能够解决需要支持强一致的事务的在线业务，但海量的订单关系型数据无法保存全量数据，因此需要支持数据分层存储。

订单数据场景中的核心需求如下：

- 在线数据同步：实时数据与历史数据分层存储，需要支持实时同步在线业务。
- 历史数据存储：历史订单数据存储，需要支持低延迟数据点查与搜索以及高性价比海量数据存储。
- 数据分析：针对历史库进行报表统计分析，需要支持计算组件实现分析统计。

订单数据场景中可以使用表格存储互联网应用架构中的数据库分层架构来实现。场景架构如下图所示。

表格存储作为关系数据库的存储补充用于存储历史订单数据并通过DTS实时同步关系数据库中的数据，同时支持使用索引实现多字段组合查询以及支持对接流批计算引擎实现数据统计分析。



#### IM场景

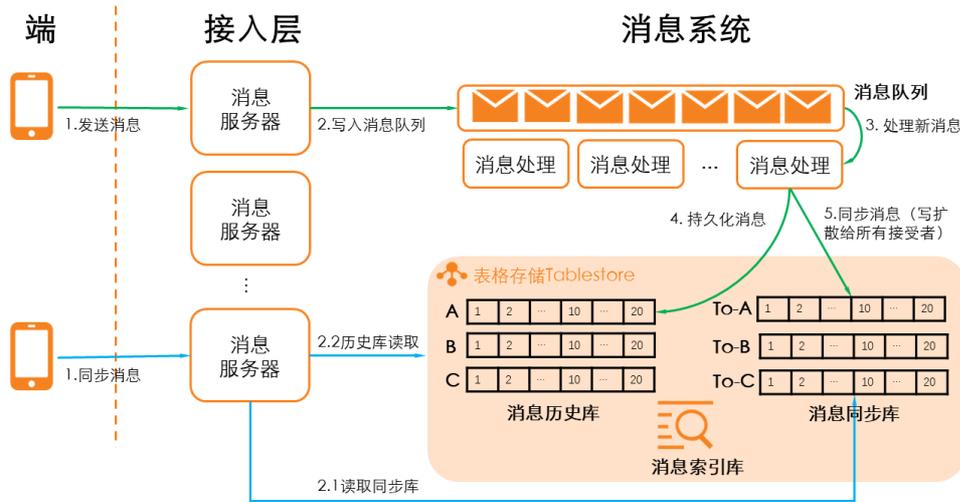
IM (Instant Messaging, 即时通讯) 已成为当前互联网业务的基础组件，广泛应用在社交、游戏、直播等场景，具有数据量大、实时性要求高、数据增长快等特点，因此需要支持海量消息的存储、同步和检索。

IM场景中的核心需求如下：

- 消息历史库：按照对话存储历史消息，需要存储海量数据且存储易拓展。
- 消息同步库：按照接收者存储同步消息，需要支持高并发写入以及实时拉取（写扩散）。
- 消息索引：对历史库数据进行数据检索，需要支持数据更新同步。

IM场景中可使用表格存储互联网应用架构中的分布式结构化数据存储架构来实现。场景架构如下图所示。

表格存储专为IM场景和Feed流场景设计了Timeline轻量级消息模型，能够支持同步表百TB存储、存储表PB级存储、每秒百万写扩散消息写入以及毫秒级同步库拉取。



### Feed流场景

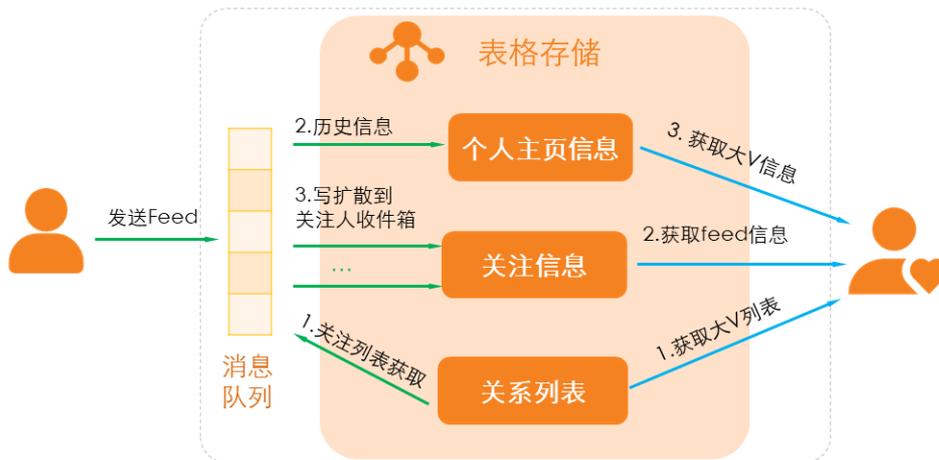
Feed流成为社交、媒体、新闻等领域的标准信息传递形态，产生了朋友圈、微博、头条等主流产品。由于Feed流场景的读写比一般为100:1，往往采用推模式，因此需要支持高并发主键自增消息写入。

Feed流场景的核心需求如下：

- 个人主页信息存储：按照发布者存储历史消息，需要支持海量数据存储以及内容检索。
- 关注信息存储：按照关注者存储消息数据，需要支持高并发写入以及实时拉取（写扩散）。
- 关系列表：存储关注关联关系，由于每次查看发布均会使用，需要支持数据快速更新查询。

Feed流场景中可使用表格存储互联网应用架构中的分布式结构化数据存储架构来实现。场景架构如下图所示。

表格存储专为IM场景和Feed流场景设计了Timeline轻量级消息模型，能够支持同步表百TB存储、存储表PB级存储、每秒百万写扩散消息写入以及毫秒级同步库拉取。



### 大数据

大数据具有数据规模大、数据类型多样、生成速度快、价值巨大但密度低的特点，如何高性价比存储海量持续增长的数据以及快速从海量数据中获取有价值的信息成为大数据领域的关注重点。表格存储数据湖架构能有效的解决大数据领域面临的数据存储与数据分析问题，典型应用场景包括推荐系统和舆情&风控分析场景。

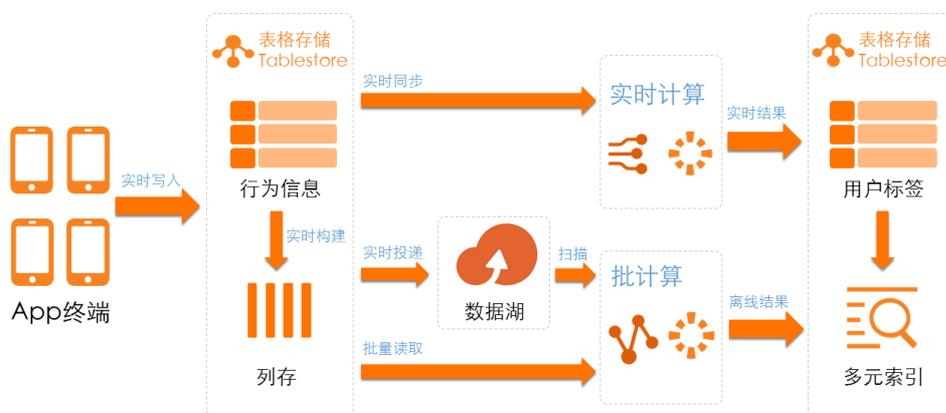
### 推荐系统

推荐系统作为当前所有业务精细化运营的主要手段，广泛应用在电商、短视频、新闻等场景，具有数据量大、实时更新、个性化推荐等特点，因此需要支持海量消息存储以及实时与离线分析。

推荐系统的核心需求如下：

- 行为日志：存储客户端写入实时数据，需要支持高并发写入以及对接流式计算进行实时分析。
- 历史数据：冷数据同步到OSS数据湖存储，需要支持数据投递、便于数据分层。
- 用户标签：对分析标签与推荐信息进行存储，需要支持属性列横向拓展以及高效检索。

推荐系统场景中可使用表格存储的数据湖架构来实现。场景架构如下图所示。



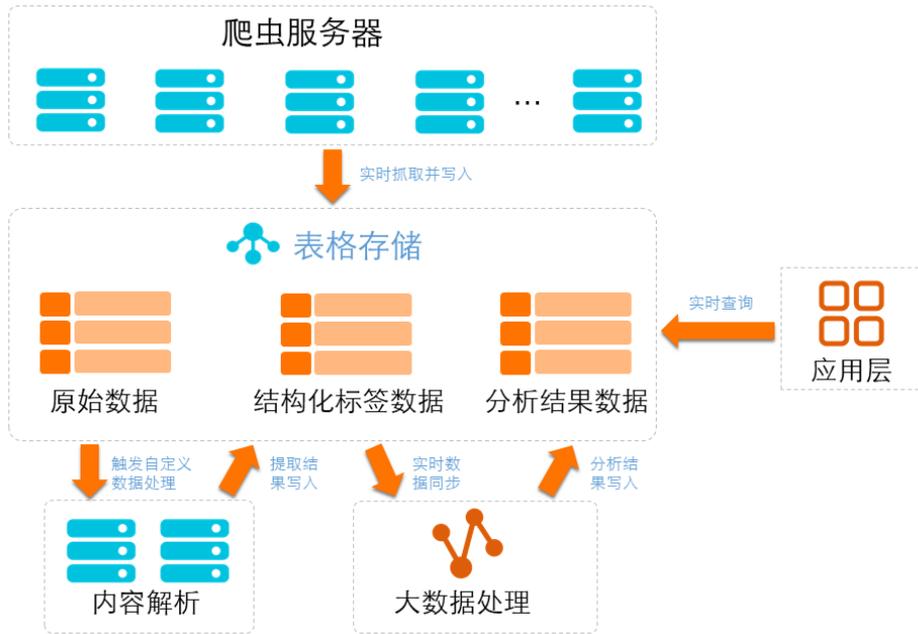
### 舆情&风控分析（数据爬虫）场景

通过对舆情信息的分析与把控，可以有效的分析与洞察市场，例如针对点评、新闻、评论等信息的收集分析，需要丰富的多类数据高并发写入以及便捷的数据流转进行计算分析。

舆情&风控分析场景中的核心需求如下：

- 原始数据写入存储：海量数据爬虫，需要支持高并发写入能力与PB级存储。
- 多数据类型存储：爬取的内容与生成的标签种类丰富，需要支持写入Schema-Free。
- 数据分析：对数据分阶段处理原始信息，生成结构化标签并存储结果，需要支持对接实时计算与离线计算。

舆情&风控分析场景中可使用表格存储的数据湖架构来实现。场景架构如下图所示。



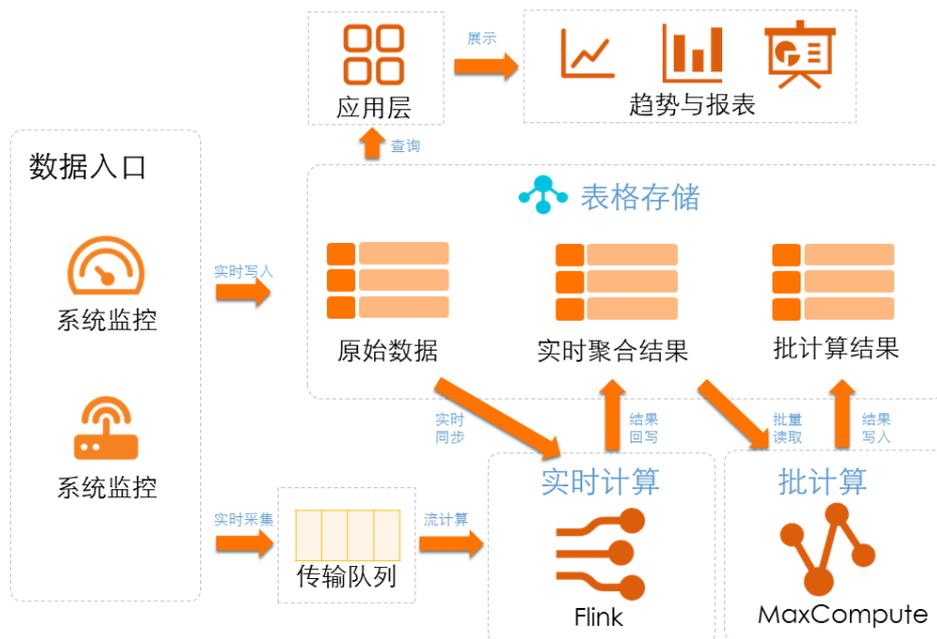
### 物联网

对系统的运维监控以及对物联网（Internet of Things，简称IoT）场景中环境与人的监控均有助于进行事实理解与决策，因此需要支持众多设备与系统的高并发写入与数据存储以及决策分析。

监控&IoT场景的核心需求如下：

- 数据高并发写入：面向众多设备与系统，需要支持百万级节点实时写入。
- 数据实时聚合：对原始数据监控预聚合，降低精度，需要支持数据实时同步对接流计算。
- 数据存储：长期保存数据，需要支持单表规模极大以及高性价比存储。

监控&IoT场景中可使用表格存储的物联网架构来实现。场景架构如下图所示。



## 6. 常见问题

本文列举了阿里云表格存储的常见问题，帮助您快速了解表格存储。

### 一般性常见问题

- 什么是表格存储？

表格存储是阿里云自研的NoSQL多模型数据库，提供海量结构化数据存储以及快速的查询和分析服务。表格存储的分布式存储和强大的索引引擎能够支持PB级存储、千万TPS以及毫秒级延迟的服务能力。具有全托管、多模型、无限存储等特点，具体请参见[什么是表格存储](#)。

- 为什么选择表格存储？

表格存储具有以下优势：

- 全托管

表格存储是一种全托管的数据库。使用表格存储您只需专注于业务研发，无需担心软硬件预置、配置、故障、集群扩展、安全等问题，在保证高服务可用性的同时，极大地减少了管理及运维成本。

- 模型丰富

表格存储支持多种数据库模型，包括Wide column、Timeline、Timestream、Grid。其中，Wide column模型是一款经典模型，目前绝大部分半结构化、结构化数据都存储在Wide column模型系统中；Timeline模型是表格存储自研的模型，主要用于消息数据，适用于IM、Feed和物联网设备消息下推等消息系统中消息的存储和同步，目前已被广泛使用；Timestream模型可应用于时序数据、时空数据等核心数据场景。

- 无缝扩展

表格存储通过数据分片和负载均衡技术，实现了存储无缝扩展。随着表数据量的不断增大，表格存储会进行数据分区的调整从而为该表配置更多的存储。

- 查询能力强

除了支持主键查询，表格存储还支持多元索引、全局二级索引。全局二级索引相当于给主表提供了另外一种排序方式，即对查询条件预先设计了一种数据分布，可加快数据查询的效率。多元索引基于倒排索引和列式存储，支持多字段自由组合查询、模糊查询、地理位置查询、全文检索等，可解决大数据的复杂查询难题。

- 高可靠

表格存储将数据的多个备份存储在不同机架的不同机器上，并会在备份失效时进行快速恢复，提供99.99999999%（10个9）的可靠性。

- 数据强一致

表格存储保证数据写入强一致，并保证数据3副本均写入磁盘，且所有数据保持一致。写操作一旦返回成功，应用程序就能立即读到最新的数据。

- 表格存储是否有存储和扩展限制？

表格存储没有存储和扩展限制。表格存储通过数据分片和负载均衡技术，实现了存储无缝扩展。随着表数据量的不断增大，表格存储会自动进行数据分区的调整从而为该表配置更多的存储。

- 如何设计表格存储的表？

合理的表设计能够发挥出数据库的最优性能，关于如何设计表格存储的表，参见[海量结构化数据存储技术揭秘：Tablestore表设计最佳实践](#)。

## 运维管理

表格存储如何运维，运维成本高么？

表格存储是一种全托管的数据库。表格存储可以自动根据您的业务量的增长进行扩容，还可自动进行故障检测和迁移，对您的应用程序屏蔽机器和网络的硬件故障，为您完成一切运维任务。使用表格存储您只需专注于业务研发，无需担心软硬件预置、配置、故障、集群扩展、安全等问题，在保证高服务可用性的同时，为您省去运维成本。

## 使用场景

表格存储有哪些主要使用场景？

表格存储适用于以下场景：

- 元数据

用户存储海量的文档、媒体文件等数据的同时，对文件元数据的存储和分析不可或缺。此外，电商的订单、银行流水、运营商话费账单也需要存储及分析大量的元数据。表格存储单表支持PB级存储、千万QPS，以及多种索引方式（全局二级索引、全文索引、倒排索引以及时空索引），满足不同场景在线的不同的查询需求，可以帮助您轻松实现高效的元数据管理。

- 消息数据

表格存储自研的Timeline模型主要用于消息数据，能够抽象出支撑海量Topic的轻量级消息队列，可以存储大量社交信息，包括IM聊天，以及评论、跟帖和点赞等Feed流信息，接口简单易用。目前表格存储Timeline模型已被应用在众多IM系统中，例如支撑钉钉海量消息同步等。此外，表格存储采用按量付费，能够以较低的成本满足访问波动明显、高并发、低延时的需求。

- 轨迹溯源

表格存储提供了面向轨迹类场景的Timestream模型，可提供PB级存储、千万TPS以及毫秒级延迟的服务能力、以及多种索引方式（全局二级索引、全文索引、倒排索引以及时空索引）。使用表格存储Timestream模型，您可以轻松管理、分析跑步、骑行、健走、外卖等轨迹数据。

- 科学大数据

多维网格数据是一种科学大数据，在地球科学领域（气象、海洋、地质、地形等）应用非常广泛，且数据规模也越来越大。相关的科学工作者有快速浏览数据的需求以及在线查询的需求，查询种类丰富、延迟要求高。表格存储是一款阿里云自研的分布式NoSQL服务，可以提供超大规模的存储容量，支撑超大规模的并发访问和低延迟的性能，可以轻松解决科学大数据的海量存储规模和查询性能问题。

- 互联网大数据

热点新闻及娱乐八卦可以在短短数分钟内，有数万计转发，数百万的阅读，如何能够实时的把握民情并作出对应的处理对很多企业来说都是至关重要的。此外，商品在各类电商平台的订单量，用户的购买评论也都对后续的消费者产生很大的影响。商家的产品设计者需要汇总统计和分析各类平台的数据做为依据，决定后续的产品发展，公司的公关和市场部门也需要根据舆情作出相应的及时处理。表格存储单表提供PB级存储、千万QPS，以及多种索引方式，可以帮助您轻松实现百亿级互联网舆情存储及分析。

- 物联网

表格存储单表提供PB级数据存储规模，无需分库分表，同时支持千万QPS，可以轻松满足IoT设备、监控系统等时序数据的存储需求，大数据分析SQL直读以及高效的增量流式读接口让数据轻松完成离线分析与实时流计算。

表格存储提供了各个场景下的经典使用案例供您参考，具体请参见[快速玩转Tablestore入门与实践](#)。

## 计费

表格存储的费用是多少？

---

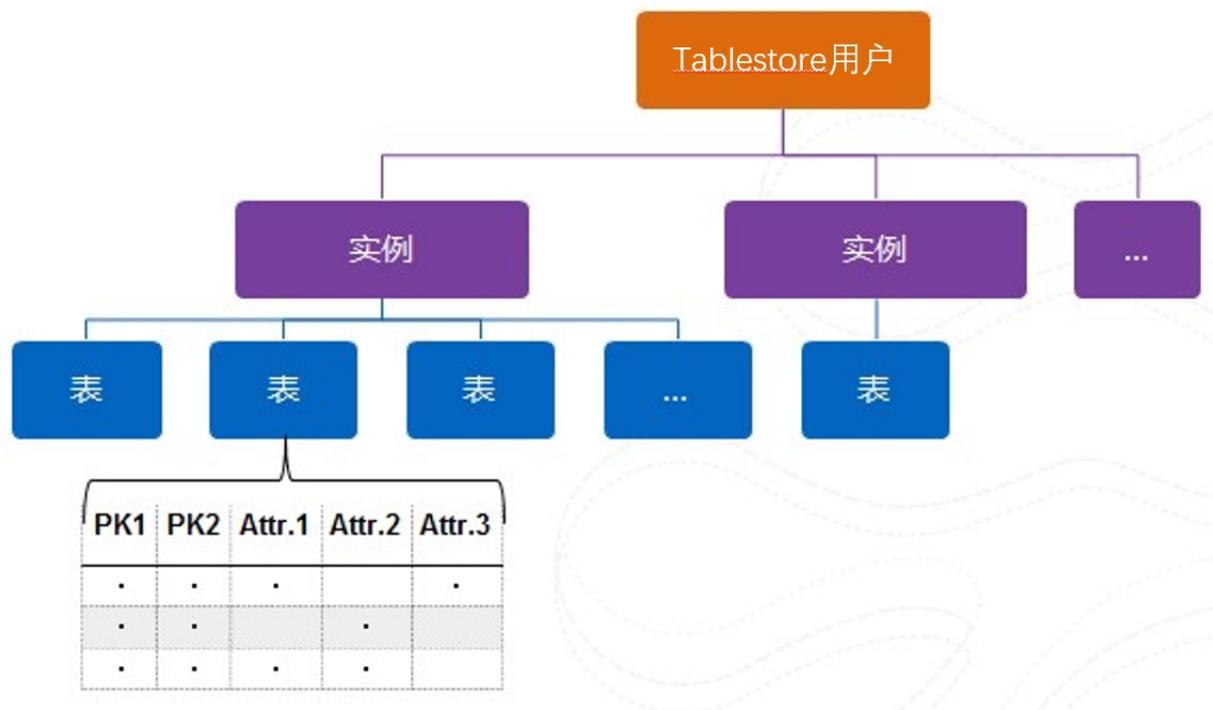
使用表格存储，您可以按实际使用量付费，先使用，后付费，以较低的成本满足访问波动明显大并发低延时的需要。您也可以预先购买资源包，之后使用资源时，扣除相应的额度。详细的价格信息请参见[阿里云产品定价](#)。

# 7.基础概念

## 7.1. 实例

实例 (Instance) 是您使用和管理表格存储服务的实体，每个实例相当于一个数据库。表格存储对应用程序的访问控制和资源计量都在实例级别完成。开通表格存储服务后，您需要通过表格存储管理控制台来创建实例，然后在实例内进行表的创建和管理。

每个云账号最多创建10个实例，每个实例内最多创建64张表。如果您需要增加限额，请[提交工单](#)。



### 实例规格

表格存储支持高性能实例和容量型实例两种实例规格。两种规格都能够支持单表PB级别的数据量，主要区别在于使用成本及适用场景，具体说明请参见下表。

**注意** 创建实例时请谨慎选择规格类型，创建后无法修改。

类型/对比项	使用场景	计费类型	读性能	写性能	并发力
高性能实例	适用于对读写性能和并发都要求非常高的场景，例如游戏、金融风控、社交应用、推荐系统等。	<ul style="list-style-type: none"> <li>预留读吞吐量或预留写吞吐量</li> <li>按量读吞吐量或按量写吞吐量</li> </ul>	高	高	高

类型/对比项	使用场景	计费类型	读性能	写性能	并发力
容量型实例	适用于对读性能不敏感，但对成本较为敏感的业务，例如日志监控数据、车联网数据、设备数据、时序数据、物流数据、舆情监控等。	按量读吞吐量或按量写吞吐量	中	高	中

## 实例命名规范

实例的名称在一个地域内必须唯一，不同的地域内实例名称可以相同。具体命名规范如下。

- 必须由英文字母、数字或短划线 (-) 组成。
- 首字符必须为英文字母。
- 末尾字符不能为短划线 (-)。
- 大小写不敏感。
- 长度在3~16字符之间。
- 实例名称不能包含['ali','ay','ots','taobao','admin']这几个单词。

## 各地域实例规格支持情况

地域名称	高性能实例	容量型实例
华东1（杭州）	支持	支持
华东1 金融云	支持	暂不支持
华东2（上海）	支持	支持
华东2 金融云	暂不支持	支持
华北1（青岛）	暂不支持	支持
华北2（北京）	支持	支持
华北3（张家口）	支持	支持
华北5（呼和浩特）	暂不支持	支持
华南1（深圳）	支持	支持
西南1（成都）	支持	支持
中国（香港）	支持	支持
新加坡	支持	暂不支持
澳大利亚（悉尼）	暂不支持	支持

地域名称	高性能实例	容量型实例
马来西亚（吉隆坡）	支持	支持
印度尼西亚（雅加达）	支持	支持
日本（东京）	暂不支持	支持
德国（法兰克福）	支持	支持
英国（伦敦）	支持	支持
美国（硅谷）	支持	暂不支持
美国（弗吉尼亚）	支持	支持
印度（孟买）	支持	支持
阿联酋（迪拜）	暂不支持	支持
菲律宾（马尼拉）	支持	暂不支持

## 7.2. 服务地址

每个表格存储实例对应一个服务地址（Endpoint），应用程序进行表和数据操作时需要指定服务地址。不同访问场景下需使用相应的服务地址格式。

 说明 各个地域对应的英文表示参考[地域](#)。

### 获取服务地址

1. 登录[表格存储控制台](#)。
2. 在概览页面，单击实例名称或在操作列单击实例管理。

在实例详情页签的实例访问地址即是该实例的Endpoint。

### 公网地址

从公网访问表格存储时使用公网地址。

服务地址格式：

```
https://instanceName.region.ots.aliyuncs.com
```

例如，华东1（杭州）节点，实例名称为myInstance的服务地址为：

```
https://myInstance.cn-hangzhou.ots.aliyuncs.com
```

### VPC地址

从VPC网络的ECS服务器访问表格存储时使用VPC地址。

服务地址格式：

```
https://instanceName.region.vpc.tablestore.aliyuncs.com
```

例如，华东1（杭州）节点，应用程序通过VPC网络访问实例名称为myInstance的服务地址为：

```
https://myInstance.cn-hangzhou.vpc.tablestore.aliyuncs.com
```

## 私网地址

从同区域经典网络的ECS服务器访问表格存储时使用私网地址。应用程序从同区域的经典网络ECS服务器上通过内网访问表格存储，可以获得更低的响应延迟，且不产生外网流量。

服务地址格式：

```
https://instanceName.region.ots-internal.aliyuncs.com
```

例如，华东1（杭州）节点，实例名称为myInstance的服务地址为：

```
https://myInstance.cn-hangzhou.ots-internal.aliyuncs.com
```

## 7.3. 读写吞吐量

读/写吞吐量的单位为读服务能力单元和写服务能力单元，简称CU（Capacity Unit），是数据读写操作的最小计费单位。当通过表格存储的API对数据表进行读写操作时，会消耗对应的写服务能力单元和读服务能力单元。

- 1单位读服务能力单元表示从数据表中读一条4 KB数据。
- 1单位写服务能力单元表示向数据表写一条4 KB数据。
- 操作数据大小不足4 KB的部分向上取整，例如写入7.6 KB数据消耗2单位写服务能力单元，读出0.1 KB数据消耗1单位读服务能力单元。

### 预留读/写吞吐量

预留读/写吞吐量是高性能实例中数据表的一个属性。在创建数据表时，可以为数据表指定预留读/写吞吐量。

- 当设置的预留读/写吞吐量大于0时，表格存储会为数据表分配和预留相应的资源，每秒对数据表的访问不超过预留读/写吞吐量时将按照预留读/写吞吐量的单价计费。
- 当设置的预留读/写吞吐量等于0时，表格存储不会为数据表分配和预留相应的资源。

 **说明** 不存在的数据表将视作预留读/写吞吐量均为0，访问不存在的数据表将根据操作类型消耗1个按量读CU或者1个按量写CU。

由于预留读/写吞吐量在单价上低于按量读/写吞吐量，配置合适的预留读/写吞吐量可以进一步降低成本。例如数据表创建后如果需要导入大量数据，可以设置较大的预留写吞吐量，能够以较低的写成本导入数据，当数据导入完成后，再将预留读/写吞吐量下调。

使用限制

- 容量型实例下的数据表不支持预留读/写吞吐量。
- 当预留读/写吞吐量大于0时，即使没有读写请求也会进行计费，所以表格存储限制用户能够自行设置的单个数据表预留读/写吞吐量最大为100000（预留读吞吐量和预留写吞吐量分别不超过100000）。当用户对单个数据表的预留读/写吞吐量需求超出100000时，可以[提交工单](#)提高预留读/写吞吐量。

预留读/写吞吐量更新规则

通过UpdateTable接口可以动态修改表的预留读/写吞吐量配置。预留读/写吞吐量的更新规则如下。

- 每个自然日内（UTC时间00:00:00到第二天的00:00:00，北京时间早上8点到第二天早上8点），上调或者下调预留读/写吞吐量的总次数不做限制，一张数据表上的两次更新的间隔必须大于1分钟。
- 预留读/写吞吐量调整完成后1分钟内生效。

## 按量读/写吞吐量

按量读/写吞吐量是数据表在每秒实际消耗的读/写吞吐量中超出预留读/写吞吐量的部分，统计周期为1秒。每个小时内，表格存储对预留读/写吞吐量取平均值，对按量读/写吞吐量取累加值来作为用户实际消耗的吞吐量。

由于按量读/写吞吐量的模式无法预估需要为数据表预留的计算资源，表格存储需要提供足够的服务能力以应对突发的访问高峰，所以按量读/写吞吐量的单价高于预留读/写吞吐量的单价。合理设置数据表的预留读/写吞吐量能够有效地降低使用成本。

假如数据表设置的预留读吞吐量为100 CU，连续3秒的访问情况如下：

- T0：读操作实际消耗120 CU读吞吐量，则这1秒内预留吞吐量为100，消耗的按量读吞吐量为20 CU。
- T1：读操作实际消耗95 CU读吞吐量，则这1秒内预留吞吐量为100，消耗的按量读吞吐量为0 CU。
- T2：读操作实际消耗110 CU读吞吐量，则这1秒内预留吞吐量为100，消耗的按量读吞吐量为10 CU。

T0至T2时刻的消耗的读吞吐量为：100 CU预留读吞吐量以及30 CU按量读吞吐量。

 **说明** 由于按量读/写吞吐量无法准确估计需要预留的资源，在某些极端访问情况下，若单个分区键每秒的访问需要消耗10000 CU，表格存储可能会返回OTSCapacityUnitExhausted错误。此时需要使用退避重试等策略来减少访问该数据表的频率。

## 7.4. 地域

地域（Region）是指阿里云物理数据中心所在的位置。表格存储部署在多个地域中，您可以根据自身的业务需求创建不同地域中的表格存储实例。

表格存储当前支持的地域与RegionID的对应关系请参见下表。

地域名称	RegionID
华东1（杭州）	cn-hangzhou
华东1 金融云	cn-hangzhou-finance
华东2（上海）	cn-shanghai
华东2 金融云	cn-shanghai-finance-1
华北1（青岛）	cn-qingdao
华北2（北京）	cn-beijing
华北3（张家口）	cn-zhangjiakou
华北5（呼和浩特）	cn-huhehaote
华南1（深圳）	cn-shenzhen
西南1（成都）	cn-chengdu
中国（香港）	cn-hongkong
新加坡	ap-southeast-1

地域名称	RegionID
澳大利亚（悉尼）	ap-southeast-2
马来西亚（吉隆坡）	ap-southeast-3
印度尼西亚（雅加达）	ap-southeast-5
日本（东京）	ap-northeast-1
德国（法兰克福）	eu-central-1
英国（伦敦）	eu-west-1
美国（硅谷）	us-west-1
美国（弗吉尼亚）	us-east-1
印度（孟买）	ap-south-1
阿联酋（迪拜）	me-east-1
菲律宾（马尼拉）	ap-southeast-6

## 8.快速玩转Tablestore入门与实战

本文为您介绍表格存储的适用场景以及可借鉴的开发指导、经典案例。通过本文提供的场景和应用案例，您可以更好地了解表格存储使用方式、设计思路、架构实现等，为您的方案选型提供参考依据。

 说明 本文中的经典场景都提供相应的实现代码。更多信息，请参见[表格存储Git hub仓库](#)。

### 基础入门-五分钟入门

- [如何免费使用一款免运维、无限容量的表存储服务](#)
- [Tablestore数据模型-WideColumn和Timeline](#)
- [亿级消息系统的核心存储：Tablestore发布Timeline 2.0模型](#)
- [Tablestore Timestream：为海量时序数据存储设计的全新数据模型](#)
- [表格存储SQL功能快速上手](#)

### 技术进阶-全方位了解

- [从SQL到NoSQL—如何使用表格存储](#)
- [深入对比HBase与阿里云的表格存储服务](#)
- [表格存储数据模型和查询操作](#)
- [Tablestore存储和索引引擎详解](#)
- [详解Tablestore模糊查询——以订单场景为例](#)
- [多元索引：如何在千亿行规模的表中快速检索数据](#)
- [通道服务：表格存储全增量一体消费通道](#)
- [Tablestore表设计最佳实践](#)
- [Tablestore+DeltaLake（快速开始）](#)

### 架构原理-架构设计思想

- [数据中台之结构化大数据存储设计](#)
- [结构化大数据分析平台设计](#)
- [云上应用系统数据存储架构演进](#)
- [Lambda plus：云上大数据解决方案](#)

### 场景实战-典型场景架构与实现

#### 元数据

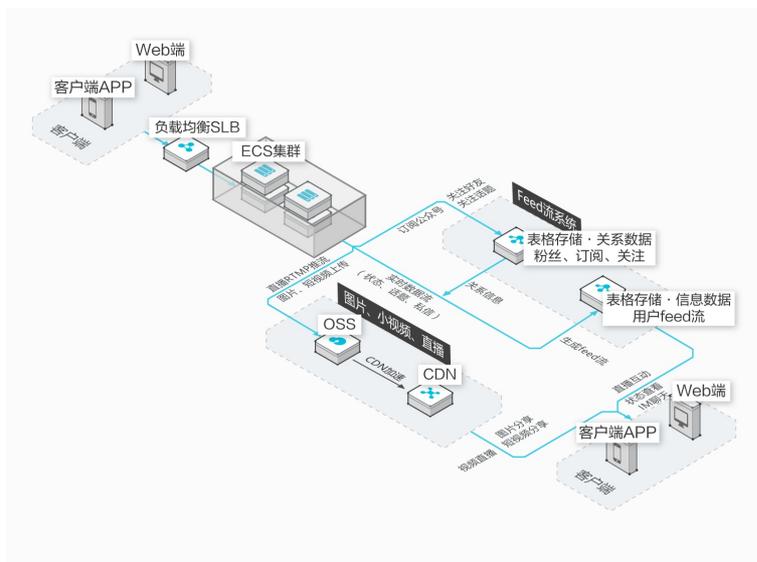
用户存储海量的文档、媒体文件等数据的同时，对文件元数据的存储和分析不可或缺。此外，电商的订单、银行流水、运营商话费账单也需要存储及分析大量的元数据。表格存储单表支持PB级存储、千万QPS，以及多种索引方式（全局二级索引、全文索引、倒排索引以及时空索引），满足不同场景在线的不同的查询需求，可以帮助您轻松实现高效的元数据管理。

- [基于MySQL+Tablestore分层存储架构的大规模订单系统实践-架构篇](#)
- [基于MySQL+Tablestore分层存储架构的大规模订单系统实践-数据同步DTS篇](#)
- [基于MySQL+Tablestore分层存储架构的大规模订单系统实践-数据同步Canal篇](#)
- [基于MySQL+Tablestore分层存储架构的大规模订单系统实践-订单搜索篇](#)

- [基于MySQL+Tablestore分层存储架构的大规模订单系统实践-SQL查询和分析](#)
- [基于MySQL+Tablestore分层存储架构的大规模订单系统实践-基于DLA的联邦查询](#)
- [基于MySQL+Tablestore分层存储架构的大规模订单系统实践-数据处理ETL篇](#)
- [基于MySQL+Tablestore分层存储架构的大规模订单系统实践-历史数据分析篇](#)
- [基于MySQL+Tablestore分层存储架构的大规模订单系统实践-数据流计算篇](#)
- [基于Tablestore打造亿量级订单管理解决方案](#)
- [基于Tablestore多元索引打造亿量级店铺搜索系统](#)
- [海量智能元数据管理系统实现解析](#)
- [基于Tablestore的海量保险单查询平台](#)

## 消息数据

表格存储自研的Timeline模型主要用于消息数据，能够抽象出支撑海量Topic的轻量级消息队列，可以存储大量社交信息，包括IM聊天，以及评论、跟帖和点赞等Feed流信息，接口简单易用。目前表格存储Timeline模型已被应用在众多IM系统中，例如支撑钉钉海量消息同步等。此外，表格存储采用按量付费，能够以较低的成本满足访问波动明显、高并发、低延时的需求。



- [现代IM系统中的消息系统架构-架构篇](#)
- [现代IM系统中的消息系统架构-模型篇](#)
- [现代IM系统中的消息系统架构-实现篇](#)
- [Tablestore Timeline：轻松构建千万级IM和Feed流系统](#)
- [Feed流系统设计-总纲](#)
- [如何打造千万级Feed流系统](#)
- [高并发IM系统架构优化实践](#)

## 轨迹溯源

表格存储提供了面向轨迹类场景的Timestream模型，可提供PB级存储、千万TPS以及毫秒级延迟的服务能力、以及多种索引方式（全局二级索引、全文索引、倒排索引以及时空索引）。使用表格存储Timestream模型，您可以轻松管理、分析跑步、骑行、健走、外卖等轨迹数据。

- [TableStore时序数据存储-架构篇](#)
- [基于Tablestore实现海量运动轨迹数据存储](#)

- [药品监管系统架构揭秘：海量溯源数据存储与查询](#)
- [基于Tablestore实现海量摩托车的轨迹管理](#)
- [超级快递——如何用系统来保证快递准时送达](#)

## 科学大数据

多维网格数据是一种科学大数据，在地球科学领域（气象、海洋、地质、地形等）应用非常广泛，且数据规模也越来越大。相关的科学工作者有快速浏览数据的需求以及在线查询的需求，查询种类丰富、延迟要求高。表格存储是一款阿里云自研的分布式NoSQL服务，可以提供超大规模的存储容量，支撑超大规模的并发访问和低延迟的性能，可以轻松解决科学大数据的海量存储规模和查询性能问题。

[基于Tablestore的海量气象格点数据解决方案实战](#)

## 互联网大数据

热点新闻及娱乐八卦可以在短短数分钟内，有数万计转发，数百万的阅读，如何能够实时的把握民情并作出对应的处理对很多企业来说都是至关重要的。此外，商品在各类电商平台的订单量，用户的购买评论也都对后续的消费者产生很大的影响。商家的产品设计者需要汇总统计和分析各类平台的数据做为依据，决定后续的产品发展，公司的公关和市场部门也需要根据舆情作出相应的及时处理。表格存储单表提供PB级存储、千万QPS，以及多种索引方式，可以帮助您轻松实现百亿级互联网舆情存储及分析。

- [Tablestore结合Spark的流批一体SQL实战](#)
- [Tablestore+Blink实战：交易数据的实时统计](#)
- [百亿级全网舆情分析系统存储设计](#)
- [海量监控日志基于EMR Spark Streaming SQL进行实时聚合](#)

## 物联网

表格存储单表提供PB级数据存储规模，无需分库分表，同时支持千万QPS，可以轻松满足IoT设备、监控系统等时序数据的存储需求，大数据分析SQL直读以及高效的增量流式读接口让数据轻松完成离线分析与实时流计算。

- [基于Tablestore的一站式物联网存储解决方案-场景篇](#)
- [基于Tablestore的一站式物联网存储解决方案-表设计篇](#)
- [基于Tablestore的一站式物联网存储解决方案-数据操作篇](#)
- [基于Tablestore的一站式物联网存储解决方案-Spark分析](#)
- [基于Tablestore的一站式物联网存储解决方案-数据湖分析](#)
- [基于Tablestore的一站式物联网存储解决方案-Flink实时计算](#)
- [基于Tablestore的共享汽车管理平台](#)
- [基于Tablestore的Wifi设备监管系统架构实现](#)
- [基于Tablestore的物联网元数据管理](#)
- [如何高效存储海量GPS数据](#)

## 控制台可视化样例

- [电商订单](#)
- [共享汽车](#)
- [快递轨迹追踪](#)
- [Wifi设备管理](#)
- [店铺搜索](#)
- [地理围栏](#)

- [智能元数据](#)

.

## 数据迁移同步

- [数据同步-从MySQL到Tablestore](#)
- [通过DataHub将本地的CSV文件导入Tablestore](#)
- [基于Tablestore Tunnel的数据复制实战](#)

## 产品技术服务

表格存储提供专业的免费的技术咨询服务，欢迎通过钉钉加入用户群11789671（表格存储技术交流群）或23307953（表格存储技术交流群-2）。

# 9. 产品技术服务

请根据实际需要联系技术支持或者申请专家服务进行技术交流和问题咨询。

## 技术支持

除了官网文档外，您还有更多途径深入了解表格存储。

- 阅读[快速玩转Tablestore入门与实践](#)。
- 关注阿里云[开发者社区](#)。
- 通过钉钉加入用户群11789671（表格存储技术交流群）或23307953（表格存储技术交流群-2）。

## 专家服务

表格存储专家服务提供架构咨询、技术答疑、费用评估、迁移护航等专家一对一支持。

## 申请方式

您可以根据业务场景需要通过[专家服务](#)申请表格存储专家一对一支持。申请通过后，我们会尽快通过手机号钉钉联系您。

## 适用场景

表格存储专家服务适用场景如下：

- 原有产品架构面临性能、扩展性等瓶颈，考虑使用表格存储优化或重构业务，应对业务发展的挑战。
- 新业务架构选型，期望详细了解产品适配度与推荐架构建议。
- 业务活动大促期间，迫切需要表格存储技术专家为业务大促活动的开展提供产品技术保障。

 **说明** 由于资源有限，表格存储专家服务团队会根据需求迫切程度、使用规模等方面评估和筛选产品技术保障需求。

## 服务方式

- 提供专属钉钉群，与表格存储技术专家直接沟通。
- 不定期现场拜访答疑。

## 服务团队

表格存储专家服务团队多年护航阿里巴巴双11活动，具有丰富的架构和运维经验。

- 表格存储技术专家，为您提供最准确的产品使用姿势。
- 存储、数据库领域专家，为您提供专业的系统架构设计建议。

## 服务内容

服务类型	服务内容
方案咨询与设计	针对用户的实际业务需求，提供表格存储技术架构选型、表结构设计、资源评估、权限安全等各方面专业建议，保障架构选型满足未来业务的快速发展。
数据迁移支持	根据不同的数据源和应用场景，推荐合适的迁移方案与配套迁移工具，保障业务的平滑迁移。

服务类型	服务内容
开发答疑支持	针对开发测试过程中产品的细节、SDK使用等问题，提供直接与产品开发工程师沟通的渠道，帮助您高效及时的获取所需准确信息，保障业务开发效率。
性能调优	针对产品过程中存在的毛刺高、延迟大等性能问题，提供分析、优化与使用建议，保障系统处于最佳状态。
项目护航	针对用户在关键业务时期例如活动大促、产品发布等，提供专家护航服务。表格存储专家服务团队会提前协助用户评估资源，参与前期产品压测和提供上线在线护航支持。
产品培训	产品最新功能更新、架构实践、业内信息动态等第一时间推送更新，提供定期的实践直播分享。