

表格存储服务

用户指南

文档版本 06
发布日期 2025-08-12



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

目录

1 创建用户并授权使用 CloudTable	1
2 使用 HBase	3
2.1 HBase 使用流程	3
2.2 创建 HBase 集群	4
2.3 连接 HBase 集群	8
2.3.1 准备弹性云服务器	8
2.3.2 使用 HBase Shell 连接 HBase 普通集群	12
2.3.3 使用 HBase Shell 连接 HBase 安全集群	15
2.3.4 通过 Thrift server 连接 HBase 集群	18
2.4 导入数据至 HBase 集群	19
2.4.1 使用 CDM 迁移数据到 CloudTable HBase 集群	19
2.4.2 使用 Import 工具将 SequenceFile 数据导入 HBase 集群	24
2.4.3 使用 CopyTable 将源数据导入 HBase 集群	24
2.5 HBase 企业级能力增强	25
2.5.1 配置 HBase 冷热分离	25
2.5.1.1 冷热分离概述	26
2.5.1.2 通过 HBase Shell 配置 HBase 冷热分离	27
2.5.1.3 使用 JAVA API 配置 HBase 冷热分离	29
2.5.2 配置 HBase 全局二级索引	32
2.5.2.1 HBase 全局二级索引介绍	32
2.5.2.2 创建 HBase 全局二级索引	34
2.5.2.3 查询 HBase 全局二级索引信息	35
2.5.2.4 修改 HBase 全局二级索引状态	36
2.5.2.5 删除 HBase 全局二级索引	38
2.6 管理 HBase 集群	38
2.6.1 查看 HBase 集群状态	38
2.6.2 查看 HBase 集群详情	40
2.6.3 重启 HBase 集群节点	43
2.6.4 重启 HBase 集群	44
2.6.5 删除 HBase 集群	45
2.7 HBase 集群运维	46
2.7.1 HBase 集群容量调整	46
2.7.1.1 HBase 集群节点扩容	46

2.7.1.2 HBase 集群磁盘扩容.....	48
2.7.1.3 HBase 集群规格变更.....	48
2.7.2 修改 HBase 参数配置对集群进行性能调优.....	49
2.7.3 使用 CES 监控 HBase 集群.....	53
2.7.3.1 HBase 集群监控指标说明.....	53
2.7.3.2 设置 HBase 集群告警规则.....	62
2.7.3.3 查看 HBase 集群监控信息.....	62
2.7.4 HBase 集群日志管理.....	64
2.7.4.1 对接 LTS 查看 HBase 集群日志.....	64
2.7.4.2 对接 CTS 查看 HBase 集群审计日志.....	65
3 使用 Doris.....	69
3.1 Doris 数据模型概述.....	69
3.2 Doris 使用流程.....	85
3.3 创建 Doris 集群.....	86
3.4 连接 Doris 集群.....	91
3.4.1 准备弹性云服务器.....	91
3.4.2 使用 MySQL 客户端连接 Doris 普通集群.....	91
3.4.3 使用 MySQL 客户端连接 Doris 安全集群.....	94
3.5 配置 Doris 用户权限.....	96
3.6 导入数据至 Doris 集群.....	99
3.6.1 导入数据至 Doris 集群方式介绍.....	99
3.6.2 使用 Broker Load 方式导入数据至 Doris 集群.....	101
3.6.3 使用 Stream Load 方式导入数据至 Doris 集群.....	108
3.7 Doris 企业级能力增强.....	115
3.7.1 配置 Doris 租户.....	115
3.7.1.1 添加租户.....	115
3.7.1.2 租户命令简介.....	117
3.7.2 配置 Doris 连接数据源.....	118
3.7.2.1 Doris 多源数据能力概述.....	118
3.7.2.2 配置 Doris 对接 Hive 数据源.....	120
3.8 管理 Doris 集群.....	126
3.8.1 查看 Doris 集群状态.....	126
3.8.2 查看 Doris 集群详情.....	128
3.8.3 重启 Doris 集群节点.....	131
3.8.4 重启 Doris 集群.....	132
3.8.5 删除 Doris 集群.....	133
3.8.6 Doris 按需集群转包周期集群.....	134
3.9 Doris 集群运维.....	134
3.9.1 Doris 集群容量调整.....	134
3.9.1.1 Doris 扩容说明.....	134
3.9.1.2 Doris 集群节点扩容.....	135
3.9.1.3 Doris 集群磁盘扩容.....	136

3.9.1.4 Doris 集群规格变更.....	137
3.9.2 修改 Doris 参数配置对集群进行性能调优.....	138
3.9.3 使用 CES 监控 Doris 集群.....	142
3.9.3.1 Doris 集群监控指标说明.....	143
3.9.3.2 设置 Doris 集群告警规则.....	167
3.9.3.3 查看 Doris 集群的监控信息.....	168
3.9.4 Doris 集群日志管理.....	169
3.9.4.1 对接 LTS 查看 Doris 集群日志.....	169
3.9.4.2 对接 CTS 查看 Doris 集群审计日志.....	171
3.10 Doris 常用 SQL 命令.....	174
3.10.1 创建数据库.....	174
3.10.2 创建表.....	175
3.10.3 插入数据.....	176
3.10.4 查询数据.....	177
3.10.5 修改表结构.....	177
3.10.6 删除表.....	178
4 使用 ClickHouse.....	179
4.1 ClickHouse 存算一体表引擎概述.....	179
4.2 ClickHouse 使用流程.....	188
4.3 创建 ClickHouse 集群.....	190
4.4 连接 ClickHouse 集群.....	194
4.4.1 准备弹性云服务器.....	194
4.4.2 使用客户端连接 ClickHouse 普通集群.....	194
4.4.3 使用客户端连接 ClickHouse 安全集群.....	196
4.4.4 使用 HTTPS 协议连接 ClickHouse 安全集群.....	199
4.5 配置 ClickHouse 用户权限.....	201
4.6 导入导出数据至 ClickHouse 集群.....	204
4.6.1 数据导入导出.....	204
4.6.2 ClickHouse 访问 RDS MySQL 服务.....	207
4.7 ClickHouse 企业级能力增强.....	209
4.7.1 查看 ClickHouse 慢查询语句.....	209
4.7.2 ClickHouse 数据迁移.....	210
4.8 管理 ClickHouse 集群.....	215
4.8.1 查看 ClickHouse 集群状态.....	215
4.8.2 查看 ClickHouse 集群详情.....	217
4.8.3 重启 ClickHouse 集群节点.....	220
4.8.4 重启 ClickHouse 集群.....	220
4.8.5 删除 ClickHouse 集群.....	221
4.8.6 ClickHouse 按需集群转包周期集群.....	222
4.9 ClickHouse 集群运维.....	223
4.9.1 ClickHouse 集群容量调整.....	223
4.9.1.1 ClickHouse 集群节点扩容.....	223

4.9.1.2 ClickHouse 集群磁盘扩容.....	224
4.9.1.3 ClickHouse 集群规格变更.....	225
4.9.2 使用 CES 监控 ClickHouse 集群.....	227
4.9.2.1 ClickHouse 集群监控指标说明.....	227
4.9.2.2 设置 ClickHouse 集群监控规则.....	235
4.9.2.3 查看 ClickHouse 集群监控信息.....	236
4.9.3 ClickHouse 集群日志管理.....	237
4.9.3.1 对接 LTS 查看 ClickHouse 集群日志.....	237
4.9.3.2 对接 CTS 查看 ClickHouse 集群审计日志.....	239
4.10 ClickHouse 存算一体常用 SQL 命令.....	242
4.10.1 数据类型.....	242
4.10.2 CREATE DATABASE.....	245
4.10.3 CREATE TABLE.....	246
4.10.4 DESC 查询表结构.....	248
4.10.5 CREATE VIEW.....	249
4.10.6 CREATE MATERIALIZED VIEW.....	250
4.10.7 INSERT INTO.....	251
4.10.8 SELECT.....	251
4.10.9 ALTER TABLE 修改表结构.....	252
4.10.10 DROP 删除表.....	253
4.10.11 SHOW 显示数据库和表信息.....	253

1 创建用户并授权使用 CloudTable

使用CloudTable服务之前，请先注册华为账号并开通华为云，有关账号的详细介绍请参见[账号中心](#)。本文介绍如何注册一个华为账号，如果您已有华为账号，可直接登录管理控制台。

注册华为云并实名认证

- 步骤1** 进入云服务[网站](#)，单击页面右上角的“注册”，进入注册页面。
- 步骤2** 填写相关信息，根据提示完成注册。注册后系统自动登录华为云。
- 步骤3** 单击页面右上角的“用户名 > 基本信息 > 实名认证”，进入实名认证页面。
- 步骤4** 根据页面提示完成认证，请参考[实名认证](#)。

---结束

通过 IAM 账户使用 CloudTable

如果您需要对您所拥有的云服务进行精细的权限管理，您可以使用[统一身份认证服务](#)（Identity and Access Management，简称IAM）。通过IAM，您可以：

- 根据企业的业务组织，在您的云账号中，给企业中不同职能部门的员工创建IAM用户，让员工拥有唯一安全凭证，并使用云服务资源。
- 根据企业用户的职能，设置不同的访问权限，以达到用户之间的权限隔离。
- 将云服务资源委托给更专业、高效的其他云账号或者云服务，这些账号或者云服务可以根据权限进行代运维。

如果云账号已经能满足您的要求，不需要创建独立的IAM用户，您可以跳过本章节，不影响您使用CloudTable服务的其它功能。

本章节为您介绍对用户授权的方法，操作流程如[图1-1](#)所示。

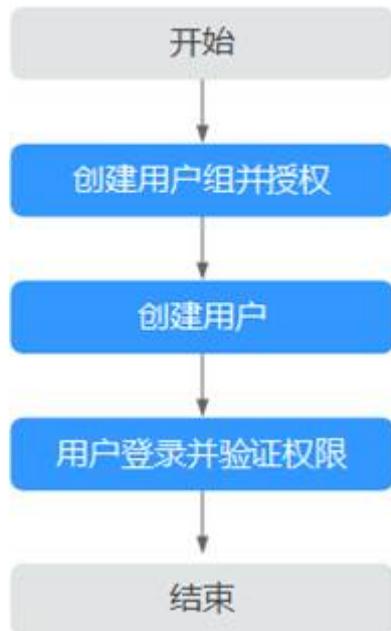
前提条件

- CloudTable服务暂不支持细粒度策略，仅支持系统角色“cloudtable Administrator”。当前CloudTable默认的系统角色“cloudtable Administrator”仅适用于未开通企业项目的账号，如已开通，请使用IAM自定义策略并直接给用户授权。

- 给用户组授权之前，请您了解用户组可以添加的CloudTable权限，并结合实际需求进行选择，CloudTable支持的系统权限，请参见：[权限管理](#)。如果您需要对除CloudTable之外的其它服务授权，IAM支持服务的所有权限请参见[权限策略](#)。

示例流程

图 1-1 给用户授权 CloudTable 权限流程



1. **创建用户组并授权。**
使用云登录IAM控制台，创建用户组，并授予表格存储服务的管理员权限“cloudtable Administrator”。
2. **创建用户并加入用户组。**
在IAM控制台创建用户，并将其加入步骤1中创建的用户组。
3. **用户登录并验证权限。**
使用新创建的用户登录控制台，切换至授权区域，验证权限：
 - 在“服务列表”中选择表格存储服务，进入CloudTable主界面，单击右上角“购买集群”，尝试创建表格存储服务集群，如果可以创建（假设当前权限仅包含cloudtable Administrator），表示“cloudtable Administrator”已生效。
 - 在“服务列表”中选择除表格存储服务之外（假设当前策略仅包含cloudtable Administrator）的任一服务，如果提示权限不足，表示“cloudtable Administrator”已生效。

2 使用 HBase

2.1 HBase 使用流程

CloudTable HBase集群模式提供了基于Apache HBase的分布式、可伸缩、全托管的NoSQL数据存储系统，它提供了毫秒级的随机读写能力，适用于海量结构化数据、半结构化数据存储和查询应用。

HBase集群创建成功后，用户可通过以下方式访问HBase集群：

- 使用客户端访问集群。
详情请参见此章节：[使用HBase](#)：介绍了如何使用HBase Shell访问集群等内容。
- 调用开源原生接口，开发HBase应用，通过CloudTable HBase链接地址访问集群。
详情请参见《表格存储服务开发指南》。

图 2-1 HBase 使用流程

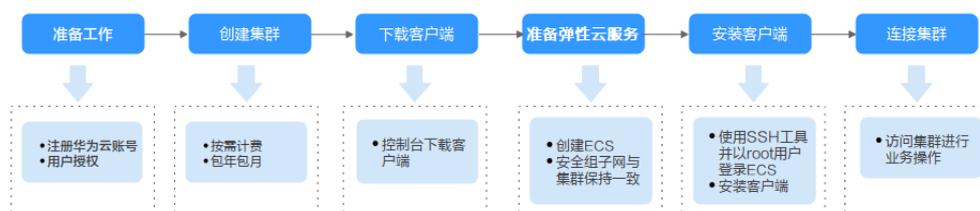


表 2-1 HBase 使用流程

流程	子任务	说明	详细指导
准备工作	创建用户并授权使用 CloudTable	<ul style="list-style-type: none"> 使用 CloudTable HBase 之前，您需要注册华为云账号并进行实名认证，给新账号授权。 将策略授予用户组，并将用户添加至用户组中从而使用户拥有对应的服务权限。 	创建用户并授权使用 CloudTable
创建集群	创建 HBase 集群	使用 HBase 执行任务前，需要创建一个 HBase 集群。	创建 HBase 集群
下载客户端	下载 HBase 客户端	创建集群后，需要下载客户端并安装客户端，安装客户端后，用户才可以借助 SSH 工具连接集群。	使用 HBase Shell 连接 HBase 普通集群
准备弹性云服务	-	<p>如果客户端工具的运行环境为 Linux 环境，您需要准备一台和 HBase 集群在相同虚拟私有云的 Linux 弹性云服务器作为客户端主机。</p> <p>如果客户端工具的运行环境为 Windows 环境，您需要准备一台和 HBase 集群在相同虚拟私有云的 Windows 弹性云服务器作为客户端主机。</p>	准备弹性云服务器
安装客户端	-	将下载的客户端放入 ECS，解压安装客户端。	手动安装客户端
连接集群	-	在 ECS 安装好 MySQL 客户端后，可用命令连接集群并进行业务操作。	连接 HBase 集群

2.2 创建 HBase 集群

通过表格存储服务可以集中管理集群，如果需要使用 CloudTable 服务，首要操作就是创建集群，本章节为您介绍如何在表格存储服务管理控制台创建一个新的集群。

HBase 集群目前支持的计费模式有按需付费、包年包月付费。创建集群页面默认采用按需付费方式，这种购买方式比较灵活，可以即开即停，按实际使用时长计费。包年包月可通过预付费，以包年包月模式购买云服务，这种购买方式相对于按需付费有更大的折扣，对于长期使用者，推荐该计费模式。您可根据业务需要，定制相应计算能力和存储空间的 CloudTable HBase 集群。

说明

当前支持创建的集群无需通过Kerberos认证，就可以访问HBase集群。如果对集群访问的安全性有要求，推荐使用MRS服务上的HBase组件。

前提条件

- 创建集群的虚拟私有云、安全组需要和公网ECS保持一致，否则客户端无法访问集群。
- 用户创建集群前，必须配置所使用主机的入方向安全组规则，请参考[安全组规则配置](#)章节，安全组端口请参考[HBase安全规则组表](#)。
- 用户创建集群前安全组规则必须添加ICMP协议，方便管理面通过ping方式查看各节点状态，请参考[安全组规则配置](#)章节。

创建集群

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击“购买集群”，进入“购买集群”页面。

步骤4 参考下列表格配置集群基本信息。

表 2-2 区域

参数	参数说明
区域	集群的工作区域。 有关区域的详细信息，请参见 地区和终端节点 。
可用区	选择集群工作区域下关联的可用区。 关于区域和可用区的描述，请参见 区域和可用区 。
计费模式	计费模式有按需付费和包年包月。
订购周期	当计费模式选择包年包月时，才会出现此选项。客户根据需求选择订购时间。
自动续费	在购买页面勾选此选项，集群到期后会自动续费，不勾选集群到期需要手动续费。

表 2-3 网络配置

参数	参数说明
名称	集群的名称。 集群名称在4位到32位之间，必须以字母开头，可以包含字母、数字或者中划线，不能包含其他的特殊字符，不区分大小写。

参数	参数说明
虚拟私有云	<p>VPC即虚拟私有云，是通过逻辑方式进行网络隔离，提供安全、隔离的网络环境。</p> <p>您可以选择使用已有的虚拟私有云网络，或者单击“查看虚拟私有云”创建新的虚拟私有云。</p> <p>关于虚拟私有云的更多信息，请参见虚拟私有云的“用户指南 > 虚拟私有云和子网 > 创建虚拟私有云和子网”。</p>
子网	<p>通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全。</p> <p>创建虚拟私有云时会创建一个子网。如需创建新的子网，请参见虚拟私有云的“用户指南 > 虚拟私有云和子网 > 为虚拟私有云创建新的子网”。</p>
安全组	<p>安全组用来实现安全组内和安全组间弹性云服务器的访问控制，加强弹性云服务器的安全保护。用户可以在安全组中定义各种访问规则，这些规则可以限定弹性云服务器的哪些端口或者协议可以访问，这些访问规则可以作用在进出虚拟机的网络访问上。当弹性云服务器加入该安全组后，即受到这些访问规则的保护。不属于该安全组的弹性云服务器，是无法同安全组内的弹性云服务器通信的。</p> <p>由于CloudTable集群的底层计算单元就是弹性云服务器，从安全和业务的角度来看，需要将这些弹性云服务器加入同一个VPC下的相同安全组中。VPC作为网络隔离，安全组则是控制哪些端口和协议可以在VPC范围内开放访问权限。</p> <p>您可以选择使用已有的安全组，或者单击“查看安全组”创建新的安全组。</p> <p>关于安全组的更多信息，请参见《虚拟私有云用户指南》中的安全组。</p>
数据引擎	选择需要创建的数据组件类型。

图 2-2 Master/Core 节点

Master节点

★ 计算规格 32*CU2 (1个CU2 = 1核2GB)

集群默认配置为高可用，使用2个master节点，额外免费赠送1个master，用于集群监控

Core节点

★ 计算规格 32*CU2 (1个CU2 = 1核2GB)

★ 存储规格 400 GB / 节点

★ 节点数量 2

默认双副本，存储容量为双副本所需容量

高级特性 开启冷热分离 [?](#)

表 2-4 Master 节点配置

参数	参数说明
计算规格	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明</p> <ul style="list-style-type: none"> 支持的计算规格： <ul style="list-style-type: none"> - 4U16G。 - 8U16G。 - 8U32G。 - 16U32G。 - 16U64G。 - 32U64G。
存储规格	默认超高IO。
节点数量	默认2个master节点。

表 2-5 Core 节点配置

参数	参数说明
计算节点	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明</p> <ul style="list-style-type: none"> 支持的计算规格： <ul style="list-style-type: none"> - 4U16G - 8U16G - 8U32G - 16U32G - 16U64G - 32U64G - 32U128G - 64U128G
存储规格	<p>选择HBase计算节点的磁盘规格、容量。</p> <p>说明</p> <ul style="list-style-type: none"> 支持的存储规格： <ul style="list-style-type: none"> - 普通IO。 - 高IO。 - 超高IO。 容量范围为400~30000GB/节点。
节点数量	选择集群中的节点个数，节点范围2~20。
高级特性	冷热分离：将数据分类存储在不同介质上，有效降低存储成本。仅支持HBase 2.4.14及以上版本，勾选表示开启。

参数	参数说明
是否开启安全通道	默认开启。集群开启安全通道可能会导致性能下降。
是否开启Thrift Server	默认关闭。开启此功能，HBase支持多语言访问。
企业项目	<p>企业项目管理帮助您将相关的资源（如具有相同使用用途的资源）集中在一起，按企业项目的方式来管理云资源。</p> <p>说明</p> <ul style="list-style-type: none"> 当企业业务发生变化，原用户不再拥有企业项目的使用权限时，可将这些用户与该企业项目的关联关系删除。 可进行单个删除和批量删除。 当用户与企业项目的关联关系被删除后，该用户将无法管理该企业项目，如需再次使用，需要重新给该用户关联企业项目。

步骤5 单击“立即购买”。

步骤6 确认集群规格的订单详细信息，单击“提交”，成功提交集群创建任务。

步骤7 单击“返回集群管理列表”，可以查看到集群创建的状态。

集群创建需要时间，请您耐心等待。所创集群的初始状态为“创建中”，创建成功后状态更新为“服务中”。

步骤8 如果创建的集群计费模式为包年/包月，单击提交，成功提交集群创建任务。

单击“去支付”，跳转到购买页面，确认信息后，选择合适的支付方式，确认付款。

返回控制台，查看集群状态，集群创建需要时间，请您耐心等待。所创集群的初始状态为“创建中”，创建成功后状态更新为“服务中”。

----结束

2.3 连接 HBase 集群

2.3.1 准备弹性云服务器

如果客户端工具的运行环境为Linux环境，您需要准备一台和CloudTable集群在相同虚拟私有云的Linux弹性云服务器作为客户端主机。例如，使用以下客户端工具访问集群，建议使用Linux弹性云服务器：HBase shell。

如果客户端工具的运行环境为Windows环境，您需要准备一台和CloudTable集群在相同虚拟私有云的Windows弹性云服务器作为客户端主机。

准备弹性云服务器

购买Linux或Windows弹性服务器的操作步骤，请参见《弹性云服务器快速入门》中的[购买弹性云服务器](#)。

购买的弹性云服务器需要满足如下要求。

- 弹性云服务器需要与CloudTable集群具有相同的区域、可用区、虚拟私有云和子网。
关于虚拟私有云的更多信息，请参见[虚拟私有云](#)的“用户指南 > 虚拟私有云和子网”。
- 弹性云服务器需要与CloudTable集群具有相同的安全组。
关于安全组的更多信息，请参见《虚拟私有云用户指南》中的[安全组](#)。

📖 说明

在使用跨VPC通信方式访问CloudTable集群时，需要由网络管理员对集群所在VPC，安全组和子网的访问进行授权。

请确认安全组的“出方向规则”中是否存在如下规则：

- 协议端口：全部
- 类型：IPv4
- 目的地址：0.0.0.0/0

如上述规则不存在，则按[图2-3](#)所示添加如下出方向规则。

- 协议：TCP
- 端口：80
- 目的地址：选择“IP地址”，并设置为0.0.0.0/0

图 2-3 添加出方向规则



- 购买弹性云服务器时，“弹性公网IP”需选择“现在购买”，或者在ECS创建完成后再为该ECS绑定一个弹性公网IP。
- 访问Linux弹性云服务器，建议使用SSH密码方式登录Linux弹性云服务器。
具体登录操作步骤请参见《弹性云服务器用户指南》中的[登录Linux弹性云服务器](#)。
- 访问Windows弹性云服务器，建议使用MSTSC远程桌面连接的方式登录。
具体登录操作步骤请参见《弹性云服务器用户指南》中的[登录Windows弹性云服务器](#)。

为 Linux 弹性云服务器配置 DNS 地址和 hosts 文件

当您通过一键部署客户端的方法安装HBase Shell时，不需要执行本操作。其他情况下，请参照以下方法为Linux弹性云服务器配置DNS地址和hosts。

步骤1 配置/etc/hosts。

将主机IP配置到/etc/hosts文件，可以加速访问CloudTable集群。

为Linux弹性云服务器配置"/etc/hosts"的操作方法如下：

1. 以root用户登录Linux弹性云服务器。
2. 执行“hostname”，查询主机名。
[root@euler ~]# hostname
euler.novalocal
3. 执行“ifconfig”，查询本机IP。

图 2-4 查询 IP

```
[root@euler ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.58 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::f816:3eff:fe8d:7acb prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:8d:7a:cb txqueuelen 1000 (Ethernet)
    RX packets 379443 bytes 522261185 (498.0 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 148302 bytes 10571485 (10.0 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 0 (Local Loopback)
    RX packets 23819 bytes 16629808 (15.8 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 23819 bytes 16629808 (15.8 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@euler ~]#
```

4. 执行“vi /etc/hosts”命令，编辑文件，增加主机配置。
192.168.0.58 euler.novalocal
127.0.0.1 euler.novalocal
5. 按“Esc”键，并输入“:wq”，保存退出。
6. 执行如下命令，查看IP地址是否写入成功。
cat /etc/hosts
7. 执行如下命令，验证是否能够解析主机名。
ping 主机名

步骤2 配置DNS。

DNS服务器用于解析CloudTable集群相关链接地址（例如ZK链接地址）中的域名。请根据CloudTable集群所在的区域，设置相应区域的内网DNS地址，各个区域的DNS地址请参见《云解析服务常见问题》中的[内网DNS地址是多少？](#)。

为Linux弹性云服务器配置DNS的操作方法如下：

1. 以root用户登录Linux弹性云服务器。
2. 执行“vi /etc/resolv.conf”命令，编辑“/etc/resolv.conf”文件，增加“nameserver”配置。

在已有的公网IP地址之前新增两行写入DNS服务器IP地址，格式如下：
nameserver 100.125.1.250

3. 按“Esc”键，并输入“:wq”，保存退出。
4. 执行如下命令，查看IP地址是否写入成功。
cat /etc/resolv.conf
5. 执行如下命令，验证访问域名是否可以解析到IP地址。
ping 访问域名

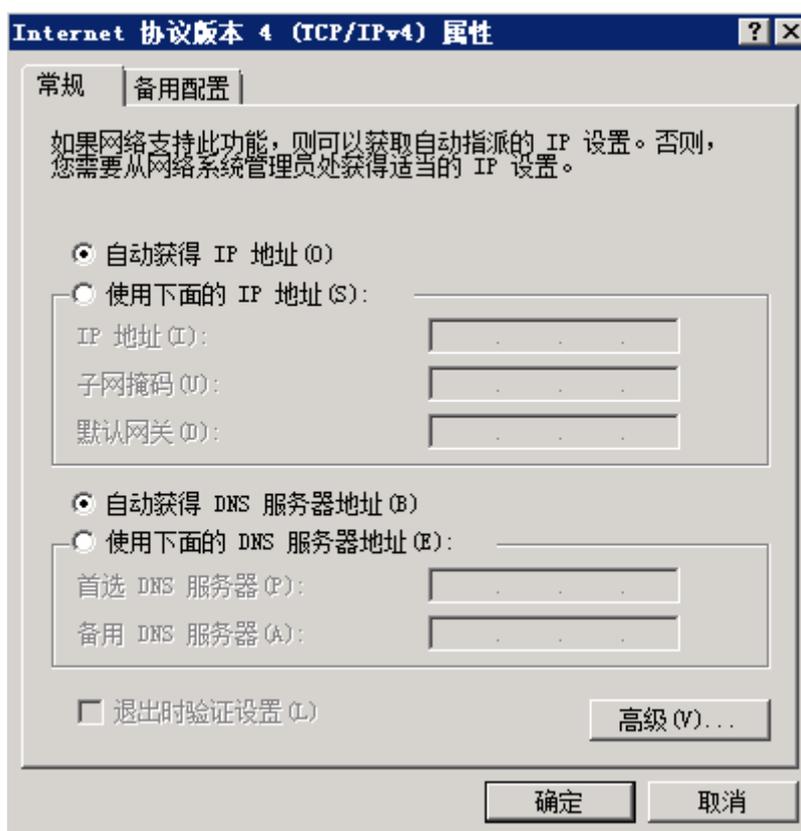
访问域名请输入CloudTable集群的相关链接地址（例如ZK链接地址）中的域名。请在表格存储服务管理控制台，单击集群名称，进入集群基本信息页面，获取CloudTable的相关链接地址。所获取的链接地址包含了以逗号分隔的3个内网域名，您可以ping其中任意一个内网域名。

----结束

为 Windows 弹性云服务器配置 DNS 地址

1. 使用Administrator账号登录Windows弹性云服务器。
2. 在Windows弹性云服务器上选择“开始 > 控制面板 > 网络和共享中心”。
3. 在“网络和共享中心”窗口，单击右边的“更改适配器设置”。
4. 右击网络适配器名，在弹出的快捷菜单中，选择“属性”，弹出属性窗口。
5. 单击“Internet协议版本4（TCP/IPv4）”，弹出“Internet协议版本4（TCP/IPv4）属性”窗口。

图 2-5 Internet 协议版本 4（TCP/IPv4）属性



6. 在图2-5中勾选“使用下面的DNS服务器地址”，将DNS服务器IP地址填入“首选DNS服务器”框，并单击“确定”。

DNS服务器用于解析CloudTable集群相关链接地址（例如ZK链接地址）中的域名。请根据CloudTable集群所在的区域，设置相应区域的内网DNS地址，各个区域的DNS地址请参见《云解析服务常见问题》中的[内网DNS地址是多少？](#)。

7. 单击“开始”，在搜索框中输入“cmd”并按“Enter”，进入命令行对话框。
8. 在命令行中执行如下命令，验证配置DNS是否成功。
ping 访问域名

访问域名请输入CloudTable集群的相关链接地址（例如ZK链接地址）中的域名。请在表格存储服务管理控制台，单击集群名称，进入集群基本信息页面，获取CloudTable的相关链接地址。所获取的链接地址包含了以逗号分隔的3个内网域名，您可以ping其中任意一个内网域名。

2.3.2 使用 HBase Shell 连接 HBase 普通集群

用户在弹性云服务器里可以通过[手动安装客户端](#)或[一键部署客户端](#)的方法，使用 HBase Shell 访问集群。推荐使用[一键部署客户端](#)的方法。如果集群开启了安全通道，连接集群的方法请参考[HBase安全集群连接步骤](#)。

使用限制

- HBase集群与弹性云服务器具有相同区域、可用区、VPC。
- HBase集群与弹性云服务器具有相同安全组。
- 弹性云服务器安全组中添加本地主机IP地址。

一键部署客户端

步骤1 准备Linux弹性云服务器。

使用一键部署客户端工具，建议Linux弹性云服务器的操作系统类型为EulerOS，CentOS，Ubuntu和SUSE。具体操作请参见[准备弹性云服务器](#)。

步骤2 下载客户端一键部署工具。

使用SSH登录工具（如PuTTY）通过“弹性IP”远程登录到Linux弹性云服务器，然后执行如下命令，获取客户端一键部署工具：

```
curl -O -k "https://cloudtable-publish.obs.myhuaweicloud.com/quick_start_hbase_shell.sh"
```

说明

此命令适用于HBase 1.x版本。

```
curl -O -k "https://cloudtable-publish.obs.myhuaweicloud.com/cloudtable-client/quick_start_hbase_shell.sh"
```

说明

- 此命令适用于HBase 2.x版本。
- 一键部署包内置校验文件。

步骤3 准备集群访问地址。

登录表格存储服务管理控制台，在左侧导航树单击集群管理，然后在集群列表中找到所需要的集群，并获取相应的“ZK链接地址（内网）”。该参数值就是集群访问地址，如[图2-6](#)所示。

图 2-6 ZK 链接地址



步骤4 使用工具部署客户端。

请将以下命令中的"\$zookeeper_address"参数替换为[步骤3](#)获取的ZK链接地址，然后在弹性云服务器命令行窗口，执行该命令一键部署客户端。

1. 普通集群一键部署命令。

```
source quick_start_hbase_shell.sh $zookeeper_address
```
2. 安全集群一键部署命令。

```
source quick_start_hbase_shell $zookeeper_address enable
```

步骤5 启动Shell访问集群。

在上一步执行source命令自动部署客户端后，就已自动启动了HBase Shell。在后续使用过程中，您也可以执行“bin/hbase shell”命令启动Shell访问集群。

---结束

手动安装客户端

步骤1 准备Linux弹性云服务器。

具体操作请参见[准备弹性云服务器](#)章节中的以下内容：

步骤2 下载客户端和客户端校验文件。

登录表格存储服务管理控制台，在左侧导航树单击“帮助”，然后在页面右侧单击“客户端下载”和“客户端校验文件”，下载客户端安装包和客户端校验文件。

步骤3 安装客户端并校验客户端。

1. 使用文件传输工具（如WinSCP工具）将客户端安装包上传到Linux弹性云服务器。
2. 使用SSH登录工具（如PuTTY）通过“弹性IP”远程登录到Linux弹性云服务器。具体登录操作步骤请参见《弹性云服务器用户指南》中的“[登录Linux弹性云服务器](#) > SSH密码方式登录”。

登录后执行如下命令解压安装包：

```
cd <客户端安装包存放路径>
tar xzvf hbase-1.3.1-bin.tar.gz
cd <客户端安装包存放路径>
tar xzvf hbase-2.4.14-bin.tar.gz
```

其中，<客户端安装包存放路径>请替换为实际的客户端存放路径。

3. 解压“客户端校验文件”，与客户端解压在同一个目录下。
 - a. 解压“客户端校验文件”。

```
cd <客户端校验文件存放路径>
tar xzvf Client_sha256.tar.gz
```
 - b. 获取客户端校验码。

```
sha256sum HBase_Client_2.4.14.tar.gz
```
 - c. 查看客户端校验文件中的校验码，并与客户端校验码进行对比。如果对比结果一致，代表客户端未被篡改，如果对比结果不一致，则代表客户端篡改。

```
less HBase_Client_2.4.14.tar.gz.sha256
```
4. 配置zookeeper地址到配置文件。

在客户端解压目录下，打开“hbase/conf/hbase-site.xml”文件，配置如下参数：

- “hbase.zookeeper.quorum”：该参数值为在集群管理列表中查找的对应集群的“ZK链接地址（内网）”。

登录表格存储服务管理控制台，在左侧导航树单击集群管理，然后在集群列表中找到所需要的集群，并获取相应的“ZK链接地址（内网）”。如下图所示：

图 2-7 查看 ZK 链接地址



- “mapreduce.cluster.local.dir”：请先检查是否存在该配置项，如果不存在，请添加该配置项。

未开启安全通道加密集群客户端配置示例如下：

```
<configuration>
<property>
<name>hbase.zookeeper.quorum</name>
<value>xxx-zk1.cloudtable.com:2181,xxx-zk2.cloudtable.com:2181,xxx-zk3.cloudtable.com:2181</value>
</property>
<property>
<name>mapreduce.cluster.local.dir</name>
<value>${hadoop.tmp.dir}/mapred/local</value>
</property>
</configuration>
```

开启安全通道加密集群客户端配置请参见[HBase安全通道加密](#)章节。

步骤4 启动Shell访问集群。

执行“bin/hbase shell”，启动Shell访问集群。

---结束

HBase 入门使用

本节介绍常用的HBase Shell命令。更多的HBase Shell命令，请参见<https://learnhbase.wordpress.com/2013/03/02/hbase-shell-commands/>。

1. 获取帮助。

在HBase shell执行“help”命令，将会返回所有命令信息，以及基本命令的说明和使用方法。

```
hbase(main):001:0> help
```

2. 创建表。

执行“create”命令创建一张表，创建表时必须指定表的名称和ColumnFamily名称。

```
hbase(main):007:0> create 'cloudtable','cf'
0 row(s) in 1.5530 seconds
```

```
=> Hbase::Table - cloudtable
```

3. 查询表。

```
hbase(main):009:0> list
TABLE
cloudtable
1 row(s) in 0.0060 seconds
```

```
=> ["cloudtable"]
```

4. 插入记录。

执行“put”命令往指定表插入一条记录，需要指定表的名称，主键，自定义列，以及插入的具体值。

```
hbase(main):004:0> put 'cloudtable','row1','cf:a','value1'
0 row(s) in 0.2720 seconds
```

命令中各参数分别代表如下含义：

- cloudtable：表的名称
- row1：主键
- cf: a：自定义的列
- value1：插入的值

5. 扫描记录。

执行“scan”命令扫描表，扫描时需要指定表的名称，即可以全表扫描，也可以指定范围扫描。

```
hbase(main):001:0> scan 'cloudtable'
ROW                                COLUMN+CELL
row1                                column=cf:a, timestamp=1504866237162,
value=value1
1 row(s) in 0.2420 seconds
```

说明

- 如果插入数据时，设置了CELL的TTL，TTL属性不支持查看，但是可以通过验证设置的TTL是否生效来确认。
- 如果插入数据时未设置CELL的TTL，此时系统会自动插入当前时间为时间戳。

6. 查询单条记录。

执行“get”命令查询单条记录，必须指定查询表的名称和主键。

```
hbase(main):001:0> get 'cloudtable','row1'
COLUMN                                CELL
cf:a                                timestamp=1504866237162, value=value1
1 row(s) in 0.2280 seconds
```

7. 禁用表。

如果需要修改表，比如删除表，修改表的设置，需要先对表进行禁用。执行“disable”命令禁用表。在禁用表时执行操作会报ERROR，并提示表被禁用。

```
hbase(main):002:0> disable 'cloudtable'
0 row(s) in 2.3550 seconds
```

8. 使能表。

与禁用表相对应的，在禁用表后，如果需要使用表，就需要执行“enable”命令使能表。

```
hbase(main):004:0> enable 'cloudtable'
0 row(s) in 1.2500 seconds
```

9. 删除表。

当不再需要表时，可以执行“drop”命令删除表。删除表之前需要禁用表，否则会报ERROR，并提示表被使能。删除表将会导致数据丢失，请谨慎操作。

```
hbase(main):007:0> disable 'cloudtable'
0 row(s) in 2.2380 seconds

hbase(main):008:0> drop 'cloudtable'
0 row(s) in 1.2600 seconds
```

10. 退出HBase shell。

执行“quit”命令退出HBase Shell。

```
hbase(main):009:0> quit
```

2.3.3 使用 HBase Shell 连接 HBase 安全集群

您可以通过创建集群开启加密通道，实现数据加密传输。此章节介绍如何开通HBase安全通道。

使用限制

- 集群开启安全通道可能会导致性能下降。
- HBase安全通道只能在创建集群时开启。
- HBase集群与弹性云服务器具有相同区域、可用区、VPC。
- HBase集群与弹性云服务器具有相同安全组。

- 弹性云服务器安全组中添加本地主机IP地址。

创建集群并打开安全通道

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 单击右上角的购买集群按钮，进入“购买集群”页面。
- 步骤4** 完成其他配置后，查看安全通道加密按钮是否开启，默认开启。
- 步骤5** 配置好参数，单击“立即购买”。
- 步骤6** 进入“规格详情”页面，确认集群规格订单信息，单击“提交”，成功提交集群创建任务，集群创建成功后，进入详情页面查看通道开启状态。

---结束

连接 HBase 安全集群

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 单击“帮助 > hbase2.x客户端下载”，下载客户端。
- 步骤4** [准备弹性云服务](#)安装客户端。
- 步骤5** 使用SSH登录工具（如PuTTY）通过“弹性IP”远程登录到Linux弹性云服务器。

具体登录操作步骤请参见《弹性云服务器用户指南》中的“[登录Linux弹性云服务器 > SSH密码方式登录](#)”。

- 步骤6** 安装[步骤3](#)中下载的客户端，请参见[安装客户端并校验客户端](#)。

- 步骤7** 进入安装客户端的目录，修改hbase-env.sh文件内容。

- 进入hbase-client文件夹，查看conf文件夹。

```
cd hbase-client
```
- 进入conf文件夹编辑hbase-env.sh文件。
 - 进入conf文件夹。

```
cd conf/
```
 - 编辑hbase-env.sh文件。

```
vi hbase-env.sh
```
 - 在hbase-env.sh文件最后添加内容：

```
CLIENT_JVMFLAGS="-Dzookeeper.clientCnxnSocket=org.apache.zookeeper.ClientCnxnSocketNetty -Dzookeeper.client.secure=true"
export HBASE_OPTS="${CLIENT_JVMFLAGS} ${HBASE_OPTS}"
```
 - 单击“Ecs”退出编辑界面。
 - 保存添加内容。

```
:wq
```

- 步骤8** 进入安装客户端目录下进行客户端配置，将如下配置添加到客户端conf目录下的hbase-site.xml文件中。

- 编辑hbase-site.xml文件。

```
vi hbase-site.xml
```

- 在hbase-site.xml文件最后添加内容：

```
<property>
  <name>hbase.rpc.protection</name>
  <value>privacy</value>
</property>
```

- 单击“Ecs”退出编辑界面。

- 保存添加内容。

```
:wq
```

步骤9 连接HBase集群。

- 配置zookeeper地址到配置文件，请参见[配置zookeeper地址](#)。

- 启动Shell访问集群。

执行“bin/hbase shell”，启动Shell访问集群。

----结束

HBase 入门使用

本节介绍常用的HBase Shell命令。更多的HBase Shell命令，请参见<https://learnhbase.wordpress.com/2013/03/02/hbase-shell-commands/>。

1. 获取帮助。

在HBase shell执行“help”命令，将会返回所有命令信息，以及基本命令的说明和使用方法。

```
hbase(main):001:0> help
```

2. 创建表。

执行“create”命令创建一张表，创建表时必须指定表的名称和ColumnFamily名称。

```
hbase(main):007:0> create 'cloudtable','cf'
0 row(s) in 1.5530 seconds
```

```
=> Hbase::Table - cloudtable
```

3. 查询表。

```
hbase(main):009:0> list
TABLE
cloudtable
1 row(s) in 0.0060 seconds
```

```
=> ["cloudtable"]
```

4. 插入记录。

执行“put”命令往指定表插入一条记录，需要指定表的名称，主键，自定义列，以及插入的具体值。

```
hbase(main):004:0> put 'cloudtable','row1','cf:a','value1'
0 row(s) in 0.2720 seconds
```

命令中各参数分别代表如下含义：

- cloudtable：表的名称
- row1：主键
- cf: a：自定义的列
- value1：插入的值

5. 扫描记录。

执行“scan”命令扫描表，扫描时需要指定表的名称，即可以全表扫描，也可以指定范围扫描。

```
hbase(main):001:0> scan 'cloudtable'
ROW                                COLUMN+CELL
row1                                column=cf:a, timestamp=1504866237162,
value=value1
1 row(s) in 0.2420 seconds
```

说明

- 如果插入数据时，设置了CELL的TTL，TTL属性不支持查看，但是可以通过验证设置的TTL是否生效来确认。
- 如果插入数据时未设置CELL的TTL，此时系统会自动插入当前时间为时间戳。

6. 查询单条记录。

执行“get”命令查询单条记录，必须指定查询表的名称和主键。

```
hbase(main):001:0> get 'cloudtable','row1'
COLUMN                                CELL
cf:a                                timestamp=1504866237162, value=value1
1 row(s) in 0.2280 seconds
```

7. 禁用表。

如果需要修改表，比如删除表，修改表的设置，需要先对表进行禁用。执行“disable”命令禁用表。在禁用表时执行操作会报ERROR，并提示表被禁用。

```
hbase(main):002:0> disable 'cloudtable'
0 row(s) in 2.3550 seconds
```

8. 使能表。

与禁用表相对应的，在禁用表后，如果需要使用表，就需要执行“enable”命令使能表。

```
hbase(main):004:0> enable 'cloudtable'
0 row(s) in 1.2500 seconds
```

9. 删除表。

当不再需要表时，可以执行“drop”命令删除表。删除表之前需要禁用表，否则会报ERROR，并提示表被使能。删除表将会导致数据丢失，请谨慎操作。

```
hbase(main):007:0> disable 'cloudtable'
0 row(s) in 2.2380 seconds
```

```
hbase(main):008:0> drop 'cloudtable'
0 row(s) in 1.2600 seconds
```

10. 退出HBase shell。

执行“quit”命令退出HBase Shell。

```
hbase(main):009:0> quit
```

2.3.4 通过 Thrift server 连接 HBase 集群

HBase Thrift server是一个运行在HBase集群上的服务程序，它通过Thrift接口提供对HBase数据库的访问，它可以将不同语言的代码生成为对应的客户端和服务端代码。

使用限制

- Thrift Server支持的语言：Python、C++。
- Thrift Server功能不支持冷热分离、热点自愈和二级索引。
- Thrift Server只在Core节点中部署。
- Thrift Server服务扩容依赖Core节点扩容。
- 开启Thrift Server后，用IP连接Thrift Server节点，如果Thrift Server进程故障会影响Thrift Server的业务。

- Thrift Serve特性仅支持HBase 2.4.14及以上版本。
- 默认关闭Thrift Server。

前提条件

- 已经[创建HBase集群](#)。
- 集群状态“服务中”。

开启 Thrift Server 步骤

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角，选择区域。

步骤3 单击集群管理界面右上角的“购买集群”。

步骤4 进入创建集群界面，在集群购买页面选择HBase数据库引擎并开启Thrift Server，创建后的集群就开启了Thrift Server。

也可在集群管理界面，单击操作列的“更多 > 开启Thrift Server”开启Thrift Server。

步骤5 单击目标集群名称，进入集群详情页“集群信息”区域获取Thrift Server IP地址。

步骤6 选择运行语言并配置IP，具体请参考[HBase多语言访问](#)。

----结束

关闭 Thrift Server 步骤

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角，选择区域。

步骤3 选择需要关闭Thrift Server的集群，单击“更多 > 关闭Thrift Server”。

步骤4 弹出关闭Thrift Server服务窗口，单击“确定”。

----结束

2.4 导入数据至 HBase 集群

2.4.1 使用 CDM 迁移数据到 CloudTable HBase 集群

通过云数据迁移（Cloud Data Migration，以下简称CDM），可以将云上云下或第三方云上的多种数据源的数据迁移到CloudTable集群的HBase表中。支持的数据源请参见《云数据迁移产品介绍》中的[CDM支持的数据源](#)章节。

使用CDM迁移数据到CloudTable的流程如下：

1. [创建CloudTable集群](#)
2. [使用CDM迁移数据到CloudTable](#)
3. [在CloudTable中查看导入的数据](#)

创建 CloudTable 集群

步骤1 登录表格存储服务管理控制台。

步骤2 在控制台左上角选择区域。

步骤3 创建一个**CloudTable HBase**集群。如果您已经有CloudTable HBase集群了，也可以跳过这一步。

例如，创建一个名为"CloudTable-demo"的CloudTable集群。

----结束

使用 CDM 迁移数据到 CloudTable

步骤1 登录云数据迁移管理控制台，创建一个CDM集群。

创建的CDM集群需要与CloudTable集群具有相同的区域、可用区、虚拟私有云、子网以及安全组。

详细的操作步骤，请参见《云数据迁移用户指南》中的[创建集群](#)章节。

步骤2 在新建的CDM集群中，创建一个源连接。

CDM支持从多种数据源迁移数据到CloudTable，请根据实际的数据源创建一个连接作为源连接。该连接用于CDM连接数据源，从数据源读取数据。

有关新建连接的详细操作，请参见[创建CDM连接](#)章节。

步骤3 在新建的CDM集群中，创建一个目的连接。

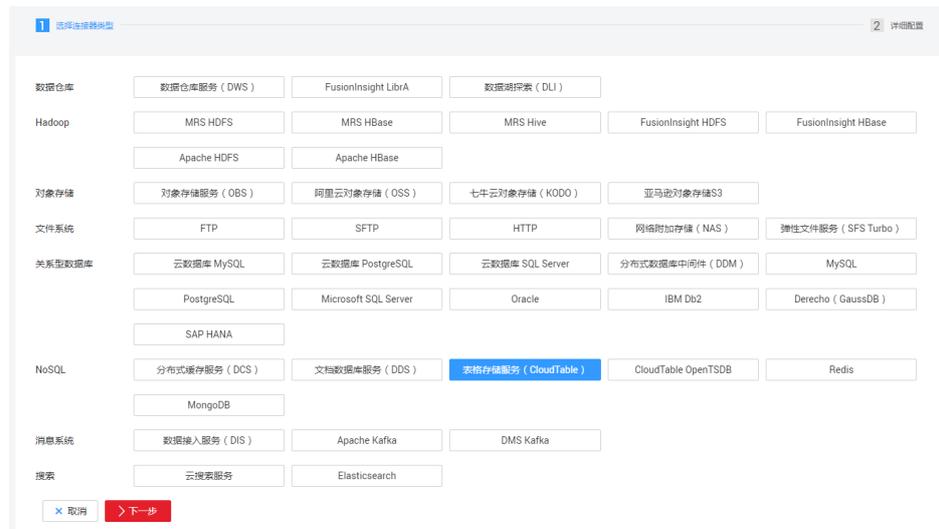
在CDM集群中创建一个连接类型为“表格存储服务（CloudTable）”的连接作为目的连接。该连接用于CDM连接CloudTable集群，将从数据源读取的数据导入到CloudTable集群的HBase表中。

有关新建连接的详细操作，请参见《云数据迁移用户指南》中的[创建CDM连接](#)章节。

新建目的连接的示例如下：

1. 在集群管理页面，找到已创建的CDM集群，单击该CDM集群后的“作业管理”，进入作业管理界面。
2. 选择“连接管理 > 新建连接”，进入选择连接器类型的页面，选择“表格存储服务（CloudTable）”，然后单击“下一步”。

图 2-8 选择连接器类型



3. 配置连接参数。

- “名称”：输入连接名称，根据连接的数据源类型，用户可自定义便于记忆、区分的连接名。例如“cloudtable_connect”。
- “ZK连接地址”：需配置为CloudTable的“ZK连接地址（内网）”。登录表格存储服务管理控制台，在左侧导航树单击集群管理，然后在集群列表中找到所需要的集群，并获取相应的“ZK连接地址（内网）”。
- “IAM统一身份认证”：设置为否。
- “IAM统一身份认证”：表格存储服务已经不使用IAM认证功能，所以填“否”。
- “运行模式”：保留默认值即可。

图 2-9 连接参数配置

* 名称	<input type="text" value="cloudtable_connector"/>
* 连接器	<input type="text" value="HBase"/>
* HBase类型	<input type="text" value="CloudTable"/>
* ZK链接地址	<input type="text" value="cloudtable-demo-zk1.cloudt: 选择"/>
IAM统一身份认证	<input type="radio" value="是"/> <input checked="" type="radio" value="否"/>
* 运行模式	<input type="text" value="STANDALONE"/>

4. 单击“保存”。

步骤4 在CDM集群中，在“表/文件迁移”页面，新建一个作业。

有关新建“表/文件迁移”作业的详细操作，请参见《云数据迁移用户指南》的[表/文件迁移](#)章节。

说明

在新建“表/文件迁移”类型的作业时，如果待迁移的源端的表/文件在目的端CloudTable集群中不存在相应的HBase表，在作业中为目的端设置表名和字段映射（支持从源端拷贝字段名）后，作业运行时会自动在目的端新建对应的HBase表。

下面以ECS上的SFTP数据源为例，创建作业步骤如下：

1. 在集群管理页面，找到刚创建的CDM集群，单击该CDM集群后的“作业管理”，进入作业管理界面。
2. 选择“表/文件迁移 > 新建作业”。
3. 基本信息配置如下：

图 2-10 基本信息配置

作业配置

* 作业名称

源连接配置

* 源连接名称

* 源目录或文件

* 文件格式

[显示高级属性](#)

目的连接配置

* 目的连接名称

* 表名

* 导入前清空数据 是 否

[显示高级属性](#)

4. 字段映射配置如下：

图 2-11 字段映射

源字段			目的字段			
列号	样值	操作	列名	行键	操作	
1	i	<input type="checkbox"/> <input type="checkbox"/>	info	c1	<input checked="" type="checkbox"/> <input type="checkbox"/>	
2	b	<input type="checkbox"/> <input type="checkbox"/>	info	c2	<input type="checkbox"/> <input type="checkbox"/>	
3	c	<input type="checkbox"/> <input type="checkbox"/>	info	c3	<input type="checkbox"/> <input type="checkbox"/>	

[添加字段+](#)

5. 任务配置如下：

图 2-12 任务配置

任务配置

抽取并发数

是否定时执行

[显示高级属性](#)

6. 单击“保存并运行”创建并运行作业。

----结束

在 CloudTable 中查看导入的数据

步骤1 准备Linux弹性云服务器。

假设该弹性云服务器名称为“ecs_20170916”，准备的具体操作请参见[准备弹性云服务器](#)章节。

步骤2 安装客户端并启动Shell访问CloudTable集群。

使用HBase Shell访问集群的具体操作请参见[使用HBase Shell连接HBase普通集群](#)章节。

- 步骤3** 在CloudTable客户端中执行查询命令，查看使用CDM迁移到CloudTable的数据。
命令示例如下，其中table_name请替换为在CDM集群中新建作业时指定的表名。

```
scan 'table_name'
```

----结束

2.4.2 使用 Import 工具将 SequenceFile 数据导入 HBase 集群

使用Import工具，可以导入HBase的SequenceFile数据文件到CloudTable的HBase中。Import工具是客户端工具安装包中自带的一个工具。

说明

SequenceFile文件是指使用Export工具从HBase导出的数据文件。

- 步骤1** 准备一台安装了CloudTable HBase客户端的服务器。客户端安装方法请参考[使用HBase Shell连接HBase普通集群](#)
- 步骤2** 把SequenceFile文件目录上传到HBase的客户端所在服务器，假设存放目录为“/tmp/sequencefile”。
- 步骤3** Import工具需要在客户端主机的操作系统的shell界面中执行。在客户端机器上进入HBase目录后执行Import工具，Import工具的命令格式如下：

```
./bin/hbase org.apache.hadoop.hbase.mapreduce.Import <TableName>  
<InputPath>
```

“TableName”：要导入数据的表名。

“InputPath”：SequenceFile文件的目录。在本地文件系统时必须要以file://开头。

例如：

```
./bin/hbase org.apache.hadoop.hbase.mapreduce.Import t1 file:///tmp/sequencefile
```

----结束

2.4.3 使用 CopyTable 将源数据导入 HBase 集群

CopyTable是HBase提供的一个实用工具，可以将部分或全部表复制到同一个集群或另一个集群中。目标表必须预先存在。在CloudTable客户端工具中已经包含了CopyTable工具，用户部署客户端工具后，即可使用CopyTable工具导入数据到CloudTable集群。

使用 CopyTable 工具导入数据

- 步骤1** 准备Linux弹性云服务器作为客户端主机，并在客户端主机上部署CloudTable HBase客户端工具。

具体操作请参见[使用HBase Shell连接HBase普通集群](#)。

在部署客户端工具时，ZK链接地址请配置为源表所在的CloudTable HBase集群的“ZK链接地址（内网）”。

- 步骤2** （可选）如果要将表复制到另一个集群，请获取HBase目标集群的“ZK链接地址（内网）”。

登录表格存储服务管理控制台，在左侧导航树单击集群管理，然后在集群列表中找到所需要的集群，并获取相应的“ZK链接地址（内网）”。

步骤3 在使用CopyTable复制表数据前，需要先确保在CloudTable HBase目标集群中目标表已经存在。如果目标表不存在，请先创建目标表。

创建表的具体步骤，请参考[HBase入门使用](#)。

步骤4 在客户端主机，打开命令行终端窗口，进入客户端工具安装目录的hbase目录下，然后执行CopyTable命令导入数据到CloudTable集群。

命令示例如下，该示例是将“TestTable”表指定1小时范围内的数据复制到目标集群：

```
cd ${客户端工具的安装目录}/hbase
./bin/hbase org.apache.hadoop.hbase.mapreduce.CopyTable --starttime=1265875194289 --
endtime=1265878794289 --peer.adr=${CloudTable目标集群的“ZK链接地址”}/hbase --
families=myOldCf:myNewCf,cf2,cf3 TestTable
```

----结束

CopyTable 命令简介

CopyTable的命令格式如下：

```
CopyTable [general options] [--starttime=X] [--endtime=Y] [--new.name=NEW] [--peer.adr=ADR]
<tablename>
```

有关上述CopyTable命令的详细信息，请参见HBase官方文档中的[CopyTable](#)。

常用选项说明如下：

- startrow：开始行。
- stoprow：停止行。
- starttime：开始时间（以毫秒为单位的unixtime）。如果未指定endtime意味着从开始时间到永久。
- endtime：结束时间。如果未指定starttime，则忽略。
- versions：要复制的单元格版本数。
- new.name：新表的名称。
- peer.adr：目标集群的地址，格式为
hbase.zookeeper.quorum:hbase.zookeeper.client.port:zookeeper.znode.paren，
对于HBase集群而言就是“\${CloudTable目标集群的“ZK链接地址”}/hbase”。
- families：要复制的列族列表，多个列族之间以逗号分隔。
如果要从sourceCfName复制到destCfName，请指定为
sourceCfName:destCfName。
如果复制后列族名称保持不变，只需指定cfName。
- all.cells：对删除标记和已删除的单元格也进行复制。

参数说明如下。

tablename：要复制的表的名称。

2.5 HBase 企业级能力增强

2.5.1 配置 HBase 冷热分离

2.5.1.1 冷热分离概述

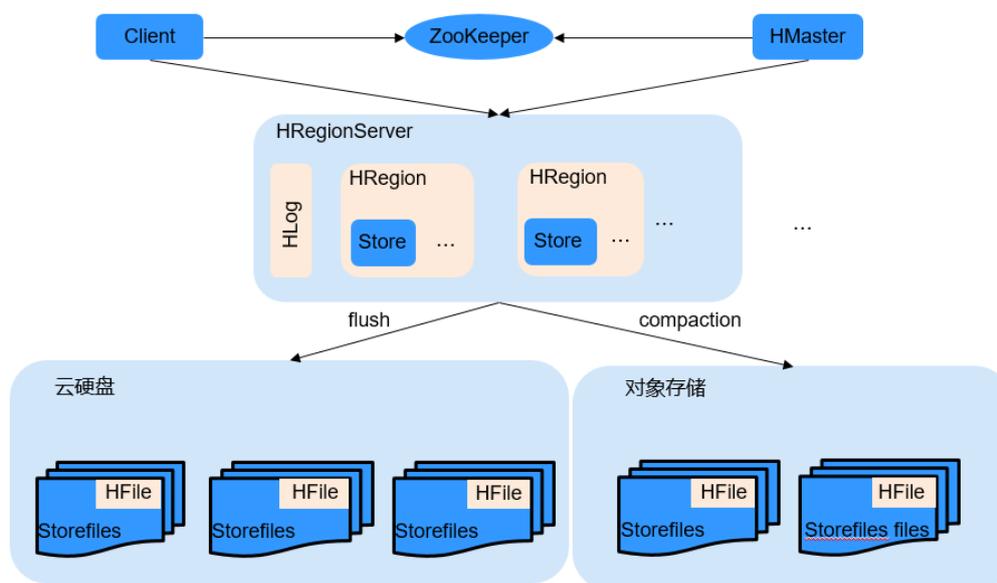
在海量大数据场景下，表中的部分业务数据随着时间的推移仅作为归档数据或者访问频率很低，同时这部分历史数据体量非常大，比如订单数据或者监控数据，如果降低这部分数据的存储成本将会极大的节省企业的成本。CloudTable HBase支持冷热分离功能，将数据分类存储在不同介质上，有效降低存储成本。针对数据使用场景，可以将数据按照时间访问频率分为热数据和冷数据。冷热数据主要从数据访问频率、更新频率划分。

- Hot（热数据）：访问、更新频率较高，对访问的响应时间要求很高的数据。
- Cold（冷数据）：不允许更新或更新访问频率较低，对访问的响应时间要求不高的数据。

原理介绍

CloudTable HBase支持对同一张表的数据进行冷热分离存储。用户在表上配置数据冷热时间分界点后，HBase会依赖用户写入数据的时间戳（毫秒）和时间分界点来判断数据的冷热。数据开始存储在热存储上，随着时间的推移慢慢往冷存储上迁移。同时用户可以任意变更数据的冷热分界点，数据可以从热存储到冷存储，也可以从冷存储到热存储。

图 2-13 原理图



使用限制

- 普通IO读IOPS的能力降低，因此只适用于低频查询的场景。
- 普通IO不适合并发大量读请求的场景，并发大量读请求可能会导致请求异常。
- 冷热分离特性仅支持HBase 2.4.14及以上版本。
- 冷热分离场景下，热盘不支持扩容。
- 普通IO集群不支持冷热分离。
- HBase冷热分离只能在创建集群时开启。

2.5.1.2 通过 HBase Shell 配置 HBase 冷热分离

HBase支持冷热分离功能，可以将冷热数据存储在不同的介质中，有效提升数据的查询效率，同时降低数据存储成本。此章节主要介绍通过HBase Shell配置HBase冷热分离。

前提条件

- 已创建**HBase集群**。
- 已安装HBase客户端。

步骤一：开启 HBase 冷热分离

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角，选择区域。

步骤3 单击集群管理界面右上角的“购买集群”，

步骤4 进入创建集群界面，在集群购买页面选择HBase数据库引擎并勾选高级特性“冷热分离”，创建后的集群就开启了冷热分离特性。

图 2-14 开启冷热分离



步骤二：设置冷热分离界线

步骤1 连接HBase集群，请参见[使用HBase Shell连接HBase普通集群](#)。

步骤2 设置表的冷热分界线。冷热分离时间点要大于Major Compaction执行周期。Major Compaction默认执行周期为7天。

- 创建冷热分离表。

```
hbase(main):002:0> create 'hot_cold_table', {NAME=>'f', COLD_BOUNDARY=>'86400'}
```

参数说明。

 - NAME：需要冷热分离的列族。
 - COLD_BOUNDARY：冷热分离时间点，单位为秒（s）。例如 COLD_BOUNDARY为86400，代表86400秒（一天）前写入的数据会被自动归档到冷存储。
- 取消冷热分离。

```
hbase(main):004:0> alter 'hot_cold_table', {NAME=>'f', COLD_BOUNDARY=>''}
```
- 为已经存在的表设置冷热分离，或者修改冷热分离分界线，单位为秒。

```
hbase(main):005:0> alter 'hot_cold_table', {NAME=>'f', COLD_BOUNDARY=>'86400'}
```

步骤3 查询冷热分离是否设置或者修改成功。

```
hbase(main):002:0> desc 'hot_cold_table'
Table hot_cold_table is ENABLED
hot_cold_table
COLUMN FAMILIES DESCRIPTION
{NAME => 'f', VERSIONS => '1', KEEP_DELETED_CELLS => 'FALSE', DATA_BLOCK_ENCODING => 'NONE', TTL
=> 'FOREVER', MIN_VERSIONS => '0', REPLICATION_SCOPE => '0', BLOOMFILTER => 'ROW', IN_MEMORY =>
```

```
'false', COMPRESS => 'NONE', BLOCKCACHE => 'true', BLOCKSIZE => '65536', METADATA => {'COLD_BOUNDARY' => '86400'}}
1 row(s)
Quota is disabled
Took 0.0339 seconds
```

----结束

步骤三：插入数据

冷热分离的表与普通表的数据写入方式完全一致，数据会先存储在热存储（云硬盘）中。随着时间的推移，如果一行数据满足：当前时间-时间列值>COLD_BOUNDARY设置的值条件，则会在执行Compaction时被归档到冷存储（对象存储）中。

执行“put”命令往指定表插入一条记录，需要指定表的名称，主键，自定义列，以及插入的具体值。

```
hbase(main):004:0> put 'hot_cold_table','row1','cf:a','value1'
0 row(s) in 0.2720 seconds
```

命令中各参数分别代表如下含义：

- hot_cold_table：表的名称。
- row1：主键。
- cf: a：自定义的列。
- value1：插入的值。

步骤四：数据查询

由于冷热数据都在同一张表中，因此用户所有的查询操作都只需在一张表内进行。在查询时，建议通过配置TimeRange来指定查询的时间范围，系统将会根据指定的时间范围决定查询模式，即仅查询热存储、仅查询冷存储或同时查询冷存储和热存储。如果查询时未限定时间范围，则会导致查询冷数据。在这种情况下，查询吞吐量会受到冷存储的限制。

冷存储中的数据常用于归档，会很少访问。如果冷存储中的数据被大量频繁请求访问，请检查冷热数据边界（COLD_BOUNDARY）配置是否正确。如果频繁查询的大量数据在冷存储中将会限制查询的性能。

如果冷存储中存储的一行数据中的某个字段更新，则更新的字段存储在热存储中。如果指定HOT_ONLY或TimeRange参数仅查询热存储中的数据，则只返回更新的字段。如果要返回整行的数据，则必须在不指定HOT_ONLY或TimeRange参数的情况下执行查询，或者确保TimeRange指定的时间范围涵盖从插入行的时间点到最后更新行的时间点的时间段。因此，建议您不要更新存储在冷存储中的数据。

- 随机查询Get。
 - 不指定HOT_ONLY参数来查询数据。在这种情况下，将会查询冷存储中的数据。

```
hbase(main):001:0> get 'hot_cold_table', 'row1'
```
 - 通过指定HOT_ONLY参数来查询数据。在这种情况下，只会查询热存储中的数据。

```
hbase(main):002:0> get 'hot_cold_table', 'row1', {HOT_ONLY=>true}
```
 - 通过指定TimeRange参数来查询数据。在这种情况下，CloudTable将会比较TimeRange和冷热边界值，以确定是只查询热存储还是冷存储中的数据，还是同时查询热冷存储中的数据。

```
hbase(main):003:0> get 'hot_cold_table', 'row1', {TIMERANGE => [0, 1568203111265]}
```

说明

TimeRange: 查询的时间范围。范围中的时间是UNIX时间戳,表示自1970年1月1日00:00 UTC以来经过的毫秒数。

- 范围查询Scan。
 - 不指定HOT_ONLY参数来查询数据。在这种情况下,将会查询冷存储中的数据。

```
hbase(main):001:0> scan 'hot_cold_table', {STARTROW =>'row1', STOPROW=>'row9'}
```
 - 通过指定HOT_ONLY参数来查询数据。在这种情况下,只会查询热存储中的数据。

```
hbase(main):002:0> scan 'hot_cold_table', {STARTROW =>'row1', STOPROW=>'row9', HOT_ONLY=>true}
```
 - 通过指定TimeRange参数来查询数据。在这种情况下,CloudTable将会比较TimeRange和冷热边界值,以确定是只查询热存储还是冷存储中的数据,还是同时查询热冷存储中的数据。

```
hbase(main):003:0> scan 'hot_cold_table', {STARTROW =>'row1', STOPROW=>'row9', TIMERANGE => [0, 1568203111265]}
```

说明

TimeRange: 查询的时间范围。范围中的时间是UNIX时间戳,表示自1970年1月1日00:00 UTC以来经过的毫秒数。

- 优先查询热数据。

在查询客户所有记录等信息的范围查询中,CloudTable可以扫描热存储和冷存储中的数据。查询结果将根据数据行按写入表时的时间戳降序返回。在大多数情况下,热数据出现在冷数据之前。如果在范围查询中没有配置HOT_ONLY参数,CloudTable将会扫描热存储和冷存储中的数据,查询响应时间将会增加。如果启用热数据优先特性,CloudTable会优先查询热存储中的数据。只有当热存储中的行数小于要查询的最小行数时,才会查询冷存储中的数据。这样,减少了冷存储的访问提高了响应速度。

```
hbase(main):001:0> scan 'hot_cold_table', {STARTROW =>'row1', STOPROW=>'row9',COLD_HOT_MERGE=>true}
```
- Major Compaction命令。
 - 合并表所有分区的热数据区。

```
hbase(main):002:0> major_compact 'hot_cold_table', nil, 'NORMAL', 'HOT'
```
 - 合并表所有分区的冷数据区。

```
hbase(main):002:0> major_compact 'hot_cold_table', nil, 'NORMAL', 'COLD'
```
 - 合并表所有分区的热冷数据区。

```
hbase(main):002:0> major_compact 'hot_cold_table', nil, 'NORMAL', 'ALL'
```

2.5.1.3 使用 JAVA API 配置 HBase 冷热分离

HBase支持冷热分离功能,可以将冷热数据存储在不同的介质中,有效提升数据的查询效率,同时降低数据存储成本。此章节主要介绍通过JAVA API配置HBase冷热分离。

前提条件

- 已创建[HBase集群](#)。
- 已安装HBase客户端。

步骤一：开启 HBase 冷热分离

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角，选择区域。

步骤3 单击集群管理界面右上角的“购买集群”，

步骤4 进入创建集群界面，在集群购买页面选择HBase数据库引擎并勾选高级特性“冷热分离”，创建后的集群就开启了冷热分离特性。

图 2-15 开启冷热分离



----结束

步骤二：设置表的冷热分界线

步骤1 通过Java API接口访问HBase集群。

步骤2 设置冷热分界线。

- 新建冷热分离表。COLD_BOUNDARY设置冷热分离时间分界点，单位为秒，示例表示1天之前的数据归档为冷数据。

```
Admin admin = connection.getAdmin();
TableName tableName = TableName.valueOf("hot_cold_table");
HTableDescriptor descriptor = new HTableDescriptor(tableName);
HColumnDescriptor cf = new HColumnDescriptor("f");
cf.setValue(HColumnDescriptor.COLD_BOUNDARY, "86400");
descriptor.addFamily(cf);
admin.createTable(descriptor);
```

- 取消冷热分离。

```
HTableDescriptor descriptor = admin.getTableDescriptor(tableName);
HColumnDescriptor cf = descriptor.getFamily("f").getBytes();
cf.setValue(HColumnDescriptor.COLD_BOUNDARY, null);
admin.modifyTable(tableName, descriptor);
```

- 为已经存在的表设置冷热分离功能，或者修改冷热分离分界线。COLD_BOUNDARY设置冷热分离时间分界点，单位为秒，示例表示1天之前的数据归档为冷数据。

```
HTableDescriptor descriptor = admin.getTableDescriptor(tableName);
HColumnDescriptor cf = descriptor.getFamily("f").getBytes();
cf.setValue(HColumnDescriptor.COLD_BOUNDARY, "86400");
admin.modifyTable(tableName, descriptor);
```

----结束

步骤三：插入数据

冷热分离的表与普通表的数据写入方式完全一致，数据会先存储在热存储（云硬盘）中。随着时间的推移，如果一行数据满足：当前时间-时间列值>COLD_BOUNDARY设置的值条件，则会在执行Compaction时被归档到冷存储（对象存储）中。

使用Java API写入数据。

步骤四：数据查询

由于冷热数据都在同一张表中，因此用户所有的查询操作都只需在一张表内进行。在查询时，建议通过配置TimeRange来指定查询的时间范围，系统将会根据指定的时间范围决定查询模式，即仅查询热存储、仅查询冷存储或同时查询冷存储和热存储。如果查询时未限定时间范围，则会导致查询冷数据。在这种情况下，查询吞吐量会受到冷存储的限制。

冷存储中的数据常用于归档，会很少访问。如果冷存储中的数据被大量频繁请求访问，请检查冷热数据边界（COLD_BOUNDARY）配置是否正确。如果频繁查询的大量数据在冷存储中将会限制查询的性能。

如果冷存储中存储的一行数据中的某个字段更新，则更新的字段存储在热存储中。如果指定HOT_ONLY或TimeRange参数仅查询热存储中的数据，则只返回更新的字段。如果要返回整行的数据，则必须在不指定HOT_ONLY或TimeRange参数的情况下执行查询，或者确保TimeRange指定的时间范围涵盖从插入行的时间点到最后更新行的时间点的时间段。因此，建议您不要更新存储在冷存储中的数据。

- 随机查询Get。

- 不指定HOT_ONLY参数来查询数据。在这种情况下，将会查询冷存储中的数据。

```
Get get = new Get("row1".getBytes());
```

- 通过指定HOT_ONLY参数来查询数据。在这种情况下，只会查询热存储中的数据。

```
Get get = new Get("row1".getBytes());  
get.setAttribute(HBaseConstants.HOT_ONLY, Bytes.toBytes(true));
```

- 通过指定TimeRange参数来查询数据。在这种情况下，CloudTable将会比较TimeRange和冷热边界值（COLD_BOUNDARY），以确定是只查询热存储还是冷存储中的数据，还是同时查询热冷存储中的数据。

```
Get get = new Get("row1".getBytes());  
get.setTimeRange(0, 1568203111265)
```

TimeRange：查询的时间范围。范围中的时间是UNIX时间戳，表示自1970年1月1日00:00 UTC以来经过的毫秒数。

- 范围查询Scan。

- 不指定HOT_ONLY参数来查询数据。在这种情况下，将会查询冷存储中的数据。

```
TableName tableName = TableName.valueOf("chsTable");  
Table table = connection.getTable(tableName);  
Scan scan = new Scan();  
ResultScanner scanner = table.getScanner(scan);
```

- 通过指定HOT_ONLY参数来查询数据。在这种情况下，只会查询热存储中的数据。

```
Scan scan = new Scan();  
scan.setAttribute(HBaseConstants.HOT_ONLY, Bytes.toBytes(true));
```

- 通过指定TimeRange参数来查询数据。在这种情况下，CloudTable将会比较TimeRange和冷热边界值（COLD_BOUNDARY），以确定是只查询热存储还是冷存储中的数据，还是同时查询热冷存储中的数据。

```
Scan scan = new Scan();  
scan.setTimeRange(0, 1568203111265);
```

TimeRange：查询的时间范围。范围中的时间是UNIX时间戳，表示自1970年1月1日00:00 UTC以来经过的毫秒数。

- 优先查询热数据。

在查询客户所有记录等信息的范围查询中，CloudTable可以扫描热存储和冷存储中的数据。查询结果将根据数据行按写入表时的时间戳降序返回。在大多数情况

下，热数据出现在冷数据之前。如果在范围查询中没有配置HOT_ONLY参数，CloudTable将会扫描热存储和冷存储中的数据，查询响应时间将会增加。如果启用热数据优先特性，CloudTable会优先查询热存储中的数据。只有当热存储中的行数小于要查询的最小行数时，才会查询冷存储中的数据。这样，减少了冷存储的访问提高了响应速度。

```
TableName tableName = TableName.valueOf("hot_cold_table");
Table table = connection.getTable(tableName);
Scan scan = new Scan();
scan.setAttribute(HBaseConstants.COLD_HOT_MERGE, Bytes.toBytes(true));
scanner = table.getScanner(scan);
```

- Major Compaction命令。

- 合并表所有分区的热数据区。

```
Admin admin = connection.getAdmin();
TableName tableName = TableName.valueOf("hot_cold_table");
admin.majorCompact(tableName, null, CompactType.NORMAL, CompactionScopeType.HOT);
```

- 合并表所有分区的冷数据区。

```
Admin admin = connection.getAdmin();
TableName tableName = TableName.valueOf("hot_cold_table");
admin.majorCompact(tableName, null, CompactType.NORMAL, CompactionScopeType.COLD);
```

- 合并表所有分区的热冷数据区。

```
Admin admin = connection.getAdmin();
TableName tableName = TableName.valueOf("hot_cold_table");
admin.majorCompact(tableName, null, CompactType.NORMAL, CompactionScopeType.ALL);
```

2.5.2 配置 HBase 全局二级索引

2.5.2.1 HBase 全局二级索引介绍

场景介绍

使用HBase二级索引可以加速带Filter的条件查询，支持HIndex（本地索引，即Local Secondary Index）和全局二级索引（Global Secondary Index）。全局二级索引相较于本地索引（HIndex），查询性能更好，适合读时延要求高的场景。

HBase全局二级索引，使用独立的索引表存储索引数据。当给定的查询条件可以命中索引时，可以将对数据表的全表查询转换为对索引表的精确范围查询，提升查询速度。开启全局二级索引特性后，应用侧代码无需特殊修改，简单易用。

HBase全局二级索引支持以下重点特性：

- **复合索引**

支持指定多个列作为索引列（支持跨列族）。

- **覆盖索引**

支持指定多个列/列族作为覆盖列/列族冗余存储到索引表中，用于支持索引查询中对非索引列的快速查询。

- **索引TTL**

支持索引表TTL，用于支持数据表开启TTL的场景，为了保障与数据表的一致性，索引表TTL将自动继承数据表索引列和覆盖列的TTL，不支持手动指定。

- **索引在线变更**

支持索引在线创建、删除和修改状态，不影响数据表读写。

- **索引在线修复**

当查询命中的索引数据无效时，可以触发索引修复，保障最终查询结果正确。

- **索引工具**

支持索引一致性检查、索引修复、索引创建/删除/修改状态、索引数据重建等功能。

HBase 全局索引/二级索引限制与约束

- **使用场景限制**

- GSI不支持与HIndex (LSI) 同时使用，即不支持在同一个数据表上同时创建本地索引与全局索引。
- 索引表不支持容灾。
- 不支持直接对索引表执行DISABLE、DROP、MODIFY和TRUNCATE操作。
- 索引DDL操作支持修改索引状态、删除索引、创建索引；不支持修改索引定义，如需修改，请先删除后重新创建。
- 已存在数据的表不能创建全局二级索引。

- **索引创建约束**

- 索引名需要符合正则要求，不支持其他字符。正则要求支持的字符为：**[a-zA-Z_0-9-.]**：
- 数据表必须存在，要创建的索引不能已存在。
- 索引表不支持多版本。
不支持在多版本（VERSION>1）的数据表上创建索引，且索引表的版本VERSION=1。
- 单个数据表的索引个数不能超过5个。
不建议为单个数据表创建过多索引，索引数量过多会造成存储成本较高，写入耗时大。如果需创建超过5个索引，请在HMaster的自定义配置“hbase.hmaster.config.expandor”中新增参数“hbase.gsi.max.index.count.per.table”，设置值大于5，并重启HMaster使配置生效。
- 索引名长度不能超过18个字符。
不建议使用过长的索引名。如果需创建较长的索引名，请在HMaster的自定义配置“hbase.hmaster.config.expandor”中新增参数“hbase.gsi.max.index.name.length”，设置值大于18，并重启HMaster使配置生效。
- 不支持为索引表创建索引。
不支持嵌套创建多个索引，索引表仅用于加速查询，不承担数据表功能。
- 不支持创建可以被已有索引覆盖的索引。
新建索引时，如果之前已存在的索引能够完全覆盖新建的索引（即创建的索引是已有索引的子集），则无法创建此索引，重复功能的索引会造成存储浪费。例如，以下操作将无法创建索引2：

创建数据表：

```
create 't1','cf1'
```

创建索引1：

```
hbase org.apache.hadoop.hbase.index.global.mapreduce.GlobalTableIndexer - Dtablename.to.index='t1' -Dindexspecs.to.add='idx1=>cf1:[q1],[q2]'
```

创建索引2：

```
hbase org.apache.hadoop.hbase.index.global.mapreduce.GlobalTableIndexer - Dtablename.to.index='t1' -Dindexspecs.to.add='idx2=>cf1:[q1]'
```

- 不支持在同一张数据表上创建同名索引，支持在不同数据表上创建同名索引。
- 索引表列族TTL继承原表，索引列族TTL必须一致。
索引表所有列族TTL相同，继承自数据表，要求数据表中相关列族TTL必须一致，否则无法创建相关索引。
- 不支持自定义索引表其他属性。

2.5.2.2 创建 HBase 全局二级索引

场景介绍

对于未建立索引的用户表，该工具允许用户同时添加和构建索引。

使用方法

在HBase客户端执行以下命令即可添加或创建索引，执行命令后，指定的索引将被添加到表中并且将处于ACTIVE状态：

```
hbase org.apache.hadoop.hbase.hindex.global.mapreduce.GlobalTableIndexer -
Dtablename.to.index='table' -Dindexspecs.to.add='idx1=>cf1:[c1->string],[c2]#idx2=>cf2:[c1->string],
[c2]#idx3=>cf1:[c1];cf2:[c1]' -Dindexspecs.covered.family.to.add='idx2=>cf1' -
Dindexspecs.covered.to.add='idx1=>cf1:[c3],[c4]' -Dindexspecs.coveredallcolumn.to.add='idx3=>true' -
Dindexspecs.splitkeys.to.set='idx1=>[\x010,\x011,\x012]#idx2=>[\x01a,\x01b,\x01c]#idx3=>[\x01d,\x01e,\x01f]'
```

相关参数介绍如下：

- **tablename.to.index**：表示创建索引的数据表的名称。
- **indexspecs.to.add**：表示与索引名与对应数据表的列的映射（索引列定义）。
- **indexspecs.covered.to.add**（可选）：表示索引中冗余存储的数据表的列（覆盖列定义）。
- **indexspecs.covered.family.to.add**（可选）：表示索引表冗余存储的数据表的列族（覆盖列族定义）。
- **indexspecs.coveredallcolumn.to.add**（可选）：表示索引表冗余存储数据表中的所有数据（覆盖所有列）。
- **indexspecs.splitkeys.to.set**（可选）：表示索引表预分区切分点，**建议指定**，避免索引表Region成为热点。预分区指定格式为：
 - '#'用于分隔索引。
 - splitkey包含在'[]'中。
 - ','用于分隔splitkey。

说明

预分区每个splitkey必须由\x01开头。

- **indexspecs.to.addandbuild**（可选）：表示创建时同时生成索引数据，数据表数据量较大时**不建议使用**，建议使用索引数据生成工具完成索引数据生成。

上述命令中的参数描述如下：

- **idx1**、**idx2**、**idx3**：表示索引名称。
- **cf1**、**cf2**：表示列族名称。
- **c1**、**c2**、**c3**、**c4**：表示列名称。

- **string**: 表示数据类型。支持STRING、INTEGER、FLOAT、LONG、DOUBLE、SHORT、BYTE和CHAR。

说明

- '#'用于分隔索引, ';'用于分隔列族, ':'用于分隔列限定符。
- 列名及其数据类型应包含在'[]'中。
- 列名及其数据类型通过' ->'分隔。
- 如果未指定具体列的数据类型, 则使用默认数据类型 (string)。

2.5.2.3 查询 HBase 全局二级索引信息

查询 HBase 全局二级索引信息

用户可以使用全局二级索引工具批量查看某个数据表相关索引的定义及状态。

在HBase客户端执行以下命令可索引的定义及状态:

```
hbase org.apache.hadoop.hbase.hindex.global.mapreduce.GlobalTableIndexer -
Dtablename.to.show='table'
```

相关参数介绍如下:

tablename.to.show: 表示需查看相关索引定义的数据表名称。

查询结果如图2-16所示, 会打印索引列定义、覆盖列定义、TTL、预分区信息、索引状态等:

图 2-16 索引查询结果

```
9888: negotiated timeout = 90000
2023-08-18 10:47:59,784 INFO [main] client.GlobalIndexTracker: GlobalIndexTracker started successfully
IndexName : idx1, IndexColumns : [cf1:c1 -> type:STRING, cf1:c2 -> type:STRING, CoveredColumns : [cf1:c3 -> type:STRING, CoveredFamilies : [], CoveredAllColumns : false, TTL : 2147483647, SplitKeys : [\u001d,\u0011,\u0012], IndexState : ACTIVE
IndexName : idx2, IndexColumns : [cf2:c1 -> type:STRING, cf2:c2 -> type:STRING, CoveredColumns : [], CoveredFamilies : [cf1], CoveredAllColumns : false, TTL : 2147483647, SplitKeys : [\u001a,\u001b,\u001c], IndexState : ACTIVE
IndexName : idx3, IndexColumns : [cf1:c1 -> type:STRING, cf2:c1 -> type:STRING, CoveredColumns : [], CoveredFamilies : [], CoveredAllColumns : true, TTL : 2147483647, SplitKeys : [\u001d,\u001e,\u001f], IndexState : ACTIVE
```

基于索引查询 HBase 表数据

在具有索引的用户表中, 可以使用SingleColumnValueFilter来查询数据。当查询条件可以命中索引时, 查询速度远快于原表查询。

索引的命中规则如下:

- 多个AND条件查询。
 - 当用于查询的列至少包含索引第一个列时, 使用索引会提高查询性能。例如, 为C1、C2和C3创建组合索引。

该索引在以下情况下生效:

```
Filter_Condition ( IndexCol1 ) AND Filter_Condition ( IndexCol2 ) AND Filter_Condition ( IndexCol3 )
Filter_Condition ( IndexCol1 ) AND Filter_Condition ( IndexCol2 )
Filter_Condition ( IndexCol1 ) AND Filter_Condition ( IndexCol3 )
Filter_Condition ( IndexCol1 )
```

该索引在下列情况下不生效:

```
Filter_Condition ( IndexCol2 ) AND Filter_Condition ( IndexCol3 )
Filter_Condition ( IndexCol2 )
Filter_Condition ( IndexCol3 )
```

- 当在查询中使用“索引列和非索引列”进行过滤时, 使用索引可提高查询性能。当非索引列命中覆盖列时, 查询性能最优; 如果有需经常查询的非索引列, 建议定义为覆盖列。例如:

```
Filter_Condition ( IndexCol1 ) AND Filter_Condition ( NonIndexCol1 )
Filter_Condition ( IndexCol1 ) AND Filter_Condition ( IndexCol2 ) AND Filter_Condition
( NonIndexCol1 )
```

- 当多个列用于查询时，只能为组合索引中的最后一列指定值范围，而其他列只能设置为指定值。
例如，为C1、C2和C3创建组合索引。在范围查询中，只能为C3设置数值范围，过滤条件为“C1 = XXX, C2 = XXX, C3 = 数值范围”。
- 多个OR条件查询。
例如，为C1、C2和C3创建组合索引。
 - 仅对索引列首个字段进行进行过滤时（支持范围过滤），使用索引可提高查询性能。
Filter_Condition (IndexCol1) OR Filter_Condition (IndexCol1) OR Filter_Condition (IndexCol1)
 - 对非索引和非索引列进行过滤时，无法命中索引，查询性能不会提高。
Filter_Condition (IndexCol1) OR Filter_Condition (NonIndexCol1)
 - 组合查询时，最外层包含OR条件时无法命中索引，查询性能不会提高。
Filter_Condition (IndexCol1) OR Filter_Condition (NonIndexCol1)
(Filter_Condition (IndexCol1) AND Filter_Condition (IndexCol2)) OR (Filter_Condition (NonIndexCol1)

📖 说明

减少OR条件使用，尤其是OR条件+范围条件，命中索引的情况下也会造成查询大范围查询，速度较慢。

2.5.2.4 修改 HBase 全局二级索引状态

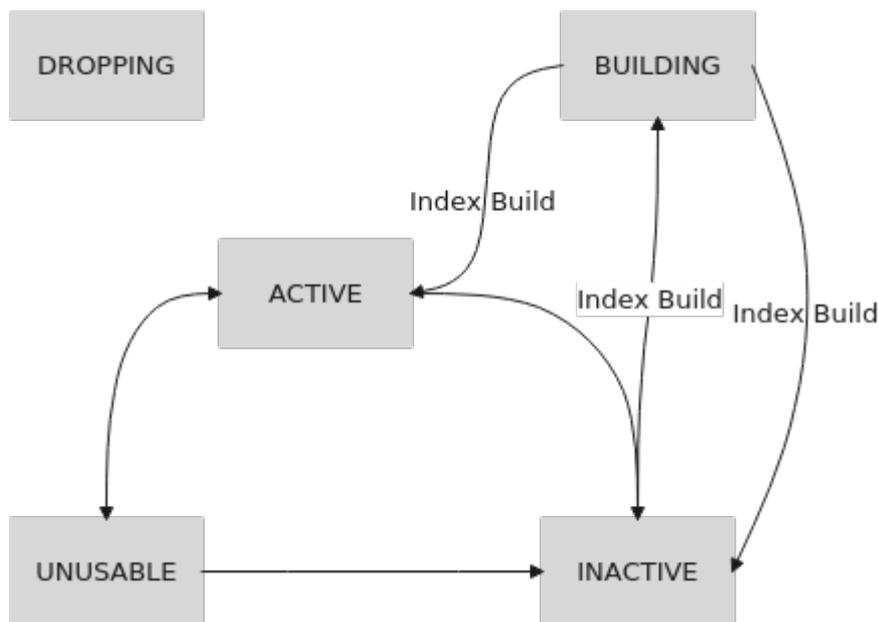
索引状态介绍

索引状态反映了索引当前的使用情况，全局二级索引支持以下五种状态：

- ACTIVE：索引正常，可以正常读写。
- UNUSABLE：索引被禁用，索引数据会正常写入，查询时无法使用这个索引。
- INACTIVE：索引异常，索引数据与数据表不一致，跳过生成这个索引的索引数据，查询数据时无法使用这个索引。
- BUILDING：索引数据正常批量生成，索引数据生成工具执行结束会自动转换到ACTIVE状态，此状态下可以正常读写。
- DROPPING：索引正在被删除，跳过生成这个索引的索引数据，查询数据时无法使用这个索引。

基于工具的索引状态修改，支持图2-17所示的状态转换。

图 2-17 索引状态转换图



场景介绍

用户可以使用全局二级索引工具禁用/启用某个索引。

使用方法

在HBase客户端执行以下命令可禁用/启用某个索引：

```
hbase org.apache.hadoop.hbase.hindex.global.mapreduce.GlobalTableIndexer -
Dtablename.to.index='table' -D[idx_state_opt]='idx1'
```

相关参数介绍如下：

- **tablename.to.index**：表示需修改索引状态的数据表的名称。
- **idx_state_opt**：表示修改索引的目标状态，可选参数如下：
 - **indexnames.to.inactive**：表示将指定的索引转换为INACTIVE状态。
 - **indexnames.to.active**：表示将指定的索引转换为ACTIVE状态。
 - **indexnames.to.unusable**：表示将指定的索引转换为UNUSABLE状态。

例如：修改table表的索引idx1的状态为UNUSABLE：

```
hbase org.apache.hadoop.hbase.hindex.global.mapreduce.GlobalTableIndexer -
Dtablename.to.index='table' -Dindexnames.to.unusable='idx1'
```

执行成功后，再次查看索引信息：

```
hbase org.apache.hadoop.hbase.hindex.global.mapreduce.GlobalTableIndexer -
Dtablename.to.show='table'
```

如图2-18所示，idx1的索引状态已被修改：

图 2-18 idx1 索引状态

```

hbase> hbase org.apache.hadoop.hbase.hindex.global.mapreduce.GlobalTableIndexer -
Dtablename.to.show='table'
IndexName : idx1, IndexColumns : [cf1:c1 -> type:STRING, cf1:c2 -> type:STRING], CoveredColumns : [cf1:c3 -> type:STRING, cf1:c4 -> type:STRING], CoveredFamilies : [], CoveredAllColumns : false, TTL : 2147483647,
SplitKeys : [\x0b,\x011,\x012], IndexState : UNUSABLE
IndexName : idx2, IndexColumns : [cf2:c1 -> type:STRING, cf2:c2 -> type:STRING], CoveredColumns : [], CoveredFamilies : [cf1], CoveredAllColumns : false, TTL : 2147483647, SplitKeys : [\x0a,\x01b,\x01c], IndexSt
ate : ACTIVE
IndexName : idx3, IndexColumns : [cf1:c1 -> type:STRING, cf2:c1 -> type:STRING], CoveredColumns : [], CoveredFamilies : [], CoveredAllColumns : true, TTL : 2147483647, SplitKeys : [\x0d,\x01e,\x01f], IndexState
: ACTIVE
  
```

2.5.2.5 删除 HBase 全局二级索引

场景介绍

用户可以使用全局二级索引工具删除某个索引。

使用方法

在HBase客户端执行以下命令可删除某个索引：

```
hbase org.apache.hadoop.hbase.hindex.global.mapreduce.GlobalTableIndexer -
Dtablename.to.index='table' -Dindexnames.to.drop='idx1#idx2'
```

相关参数介绍如下：

- **tablename.to.index**：表示需删除的索引所在的表名称。
- **indexnames.to.drop**：表示需要删除的索引名称，可以同时指定多个，用#号分隔。

2.6 管理 HBase 集群

2.6.1 查看 HBase 集群状态

登录表格存储服务，在左侧导航树，单击集群管理，在集群管理列表中会显示 CloudTable所有的集群，集群数量较多时，可采用翻页显示，您可以查看任何状态下的集群。

集群管理列表默认按时间顺序排列，时间最近的集群显示在最前端。集群管理列表参数说明如表2-6和表2-7所示。

表 2-6 集群管理列表参数

参数	参数说明
集群名称	集群的名称，创建集群时设置。
集群状态	集群的运行状态信息。集群状态包括创建中、服务中、亚健康、创建失败和冻结。
任务状态	集群任务的状态信息。请参见表2-9。
引擎类型	HBase
集群版本	集群内核版本。
企业项目	<p>企业项目管理帮助您将相关的资源（如具有相同使用用途的资源）集中在一起，按企业项目的方式来管理云资源。</p> <p>说明</p> <ul style="list-style-type: none"> • 当企业业务发生变化，原用户不再拥有企业项目的使用权限时，可将这些用户与该企业项目的关联关系删除。 • 可进行单个删除和批量删除。 • 当用户与企业项目的关联关系被删除后，该用户将无法管理该企业项目，如需再次使用，需要重新给该用户关联企业项目。

参数	参数说明
创建时间	集群创建的时间。
计费模式	集群的计费模式分为按需和包年/包月。
ZK链接地址 (内网)	Zookeeper地址。
操作	<ul style="list-style-type: none"> ● 查看监控信息：单击“查看监控信息”，进入“云服务监控 > 表格存储服务”的监控图表页面。 ● 节点扩容：扩容集群中的计算单元。具体操作，请参见HBase 集群节点扩容。 ● 更多 <ul style="list-style-type: none"> - 磁盘扩容：单击“磁盘扩容”进行扩容。具体操作方法，请参见HBase 集群磁盘扩容。 - 规格扩容：单击“规格扩容”进行扩容。具体操作方法，请参见HBase 集群规格变更。 - 转包周期：单击操作列“更多 > 转包周期”，计费模式转为包年/包月。 - 退订/释放：单击操作列“更多 > 退订”，释放包周期。 - 重启：单击“重启”，重启集群。操作方法，请参见重启 HBase 集群。 - 删除：单击“删除”，删除一个集群。操作方法，请参见删除 HBase 集群。 - 开启Thrift Server：单击“开启Thrift Server”，集群开启Thrift Server。 - 关闭Thrift Server：单击“关闭Thrift Server”，集群关闭Thrift Server。

表 2-7 按钮说明

按钮	说明
	单击  查看所有项目。
	在搜索框中输入集群名称，单击  ，搜索集群。
	单击  ，手动刷新集群列表。

集群状态

表 2-8 集群状态说明

状态	说明
创建中	表示集群正在创建中。
服务中	当集群创建成功并正常提供服务时，其集群状态为服务中。
亚健康	当超过指定时间监控不到集群状态时，集群状态将显示为“亚健康”。“亚健康”状态的集群需要人工干预让集群恢复正常，比如，可以通过重启集群尝试恢复正常状态。
创建失败	表示集群创建失败。
冻结	续费时所支付的余额已不足，则其状态为“冻结”。 当集群状态为“冻结”时，您需要通过续费保证账户余额不为0才能解冻。 说明 冻结期间集群不可用且关闭集群中所有云主机，解冻后集群自动恢复为“服务中”状态。如果用户没有续费，冻结期限到期后集群会被删除。

任务状态

表 2-9 任务状态说明

状态	说明
删除中	表示集群正在删除中。
重启中	表示集群正在重启中。
开启Thrift Server中	表示集群正在开启Thrift Server。
关闭Thrift Server中	表示集群正在关闭Thrift Server。
节点扩容中	表示集群正在节点扩容。
磁盘扩容中	表示集群正在磁盘扩容。
规格变更中	表示集群正在变更规格。

2.6.2 查看 HBase 集群详情

集群创建完成后，可对集群进行监控和管理。在CloudTable管理控制台单击集群管理，在集群列表中找到所要查看的集群并单击集群名称，进入集群基本信息页面，用户可查看集群信息和集群的网络配置。

集群基本信息参数说明如[表2-10](#)和[表2-11](#)所示。

表 2-10 集群信息

参数	参数说明
集群名称	集群的名称，创建集群时设置。
集群ID	集群的ID。
集群状态	集群状态信息。
已使用存储容量 (GB)	显示已经使用的存储容量。
计费模式	显示集群的计费模式。
集群版本	集群内核版本。
创建时间	显示集群创建的时间。
企业项目	<p>企业项目管理帮助您将相关的资源（如具有相同使用用途的资源）集中在一起，按企业项目的方式来管理云资源。</p> <p>说明</p> <ul style="list-style-type: none"> 当企业业务发生变化，原用户不再拥有企业项目的使用权限时，可将这些用户与该企业项目的关联关系删除。 可进行单个删除和批量删除。 当用户与企业项目的关联关系被删除后，该用户将无法管理该企业项目，如需再次使用，需要重新给该用户关联企业项目。
ZK链接地址 (内网)	<p>Zookeeper地址。</p> <p>用户可以单击 ，将ZK链接地址复制到剪切板。</p>
热存储已使用容量	集群开启冷热分离高级特性，热数据的存储容量。
冷存储已使用容量	集群开启冷热分离高级特性，冷数据的存储容量。
是否开启安全通道	如果界面显示“否”，表示未开启安全通道；如果界面显示“是”，表示开启安全通道。
是否开启Thrift Server	如果界面显示“否”，表示未开启Thrift Server；如果界面显示“是”，表示开启Thrift Server，并且显示Thrift Server的IP地址。
Thrift Server IP	如果开启Thrift Server，集群详情页面会显示此字段；如果未开启Thrift Server，集群详情页面不显示此字段。
数据库引擎	HBase。
进入只读模式阈值	<p>当集群存储达到某一阈值后，如果继续向存储写入数据，存储会写满，可能会导致性能以及其他功能问题。因此需要给集群设置阈值，当达到设置的阈值后，集群进入只读模式，并发出告警提示用户及时处理集群状态。阈值默认值是85%，取值范围70到90之间。</p> <p>说明</p> <p>Core节点存储规格为“普通IO”时，不支持只读模式。</p>

参数	参数说明
是否为只读模式	“是”代表阈值达到设置的值；“否”代表阈值未达到设置的值。

表 2-11 网络配置

参数	参数说明
区域	集群的工作区域，在创建集群时设置。
可用区	显示创建集群时所选择的可用区信息。
虚拟私有云	VPC信息，创建集群时所选。 VPC即虚拟私有云，是通过逻辑方式进行网络隔离，提供安全、隔离的网络环境。
子网	子网信息，创建集群时所选。 通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全。
安全组	显示创建集群时所选择的安全组信息。

表 2-12 Master 节点配置

参数	参数说明
计算规格	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明</p> <ul style="list-style-type: none"> 支持的计算规格： <ul style="list-style-type: none"> 4U16G。 8U16G。 8U32G。 16U32G。 16U64G。 32U64G。
存储规格	默认超高IO。
节点数量	默认2个master节点。

表 2-13 Core 节点配置

参数	参数说明
计算节点	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明</p> <ul style="list-style-type: none"> 支持的计算规格： <ul style="list-style-type: none"> 4U16G 8U16G 8U32G 16U32G 16U64G 32U64G
存储规格	<p>选择HBase计算节点的磁盘规格、容量。</p> <p>说明</p> <ul style="list-style-type: none"> 支持的存储规格： <ul style="list-style-type: none"> 普通IO。 高IO。 超高IO。 容量范围为400~30000GB/节点。
节点数量	选择集群中的节点个数，节点范围2~20。

2.6.3 重启 HBase 集群节点

当CloudTable集群节点状态异常时，可通过执行重启节点操作尝试将该节点状态恢复正常。

注意事项

- 重启节点过程中，节点不可用。
- 重启后节点请在业务低谷期进行重启，避免对高峰期业务造成影响。
- 节点重启时，磁盘扩容、节点扩容、规格变更按钮置灰，不可操作。
- 节点重启是进程重启，不是整个节点重启。

节点重启操作步骤

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 单击集群管理，进入集群列表。
- 步骤4** 选择需要操作的集群，单击集群名称，进入集群详情页。
- 步骤5** 在集群详情页单击“操作 > 重启”，弹出节点重启窗口。

步骤6 输入“RESTART”或者单击“一键输入”，输入后单击“确定”，节点重启。

节点重启后，“重启”按钮置灰。

步骤7 查看节点重启结果，如果重启成功，则“重启”按钮高亮显示。如果节点重启失败，则任务状态显示“节点重启失败”，“重启”按钮高亮显示，节点可继续重启。

---结束

节点重启状态

表 2-14 重启状态

状态	状态描述
重启	重启按钮高亮显示，一种是节点未重启，一种是节点已经重启。
重启中	集群节点正在重启，且在“集群信息 > 任务状态”显示“节点重启中”。
重启失败	集群节点重启失败，且在“集群信息 > 任务状态”显示“节点重启失败”，重启失败的节点可继续重启。

2.6.4 重启 HBase 集群

当集群处于非均衡或不能正常工作时，可能需要通过重启集群进行恢复。当您修改完配置，例如修改集群安全设置、参数修改相关配置，未立即重启集群的情况下，您可以通过手动重启集群使配置生效。

📖 说明

- 如果集群欠费，可能会导致该功能被限制不可用，请及时充值确保集群可正常使用。
- 如果集群状态亚健康，也会限制该功能不可用，请及时联系技术工程师处理使集群可正常使用。

对系统的影响

- 重启期间集群将无法提供服务。因此，在重启前，请确定集群中没有正在运行的任务，并且所有数据都已经保存。
- 如果集群正在处理业务数据，如导入数据、查询数据，一旦重启集群，有可能导致文件损坏或重启失败。因此，建议停止所有集群任务后，再重启集群。
- 如果重启失败，将有可能导致集群不可用，建议联系技术支持人员进行处理或稍后重试。

操作步骤

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 在左侧导航树，单击集群管理。

- 步骤4** 在集群管理列表中的右上角，输入要搜索的集群名称，单击 。
- 步骤5** 选中需要重启的集群名，在对应的“操作”列中，单击“更多 > 重启”。
- 步骤6** 在弹出的对话框中，勾选了解重启集群的影响后，单击“确定”，重启集群。

----结束

2.6.5 删除 HBase 集群

如果不再需要集群时，以删除或退订/释放集群。删除或退订/释放CloudTable集群将会清空所有与该集群相关的资源及数据，且无法恢复，请谨慎操作。

- 按需集群支持直接删除，详情请参见[删除HBase按需集群](#)。
- 包年/包月集群不能直接删除，需要退订（未超期集群）/释放（已超期未续费的集群），详情请参见[退订/释放包周期集群](#)。

说明

- 集群处于扩容状态时无法删除，需要等待扩容完成后删除。
- 如果集群欠费，可能会导致该功能被限制不可用，请及时充值确保集群可正常使用。

删除 HBase 按需集群

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 选中需要删除的集群名，在对应的“操作”列中，单击“更多 > 删除”。
- 步骤4** 在弹出的对话框中，输入“DELETE”或者单击“一键输入”，输入后单击“确定”，删除集群。

图 2-19 确认删除窗口



----结束

退订/释放包周期集群

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。

- 步骤3** 选中需要删除的集群名，在对应的“操作”列中，单击“更多 > 退订/释放”。
- 步骤4** 在弹出的对话框中，输入“DELETE”或者单击“一键输入”，输入后单击“确定”，“退订/释放”集群。
- 步骤5** 进入“退订资源”页面，确认待退订集群信息，选择退订原因，并确认退订金额和相关费用。
- 步骤6** 勾选“我已确认本次退订的资源已完成数据备份或不再使用，未放入回收站的资源退订后无法恢复。查看回收站说明”，然后单击“退订”。
- 步骤7** 返回控制台查看集群已经退订/释放。
- 结束

2.7 HBase 集群运维

2.7.1 HBase 集群容量调整

2.7.1.1 HBase 集群节点扩容

为了提升系统的计算能力、存储能力，用户可以通过节点扩容来扩容集群。节点扩容将会产生相应的费用。用户可根据实际需要或业务情况，动态的增加节点的个数，集群将自适应的实现负载均衡，保证业务不中断，平滑扩容。

注意事项

- 扩容过程中，不会自动重启集群，保证业务不中断，平滑扩容。
- 如果扩容失败，“任务状态”将显示为“节点扩容失败”，集群会自动回滚到扩容前的状态。用户可以再次重试扩容操作。如有问题请及时联系技术支持人员进行处理。
- 只有Core节点可以节点扩容。

节点扩容操作步骤

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 在左侧导航树，选择集群管理。
- 步骤4** 在集群列表中，找到需要扩容的集群名称，单击操作列的“节点扩容”，进入“扩容”页面。

图 2-20 节点扩容

节点扩容

1、请在业务低峰期执行扩容

集群ID/名称	f56157c7-aa0b-4a90-8db9-d667d2a...
地域/可用区	
计费模式	按需
Master节点规格	
Master存储规格	
Master节点数量	3
Core节点规格	
Core存储规格	
Core节点数量	2
变配节点	<input type="radio"/> Master节点 <input checked="" type="radio"/> Core节点
Core节点扩容至	<input type="button" value="-"/> <input type="text" value="3"/> <input type="button" value="+"/>
整体费用 ?	

说明

节点扩容操作也可以在集群详情页面单击“节点扩容”，跳转节点扩容页面。

步骤5 选择变配节点为Core节点。

步骤6 按“+”增加扩容节点。

步骤7 确定费用以及资源配额后，单击确定。

----结束

2.7.1.2 HBase 集群磁盘扩容

磁盘扩容，即纵向扩容，是一种扩容节点内磁盘容量的操作。一般来说，具体的数据存储的计算节点，当计算节点磁盘容量不足时，业务数据将无法存储，因此需要对节点进行纵向扩容。

使用限制

- 当所有的Core节点磁盘占用90%时，集群异常且读写不可用。
- 当集群状态处于“服务中”，且没有任何任务操作（如：节点扩容、规格变更等）时，可以执行扩容操作。
- 磁盘扩容过程中，集群业务可能存在抖动。
- Core节点存储规格为“普通IO”时，不支持磁盘扩容。

磁盘扩容操作步骤

步骤1 登录CloudTable管理控制台。

步骤2 单击集群管理，显示当前集群列表。

步骤3 选择操作集群，单击“操作 > 更多 > 磁盘扩容”，进入扩容页面。

说明

磁盘操作也可以在集群详情页面单击“磁盘扩容”，跳转磁盘扩容页面。

步骤4 单击“+”增加磁盘容量，确定费用以及资源配额后，单击确定。

----结束

磁盘扩容状态

表 2-15 磁盘扩容任务状态说明

状态	说明
磁盘扩容中（任务状态）	表示集群正在扩容中。
服务中（集群状态）	当集群扩容成功并正常提供服务时，其集群状态为服务中。
磁盘扩容失败（任务状态）	表示扩容失败。

2.7.1.3 HBase 集群规格变更

前提条件

当集群状态处于“服务中”，且没有任何任务操作（如：节点扩容、磁盘扩容等）时，可以执行变更操作。

📖 说明

- 规格变更只支持从小规格变更为大规格，如果要大规格更改为小规格，建议新建小规格集群，采用数据迁移进行切换。
- 规格变更过程中，整个集群不可用，变更结束前不能再次变更。
- 规格变更过程中系统不可用。
- 一次只支持变更一种类型的节点规格，且变更完成后只生效所选类型的节点规格。

规格变更操作步骤

步骤1 登录CloudTable管理控制台。

步骤2 单击集群管理，显示当前集群列表。

步骤3 选择操作集群，单击“操作 > 更多 > 规格变更”。

📖 说明

规格变更操作也可以在集群详情页面单击“规格变更”，跳转规格变更页面。

步骤4 选择变配规格，单击 ▾，选择规格。

步骤5 确定费用以及资源配额后，单击确定。

单击确定后，集群状态变更为“亚健康”，任务状态为“规格变更中”，当集群变更完毕后，集群状态更新为“服务中”，任务状态清空。

----结束

规格变更状态

表 2-16 规格变更任务状态说明

状态	说明
规格变更中	表示集群正在变更中。
服务中	当集群变更成功并正常提供服务时，其集群状态为服务中。
规格变更失败	表示集群变更失败。

2.7.2 修改 HBase 参数配置对集群进行性能调优

操作场景

创建完CloudTable HBase集群后，由于HBase有很多与读写性能相关的配置参数，读写请求负载不同的情况下，用户可以在表格存储服务控制台中多次修改集群HBase配置参数，对集群的性能进行调优。当修改完集群HBase参数后，用户需要重启集群才能生效。

使用限制

- 修改CloudTable HBase集群参数需要重启集群，可能会导致业务的中断。

- CloudTable HBase集群重启过程中，不允许修改集群参数。

前提条件

只有当集群无运行中的任务时，才能修改参数。

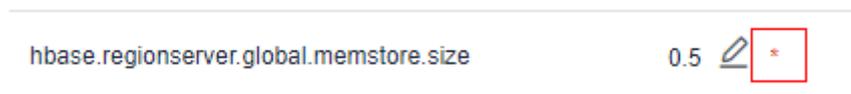
操作步骤

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 在左侧导航树，单击集群管理。
- 步骤4** 选中需要修改参数的集群，单击对应的集群名称，跳转到该集群详情页面。
- 步骤5** 在“参数配置”中单击“参数配置”页签，对HBase参数进行修改。
CloudTable集群当前可以修改的HBase参数请参见[HBase参数说明](#)。

1. 选中需要修改的参数，在“参数值”列单击 。
2. 在修改框中输入新值，单击 ，系统提示“参数修改为xx成功，请保存本次修改值”，并且修改的参数值后显示红色*。

如果单击 ，则取消修改。

图 2-21 修改 HBase 参数



3. 单击参数列表左上方的“保存修改”，弹出“保存修改”对话框。
4. 在“保存修改”对话框中，确认参数修改信息正确后，勾选“是否立即重启”，然后单击“确定”。

图 2-22 保存修改



- 当勾选“是否立即重启”并单击“确定”后，集群将立即重启，重启完成

后，修改后的参数才能生效。单击参数列表右上方的 ，参数列表左上方将显示“新值已应用”。

- 在“保存修改”对话框中可以先不勾选“是否立即重启”，单击“确定”后，参数列表上方将显示“新值未应用”。此时用户需要自行重启集群，才能使修改的参数值生效。重启集群操作请参见[重启HBase集群](#)。
- 如果修改的参数值不正确，可以在“保存修改”对话框中单击“一键还原”，取消修改参数值。

步骤6 参数修改完成后，可以单击“修改历史”页签查看参数修改的历史。

在“修改历史”页签中可以查看如下信息：

- 参数名：被修改过的参数名。
- 旧值：修改前的参数值。
- 新值：修改后的参数值。
- 修改时间：记录用户修改参数值的时间。

----结束

HBase 参数说明

CloudTable集群当前可以修改的参数如[表2-17](#)所示。

说明

hbase.regionserver.global.memstore.size和hfile.block.cache.size两个参数值相加不能超过0.8。

表 2-17 HBase 参数说明

参数名	参数范围	参考缺省值	参数描述
hbase.regionserver.thread.compaction.small	[1,20]	1	HFile compaction线程数，在put高负载情况下可以适当调大。
hbase.regionserver.global.memstore.size	(0,0.8)	0.4	建议设置为“hbase.hregion.memstore.flush.size * 写活跃region数 / RegionServer GC -Xmx”。默认值为“0.4”，表示使用RegionServer GC -Xmx的40%。
hbase.hstore.blockingStoreFiles	[1,2147483647]	60	当列簇的HFile数达到该阈值，阻塞该region的所有操作，直到compaction完成，在put高负载场景下可以适当调大。
hbase.client.scanner.timeout.period	[1,2147483647]	60000	客户端和RegionServer端参数，表示scan租约的时间，建议设置为60000ms的整数倍，在读高负载情况下可以适当调大。单位：毫秒。
hfile.block.cache.size	(0,0.8)	0.2	数据缓存所占的RegionServer GC -Xmx百分比，在读高负载情况下可以适当调大以增大缓存命中率以提高性能。默认值为“0.2”，表示使用RegionServer GC -Xmx的20%。

参数名	参数范围	参考缺省值	参数描述
hbase.regionserver.handler.count	[1,300]	100	RegionServer上的RPC服务器实例数，建议设置为100 ~ 300之间。
hbase.regionserver.metahandler.count	[1,100]	50	RegionServer中处理优先请求的程序实例的数量，建议设置为20 ~ 100之间。
hbase.hstore.flusher.count	[1,10]	2	memstore的flush线程数，在put高负载场景下可以适当调大。
hbase.ipc.server.callqueue.read.ratio	[0,1]	0.5	该配置项用于不同负载模型时，控制读和写操作的RPC队列数量的比例。取值范围为0到1.0，默认值为0.5。 <ul style="list-style-type: none"> 0表示所有RPC队列不区分读和写操作； 小于0.5表示读操作负载少于写操作负载； 0.5表示读操作负载与写操作负载相同； 大于0.5表示读操作负载大于写操作负载； 1.0表示除了一个RPC队列外，其余所有RPC队列都用于读操作。
hbase.regionserver.hotregion.handler.count	[1,65535]	66	在RegionServer上启动的用于热点Region的RPC侦听器实例数。
hbase.ipc.server.hotregion.max.callqueue.length	[1,65535]	330	RegionServer处理热点Region的请求队列长度最大值，当接收到新的请求时，会检查是否会导致队列长度超过此阈值，如果超过阈值，会直接丢弃此请求。
hbase.metric.controller.analysis.period	[1,2147483647]	60	MetricController服务热点分析周期，单位：秒。
hbase.metric.controller.analysis.threads.max	[1,100]	10	热点分析任务线程池最大线程数。
hbase.metric.controller.collect.threads.max	[1,100]	16	热点分析流量收集线程池最大线程数。
hbase.metric.regionserver.hotspot.threshold	[1,2147483647]	20000	单RegionServer热点阈值，单位为每秒请求数。

参数名	参数范围	参考缺省值	参数描述
hbase.metric.region.hotspot.threshold	[1,2147483647]	10000	单Region热点阈值，单位为每秒请求数。
hbase.hotspot.enabled	[true,false]	true	是否开启热点自愈功能，true为开启，false为关闭。开启后如有出现访问热点，会自动处理热点。
hbase.tries.cache.enabled	[true,false]	false	设置为true时，使用LoudsTriesLruBlockCache缓存索引块和数据块。
hbase.write.tries	[true,false]	false	设置为true时，开启succinct tries特性，它将使用一种新的数据结构来提升index block的利用率。
hbase.hfile.hsync	[true,false]	false	设置是否启用Hfile耐久性以将数据持久化到磁盘。若将该参数设置为true，则性能将受到影响，原因是每个Hfile写入时都会被同步到磁盘上。
hbase.wal.hsync	[true,false]	false	设置是否启用WAL文件耐久性以将WAL数据持久化到磁盘。若将该参数设置为true，则性能将受到影响，原因是每个WAL的编辑都会被同步到磁盘上。

2.7.3 使用 CES 监控 HBase 集群

2.7.3.1 HBase 集群监控指标说明

功能说明

监控是保持CloudTable服务可靠性、可用性和性能的重要部分，通过监控，用户可以观察CloudTable服务器的运行状态。

本章节定义了表格存储服务上报云监控服务的监控指标的命名空间、监控指标列表和维度定义。

命名空间

SYS.CloudTable

CloudTable HBase HMaster 实例支持的监控指标

表 2-18 CloudTable HBase HMaster 实例支持的监控指标

指标ID	指标名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)
disk_throughput_write_rate	磁盘读速率	统计每秒从测量对象读出数据量	>= 0	Byte/s	1024(IEC)	CloudTable实例节点	1分钟
disk_throughput_read_rate	磁盘写速率	统计每秒写到测量对象的数据	>= 0	Byte/s	1024(IEC)	CloudTable实例节点	1分钟
cmdForTotalMemory	总内存大小	统计测量对象的总内存大小	> 0	Byte	1024(IEC)	CloudTable实例节点	1分钟
cmdProcessCPU	CPU使用率	统计测量对象的CPU使用率	0~100	%	不涉及	CloudTable实例节点	1分钟
cmdProcessMem	内存使用率	统计测量对象的内存使用率	0~100	%	不涉及	CloudTable实例节点	1分钟
hm_deadregionservernum	故障的Region Server个数	统计测量对象所在集群中存在故障的RegionServer数	0~RegionServer总数	Count	不涉及	CloudTable实例节点	1分钟
hm_regionservernum	正常的Region Server个数	统计测量对象所在集群中运行良好的RegionServer数	0~RegionServer总数	Count	不涉及	CloudTable实例节点	1分钟
hm_ritCount	处在RIT状态的region个数	统计测量对象所在集群中处在RIT(Region In Transaction)状态的region个数	0~region总数	Count	不涉及	CloudTable实例节点	1分钟

指标ID	指标名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期(原始指标)
hm_ritCountOverThreshold	处在RIT状态达到阈值时长的region个数	统计测量对象所在集群中处在RIT(Region In Transaction)状态达到阈值时长的region个数	0~region总数	Count	不涉及	CloudTable实例节点	1分钟
rs_queuecalltime_max	RPC队列最大等待时间	统计测量对象的RPC队列最大等待时间	>= 0	ms	不涉及	CloudTable实例节点	1分钟
rs_queuecalltime_mean	RPC队列平均等待时间	统计测量对象的RPC队列平均等待时间	>= 0	ms	不涉及	CloudTable实例节点	1分钟
nn_percentallused	磁盘空间使用率	集群磁盘空间使用率	0~100	%	不涉及	CloudTable实例节点	1分钟
nn_capacityremaining	剩余的存储空间	集群剩余磁盘空间	取决于集群磁盘容量	GB	不涉及	CloudTable实例节点	1分钟
nn_capacityused	已使用的存储空间	集群已使用磁盘空间	取决于集群磁盘容量	GB	不涉及	CloudTable实例节点	1分钟
cmdForUsedStorageRate	已用存储空间比率	统计测量对象所在集群的已用存储空间大小占总配额的比率	0~100	%	不涉及	CloudTable实例节点	1分钟
network_throughput_inbound_rate	网络吞吐流入速率	每个节点每秒的网络流入数据量。	>= 0	KB/s	不涉及	CloudTable实例节点	1分钟
network_throughput_outgoing_rate	网络吞吐流出速率	每个节点每秒的网络流出数据量。	>= 0	KB/s	不涉及	CloudTable实例节点	1分钟

指标ID	指标名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)
disk_throughput_read_rate	磁盘吞吐读速率	磁盘读取吞吐速率。	>= 0	Byte/s	1024(IEC)	CloudTable实例节点	1分钟
disk_throughput_write_rate	磁盘吞吐写速率	磁盘写入吞吐速率。	>= 0	Byte/s	1024(IEC)	CloudTable实例节点	1分钟

说明

hmaster实例分为hmaster-standby (备) hmaster-active (主) 实例，当hmaster-active (主) 出现故障的时候，hmaster-standby (备) 会升主对外提供服务。

HBase集群磁盘默认10%预留空间，所以集群磁盘告警值不等于磁盘使用率。

CloudTable HBase Regionserver 实例支持的监控指标

CloudTable HBase Regionserver实例支持的监控指标如表2-19所示。

表 2-19 支持的监控指标

指标ID	指标名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)
cmdProcessCPU	CPU使用率	统计测量对象的CPU使用率。	0~100	%	不涉及	CloudTable实例节点	1分钟
cmdForTotalMemory	总内存大小	统计测量对象的总内存大小。	> 0	Byte	1024(IEC)	CloudTable实例节点	1分钟
cmdProcessMem	内存使用率	统计测量对象的内存使用率。	0~100	%	不涉及	CloudTable实例节点	1分钟

指标ID	指标名称	含义	取值范围	单位	进制	测量对象(维度)	监控周期(原始指标)
disk_throughput_write_rate	磁盘写速率	统计每秒写到测量对象的数据量。	>= 0	Byte/s	1024(IEC)	Cloud Table实例节点	1分钟
disk_throughput_read_rate	磁盘读速率	统计每秒从测量对象读出数据量。	>= 0	Byte/s	1024(IEC)	Cloud Table实例节点	1分钟
hm_regionservernum	正常的RegionServer个数	统计测量对象所在集群中运行良好的RegionServer数。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
hm_deadregionservernum	故障的RegionServer个数	统计测量对象所在集群中存在故障的RegionServer数。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
hm_ritCountOverThreshold	处在RIT状态达到阈值时长的region个数	统计测量对象所在集群中处在RIT(Region In Transaction)状态达到阈值时长的region个数。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
hm_ritCount	处在RIT状态的region个数	统计测量对象所在集群中处在RIT(Region In Transaction)状态的region个数。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
rs_requests	单个RegionServer每秒请求次数	统计测量对象的单个RegionServer每秒请求次数。	>= 0	requests/s	不涉及	Cloud Table实例节点	1分钟
rs_regions	单个RegionServer的region个数	统计测量对象的单个RegionServer的region个数。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟

指标ID	指标名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)
rs_writerequestscount	单个 RegionServer 写请求次数	统计测量对象的单个 RegionServer 的写请求次数。	>= 0	Count	不涉及	Cloud Table 实例节点	1分钟
rs_readrequestscount	单个 RegionServer 读请求次数	统计测量对象的单个 RegionServer 的读请求次数。	>= 0	Count	不涉及	Cloud Table 实例节点	1分钟
rs_blockcachehitcachinratio	缓存块命中缓存的比率	统计测量对象的缓存块命中缓存的比率。	0~100	%	不涉及	Cloud Table 实例节点	1分钟
rs_blockCacheCountHitPercent	缓存命中比率	统计测量对象的缓存命中比率。	0~100	%	不涉及	Cloud Table 实例节点	1分钟
rs_getavgtime	Get操作延迟	统计测量对象的 RegionServer 单位时间内 Get 操作的延迟时间均值。	>= 0	ms	不涉及	Cloud Table 实例节点	1分钟
rs_putavgtime	Put操作延迟	统计测量对象的 RegionServer 单位时间内 Put 操作的延迟时间均值。	>= 0	ms	不涉及	Cloud Table 实例节点	1分钟
rs_deleteavgtime	Delete操作延迟	统计测量对象的 RegionServer 单位时间内 Delete 操作的延迟时间均值。	>= 0	ms	不涉及	Cloud Table 实例节点	1分钟
rs_getnumops	Get操作数	统计测量对象的 RegionServer 单位时间内 Get 操作数。	>= 0	Count	不涉及	Cloud Table 实例节点	1分钟

指标ID	指标名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)
rs_putnumops	Put操作数	统计测量对象的RegionServer单位时间内Put操作数。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
rs_deletenumops	Delete操作数	统计测量对象的Regionserver单位时间内Delete操作数。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
rs_queuecalltime_max	RPC队列最大等待时间	统计测量对象的RPC队列最大等待时间。	>= 0	ms	不涉及	Cloud Table实例节点	1分钟
rs_queuecalltime_mean	RPC队列平均等待时间	统计测量对象的, RPC队列平均等待时间。	>= 0	ms	不涉及	Cloud Table实例节点	1分钟
rs_flush_time_mean	Flush操作的平均耗时	统计测量对象的Flush操作的平均耗时。	>= 0	ms	不涉及	Cloud Table实例节点	1分钟
rs_compactionqueue_size	Compaction操作队列大小	统计测量对象的RegionServer中Compaction操作队列大小。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
rs_flushqueue_size	Flush操作队列大小	统计测量对象的RegionServer中Flush操作队列大小。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
rs_compaction_completedcount	Compaction次数	统计测量对象的Compaction次数。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
rs_flush_time_ops_num	Flush次数	统计测量对象的Flush次数。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟

指标ID	指标名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)
rs_blockcacheevictedcount	淘汰的缓存数量	统计测量对象的淘汰的缓存数量。	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
rs_syncTime_max	Hlog Sync最大时长	统计测量对象的Hlog Sync最大时长。	>= 0	ms	不涉及	Cloud Table实例节点	1分钟
rs_syncTime_mean	Hlog Sync平均时长	统计测量对象的Hlog Sync平均时长。	>= 0	ms	不涉及	Cloud Table实例节点	1分钟
dn_byteswritten_speed	每秒写入字节数	节点每秒写入字节数	>=0	Byte	1024(IEC)	Cloud Table实例节点	1分钟
dn_bytesread_speed	每秒读取字节数	节点每秒读取字节数	>=0	Byte	1024(IEC)	Cloud Table实例节点	1分钟
rs_numActiveHandler	RegionServer的活跃handler数	RegionServer的活跃handler数 (处理用户表请求的handler数、处理meta表请求的handler数和处理replication请求的handler请求数的总和)	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
rs_numActiveGeneralHandler	RegionServer处理用户表请求的活跃handler数	RegionServer处理用户表请求的活跃handler数	>= 0	Count	不涉及	Cloud Table实例节点	1分钟
rs_scanTime_p999	P999 Scan操作延迟	RegionServer Scan时延的P999	>= 0	ms	不涉及	Cloud Table实例节点	1分钟

指标ID	指标名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)
rs_syncTime_p999	P999 WAL Sync操作延迟	RegionServer WAL Sync时延的P999	≥ 0	ms	不涉及	Cloud Table实例节点	1分钟
rs_Get_99th_percentile	P99 Get操作延迟	RegionServer Get时延的P99	≥ 0	ms	不涉及	Cloud Table实例节点	1分钟
rs_Put_99th_percentile	P99 Put操作延迟	RegionServer Put时延的P99	≥ 0	ms	不涉及	Cloud Table实例节点	1分钟
rs_Delete_99th_percentile	P99 Delete操作延迟	RegionServer Delete时延的P99	≥ 0	ms	不涉及	Cloud Table实例节点	1分钟
rs_Get_999th_percentile	P999 Get操作延迟	P999 Get操作延迟	≥ 0	ms	不涉及	Cloud Table实例节点	1分钟
rs_Put_999th_percentile	P999 Put操作延迟	RegionServer Put时延的P999	≥ 0	ms	不涉及	Cloud Table实例节点	1分钟
rs_Delete_999th_percentile	P999 Delete操作延迟	RegionServer Delete时延的P999	≥ 0	ms	不涉及	Cloud Table实例节点	1分钟

维度

Key	Value
cluster_id	CloudTable集群ID。
instance_name	CloudTable集群节点名称。

2.7.3.2 设置 HBase 集群告警规则

通过设置CloudTable的告警规则，用户可以自定义监控目标和通知策略，及时了解CloudTable HBase集群运行状态，从而起到预警作用。

设置CloudTable HBase的告警规则包括设置告警规则名称、监控对象、监控指标、告警阈值、监控周期和是否发送通知等参数。本节介绍了设置CloudTable规则的具体方法。

设置 HBase 集群告警规则

步骤1 登录表格存储服务管理控制台。

步骤2 在集群管理列表中的右上角，输入要搜索的集群名称，单击.

步骤3 在集群列表中，找到所需要集群，在对应的“操作”列中，单击“查看监控信息”，系统将跳转到该集群的“云服务监控”页面。

待查看监控信息的集群状态必须为“服务中”。

步骤4 单击“创建告警规则”，进入创建告警规则页面进行添加。

具体参数设置请参考[创建告警规则和通知](#)。

步骤5 配置完成后，单击“立即创建”，完成告警规则的创建。告警规则添加完成后，当监控指标触发设定的阈值时，云监控会在第一时间通过消息通知服务实时告知您资源异常，以免因此造成业务损失。

说明

如果想设置单个指标的告警规则，进入查看监控信息页面：

- 查看监控指标，单击指标图。
- 单击右上角的“+”。
- 进入创建告警规则页面，设置规则。

---结束

2.7.3.3 查看 HBase 集群监控信息

操作场景

云平台提供的云监控服务，可以对HBase集群的运行状态进行日常监控。您可以通过管理控制台，直观地查看集群的各项监控指标。通过查看集群的监控信息可以快速掌握集群的健康状态并及时获取系统的关键信息。

背景信息

- 处于不可用状态的集群节点，无法在云监控中查看其监控指标。当其再次启动或恢复后，即可正常查看。
- 故障1小时的集群节点，云监控服务将默认该节点不存在，并在监控列表中删除，不再对其进行监控，但告警规则需要用户手动清理。

在 CES 界面查看 HBase 集群监控信息

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 在左侧导航树，单击集群管理。

步骤4 在集群列表中找到需要查看的集群，在对应的操作列中，单击查看监控信息，系统会跳到该集群的“云服务监控”页面。

待查看集群，其状态必须为“服务中”。

步骤5 在集群的“云服务监控”页面中，单击集群ID前的按钮  展开计算单元列表，选择对应节点查看监控信息。

- **ID**：所监控的实例ID，即CloudTable集群ID。
- **查看监控指标**：找到所需查看的计算单元，单击“查看监控指标”进入该计算单元的监控指标页面。

步骤6 如果您查看监控页面的指标太多，可以对指标进行设置，设置自己需要查看的指标。

1. 如果指标太多，可以通过设置监控指标页面对指标进行删除。
2. 如果页面显示的指标没有您想看的指标，可以在设置监控指标页面进行添加。
3. 选择您在页面中要展示的指标名称，拖动选中指标可以对指标进行排序。

图 2-23 设置指标图

设置监控指标



----结束

在集群详情页查看 HBase 集群监控信息

集群详情页面主要展示HBase每个节点的CPU使用率、内存使用率、磁盘使用率、磁盘使用量、磁盘总量。

步骤1 登录表格存储服务管理控制台。

步骤2 进入集群管理页面，选择目标HBase集群，单击“集群名称 > 详情”。

步骤3 进入详情页面，详情页面显示监控页面的指标。

图 2-24 HBase 监控页面

Master节点						
计算规格	4 vCPU 16 GB 4*CPU (1个CPU+16GB)					
存储规格	SSD 200GB					
节点数量	3					
节点名称	CPU使用率	内存使用率	磁盘使用率	磁盘使用量(GB)	磁盘总量(GB)	监控
hmaster-1-1	16.38%	22%	0.05%	0.10	200	📊
hmaster-2-1	28.47%	21%	0.05%	0.12	200	📊
hmaster-3-1	18.84%	20%	0.04%	0.08	200	📊
Core节点						
计算规格	4 vCPU 16 GB 4*CPU (1个CPU+16GB)					
存储规格	SSD 400GB					
节点数量	2					
节点名称	CPU使用率	内存使用率	磁盘使用率	磁盘使用量(GB)	磁盘总量(GB)	监控
regionserver-1-1	5.63%	15%	0.00%	0.32	400	📊
regionserver-2-1	7.72%	16%	0.00%	0.32	400	📊

步骤4 如要想要查看此节点所有指标信息，请单击详情页面的监控小图标 ，可以跳转CES云服务监控页面查看HBase集群单节点详细监控指标。

----结束

2.7.4 HBase 集群日志管理

2.7.4.1 对接 LTS 查看 HBase 集群日志

集群日志服务用于采集集群日志并报送云日志服务（Log Tank Service，以下简称 LTS），用户可以在LTS云日志服务查看采集的集群日志或进行日志转储。

📖 说明

当前支持的日志类型：

HBase集群存放日志路径：/var/chroot/cloudtable/hbase/logs/

- hbase-Ruby-master- $\$hostname$.log
- hbase-Ruby-regionserver- $\$hostname$.log
- hbase-audit.log

开启云日志服务对接步骤

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群，进入集群列表。

步骤4 在集群列表中单击集群名称，进入集群详情页面，左侧导航栏单击“日志服务”，进入日志服务页面。

步骤5 单击页面左上角的 ，开启云日志服务对接。

说明

- 如果首次开启，会弹出创建委托窗口，单击“确定”，进行授权委托。
- 若已开启过LTS云日志服务对接，并授权该服务创建委托，再次开启时则无需授权操作。

----结束

查看集群日志

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群，进入集群列表。

步骤4 在集群列表中单击集群名称，进入集群详情页面，左侧导航栏单击“日志服务”，进入日志服务页面。

步骤5 选择查看日志，单击“操作 > 查看日志”。即可跳转到LTS云日志服务查看日志。

----结束

关闭日志

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群，进入集群列表。

步骤4 在集群列表中单击集群名称，进入集群详情页面，左侧导航栏单击“日志服务”，进入日志服务页面。

步骤5 单击“云日志服务对接”开关，弹出“关闭云日志服务”提示框。

步骤6 确认无误后，单击提示框中的“确认”按钮，即可关闭日志采集。

----结束

2.7.4.2 对接 CTS 查看 HBase 集群审计日志

CloudTable使用云审计服务，可以记录与CloudTable服务相关的操作事件，便于日后的查询、审计和回溯。

CloudTable的以下关键操作事件将记录审计日志，详细内容如表2-20所示。

表 2-20 云审计服务支持的 CloudTable HBase 集群操作事件列表

操作名称	事件名称	资源类型
创建集群	createCloudTableClusterV3	cluster
节点扩容	growCloudTableCluster	cluster
重启集群	rebootCloudTableCluster	cluster

操作名称	事件名称	资源类型
设置存储配额	storageClusterAction	cluster
特性开关	modifyClusterFeatures	cluster
参数配置	modifyClusterSetting	cluster
创建数据迁移任务	copierCreateTask	cluster
枚举数据库信息	copierListDatabaseInfo	cluster
枚举集群节点信息	copierListNodeInfo	cluster
删除集群	deleteCloudTableClusterV2	cluster
关闭集群日志	disableLTSAccess	cluster
打开集群日志	enableLTSAccessc	cluster
获取集群信息	getClusterInfo	cluster
获取数据库信息	getDatabases	cluster
获取角色信息	getRoles	cluster
获取表信息	getTables	cluster
磁盘扩容	growCloudTableDisk	cluster
规格扩容	growCloudTableFlavor	cluster
HBase修改配置参数	modifyClusterSetting	cluster
重启节点	restartInstance	cluster
重启	REBOOTING	cluster
扩容	GROWING	cluster
删除	DELETING	cluster
开启或关闭HBase thrift server	switchThriftServer	cluster
包周期集群扩容、规格变更	changeCloudTableCluster	cluster
开启冷热分离	switchHotColdFeature	cluster
查杀sql	killQueryBySqlId	cluster

开启云审计服务

使用云审计服务前需要开启云审计服务，开启云审计服务后系统会自动创建一个追踪器，系统记录的所有操作将关联在该追踪器中。目前，一个云账户系统仅支持创建一个追踪器。

步骤1 在CloudTable管理控制台，选择“服务列表 > 管理与部署 > 云审计服务”，进入云审计服务信息页面。

步骤2 开通云审计服务。

如果您是首次使用云审计服务，在追踪器列表中还没有已创建的追踪器，则请参考《云审计服务快速入门》中的“开通云审计服务”开启云审计服务。

如果您已开通过云审计服务了，开通时系统已为您自动创建了一个管理事件追踪器，管理事件追踪器只能有一个且不可删除。您也可以自行创建数据事件追踪器，详细内容请参见《云审计服务用户指南》中的[创建追踪器](#)。

----结束

关闭审计日志

如果用户想关闭审计日志，需要在云审计服务中停用追踪器。

步骤1 在CloudTable管理控制台，选择“服务列表 > 管理与监管 > 云审计服务”，进入云审计服务信息页面。

步骤2 通过停用追踪器，关闭审计日志。如需重新开启审计日志，只要启用追踪器即可。

有关停用/启用追踪器的更多信息，请参考《云审计服务用户指南》中的[启用/停用追踪器](#)。

----结束

查看 CloudTable 的云审计日志

步骤1 登录管理控制台。

步骤2 单击页面上方的“服务列表”，选择“管理与监管 > 云审计服务”，进入云审计服务信息页面。

步骤3 单击左侧导航树的“事件列表”，进入事件列表信息页面。

步骤4 单击事件列表右上方的“筛选”，设置对应的操作事件条件。

当前事件列表支持四个维度的组合查询，详细信息如下：

- “事件来源”、“资源类型”和“筛选类型”。
 - “事件来源”：选择“CloudTable”。
 - “资源类型”：选择“所有资源类型”，或者指定具体的资源类型。
 - “筛选类型”：选择“所有筛选类型”，或者选择以下任一选项。
 - “按事件名称”：选择该选项时，还需选择某个具体的事件名称。
 - “按资源ID”：选择该选项时，还需选择或者手动输入某个具体的资源ID。
 - “按资源名称”：选择该选项时，还需选择或手动输入某个具体的资源名称。
- “操作用户”：在下拉框中选择某一具体的操作用户，此操作用户指用户级别，而非租户级别。
- “事件级别”：可选项为“所有事件级别”、“normal”、“warning”、“incident”，只可选择其中一项。

- “起始时间”、“结束时间”：可通过选择时间段查询操作事件。

步骤5 单击“查询”，查看对应的操作事件。

步骤6 在需要查看的事件左侧，单击  展开该记录的详细信息。

图 2-25 事件



步骤7 在需要查看的事件右侧，单击“查看事件”，弹出一个窗口，显示了该操作事件结构的详细信息。

图 2-26 查看事件

查看事件

```
{
  "service_type": "CloudTable",
  "user": {
    "name": " ",
    "id": "2f210e62179a425c96e5b57f25d1d492",
    "domain": {
      "name": " ",
      "id": "760dfba8429649e7a477445e1033a839"
    }
  },
  "time": "2018/05/29 14:50:16 GMT+08:00",
  "code": 200,
  "resource_type": "cluster",
  "resource_name": "cloudtable ",
  "resource_id": "689ae304-086b-4fa9-984e-16c58e3d75bf",
  "source_ip": " ",
  "trace_name": "rebootCloudTableCluster",
  "trace_type": "ConsoleAction",
  "api_version": "v1.0",
  "record_time": "2018/05/29 14:50:16 GMT+08:00",
  "trace_id": "8b253930-630c-11e8-82ae-286ed488cbe3",
  "trace_status": "normal"
}
```

关于云审计服务事件结构的关键字段详解，请参见《云审计服务用户指南》的[事件结构](#)章节。

----结束

3 使用 Doris

3.1 Doris 数据模型概述

在Doris中，数据以表的形式进行逻辑上的描述。表是具有相同模式的同质数据的集合。一张表包括行（Row）和列（Column）。Row即用户的一行数据。Column用于描述一行数据中不同的字段，可以根据实际情况采用不同的数据类型（如整型、字符串、布尔型等）。

从OLAP场景看，Column可以分为两大类：Key和Value。Key表示维度列，Value表示指标列。

Doris的数据模型主要分为3类：

- **Aggregate模型。**
- **Unique模型。**
- **Duplicate。**

Aggregate 模型

以实际的例子来说明什么是聚合模型，以及如何正确的使用聚合模型。

- 示例1：导入数据聚合
假设业务有以下模式：

表 3-1 参数说明

ColumnName	Type	AggregationType	Comment
user_id	LARGEINT	-	用户ID
date	DATE	-	数据导入日期
city	VARCHAR(20)	-	用户所在城市
age	SMALLINT	-	用户年龄
sex	TINYINT	-	用户性别

ColumnName	Type	AggregationType	Comment
last_visit_date	DATETIME	REPLACE	用户最后一次访问时间
cost	BIGINT	SUM	用户总消费
max_dwell_time	INT	MAX	用户最大停留时间
min_dwell_time	INT	MIN	用户最小停留时间

转换成建表语句，如下所示。

```
CREATE TABLE IF NOT EXISTS demo.example_tbl
(
  `user_id` LARGEINT NOT NULL COMMENT "用户id",
  `date` DATE NOT NULL COMMENT "数据灌入日期时间",
  `city` VARCHAR(20) COMMENT "用户所在城市",
  `age` SMALLINT COMMENT "用户年龄",
  `sex` TINYINT COMMENT "用户性别",
  `last_visit_date` DATETIME REPLACE DEFAULT "1970-01-01 00:00:00" COMMENT "用户最后一次访问时间",
  `cost` BIGINT SUM DEFAULT "0" COMMENT "用户总消费",
  `max_dwell_time` INT MAX DEFAULT "0" COMMENT "用户最大停留时间",
  `min_dwell_time` INT MIN DEFAULT "99999" COMMENT "用户最小停留时间"
)
AGGREGATE KEY(`user_id`, `date`, `city`, `age`, `sex`)
DISTRIBUTED BY HASH(`user_id`) BUCKETS 1
PROPERTIES (
  "replication_allocation" = "tag.location.default: 3"
);
```

可以看到，这是一个典型的用户信息和访问行为的事实表。在一般星型模型中，用户信息和访问行为一般分别存放在维度表和事实表中。这里我们为了更加方便的解释Doris的数据模型，将两部分信息统一存放在一张表中。

表中的列按照是否设置了AggregationType，分为Key（维度列）和Value（指标列）。没有设置AggregationType的，如user_id、date、age、sex称为Key，而设置了AggregationType的称为Value。

当导入数据时，对于Key列相同的行会聚合成一行，而Value列会按照设置的AggregationType进行聚合。AggregationType目前有以下四种聚合方式：

- SUM：求和，多行的Value进行累加。
- REPLACE：替代，下一批数据中的Value会替换之前导入过的行中的Value。
- MAX：保留最大值。
- MIN：保留最小值。

表 3-2 原始数据

user_id	date	city	age	sex	last_visit_date	cost	max_dwell_time	min_dwell_time
10000	2017-10-01	A	20	0	2017-10-01 06:00:00	20	10	10
10000	2017-10-01	A	20	0	2017-10-01 07:00:00	15	2	2
10001	2017-10-01	A	30	1	2017-10-01 17:05:45	2	22	22
10002	2017-10-02	B	20	1	2017-10-02 12:59:12	200	5	5
10003	2017-10-02	C	32	0	2017-10-02 11:20:00	30	11	11
10004	2017-10-01	D	35	0	2017-10-01 10:00:15	100	3	3
10004	2017-10-03	D	35	0	2017-10-03 10:20:22	11	6	6

我们假设这是一张记录用户访问某商品页面行为的表。我们以第一行数据为例，解释如下：

表 3-3 参数说明

数据	说明
10000	用户id，每个用户唯一识别id
2017-10-01	数据入库时间，精确到日期
A	用户所在城市
20	用户年龄

数据	说明
0	性别男（1 代表女性）
2017-10-01 06:00:00	用户本次访问该页面的时间，精确到秒
20	用户本次访问产生的消费
10	用户本次访问，驻留该页面的时间
10	用户本次访问，驻留该页面的时间（冗余）

那么当这批数据正确导入到Doris中后，Doris中最终存储如下：

表 3-4 插入数据

user_id	date	city	age	sex	last_visit_date	cost	max_dwelling_time	min_dwelling_time
10000	2017-10-01	A	20	0	2017-10-01 07:00:00	35	10	2
10001	2017-10-01	A	30	1	2017-10-01 17:05:45	2	22	22
10002	2017-10-02	B	20	1	2017-10-02 12:59:12	200	5	5
10003	2017-10-02	C	32	0	2017-10-02 11:20:00	30	11	11
10004	2017-10-01	D	35	0	2017-10-01 10:00:15	100	3	3
10004	2017-10-03	D	35	0	2017-10-03 10:20:22	11	6	6

可以看到，用户10000只剩下了一行聚合后的数据。而其余用户的数据和原始数据保持一致。这里先解释下用户10000 聚合后的数据：

前5列没有变化，从第6列last_visit_date开始：

- 2017-10-01 07:00:00：因为last_visit_date列的聚合方式为REPLACE，所以2017-10-01 07:00:00替换了2017-10-01 06:00:00保存了下来。

📖 说明

在同一个导入批次中的数据，对于REPLACE这种聚合方式，替换顺序不做保证。如在这个例子中，最终保存下来的，也有可能是2017-10-01 06:00:00。而对于不同导入批次中的数据，可以保证，后一批次的数据会替换前一批次。

- 35：因为cost列的聚合类型为SUM，所以由20+15累加获得35。
- 10：因为max_dwell_time列的聚合类型为MAX，所以10和2取最大值，获得10。
- 2：因为min_dwell_time列的聚合类型为MIN，所以10和2取最小值，获得2。

经过聚合，Doris中最终只会存储聚合后的数据。换句话说，即明细数据会丢失，用户不能够再查询到聚合前的明细数据了。

- 示例2：保留，明细数据。

接示例1，将表结构修改如下：

表 3-5 参数说明

ColumnName	Type	AggregationType	Comment
user_id	LARGEINT	-	用户ID
date	DATE	-	数据导入日期
timestamp	DATETIME	-	数据导入时间，精确到秒
city	VARCHAR(20)	-	用户所在城市
age	SMALLINT	-	用户年龄
sex	TINYINT	-	用户性别
last_visit_date	DATETIME	REPLACE	用户最后一次访问时间
cost	BIGINT	SUM	用户总消费
max_dwell_time	INT	MAX	用户最大停留时间
min_dwell_time	INT	MIN	用户最小停留时间

即增加了一列timestamp，记录精确到秒的数据导入时间。

同时，将AGGREGATE KEY设置为AGGREGATE KEY(user_id, date, timestamp, city, age, sex)。

导入数据如下:

表 3-6 原始数据

user_id	date	time stamp	city	age	sex	last_visit_date	cost	max_dwelling_time	min_dwelling_time
10000	2017-10-01	2017-10-01 08:00:05	A	20	0	2017-10-01 06:00:00	20	10	10
10000	2017-10-01	2017-10-01 09:00:05	A	20	0	2017-10-01 07:00:00	15	2	2
10001	2017-10-01	2017-10-01 18:12:10	A	30	1	2017-10-01 17:05:45	2	22	22
10002	2017-10-02	2017-10-02 13:10:00	B	20	1	2017-10-02 12:59:12	200	5	5
10003	2017-10-02	2017-10-02 13:15:00	C	32	0	2017-10-02 11:20:00	30	11	11
10004	2017-10-01	2017-10-01 12:12:48	D	35	0	2017-10-01 10:00:15	100	3	3
10004	2017-10-03	2017-10-03 12:38:20	D	35	0	2017-10-03 10:20:22	11	6	6

那么当这批数据正确导入到Doris中后，Doris中最终存储如下：

表 3-7 数据结果

user_id	date	timestamp	city	age	sex	last_visit_date	cost	max_dwelling_time	min_dwelling_time
10000	2017-10-01	2017-10-01 08:00:05	A	20	0	2017-10-01 06:00:00	20	10	10
10000	2017-10-01	2017-10-01 09:00:05	A	20	0	2017-10-01 07:00:00	15	2	2
10001	2017-10-01	2017-10-01 18:12:10	A	30	1	2017-10-01 17:05:45	2	22	22
10002	2017-10-02	2017-10-02 13:10:00	B	20	1	2017-10-02 12:59:12	200	5	5
10003	2017-10-02	2017-10-02 13:15:00	C	32	0	2017-10-02 11:20:00	30	11	11
10004	2017-10-01	2017-10-01 12:12:48	D	35	0	2017-10-01 10:00:15	100	3	3
10004	2017-10-03	2017-10-03 12:38:20	D	35	0	2017-10-03 10:20:22	11	6	6

- 示例3: 导入数据与已有数据聚合。
接示例1中的参数列表，插入以下表中数据。

表 3-8 原始数据

user_id	date	city	age	sex	last_visit_date	cost	max_dwell_time	min_dwell_time
10000	2017-10-01	A	20	0	2017-10-01 07:00:00	35	10	2
10001	2017-10-01	A	30	1	2017-10-01 17:05:45	2	22	22
10002	2017-10-02	B	20	1	2017-10-02 12:59:12	200	5	5
10003	2017-10-02	C	32	0	2017-10-02 11:20:00	30	11	11
10004	2017-10-01	D	35	0	2017-10-01 10:00:15	100	3	3
10004	2017-10-03	D	35	0	2017-10-03 10:20:22	11	6	6

再导入一批新的数据：

表 3-9 新数据

user_id	date	city	age	sex	last_visit_date	cost	max_dwell_time	min_dwell_time
10004	2017-10-03	D	35	0	2017-10-03 11:22:00	44	19	19
10005	2017-10-03	E	29	1	2017-10-03 18:11:02	3	1	1

那么当这批数据正确导入到Doris中后，Doris中最终存储如下：

表 3-10

user_id	date	city	age	sex	last_visit_date	cost	max_dwell_time	min_dwell_time
10000	2017-10-01	A	20	0	2017-10-01 07:00:00	35	10	2
10001	2017-10-01	A	30	1	2017-10-01 17:05:45	2	22	22
10002	2017-10-02	B	20	1	2017-10-02 12:59:12	200	5	5
10003	2017-10-02	C	32	0	2017-10-02 11:20:00	30	11	11
10004	2017-10-01	D	35	0	2017-10-01 10:00:15	100	3	3
10004	2017-10-03	D	35	0	2017-10-03 11:22:00	55	19	6
10005	2017-10-03	E	29	1	2017-10-03 18:11:02	3	1	1

可以看到，用户10004的已有数据和新导入的数据发生了聚合。同时新增了10005用户的数据。

数据的聚合，在Doris中有如下三个阶段发生：

- 每一批次数据导入的ETL阶段。该阶段会在每一批次导入的数据内部进行聚合。
- 底层BE进行数据Compaction的阶段。该阶段，BE会对已导入的不同批次的数据进行进一步的聚合。
- 数据查询阶段。在数据查询时，对于查询涉及到的数据，会进行对应的聚合。

数据在不同时间，可能聚合的程度不一致。例如一批数据刚导入时，可能还未与之前已存在的数据进行聚合。但是对于用户而言，用户只能查询到聚合后的数据。即不同的聚合程度对于用户查询而言是透明的。用户需始终认为数据以最终的完成的聚合程度存在，而不应假设某些聚合还未发生。

Unique 模型

在某些多维分析场景下，用户更关注的是如何保证Key的唯一性，即如何获得Primary Key唯一性约束。因此，我们引入了Unique的数据模型。该模型本质上是聚合模型的一个特例，也是一种简化的表结构表示方式。举例说明：

说明

Unique模型表，不推荐开启merge-on-write属性，默认使用merge-on-read。

表 3-11 参数说明

ColumnName	Type	IsKey	Comment
user_id	BIGINT	Yes	用户ID
username	VARCHAR(50)	Yes	用户昵称
city	VARCHAR(20)	No	用户所在城市
age	SMALLINT	No	用户年龄
sex	TINYINT	No	用户性别
phone	LARGEINT	No	用户电话
address	VARCHAR(500)	No	用户住址
register_time	DATETIME	No	用户注册时间

这是一个典型的用户基础信息表。这类数据没有聚合需求，只需保证主键唯一性。（这里的主键为user_id+username）。那么我们的建表语句如下：

```
CREATE TABLE IF NOT EXISTS example_db.expamle_tbl
(
  `user_id` LARGEINT NOT NULL COMMENT "用户id",
  `username` VARCHAR(50) NOT NULL COMMENT "用户昵称",
  `city` VARCHAR(20) COMMENT "用户所在城市",
  `age` SMALLINT COMMENT "用户年龄",
  `sex` TINYINT COMMENT "用户性别",
  `phone` LARGEINT COMMENT "用户电话",
  `address` VARCHAR(500) COMMENT "用户地址",
  `register_time` DATETIME COMMENT "用户注册时间"
)
UNIQUE KEY(`user_id`, `username`)
DISTRIBUTED BY HASH(`user_id`) BUCKETS 1
PROPERTIES (
  "replication_allocation" = "tag.location.default: 3"
);
```

而这个表结构，完全同等于以下使用聚合模型描述的表结构：

表 3-12 参数说明

ColumnName	Type	AggregationType	Comment
user_id	BIGINT	-	用户ID
username	VARCHAR(50)	-	用户昵称
city	VARCHAR(20)	REPLACE	用户所在城市
age	SMALLINT	REPLACE	用户年龄
sex	TINYINT	REPLACE	用户性别
phone	LARGEINT	REPLACE	用户电话
address	VARCHAR(500)	REPLACE	用户住址
register_time	DATETIME	REPLACE	用户注册时间

建表语句。

```
CREATE TABLE IF NOT EXISTS example_db.exapmle_tbl
(
  `user_id` LARGEINT NOT NULL COMMENT "用户id",
  `username` VARCHAR(50) NOT NULL COMMENT "用户昵称",
  `city` VARCHAR(20) REPLACE COMMENT "用户所在城市",
  `age` SMALLINT REPLACE COMMENT "用户年龄",
  `sex` TINYINT REPLACE COMMENT "用户性别",
  `phone` LARGEINT REPLACE COMMENT "用户电话",
  `address` VARCHAR(500) REPLACE COMMENT "用户地址",
  `register_time` DATETIME REPLACE COMMENT "用户注册时间"
)
AGGREGATE KEY(`user_id`, `username`)
DISTRIBUTED BY HASH(`user_id`) BUCKETS 1
PROPERTIES (
  "replication_allocation" = "tag.location.default: 3"
);
```

即Unique模型完全可以用聚合模型中的REPLACE方式替代。其内部的实现方式和数据存储方式也完全一样。

Duplicate 模型

在某些多维分析场景下，数据既没有主键，也没有聚合需求。因此，我们引入Duplicate数据模型来满足这类需求。

表 3-13 数据

ColumnName	Type	SortKey	Comment
timestamp	DATETIME	Yes	日志时间
type	INT	Yes	日志类型
error_code	INT	Yes	错误码
error_msg	VARCHAR(1024)	No	错误详细信息

ColumnName	Type	SortKey	Comment
op_id	BIGINT	No	负责人ID
op_time	DATETIME	No	处理时间

建表语句。

```
CREATE TABLE IF NOT EXISTS example_db.exampml_tbl
(
  `timestamp` DATETIME NOT NULL COMMENT "日志时间",
  `type` INT NOT NULL COMMENT "日志类型",
  `error_code` INT COMMENT "错误码",
  `error_msg` VARCHAR(1024) COMMENT "错误详细信息",
  `op_id` BIGINT COMMENT "负责人id",
  `op_time` DATETIME COMMENT "处理时间"
)
DUPLICATE KEY(`timestamp`, `type`)
DISTRIBUTED BY HASH(`type`) BUCKETS 1
PROPERTIES (
  "replication_allocation" = "tag.location.default: 3"
);
```

这种数据模型区别于Aggregate和Unique模型。数据完全按照导入文件中的数据进行存储，不会有任何聚合。即使两行数据完全相同，也都会保留。而在建表语句中指定的DUPLICATE KEY，只是用来指明底层数据按照哪些列进行排序。更贴切的名称应该为“Sorted Column”，这里取名“DUPLICATE KEY”只是用以明确表示所用的数据模型。在DUPLICATE KEY的选择上，我们建议适当地选择前2-4列就可以。

这种数据模型适用于既没有聚合需求，又没有主键唯一性约束的原始数据的存储。更多使用场景，可参见聚合模型的局限性。

聚合模型的局限

- Aggregate模型&Unique模型。

这里我们针对Aggregate模型（包括Unique模型），来介绍下聚合模型的局限性。

在聚合模型中，模型对外展现的，是最终聚合后的数据。也就是说，对于任何还未聚合的数据（例如说两个不同导入批次的数据），必须通过某种方式保证对外展示的一致性。

假设表结构如下：

表 3-14 数据

ColumnName	Type	AggregationType	Comment
user_id	LARGEINT	-	用户ID
date	DATE	-	数据导入日期
cost	BIGINT	SUM	用户总消费

假设存储引擎中有如下两个已经导入完成的批次的数据。

batch1

表 3-15 数据

user_id	date	cost
10001	2017-11-20	50
10002	2017-11-21	39

batch2

表 3-16 数据

user_id	date	cost
10001	2017-11-20	1
10001	2017-11-21	5
10003	2017-11-22	22

可以看到，用户10001分属在两个导入批次中的数据还没有聚合。但是为了保证用户只能查询到如下最终聚合后的数据。

表 3-17 数据

user_id	date	cost
10001	2017-11-20	51
10001	2017-11-21	5
10002	2017-11-21	39
10003	2017-11-22	22

建数据表。

```
CREATE TABLE IF NOT EXISTS example_db.exampale_tb2
(
  `user_id` LARGEINT NOT NULL COMMENT "用户id",
  `date` DATE NOT NULL COMMENT "数据导入日期时间",
  `cost` BIGINT SUM DEFAULT "0" COMMENT "用户总消费"
)
AGGREGATE KEY(`user_id`, `date`)
DISTRIBUTED BY HASH(`user_id`) BUCKETS 1
PROPERTIES (
  "replication_allocation" = "tag.location.default: 3"
);
```

插入表15、表16的数据。

```
INSERT INTO example_db.expamle_tb2 (user_id,date,cost) VALUES('10001','2017-11-20','50'),
('10002','2017-11-21','39'),('10001','2017-11-20','1'),('10001','2017-11-21','5'),
('10003','2017-11-22','22');
```

我们在查询引擎中加入了聚合算子，来保证数据对外的一致性。

另外，在聚合列（Value）上，执行与聚合类型不一致的聚合类查询时，要注意语义。例如我们在如上示例中执行如下查询：

```
mysql> SELECT MIN(cost) FROM example_db.expamle_tb2;
+-----+
| min(`cost`) |
+-----+
|          5 |
+-----+
1 row in set (0.02 sec)
```

得到的结果是5，而不是1。

同时，这种一致性保证，在某些查询中，会极大降低查询效率。

我们以最基本的count(*)查询为例：

```
SELECT COUNT(*) FROM table;
```

在其他数据库中，这类查询都会很快返回结果。因为在实现上，我们可以通过如导入时对行进行计数，保存count的统计信息，或者在查询时仅扫描某一系列数据，获得count值的方式，只需很小的开销，即可获得查询结果。但是在Doris的聚合模型中，这种查询的开销非常大。以刚才的数据为例。

batch1

表 3-18 数据

user_id	date	cost
10001	2017-11-20	50
10002	2017-11-21	39

batch2

表 3-19 数据

user_id	date	cost
10001	2017-11-20	1
10001	2017-11-21	5
10003	2017-11-22	22

最终聚合结果如下表所示。

表 3-20 数据

user_id	date	cost
10001	2017-11-20	51

user_id	date	cost
10001	2017-11-21	5
10002	2017-11-21	39
10003	2017-11-22	22

查询结果。

```
mysql> SELECT COUNT(date) FROM example_db.expamle_tb2;
+-----+
| count(`date`) |
+-----+
|          4 |
+-----+
1 row in set (0.01 sec)
```

所以，select count(*) from table;的正确结果应该为4。但如果我们只扫描user_id这一列，如果加上查询时聚合，最终得到的结果是3（10001, 10002, 10003）。而如果不加查询时聚合，则得到的结果是5（两批次一共5行数据）。可见这两个结果都是不对的。

为了得到正确的结果，我们必须同时读取user_id和date这两列的数据，再加上查询时聚合，才能返回4这个正确的结果。也就是说，在count(*)查询中，Doris必须扫描所有的AGGREGATE KEY列（这里就是user_id和date），并且聚合后，才能得到语义正确的结果。当聚合列非常多时，count(*)查询需要扫描大量的数据。

因此，当业务上有频繁(count(*)查询时，我们建议用户通过增加一个值恒为1的，聚合类型为SUM的列来模拟count(*)。如刚才的例子中的表结构，我们修改如下：

表 3-21 参数说明

ColumnName	Type	AggregateType	Comment
user_id	BIGINT	-	用户ID
date	DATE	-	数据导入日期
cost	BIGINT	SUM	用户总消费
count	BIGINT	SUM	用于计算count

增加一个count列，并且导入数据中，该列值恒为1。则select count(*) from table;的结果等价于select sum(count) from table;。而后者的查询效率将远高于前者。不过这种方式也有使用限制，就是用户需要自行保证，不会重复导入AGGREGATE KEY列都相同的行。否则，select sum(count) from table;只能表述原始导入的行数，而不是select count(*) from table;的语义，前者值会错误的增大。

另一种方式，就是将如上的count列的聚合类型改为REPLACE，且依然值恒为1。那么select sum(count) from table;和select count(*) from table;的结果将是一致的。并且这种方式，没有导入重复行的限制。

- Duplicate
Duplicate模型没有聚合模型的这个局限性。因为该模型不涉及聚合语意，在做count(*)查询时，任意选择一列查询，即可得到语意正确的结果。

数据模型选择

Doris数据模型上目前分为三类：AGGREGATE KEY，UNIQUE KEY，DUPLICATE KEY。三种模型中数据都是按KEY进行排序。

- Aggregate模型。
Aggregate模型可以通过预聚合，极大地降低聚合查询时所需扫描的数据量和查询的计算量，非常适合有固定模式的报表类查询场景。但是该模型对count(*)查询很不友好。同时因为固定了Value列上的聚合方式，在进行其他类型的聚合查询时，需要考虑语意正确性。

Aggregate Key相同时，新旧记录进行聚合，目前支持的聚合函数有SUM，MIN，MAX，REPLACE。

```
CREATE TABLE site_visit
(
  siteid INT,
  city SMALLINT,
  username VARCHAR(32),
  pv BIGINT SUM DEFAULT '0'
)
AGGREGATE KEY(siteid, city, username)
DISTRIBUTED BY HASH(siteid) BUCKETS 10;
```

- Unique模型。
Unique模型针对需要唯一主键约束的场景，Unique key相同时，新记录覆盖旧记录，可以保证主键唯一性约束。适用于有更新需求的分析业务。目前Unique key实现上和Aggregate key的REPLACE聚合方法一样，二者本质上相同。但是无法利用ROLLUP等预聚合带来的查询优势（因为本质是REPLACE，没有SUM这种聚合方式）。

```
CREATE TABLE sales_order
(
  orderid BIGINT,
  status TINYINT,
  username VARCHAR(32),
  amount BIGINT DEFAULT '0'
)
UNIQUE KEY(orderid)
DISTRIBUTED BY HASH(orderid) BUCKETS 10;
```

- Duplicate模型。
Duplicate模型相同的行不会合并，适合任意维度的Ad-hoc查询。虽然无法利用预聚合的特性，但是不受聚合模型的约束，可以发挥列存模型的优势（列裁剪、向量执行等）。

```
CREATE TABLE session_data
(
  visitorid SMALLINT,
  sessionid BIGINT,
  visitime DATETIME,
  city CHAR(20),
  province CHAR(20),
  ip varchar(32),
  browser CHAR(20),
  url VARCHAR(1024)
)
DUPLICATE KEY(visitorid, sessionid)
DISTRIBUTED BY HASH(sessionid, visitorid) BUCKETS 10;
```

3.2 Doris 使用流程

CloudTable集群模式提供了基于Doris的分布式、可伸缩、全托管的实时数据仓库，它提供了亚秒级返回海量数据查询结果的能力，可以支持高并发的点查询场景，也能支持高吞吐的复杂分析场景。

Doris集群管理下，用户申请专属集群，按需使用，专属集群是完全物理隔离且不受其他业务影响，方便用户管理。

Doris集群创建成功后，用户可通过使用客户端访问集群：详情请参见使用Doris章节，介绍如何[使用Doris集群](#)。

图 3-1 Doris 使用流程

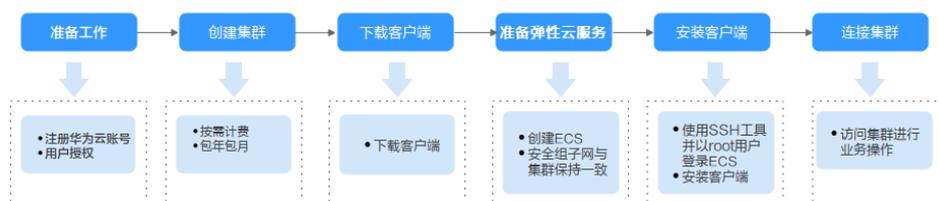


表 3-22 Doris 使用流程

流程	子任务	说明	详细指导
准备工作	创建用户并授权使用 CloudTable	<ul style="list-style-type: none"> 使用CloudTable Doris之前，您需要注册华为云账号并进行实名认证，给新账号授权。 将策略授予用户组，并将用户添加至用户组中从而使用户拥有对应的服务权限。 	创建用户并授权使用 CloudTable
创建集群	创建Doris集群	使用Doris执行任务前，需要创建一个Doris集群。	创建Doris集群
下载客户端	下载Doris客户端	创建集群后，需要下载客户端并安装客户端，安装客户端后，用户才可以借助SSH工具连接集群。	-
准备弹性云服务	-	如果客户端工具的运行环境为Linux环境，您需要准备一台和Doris集群在相同虚拟私有云的Linux弹性云服务器作为客户端主机。 如果客户端工具的运行环境为Windows环境，您需要准备一台和Doris集群在相同虚拟私有云的Windows弹性云服务器作为客户端主机。	准备弹性云服务器

流程	子任务	说明	详细指导
安装客户端	-	将下载的客户端放入ECS，解压安装客户端。	使用MySQL客户端连接Doris普通集群
连接集群	-	在ECS安装好MySQL客户端后，可用命令连接集群并进行业务操作。	使用MySQL客户端连接Doris普通集群

3.3 创建 Doris 集群

通过表格存储服务可以集中管理集群，如果需要使用CloudTable服务，首要操作就是创建集群，本章节为您介绍如何在表格存储服务管理控制台创建一个新的集群。

Doris集群目前支持的计费模式有按需付费、包年包月付费。创建集群页面默认采用按需付费方式，即计算单元按实际使用时长计费，计费周期为一小时。这种购买方式比较灵活，可以即开即停，按实际使用时长计费。包年包月可通过预付费，以包年包月模式购买云服务，这种购买方式相对于按需付费提供更大的折扣，对于长期使用者，推荐该计费模式。您可根据业务需要，定制相应计算能力和存储空间的CloudTable Doris集群。

前提条件

- 创建集群的虚拟私有云、安全组需要和公网ECS保持一致，否则客户端无法访问集群。
- 用户创建集群前，必须配置所使用主机的入方向安全组规则，请参考[安全组规则配置](#)章节。
- 用户创建集群前安全组规则必须添加ICMP协议，方便管理面通过ping方式查看各节点状态，请参考[安全组规则配置](#)章节。

步骤

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 单击右上角的购买集群按钮，进入“购买集群”页面。
- 步骤4** 参考下面表格配置集群的基本信息。

表 3-23 区域

参数	参数说明
区域	集群的工作区域。 <ul style="list-style-type: none"> ● 选择集群节点实际工作区域。 ● 有关区域的详细信息，请参见地区和终端节点。
可用区域	选择集群工作区域下关联的可用区。 关于区域和可用区的描述，请参见 区域和可用区 。

参数	参数说明
计费模式	计费模式有按需付费和包年/包月。
订购周期	当计费模式选择包年/包月时，才会出现此选项。客户根据需求选择订购时间。
自动续费	在购买页面勾选此选项，集群到期后会自动续费，不勾选集群到期需要手动续费。

表 3-24 集群和网络配置

参数	参数说明
名称	<p>集群的名称。</p> <p>集群名称在4位到32位之间，必须以字母开头，可以包含字母、数字或者中划线，不能包含其他的特殊字符，不区分大小写。</p>
虚拟私有云	<p>VPC即虚拟私有云，是通过逻辑方式进行网络隔离，提供安全、隔离的网络环境。</p> <p>默认即可。如果没有虚拟私有云，请单击“查看虚拟私有云”进入虚拟私有云，创建一个新的虚拟私有云。</p>
子网	<p>指定虚拟私有云的一个子网。</p> <p>集群使用子网实现与其他网络的隔离，并独享所有网络资源，以提高网络安全。</p>
安全组	<p>安全组用来实现安全组内和安全组间弹性云服务器的访问控制，加强弹性云服务器的安全保护。用户可以在安全组中定义各种访问规则，这些规则可以限定弹性云服务器的哪些端口或者协议可以访问，这些访问规则可以作用在进出虚拟机的网络访问上。当弹性云服务器加入该安全组后，即受到这些访问规则的保护。不属于该安全组的弹性云服务器，是无法同安全组内的弹性云服务器通信的。</p> <p>您可以选择使用已有的安全组，或者单击“查看安全组”创建新的安全组。</p> <p>关于安全组的更多信息，请参见《虚拟私有云用户指南》中的安全组。</p>
数据引擎	选择需要创建的数据组件类型。
集群存储模式	集群存储模式为存算一体。
Doris内核版本	组件引擎版本。

说明

- CloudTable Doris集群规格说明：FE节点至少为3个，从而保证高可用。
- 切换内核版本时，需要重新选择规格，避免创建集群出现问题。
- 小规格支持包周期和按需计费。

图 3-2 FE 节点配置

FE节点 ?

* 计算规格 8*CU4 (1个CU4=1核4GB)

* 存储规格 GB / 节点

* 节点数量

表 3-25 FE 节点配置

参数	参数说明
计算规格	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明</p> <p>支持的计算规格：</p> <ul style="list-style-type: none"> • 4U16G。 • 8U32G。 • 16U64G。 • 32U128G。 • 64U256G。
存储规格	<p>选Doris计算节点的磁盘规格、容量。</p> <p>说明</p> <ul style="list-style-type: none"> • 支持的存储规格： <ul style="list-style-type: none"> - 高IO。 - 超高IO。 - 极速型SSD。 • 容量范围为200~2000GB/节点。
节点数量	<p>选择集群中的节点个数。</p> <p>支持的节点数范围3个或5个。</p>

图 3-3 BE 节点配置

BE节点 ?

* 存储类型 ? 云储存 本地SSD盘

* 计算规格 8U16G 8*CU₂ (1个CU₂ = 1核2GB)

* 存储规格 高IO 400 + GB / 节点

* 节点数量 3 +

表 3-26 BE 节点云存储配置

参数	参数说明
计算规格	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明</p> <p>支持的计算规格：</p> <ul style="list-style-type: none"> • 8U32G。 • 16U64G。 • 32U128G。 • 64U256G。
存储规格	<p>选Doris计算节点的磁盘规格、容量。</p> <p>说明</p> <ul style="list-style-type: none"> • 支持的存储规格： <ul style="list-style-type: none"> - 高IO。 - 超高IO。 - 极速型SSD。 • 容量范围为400~10000GB/节点。
节点数量	<p>选择集群中的节点个数。</p> <p>支持的节点数范围3~100。</p>

表 3-27 用户名密码参数

参数	参数说明
企业项目	<p>企业项目管理帮助您将相关的资源（如具有相同使用用途的资源）集中在一起，按企业项目的方式来管理云资源。</p> <p>说明</p> <ul style="list-style-type: none"> 当企业业务发生变化，原用户不再拥有企业项目的使用权限时，可将这些用户与该企业项目的关联关系删除。 可进行单个删除和批量删除。 当用户与企业项目的关联关系被删除后，该用户将无法管理该企业项目，如需再次使用，需要重新给该用户关联企业项目。
用户名	Doris管理用户，用于连接Doris集群，默认为admin，不可修改。
密码	<p>设置admin用户密码。</p> <p>密码复杂度要求如下：</p> <ul style="list-style-type: none"> 8~12个字符。 至少包含以下字符中的4种：大写字母，小写字母、数字、特殊字符（\$@!%*?&）。 不能与用户名或倒序用户名相同。 强弱密码检查。 <p>说明 请妥善保管并定期更新密码，避免安全风险。</p>
确认密码	设置admin用户密码。
是否开启Broker进程	是否打开Broker进程，用于给客户做数据导入场景，仅支持创建时开启。
是否开启https	<p>是否开启https访问通道加密，https协议端口为8050。</p> <p>说明</p> <ul style="list-style-type: none"> 开启https时连接集群只能使用MySQL8及以上客户端，连接端口为9030，使用方法请参见连接Doris集群。 创建集群开启https选项，后续无法关闭。 创建集群未开启https选项，后续无法开启。
是否开启接口认证	<p>开启后，集群内rest接口需要添加账号密码认证，但同时会略微降低集群性能。</p> <p>说明</p> <ul style="list-style-type: none"> 开启https选项后，显示此字段。 开启该选项后，再使用Doris集群内的rest接口，如：修改参数、获取集群监控数据等接口需要添加账号密码认证。

步骤5 配置好参数，单击“立即购买”。

步骤6 进“规格详情”页面，确认集群规格订单信息，单击“提交”，成功提交集群创建任务。

步骤7 单击“返回集群管理列表”，可以查看到集群创建的状态。

集群创建需要时间，请您耐心等待。所创集群的初始状态为“创建中”，创建成功后状态更新为“服务中”。

步骤8 如果创建的集群计费模式为包年/包月，单击提交，成功提交集群创建任务。

单击“去支付”，跳转到购买页面，确认信息后，选择合适的支付方式，确认付款。

返回控制台，查看集群状态，集群创建需要时间，请您耐心等待。所创集群的初始状态为“创建中”，创建成功后状态更新为“服务中”。

----结束

3.4 连接 Doris 集群

3.4.1 准备弹性云服务器

用户创建好集群开始使用之前，需要使用客户端连接到数据库。安装客户端后可以通过集群的内网地址访问集群。

准备弹性云服务

购买弹性云服务器的操作步骤，请参见弹性云服务器[购买弹性云服务器](#)。

购买弹性云服务器需要满足以下要求：

- 弹性云服务器需要与CloudTable集群具有相同的区域、可用区、虚拟私有云和子网。
关于虚拟私有云的更多信息，请参见[虚拟私有云](#)的“用户指南 > 虚拟私有云和子网”。
- 弹性云服务器需要与CloudTable集群具有相同的安全组。
关于安全组的更多信息，请参见《虚拟私有云用户指南》中的[安全组](#)。

📖 说明

在使用跨VPC通信方式访问CloudTable集群时，需要由网络管理员对集群所在VPC，安全组和子网的访问进行授权。

- 购买弹性云服务器时，“弹性公网IP”需选择“现在购买”，或者在ECS创建完成后再为该ECS绑定一个弹性公网IP。
- 访问Linux弹性云服务器，建议使用SSH密码方式登录Linux弹性云服务器。
具体登录操作步骤请参见《弹性云服务器用户指南》中的[登录Linux弹性云服务器](#)。

3.4.2 使用 MySQL 客户端连接 Doris 普通集群

用户在弹性云服务器里可以通过[手动安装客户端](#)的方法，使用MySQL访问集群。

使用限制

- Doris集群与弹性云服务器具有相同区域、可用区、VPC。
- Doris集群与弹性云服务器具有相同安全组。
- 弹性云服务器安全组中添加本地主机IP地址。

手动安装客户端

步骤1 准备Linux弹性云服务器，具体操作请见[准备弹性云服务器](#)章节。

步骤2 安装客户端，连接集群。

1. 使用SSH登录工具，通过弹性IP远程登录到Linux弹性云服务器。

具体登录操作步骤请参见弹性云服务器《用户指南》中的“[SSH密码方式登录](#)”章节。

2. 登录后执行以下命令解压安装包。

```
cd <客户端安装包存放路径>
tar -xzf 客户端压缩包名
```

📖 说明

操作**步骤**提到的“<客户端安装包存放路径>”请替换为实际的客户端存放路径。

3. 进入bin目录。

```
cd mysql-5.7.22-linux-glibc2.12-x86_64/bin/
```

4. 然后执行下面的命令连接Doris集群。

```
./mysql -uadmin -ppassword -h集群内网地址 -P9030
```

📖 说明

- 集群内网地址：集群详情页面中集群访问地址，这里替换成您自己购买的集群的访问地址（FE节点的访问地址，都可以访问集群）。
- *password*为您购买集群时设置的密码，如果有特殊符号，特殊符号前面需要\进行转义，如果密码整体用单引号括起来，则特殊字符不需要转义。
- 端口：FE上的mysql server端口，参见[表3-28](#)。

表 3-28 自定义安全规则组

方向	策略	协议端口/范围	类型	目的/源地址	用途
出方向	允许	全部	IPv4/ IPv6	0.0.0.0/0	出方向放行。
入方向	允许	9030		CloudTable Doris集群所在的安全组。	FE上的mysql server端口。
	允许	8030		FE上的http server端口。	
	允许	8040		BE上的http server的端口。	
	允许	8050		FE上的https server端口	

----结束

Doris 入门使用

1. 创建一个数据库。

```
CREATE DATABASE demo;
```

2. 创建数据表

- 使用数据库。

```
USE demo;
```

- 创建表。

```
CREATE TABLE IF NOT EXISTS demo.example_tbl
(
  `user_id` LARGEINT NOT NULL COMMENT "用户id",
  `date` DATE NOT NULL COMMENT "数据插入日期时间",
  `city` VARCHAR(20) COMMENT "用户所在城市",
  `age` SMALLINT COMMENT "用户年龄",
  `sex` TINYINT COMMENT "用户性别",
  `last_visit_date` DATETIME REPLACE DEFAULT "1970-01-01 00:00:00" COMMENT "用户最后一次访问时间",
  `cost` BIGINT SUM DEFAULT "0" COMMENT "用户总消费",
  `max_dwll_time` INT MAX DEFAULT "0" COMMENT "用户最大停留时间",
  `min_dwll_time` INT MIN DEFAULT "99999" COMMENT "用户最小停留时间"
)
AGGREGATE KEY(`user_id`, `date`, `city`, `age`, `sex`)
DISTRIBUTED BY HASH(`user_id`) BUCKETS 1
PROPERTIES (
  "replication_allocation" = "tag.location.default: 3"
);
```

3. 插入数据。

```
INSERT INTO demo.example_tbl
(user_id,date,city,age,sex,last_visit_date,cost,max_dwll_time,min_dwll_time)
VALUES('10000','2017-10-01','A','20','0','2017-10-01 07:00:00','35','10','2'),
('10001','2017-10-01','A','30','1','2017-10-01 17:05:45','2','22','22'),
('10002','2017-10-02','B','20','1','2017-10-02 12:59:12','200','5','5'),
('10003','2017-10-02','C','32','0','2017-10-02 11:20:12','30','11','11'),
('10004','2017-10-01','D','35','0','2017-10-01 10:00:15','100','3','3'),
('10004','2017-10-03','D','35','0','2017-10-03 10:20:22','11','6','6');
```

4. 查询数据。

- 我们上面完成了建表，输数据导入，下面我们就可以体验Doris的数据快速查询分析能力。

```
mysql> SELECT * FROM
demo.example_tbl;
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
| user_id | date   | city | age | sex | last_visit_date | cost | max_dwll_time | min_dwll_time |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 10000   | 2017-10-01 | A   | 20 | 0 | 2017-10-01 07:00:00 | 35 | 10 | 2 |
| 10001   | 2017-10-01 | A   | 30 | 1 | 2017-10-01 17:05:45 | 2 | 22 | 22 |
| 10002   | 2017-10-02 | B   | 20 | 1 | 2017-10-02 12:59:12 | 200 | 5 | 5 |
| 10003   | 2017-10-02 | C   | 32 | 0 | 2017-10-02 11:20:12 | 30 | 11 | 11 |
| 10004   | 2017-10-01 | D   | 35 | 0 | 2017-10-01 10:00:15 | 100 | 3 | 3 |
| 10004   | 2017-10-03 | D   | 35 | 0 | 2017-10-03 10:20:22 | 11 | 6 | 6 |
+-----+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.02 sec)
```

- 查看指定城市信息。

```
mysql> SELECT * FROM demo.example_tbl where city='B';
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
| user_id | date   | city | age | sex | last_visit_date | cost | max_dwll_time | min_dwll_time |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 10002   | 2017-10-02 | B   | 20 | 1 | 2017-10-02 12:59:12 | 200 | 5 | 5 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.10 sec)
```

5. 删除数据。

- a. 删除指定行数据。

```
mysql> DELETE FROM demo.example_tbl WHERE user_id = 10003;
```

```
Query OK, 0 rows affected (0.04 sec)
```

```
{'label':'delete_77ed273a-a052-4d64-bac0-23916b698003', 'status':'VISIBLE', 'txnId':'39'}
```

- b. 删除表。
- ```
mysql> DROP TABLE demo.example_tbl;
Query OK, 0 rows affected (0.01 sec)
```

### 3.4.3 使用 MySQL 客户端连接 Doris 安全集群

您可以通过创建集群开启https，实现数据加密传输。此章节介绍如何开通Doris安全通道。

#### 使用限制

- 集群开启安全通道可能会导致性能下降。
- Doris集群与弹性云服务器具有相同区域、可用区、VPC。
- Doris集群与弹性云服务器具有相同安全组。
- 弹性云服务器安全组中添加本地主机IP地址。

#### 开启 HTTPS 步骤

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 单击右上角的购买集群按钮，进入“购买集群”页面。
- 步骤4** 完成其他配置后，查看https按钮是否开启，默认开启。
- 步骤5** 配置好参数，单击“立即购买”。
- 步骤6** 进入“规格详情”页面，确认集群规格订单信息，单击“提交”，成功提交集群创建任务，集群创建成功后，进入详情页面查看通道开启状态。

----结束

#### 连接 Doris 集群

- 步骤1** 安全集群创建后，打开详情页面，下载证书。
- 步骤2** 下载证书后，自定义存放路径。
- 步骤3** 连接集群。

```
./mysql -uadmin -h集群内网地址 -P端口 --ssl-ca={path}/certificate.crt --ssl-mode=VERIFY_CA -ppassword
```

- 内网地址：所连接集群的内网地址。
- path：证书存放的路径。
- 端口：FE上的mysql server端口9030。
- password：指创建集群时设置的密码。
- {path}/certificate.crt：存放下载证书的路径。

#### 说明

开启https后需要使用mysql8及以上客户端。

----结束

## Doris 入门使用

1. 创建一个数据库。

```
CREATE DATABASE demo;
```

2. 创建数据表

- 使用数据库。

```
USE demo;
```

- 创建表。

```
CREATE TABLE IF NOT EXISTS demo.example_tbl
(
 `user_id` LARGEINT NOT NULL COMMENT "用户id",
 `date` DATE NOT NULL COMMENT "数据插入日期时间",
 `city` VARCHAR(20) COMMENT "用户所在城市",
 `age` SMALLINT COMMENT "用户年龄",
 `sex` TINYINT COMMENT "用户性别",
 `last_visit_date` DATETIME REPLACE DEFAULT "1970-01-01 00:00:00" COMMENT "用户最后一次访问时间",
 `cost` BIGINT SUM DEFAULT "0" COMMENT "用户总消费",
 `max_dwll_time` INT MAX DEFAULT "0" COMMENT "用户最大停留时间",
 `min_dwll_time` INT MIN DEFAULT "99999" COMMENT "用户最小停留时间"
)
AGGREGATE KEY(`user_id`, `date`, `city`, `age`, `sex`)
DISTRIBUTED BY HASH(`user_id`) BUCKETS 1
PROPERTIES (
 "replication_allocation" = "tag.location.default: 3"
);
```

3. 插入数据。

```
INSERT INTO demo.example_tbl
(user_id,date,city,age,sex,last_visit_date,cost,max_dwll_time,min_dwll_time)
VALUES('10000','2017-10-01','A','20','0','2017-10-01 07:00:00','35','10','2'),
('10001','2017-10-01','A','30','1','2017-10-01 17:05:45','2','22','22'),
('10002','2017-10-02','B','20','1','2017-10-02 12:59:12','200','5','5'),
('10003','2017-10-02','C','32','0','2017-10-02 11:20:12','30','11','11'),
('10004','2017-10-01','D','35','0','2017-10-01 10:00:15','100','3','3'),
('10004','2017-10-03','D','35','0','2017-10-03 10:20:22','11','6','6');
```

4. 查询数据。

- 我们上面完成了建表，输数据导入，下面我们就可以体验Doris的数据快速查询分析能力。

```
mysql> SELECT * FROM
demo.example_tbl;
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+
| user_id | date | city | age | sex | last_visit_date | cost | max_dwll_time | min_dwll_time |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
10000	2017-10-01	A	20	0	2017-10-01 07:00:00	35	10	2
10001	2017-10-01	A	30	1	2017-10-01 17:05:45	2	22	22
10002	2017-10-02	B	20	1	2017-10-02 12:59:12	200	5	5
10003	2017-10-02	C	32	0	2017-10-02 11:20:12	30	11	11
10004	2017-10-01	D	35	0	2017-10-01 10:00:15	100	3	3
10004	2017-10-03	D	35	0	2017-10-03 10:20:22	11	6	6
+-----+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.02 sec)
```

- 查看指定城市信息。

```
mysql> SELECT * FROM demo.example_tbl where city='B';
```

```
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+
| user_id | date | city | age | sex | last_visit_date | cost | max_dwll_time | min_dwll_time |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 10002 | 2017-10-02 | B | 20 | 1 | 2017-10-02 12:59:12 | 200 | 5 | 5 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.10 sec)
```

5. 删除数据。

- a. 删除指定行数据。  
mysql> DELETE FROM demo.example\_tbl WHERE user\_id = 10003;  
Query OK, 0 rows affected (0.04 sec)  
{'label':'delete\_77ed273a-a052-4d64-bac0-23916b698003', 'status':'VISIBLE', 'txnId':'39'}
- b. 删除表。  
mysql> DROP TABLE demo.example\_tbl;  
Query OK, 0 rows affected (0.01 sec)

## 3.5 配置 Doris 用户权限

Doris用户权限管理实现了对集群中各个节点上用户、角色、权限的统一管理。通过在控制台创建角色、创建用户、用户绑定角色方式，实现用户权限控制，不同用户之间操作互不影响，提高业务效率。

### 注意事项

- 删除操作不可逆，即使删除后立即重新添加同名角色，也可能存在权限和以前不一致的情况。所以，删除角色前需确认此角色是否应用中。
- 删除用户前需确认此账户是否再用。删除操作不可逆，即使删除后立即重新添加同名账户，也可能存在权限和以前不一致的情况。
- 组件不支持查看已有账户的密码，如果忘记密码可以通过重置密码功能设置新密码。
- 组件支持查看账户对数据库的操作权限。
- 用户名、角色名称大小写敏感。

### 前提条件

- 已创建Doris集群，集群运行正常。
- 已安装MySQL客户端。

### 步骤一：创建 Doris 角色

- 步骤1** 登录CloudTable控制台。
- 步骤2** 在“集群管理”列表中选择目标集群，单击“集群名称 > 角色管理”，进入角色管理页面。
- 步骤3** 单击“新建角色”，弹出新建角色页面，配置表中相关参数。

表 3-29 角色权限说明

| 参数   | 说明                                    |
|------|---------------------------------------|
| 角色名  | 角色名称以字母开头，长度范围1~64。                   |
| 全局权限 | 全局权限指拥有所有数据库和表的权限，包括查询、插入、更改、创建、删除权限。 |

| 参数   | 说明                                                                                                                                                                                                                                                    |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 各级权限 | <p>数据库和表，指的是集群中所创建的数据库和表。</p> <ul style="list-style-type: none"> <li>• 数据库有“高危权限”和普通权限；表只有“普通权限”。</li> <li>• 权限类别                             <ul style="list-style-type: none"> <li>- 高危权限：创建表、删除表。</li> <li>- 普通权限：查询、插入、更改。</li> </ul> </li> </ul> |

**步骤4** 参数配置完成后，单击“确认”。

----结束

## 步骤二：创建 Doris 用户

**步骤1** 角色创建完成后，单击“用户管理”，进入用户管理页面。

**步骤2** 单击“新建用户”，弹出“新建用户”页面，配置用户名、密码。

表 3-30 用户参数说明

| 参数   | 说明                                                                                                                                                                                             |
|------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 用户名  | 新建用户名必须以字母开头，长度范围1~64。                                                                                                                                                                         |
| 密码   | <p>新建用户的密码。</p> <p><b>说明</b><br/>设置密码规则：</p> <ul style="list-style-type: none"> <li>• 8~16个字符。</li> <li>• 至少包含以下字符中的4种：大写字母、小写字母、数字、特殊字符（\$@!%*?&amp;）。</li> <li>• 不能与用户名或倒序的用户名相同。</li> </ul> |
| 确认密码 | 确认密码。                                                                                                                                                                                          |

**步骤3** 参数配置完成后，单击“确认”。

----结束

## 步骤三：Doris 用户绑定角色并访问集群

例如：角色A拥有查询、插入、更改、创建、删除。用户绑定角色A后拥有角色A权限。

**步骤1** 角色、用户创建完成后，在“用户管理”页面，选择目标用户，单击操作列“更多 > 分配角色”。

**步骤2** 弹出“分配角色”页面，选择角色。单击“确定”。

**步骤3** 单击操作列“查看权限”，用户拥有角色权限。

#### 步骤4 使用新建的用户连接集群。

非安全集群连接命令，详细操作请参见[使用MySQL客户端连接Doris普通集群](#)。

```
./mysql -uadmin -ppassword -h集群内网地址 -P9030
```

安全集群连接命令，详细操作请参见[使用MySQL客户端连接Doris安全集群](#)。

```
./mysql -uadmin -h集群内网地址 -P9030 --ssl-ca={path}/certificate.crt --ssl-mode=VERIFY_CA -ppassword
```

#### 步骤5 验证：在命令窗口中执行查询、插入、更改、创建、删除权限。

- 如果可以执行查询、插入、更改、创建、删除权限，说明绑定角色成功。
- 如果无法执行查询、插入、更改、创建、删除权限，请检查角色是否配置权限、用户是否绑定角色。以上两种情况无问题请联系技术人员处理。

----结束

## 管理用户权限

- 管理角色
  - 删除角色：单击操作列“删除”，弹出删除的页面，输入“DELETE”或者单击“一键输入”，输入后单击“确定”，删除角色。
  - 修改角色权限：单击操作列“修改”，弹出“修改角色”页面，根据用户需求选择权限，单击“确定”。
  - 查看角色权限：单击操作列“查看权限”，弹出“查看权限”页面，查看角色当下拥有使用数据库和表的权限。
- 管理用户
  - 查看用户权限：单击操作列“查看权限”，弹出“查看权限”页面，查看用户当下拥有使用数据库和表的权限。
  - 删除用户：单击操作列“删除”，弹出删除的页面，单击“确定”，删除用户。
  - 修改用户密码：单击操作列“更多 > 修改密码”，弹出“修改密码”页面，修改密码后单击“确定”。

## 用户权限常用命令

### 1. 创建角色

```
CREATE Role role_name;
```

role\_name：指新建角色名称。

### 2. 给角色赋予权限。

- 授予指定库表的权限给角色。

```
GRANT LOAD_PRIV ON ctl1.db1.* TO ROLE 'my_role';
```

- 授予指定资源的使用权限给角色。

```
GRANT USAGE_PRIV ON RESOURCE 'spark_resource' TO ROLE 'my_role';
```

### 3. 删除角色。

```
DROP ROLE role1;
```

### 4. 创建用户。

- CREATE USER命令用于创建一个Doris用户（无角色）。

```
CREATE USER 'Jack' IDENTIFIED BY 'password';
```

表 3-31 参数说明

| 参数       | 说明        |
|----------|-----------|
| Jack     | 设置的用户名。   |
| password | 指创建的用户密码。 |

#### 说明

可通过命令指定IP。

```
CREATE USER 'jack02' '@'192.168.%' identified by '123456' DEFAULT ROLE 'default_role';
```

- CREATE USER命令用于创建一个Doris用户（分配系统默认角色）。  
CREATE USER 'jeo' IDENTIFIED BY 'password' DEFAULT ROLE 'default\_role';

#### 5. 修改用户密码。

ALTER USER命令用于修改用户密码。  
ALTER USER 'Jack' IDENTIFIED BY "password";

#### 6. 给用户分配角色。

- 给用户分配角色。  
GRANT 'role1','role2' TO 'jack'@'%';

表 3-32 参数说明

| 参数   | 说明        |
|------|-----------|
| role | 代表创建的角色。  |
| jack | 代表新建的用户名。 |

- 撤销分配给用户的角色。  
REVOKE 'role1' FROM 'candy';

#### 7. 删除用户。

```
DROP USER'Jack';
```

#### 8. 查询当前用户的权限和角色。

```
SHOW GRANTS;
```

## 3.6 导入数据至 Doris 集群

### 3.6.1 导入数据至 Doris 集群方式介绍

导入（Load）功能就是将用户的原始数据导入到Doris中。导入成功后，用户即可通过Mysql客户端查询数据。Doris支持多种导入方式。

#### 支持数据源

Doris提供多种数据导入方案，可以针对不同的数据源选择不同的数据导入方式。

- [使用Broker Load方式导入数据至Doris集群](#)

- [使用Stream Load方式导入数据至Doris集群](#)

## 支持的数据格式

不同的导入方式支持的数据格式不同。

表 3-33 导入方式说明

| 导入方式        | 支持格式                 |
|-------------|----------------------|
| Broker Load | parquet、orc、obs      |
| Stream Load | csv、json、parquet、orc |

## 导入说明

Doris的数据导入实现有以下共性特征，这里分别介绍，以帮助大家更好的使用数据导入功能。

### 导入的原子性保证

Doris的每一个导入任务，不论是使用Broker Load进行批量导入，还是使用INSERT语句进行单条导入，都是一个完整的事务操作。导入事务可以保证一批次内的数据原子生效，不会出现部分数据写入的情况。

同时，每个导入任务都会有一个Label。这个Label在数据库（Database）中是唯一的，用于唯一标识一个导入任务。Label可以由用户指定，部分导入功能也会由系统自动生成。

Label是用于保证对应的导入任务，仅能成功导入一次。一个被成功导入的Label，再次使用时，会被拒绝并报错Label already used。通过这个机制，可以在Doris侧做到At-Most-Once语义。如果结合上游系统的At-Least-Once语义，则可以实现导入数据的Exactly-Once语义。

## 同步和异步

Doris目前的导入方式分为两类，同步和异步。如果是外部程序接入Doris的导入功能，需要判断使用导入方式是哪类再确定接入逻辑。

- 同步

同步导入方式即用户创建导入任务，Doris同步执行导入，执行完成后返回用户导入结果。用户可直接根据创建导入任务命令返回的结果同步判断导入是否成功。

- 异步

异步导入方式即用户创建导入任务后，Doris直接返回创建成功。创建成功不代表数据已经导入。导入任务会被异步执行，用户在创建成功后，需要通过轮询的方式发送命令查看导入作业的状态。如果创建失败，则可以根据失败信息，判断是否需要再次创建。

### 📖 说明

无论是异步还是同步的导入类型，都不应该在Doris返回导入失败或导入创建失败后，无休止的重试。外部系统在有限次数重试并失败后，保留失败信息，大部分多次重试均失败问题都是使用方法问题或数据本身问题。

### 3.6.2 使用 Broker Load 方式导入数据至 Doris 集群

Broker Load是一个异步的导入方式，支持的数据源取决于Broker进程支持的数据源。本文为您介绍Broker Load导入的基本原理、基本操作、系统配置以及最佳实践。

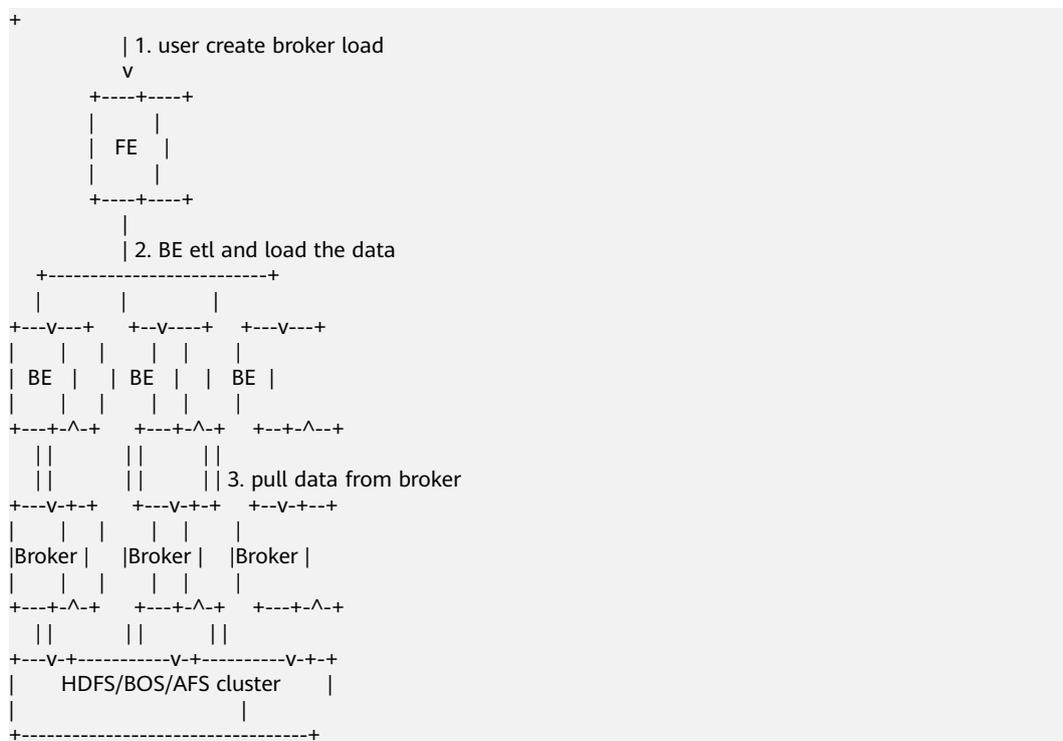
#### 适用场景

- 源数据在Broker可以访问的存储系统中，如HDFS、OBS。
- 数据量在几十到百GB级别。

#### 基本原理

用户在提交导入任务后，FE会生成对应的Plan并根据目前BE的个数和文件的大小，将Plan分给多个BE执行，每个BE执行一部分导入数据。

BE在执行的过程中会从Broker拉取数据，在对数据transform之后将数据导入系统。所有BE均完成导入，由FE最终决定导入是否成功。



#### 开始导入

下面我们通过几个实际的场景示例来看Broker Load的使用。

数据样例：

```
'100','101','102','103','104','105',100.00,100.01,100.02,'100',200,100.08,2022-04-01
'101','102','103','104','105','105',100.00,100.01,100.02,'100',200,100.08,2022-04-02
'102','103','104','105','106','105',100.00,100.01,100.02,'100',200,100.08,2022-04-03
```

准备工作：

在本地创建示例数据文件source\_text.txt，并上传至hdfs的/tmp/。

在hive中创建ods\_source表。

```
CREATE TABLE `ods_source` (
 `id` string,
 `store_id` string,
 `company_id` string,
 `tower_id` string,
 `commodity_id` string,
 `commodity_name` string,
 `commodity_price` double,
 `member_price` double,
 `cost_price` double,
 `unit` string,
 `quantity` string,
 `actual_price` double,
 `day` string
)
row format delimited fields terminated by ','
lines terminated by '\n'
stored as textfile;
```

将hdfs创建的txt文件导入到ods\_source表。

```
load data inpath '/tmp/source_text.txt' into table ods_source;
```

- 示例1，textfile格式表导入。
  - 在hive中创建分区表并写入数据。

- 创建ods\_demo\_detail表。

```
CREATE TABLE `ods_demo_detail` (
 `id` string,
 `store_id` string,
 `company_id` string,
 `tower_id` string,
 `commodity_id` string,
 `commodity_name` string,
 `commodity_price` double,
 `member_price` double,
 `cost_price` double,
 `unit` string,
 `quantity` string,
 `actual_price` double
)
PARTITIONED BY (day string)
row format delimited fields terminated by ','
lines terminated by '\n'
stored as textfile;
```

- 把ods\_source表中的数据导入到ods\_demo\_detail表中。

```
set hive.exec.dynamic.partition.mode=nonstrict;
set hive.exec.dynamic.partition=true;
insert overwrite table ods_demo_detail partition(day) select * from ods_source;
```

- 查看Hive表ods\_demo\_detail有没有数据。
- 在Doris创建数据库。
- 创建Doris表doris\_ods\_test\_detail。

#### 说明

如果创建集群的时候不是超高io，则去掉'storage\_medium' = 'SSD'。

```
use doris_demo_db;
CREATE TABLE `doris_ods_test_detail` (
 `rq` date NULL,
 `id` varchar(32) NOT NULL,
 `store_id` varchar(32) NULL,
 `company_id` varchar(32) NULL,
```

```

`tower_id` varchar(32) NULL,
`commodity_id` varchar(32) NULL,
`commodity_name` varchar(500) NULL,
`commodity_price` decimal(10, 2) NULL,
`member_price` decimal(10, 2) NULL,
`cost_price` decimal(10, 2) NULL,
`unit` varchar(50) NULL,
`quantity` int(11) NULL,
`actual_price` decimal(10, 2) NULL
) ENGINE=OLAP
UNIQUE KEY(`rq`, `id`, `store_id`)
PARTITION BY RANGE(`rq`)
(
PARTITION P_202204 VALUES [('2022-04-01'),('2022-08-30')]
DISTRIBUTED BY HASH(`store_id`) BUCKETS 1
PROPERTIES (
'replication_allocation' = 'tag.location.default: 3',
'dynamic_partition.enable' = 'true',
'dynamic_partition.time_unit' = 'MONTH',
'dynamic_partition.start' = '-2147483648',
'dynamic_partition.end' = '2',
'dynamic_partition.prefix' = 'P_',
'dynamic_partition.buckets' = '1',
'in_memory' = 'false',
'storage_format' = 'V2',
'storage_medium' = 'SSD'
);

```

- 导入数据。

```

LOAD LABEL broker_name_test01
(
DATA INFILE('hdfs://{hdfs远端ip}:{hdfs远端口}/user/hive/warehouse/ods_demo_detail/*/*')
INTO TABLE doris_ods_test_detail
COLUMNS TERMINATED BY ','
(id,store_id,company_id,tower_id,commodity_id,commodity_name,commodity_price,member_price,cost_price,unit,quantity,actual_price)
COLUMNS FROM PATH AS (`day`)
SET
(rq = str_to_date(`day`, '%Y-%m-%d'),id=id,store_id=store_id,company_id=company_id,tower_id=tower_id,commodity_id=commodity_id,commodity_name=commodity_name,commodity_price=commodity_price,member_price=member_price,cost_price=cost_price,unit=unit,quantity=quantity,actual_price=actual_price)
)
WITH BROKER 'broker1'
(
'username' = 'hdfs',
'password' = ''
)
PROPERTIES
(
'timeout'='1200',
'max_filter_ratio'=0.1'
);

```

- 查看导入状态

您可以通过下面的命令查看上面导入任务的状态信息。

```
show load order by createtime desc limit 1\G;
```

## 说明

如果状态信息出现"Scan bytes per file scanner exceed limit: 3221225472", 说明导入失败, 需要修改参数"max\_bytes\_per\_broker\_scanner", 请参见[修改Doris参数配置对集群进行性能调优](#)章节的FE节点参数说明表。

图 3-4 查看数据导入状态

```
mysql> show load order by createtime desc limit 2\G;
***** 1. row *****
JobId: 11015
Label: label_2023070614
States: CANCELED
Progress: ETL:N/A; LOAD:N/A
Type: BROKER
EtlInfo: NULL
TaskInfo: cluster:N/A; timeout(s):120000; max_filter_ratio:0.1
ErrorMsg: type:ETL_RUN_FAIL; msg:errorCode = 2, detailMessage = Scan bytes per file scanner exceed limit: 3221225472
CreateTime: 2023-07-06 11:34:04
EtlStartTime: NULL
EtlFinishTime: NULL
LoadStartTime: NULL
LoadFinishTime: 2023-07-06 11:34:08
URL: NULL
JobDetails: {"Unfinished backends": {}, "ScannedRows": 0, "TaskNumber": 0, "LoadBytes": 0, "All backends": {}, "FileNumber": 1, "FileSize": 98630126365}
TransactionId: 1007
ErrorTables: {}
```

- 示例2, ORC格式表导入。

- 在Hive中创建Hive分区表, ORC格式。

```
CREATE TABLE `ods_demo_orc_detail` (
```

```
 `id` string,
 `store_id` string,
 `company_id` string,
 `tower_id` string,
 `commodity_id` string,
 `commodity_name` string,
 `commodity_price` double,
 `member_price` double,
 `cost_price` double,
 `unit` string,
 `quantity` double,
 `actual_price` double
)
```

```
 PARTITIONED BY (day string)
 row format delimited fields terminated by ','
 lines terminated by '\n'
 STORED AS ORC;
```

- 查询Source表写入分区表。

```
insert overwrite table ods_demo_orc_detail partition(day) select * from ods_source;
```

- 创建Doris表。

```
CREATE TABLE `doris_ods_orc_detail` (
 `rq` date NULL,
 `id` varchar(32) NOT NULL,
 `store_id` varchar(32) NULL,
 `company_id` varchar(32) NULL,
 `tower_id` varchar(32) NULL,
 `commodity_id` varchar(32) NULL,
 `commodity_name` varchar(500) NULL,
 `commodity_price` decimal(10, 2) NULL,
 `member_price` decimal(10, 2) NULL,
 `cost_price` decimal(10, 2) NULL,
 `unit` varchar(50) NULL,
 `quantity` int(11) NULL,
 `actual_price` decimal(10, 2) NULL
) ENGINE=OLAP
UNIQUE KEY(`rq`, `id`, `store_id`)
PARTITION BY RANGE(`rq`)
(
 PARTITION P_202204 VALUES [('2022-04-01'), ('2022-08-30'))
)
DISTRIBUTED BY HASH(`store_id`) BUCKETS 1
PROPERTIES (
 'replication_allocation' = 'tag.location.default: 3',
 'dynamic_partition.enable' = 'true',
 'dynamic_partition.time_unit' = 'MONTH',
 'dynamic_partition.start' = '-2147483648',
 'dynamic_partition.end' = '2',
```

```
'dynamic_partition.prefix' = 'P_',
'dynamic_partition.buckets' = '1',
'in_memory' = 'false',
'storage_format' = 'V2');
```

- 导入数据。

```
LOAD LABEL orc_2022_02_17
(
 DATA INFILE("${hdfs}://{hdfs远端ip}:${hdfs远端端口号}/user/hive/warehouse/
ods_demo_orc_detail/*/*")
 INTO TABLE doris_ods_orc_detail
 COLUMNS TERMINATED BY ","
 FORMAT AS 'orc'
(id,store_id,company_id,tower_id,commodity_id,commodity_name,commodity_price,member_pric
e,cost_price,unit,quantity,actual_price)
 COLUMNS FROM PATH AS (`day`)
 SET
 (rq = str_to_date(`day`, '%Y-%m-
%d'),id=id,store_id=store_id,company_id=company_id,tower_id=tower_id,commodity_id=commodi
ty_id,commodity_name=commodity_name,commodity_price=commodity_price,member_price=m
ember_price,cost_price=cost_price,unit=unit,quantity=quantity,actual_price=actual_price)
)
WITH BROKER 'broker1'
(
 'username' = 'hdfs',
 'password' = ""
)
PROPERTIES
(
 'timeout'="1200",
 'max_filter_ratio'="0.1"
);
```

- 查询导入数据。

```
show load order by createtime desc limit 1\G;
```

• 示例3，OBS格式数据导入。

- 创建Doris表。

```
CREATE TABLE `obs_detail_test` (
 `id` varchar(32) NOT NULL,
 `store_id` varchar(32) NULL,
 `company_id` varchar(32) NULL,
 `tower_id` varchar(32) NULL,
 `commodity_id` varchar(32) NULL,
 `commodity_name` varchar(500) NULL,
 `commodity_price` decimal(10, 2) NULL,
 `member_price` decimal(10, 2) NULL,
 `cost_price` decimal(10, 2) NULL,
 `unit` varchar(50) NULL,
 `quantity` int(11) NULL,
 `actual_price` decimal(10, 2) NULL
) ENGINE=OLAP
UNIQUE KEY(`id`, `store_id`)
DISTRIBUTED BY HASH(`store_id`) BUCKETS 1
PROPERTIES (
'replication_allocation' = 'tag.location.default: 3',
'in_memory' = 'false',
'storage_format' = 'V2'
);
```

- 将OBS数据导入到Doris表。

构造text数据100条，该数据与Doris表字段对应，将数据上传到OBS桶。

```
LOAD LABEL label_2023021801
(
 DATA INFILE("${obs}://xxx/source_text2.txt")
 INTO TABLE `obs_detail_test`
 COLUMNS TERMINATED BY ","
(id,store_id,company_id,tower_id,commodity_id,commodity_name,commodity_price,member_pric
e,cost_price,unit,quantity,actual_price)
```

```

)
WITH BROKER 'broker1' (
'fs.obs.access.key' = 'xxx',
'fs.obs.secret.key' = 'xxxxxx',
'fs.obs.endpoint' = 'xxxxxx'
)
PROPERTIES
(
'timeout'='1200',
'max_filter_ratio'='0.1'
);

```

fs.obs.access.key、fs.obs.secret.key、fs.obs.endpoint获取方式如下所示：

- fs.obs.access.key与fs.obs.secret.key的获取方式请参见对象存储服务的[获取访问密钥（AK/SK）](#)章节。
  - fs.obs.endpoint的获取方式见请参见对象存储服务的[获取终端节点](#)章节。
- 查询数据。
- ```
show load order by createtime desc limit 1\G;
```
- 示例4，使用With HDFS方式将HDFS的数据导入Doris表。

- 创建Doris表。

```

CREATE TABLE `ods_dish_detail_test` (
`id` varchar(32) NOT NULL,
`store_id` varchar(32) NULL,
`company_id` varchar(32) NULL,
`tower_id` varchar(32) NULL,
`commodity_id` varchar(32) NULL,
`commodity_name` varchar(500) NULL,
`commodity_price` decimal(10, 2) NULL,
`member_price` decimal(10, 2) NULL,
`cost_price` decimal(10, 2) NULL,
`unit` varchar(50) NULL,
`quantity` int(11) NULL,
`actual_price` decimal(10, 2) NULL
) ENGINE=OLAP
UNIQUE KEY(`id`, `store_id`)
DISTRIBUTED BY HASH(`store_id`) BUCKETS 1
PROPERTIES (
'replication_allocation' = 'tag.location.default: 3',
'in_memory' = 'false',
'storage_format' = 'V2'
);

```

- 导入数据。

构造text数据100条，该数据与Doris表字段对应。

```

LOAD LABEL label_2023021703
(
DATA INFILE("hdfs://{hdfs远端ip}:{hdfs远端口号}/tmp/source_text2.txt")
INTO TABLE `ods_dish_detail_test`
COLUMNS TERMINATED BY ","
(id,store_id,company_id,tower_id,commodity_id,commodity_name,commodity_price,member_pric
e,cost_price,unit,quantity,actual_price)
)
with HDFS (
'fs.defaultFS'="hdfs://{hdfs远端ip}:{hdfs远端口号}",
'hadoop.username'="hdfs",
'password'=""
)
PROPERTIES
(
'timeout'="1200",
'max_filter_ratio'="0.1"
);

```

- 查询数据。
show load order by createtime desc limit 1\G;

取消导入

当Broker load作业状态不为CANCELLED或FINISHED时，可以被用户手动取消。取消时需要指定待取消导入任务的Label。

相关系统配置

FE配置。下面几个配置属于Broker load的系统级别配置，也就是作用于所有Broker load导入任务的配置。主要通过修改**FE配置项**来调整配置值。

max_bytes_per_broker_scanner/max_broker_concurrency

max_bytes_per_broker_scanner配置限制了单个BE处理的数据量的最大值。
max_broker_concurrency配置限制了一个作业的最大的导入并发数。最小处理的数据量（默认64M），最大并发数，源文件的大小和当前集群BE的个数共同决定了本次导入的并发数。

本次导入并发数=Math.min(源文件大小/最小处理量（默认64M），最大并发数，当前BE节点个数)。
本次导入单个BE的处理量=源文件大小/本次导入的并发数。

通常一个导入作业支持的最大数据量为max_bytes_per_broker_scanner*BE节点数。如果需要导入更大数据量，则需要适当调整max_bytes_per_broker_scanner参数的大小。

默认配置：

- 参数名：max_broker_concurrency，默认10。
- 参数名：max_bytes_per_broker_scanner，默认3G，单位bytes。

应用实践

- 应用场景。
使用Broker load最适合的场景就是原始数据在文件系统（HDFS，BOS，AFS）中的场景。其次，由于Broker load是单次导入中唯一的一种异步导入的方式，所以如果用户在导入大文件中，需要使用异步接入，也可以考虑使用Broker load。
- 数据量。
这里仅讨论单个BE的情况，如果用户集群有多个BE则下面标题中的数据量应该乘以BE个数来计算。比如：如果用户有3个BE，则3G以下（包含）则应该乘以3，也就是9G以下（包含）。
 - 3G以下（包含）：用户可以直接提交Broker load创建导入请求。
 - 3G以上：由于单个导入BE最大的处理量为3G，超过3G的待导入文件就需要通过调整Broker load的导入参数来实现大文件的导入。
 - 根据当前BE的个数和原始文件的大小修改单个BE的最大扫描量和最大并发数。
修改fe配置项。
max_broker_concurrency=BE个数。
当前导入任务单个BE处理的数据量=原始文件大小/max_broker_concurrency。
max_bytes_per_broker_scanner >=当前导入任务单个BE处理的数据量。
比如一个100G的文件，集群的BE个数为10个。
max_broker_concurrency=10。
max_bytes_per_broker_scanner >=10G=100G/10。
修改后，所有的BE会并发的处理导入任务，每个BE处理原始文件的一部分。

📖 说明

上述两个FE中的配置均为系统配置，也就是说其修改是作用于所有的Broker load的任务的。

- 在创建导入的时候自定义当前导入任务的timeout时间。

这时候不推荐用户将导入最大超时时间直接改大来解决问题。单个导入时间如果超过默认的导入最大超时时间4小时，最好是通过切分待导入文件并且分多次导入来解决问题。主要原因是：单次导入超过4小时的话，导入失败后重试的时间成本很高。

可以通过如下公式计算出Doris集群期望最大导入文件数据量：

期望最大导入文件数据量=14400s*10M/s*BE个数。

比如：集群的BE个数为10个。

期望最大导入文件数据量=14400s*10M/s*10 =1440000M≈1440G。

📖 说明

一般用户的环境可能达不到10M/s的速度，所以建议超过500G的文件都进行文件切分，再导入。

作业调度

系统会限制一个集群内正在运行的Broker Load作业数量，以防止同时运行过多的Load作业。

首先，FE的配置参数：`desired_max_waiting_jobs`会限制一个集群内未开始或正在运行（作业状态为PENDING或LOADING）的Broker Load作业数量。默认为100。如果超过这个阈值，新提交的作业将会被直接拒绝。

一个Broker Load作业会被分为pending task和loading task阶段。其中pending task负责获取导入文件的信息，而loading task会发送给BE执行具体的导入任务。

FE的配置参数`async_pending_load_task_pool_size`用于限制同时运行的pending task的任务数量。也相当于控制了实际正在运行的导入任务数量。该参数默认为10。也就是说，假设用户提交了100个Load作业，同时只会有10个作业会进入LOADING状态开始执行，而其他作业处于PENDING等待状态。

FE的配置参数`async_loading_load_task_pool_size`用于限制同时运行的loading task的任务数量。一个Broker Load作业会有1 pending task和多个loading task（等于LOAD语句中DATA INFILE子句的个数）。所以`async_loading_load_task_pool_size`应该大于等于`async_pending_load_task_pool_size`。

性能分析

可以在提交LOAD作业前，先执行`set enable_profile=true`打开会话变量。然后提交导入作业。待导入作业完成后，可以在FE的web页面的Queries标签中查看到导入作业的Profile。

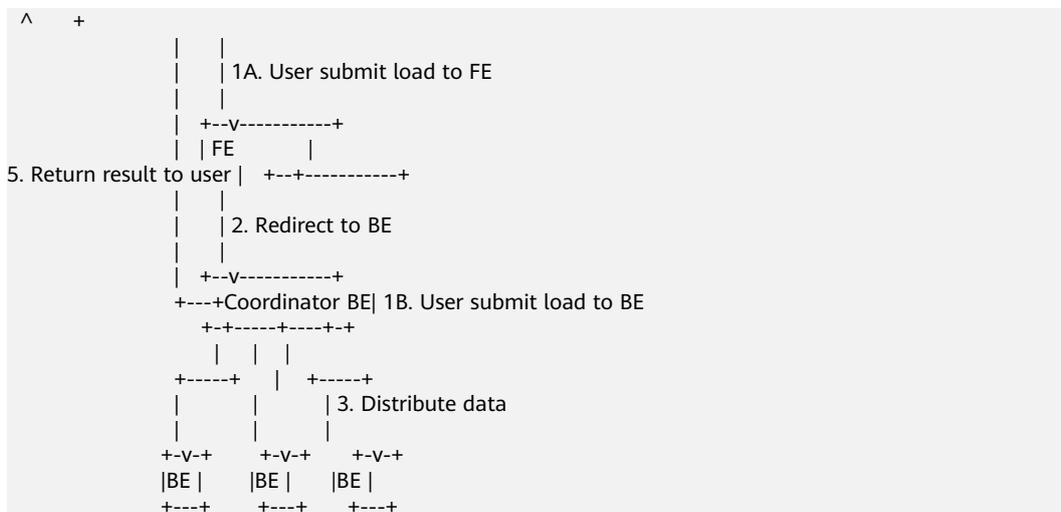
3.6.3 使用 Stream Load 方式导入数据至 Doris 集群

Stream load是一个同步的导入方式，用户通过发送HTTP协议发送请求将本地文件或数据流导入到Doris中。Stream load同步执行导入并返回导入结果。用户可直接通过请求的返回体判断本次导入是否成功。

Stream load主要适用于导入本地文件，或通过程序导入数据流中的数据。

基本原理

下图展示了Stream load的主要流程，省略了一些导入细节。



Stream load中，Doris会选定一个节点作为Coordinator节点。该节点负责接数据并分发数据到其他数据节点。您可以通过HTTP协议提交导入命令。如果提交到FE，则FE会通过HTTP redirect指令将请求转发给某一个BE。用户也可以直接提交导入命令给某一指定BE。导入的最终结果由Coordinator BE返回给用户。

基本操作

📖 说明

在执行数据导入Stream Load操作之前，必须确保Doris集群的安全组端口开放，即8030和8040端口，否则Stream Load操作将会连接超时。

● 创建导入

Stream Load通过HTTP协议提交和传输数据。这里通过curl命令展示如何提交导入。

用户也可以通过其他HTTP client进行操作。

```
curl --location-trusted -u user:passwd [-H ""...] -T data.file -XPUT http://fe_host:http_port/api/{db}/{table}/_stream_load
```

📖 说明

- Header中支持属性见下面的‘导入任务参数’说明。
- 格式为：-H "key1:value1"。
- port: HTTP的端口。

创建导入任务的详细语法可以通过HELP STREAM LOAD命令查看。Stream Load中所有与导入任务相关的参数均设置在Header中。相关参数描述如下表所示。

表 3-34 参数说明

参数		说明
签名参数	user/ passwd	Stream load由于创建导入的协议使用的是HTTP协议，通过Basic access authentication进行签名。Doris系统会根据签名验证用户身份和导入权限。

参数	说明	
导入任务参数	label	<p>导入任务的标识。每个导入任务，都有一个在单database内部唯一的label。label是用户在导入命令中自定义的名称。通过这个label，用户可以查看对应导入任务的执行情况。</p> <p>label的另一个作用，是防止用户重复导入相同的数据。强烈推荐用户同一批次数据使用相同的label。这样同一批次数据的重复请求只会被接受一次，保证了At-Most-Once。当label对应的导入作业状态为CANCELLED时，该label可以再次被使用。</p>
	column_separator	<p>用于指定导入文件中的列分隔符，默认为\t。如果是不可见字符，则需要加\x作为前缀，使用十六进制来表示分隔符。</p> <p>如hive文件的分隔符\x01，需要指定为-H "column_separator:\x01"。</p> <p>可以使用多个字符的组合作为列分隔符。</p>
	line_delimiter	<p>用于指定导入文件中的换行符，默认为\n。</p> <p>可以使用做多个字符的组合作为换行符。</p>
	max_filter_ratio	<p>导入任务的最大容忍率，默认为0容忍，取值范围是0~1。当导入的错误率超过该值，则导入失败。</p> <p>如果用户希望忽略错误的行，可以通过设置这个参数大于0，来保证导入可以成功。</p> <p>计算公式为： $(dpp.abnorm.ALL / (dpp.abnorm.ALL + dpp.norm.ALL)) > max_filter_ratio$ dpp.abnorm.ALL表示数据质量不合格的行数。如类型不匹配，列数不匹配，长度不匹配等等。 dpp.norm.ALL指的是导入过程中正确数据的条数。可以通过SHOW LOAD命令查询导入任务的正确数据量。 原始文件的行数 = dpp.abnorm.ALL + dpp.norm.ALL。</p>
	where	<p>导入任务指定的过滤条件。Stream load支持对原始数据指定where语句进行过滤。被过滤的数据将不会被导入，也不会参与filter ratio的计算，但会被计入num_rows_unselected。</p>
	Partitions	<p>待导入表的Partition信息，如果待导入数据不属于指定的Partition则不会被导入。这些数据将计入dpp.abnorm.ALL。</p>
	columns	<p>待导入数据的函数变换配置，目前Stream load支持的函数变换方法包含列的顺序变化以及表达式变换，其中表达式变换的方法与查询语句的一致。</p>

参数		说明
	exec_mem_limit	导入内存限制。默认为2GB，单位为字节。
	strict_mode	Stream Load导入可以开启strict mode模式。开启方式为在HEADER中声明strict_mode=true。默认的strict mode为关闭。 strict mode模式的意思是：对于导入过程中的列类型转换进行严格过滤。严格过滤的策略如下： 对于列类型转换来说，如果strict mode为true，则错误的的数据将被filter。这里的错误数据是指：原始数据并不为空值，在参与列类型转换后结果为空值的这一类数据。对于导入的某列由函数变换生成时，strict mode对其不产生影响。对于导入的某列类型包含范围限制的，如果原始数据能正常通过类型转换，但无法通过范围限制的，strict mode对其也不产生影响。例如：如果类型是decimal(1,0)，原始数据为10，则属于可以通过类型转换但不在列声明的范围内。这种数据strict对其不产生影响。
	merge_type	数据的合并类型，一共支持三种类型APPEND、DELETE、MERGE其中，APPEND是默认值，表示这批数据全部需要追加到现有数据中，Delete表示删除与这批数据Key相同的所有行，MERGE语义需要与Delete条件联合使用，表示满足Delete条件的数据按照Delete语义处理其余的按照APPEND语义处理。

参数	说明
two_phase_commit	<p>Stream load导入可以开启两阶段事务提交模式：在Stream load过程中，数据写入完成即会返回信息给用户，此时数据不可见，事务状态为PRECOMMITTED，用户手动触发commit操作之后，数据才可见。</p> <p>示例：</p> <ul style="list-style-type: none"> 发起Stream load预提交操作。 <pre>curl --location-trusted -u user:passwd -H "two_phase_commit:true" -T test.txt http://fe_host:http_port/api/{db}/{table}/_stream_load { "TxnId": 18036, "Label": "55c8ffc9-1c40-4d51-b75e-f2265b3602ef", "TwoPhaseCommit": "true", "Status": "Success", "Message": "OK", "NumberTotalRows": 100, "NumberLoadedRows": 100, "NumberFilteredRows": 0, "NumberUnselectedRows": 0, "LoadBytes": 1031, "LoadTimeMs": 77, "BeginTxnTimeMs": 1, "StreamLoadPutTimeMs": 1, "ReadDataTimeMs": 0, "WriteDataTimeMs": 58, "CommitAndPublishTimeMs": 0 }</pre> 对事务触发commit操作： <ul style="list-style-type: none"> 注意1：请求发往fe或be均可。 注意2：commit的时候可以省略url中的{table}。 <pre>curl -X PUT --location-trusted -u user:passwd -H "txn_id:18036" -H "txn_operation:commit" http://fe_host:http_port/api/{db}/{table}/_stream_load_2pc { "status": "Success", "msg": "transaction [18036] commit successfully." }</pre> 对事务触发abort操作： <ul style="list-style-type: none"> 注意1：请求发往FE或BE均可。 注意2：abort的时候可以省略URL中的{table}。 <pre>curl -X PUT --location-trusted -u user:passwd -H "txn_id:18037" -H "txn_operation:abort" http://fe_host:http_port/api/{db}/{table}/_stream_load_2pc { "status": "Success", "msg": "transaction [18037] abort successfully." }</pre>

- 示例1，CSV数据格式导入。

- 创建Doris表

```
CREATE TABLE clouddtable0327.doris_streamload_test01
(
  user_id bigint,
```

```

date date,
group_id bigint,
modify_date date,
keyword VARCHAR(128)
)
UNIQUE KEY(user_id, date, group_id)
DISTRIBUTED BY HASH (user_id) BUCKETS 32
PROPERTIES(
'function_column.sequence_col' = 'modify_date',
'replication_num' = '3',
'in_memory' = 'false'
);

```

- 准备数据表sequencedata01.csv。

表 3-35 sequencedata01.csv

1	2020-02-22	1	2020-02-21	a
1	2020-02-22	1	2020-02-22	b
1	2020-02-22	1	2020-03-05	c
1	2020-02-22	1	2020-02-26	d
1	2020-02-22	1	2020-02-23	e
1	2020-02-22	1	2020-02-24	b

- 执行curl命令load数据。

```

curl -k --location-trusted -u admin:passwd -T sequencedata01.csv -H 'column_separator:'
https://{fe_host}:{http_port}/api/cloudtable0327/doris_streameload_test01/_stream_load

```

- 返回结果。

由于Stream load是一种同步的导入方式，所以导入的结果会通过创建导入的返回值直接返回给用户。

```

{
  "TxnId": 1003,
  "Label": "b6f3bc78-0d2c-45d9-9e4c-faa0a0149bee",
  "Status": "Success",
  "ExistingJobStatus": "FINISHED", // optional
  "Message": "OK",
  "NumberTotalRows": 1000000,
  "NumberLoadedRows": 1000000,
  "NumberFilteredRows": 1,
  "NumberUnselectedRows": 0,
  "LoadBytes": 40888898,
  "LoadTimeMs": 2144,
  "BeginTxnTimeMs": 1,
  "StreamLoadPutTimeMs": 2,
  "ReadDataTimeMs": 325,
  "WriteDataTimeMs": 1933,
  "CommitAndPublishTimeMs": 106,
  "ErrorURL": "http://{fe_host:http_port}/api/_load_error_log?file=_shard_0/
error_log_insert_stmt_db18266d4d9b4ee5-
abb00ddd64bdf005_db18266d4d9b4ee5_abb00ddd64bdf005"
}

```

- Stream load导入结果参数如下表。

表 3-36 参数说明

参数	说明
TxnId	导入的事务ID。用户可不感知。
Label	导入Label。由用户指定或系统自动生成。
Status	<p>导入完成状态。</p> <ul style="list-style-type: none"> • Success: 表示导入成功。 • Publish Timeout: 该状态也表示导入已经完成，只是数据可能会延迟可见，无需重试。 • Label Already Exists: Label重复，需更换Label。 • Fail: 导入失败。
ExistingJobStatus	<p>已存在的Label对应的导入作业的状态。</p> <p>这个字段只有在当Status为“Label Already Exists”时才会显示。用户可以通过这个状态，知晓已存在Label对应的导入作业的状态。</p> <p>“RUNNING”表示作业还在执行，“FINISHED”表示作业成功。</p>
Message	导入错误信息。
NumberTotalRows	导入总处理的行数。
NumberLoadedRows	成功导入的行数。
NumberFilteredRows	数据质量不合格的行数。
NumberUnselectedRows	被where条件过滤的行数。
LoadBytes	导入的字节数。
LoadTimeMs	导入完成时间。单位毫秒。
BeginTxnTimeMs	向Fe请求开始一个事务所花费的时间，单位毫秒。
StreamLoadPutTimeMs	向Fe请求获取导入数据执行计划所花费的时间，单位毫秒。
ReadDataTimeMs	读取数据所花费的时间，单位毫秒。
WriteDataTimeMs	执行写入数据操作所花费的时间，单位毫秒。
CommitAndPublishTimeMs	向Fe请求提交并且发布事务所花费的时间，单位毫秒。
ErrorURL	如果有数据质量问题，通过访问这个URL查看具体错误行。

📖 说明

由于Stream load是同步的导入方式，所以并不会在Doris系统中记录导入信息，用户无法异步的通过查看导入命令看到Stream load。使用时需监听创建导入请求的返回值获取导入结果。

- 示例2，json数据格式导入。

准备json格式数据并保存为testjson.json，并将json数据上传至doris客户端:

```
{"id": 100, "city": "B", "code" : 1}
```

- 创建Doris表。

```
CREATE TABLE `doris_testjson01` (  
  `id` varchar(32) NOT NULL,  
  `city` ARRAY<int(11)>,  
  `code` int(11)  
) ENGINE=OLAP  
DUPLICATE KEY(`id`)  
COMMENT "OLAP"  
DISTRIBUTED BY HASH(`id`) BUCKETS 1  
PROPERTIES (  
  'replication_allocation' = 'tag.location.default: 3',  
  'in_memory' = 'false',  
  'storage_format' = 'V2'  
);
```

- curl命令进行load数据。

```
curl --location-trusted -u admin:{doris集群密码} -H 'format: json' -T testjson.json https://  
fe_host:http_port/api/{doris数据库}/doris_testjson01/_stream_load -k
```

- 查询数据。

```
select * from doris_testjson01;
```

取消导入

用户无法手动取消Stream Load，Stream Load在超时或者导入错误后会被系统自动取消。

查看 Stream load

用户可以通过show stream load来查看已经完成的stream load任务。

3.7 Doris 企业级能力增强

3.7.1 配置 Doris 租户

3.7.1.1 添加租户

Doris的多租户管理，目的是为了多个用户在同一Doris集群内进行数据操作时，减少相互之间的干扰，能够将集群资源更合理的分配给各用户。且集成了租户生命周期管理、租户资源配置和租户资源使用统计等功能，为企业提供了成熟的多租户管理模式，实现集中式的租户和业务管理。

新建租户步骤

- 步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击左侧集群管理，显示当前集群列表。

步骤4 选择需要操作的集群，单击“集群名称 > 租户管理”进入租户管理页面。

步骤5 单击页面右上角“新建租户”，弹出新建租户页面，参照表1为租户配置属性。

表 3-37 新建租户参数描述

参数	取值范围	描述
租户名	-	指定当前的租户名称。以字母开头，长度为4-64位，只能输入大小写字母、数字和下划线。
CPU配额权重	1~922337203685477580	租户占用CPU资源的相对权重。 例如只有租户A执行查询时，此时租户A的CPU占用率为100%。如果有两个租户A和B执行查询，对应该值分别为A: 10, B: 20, 此时租户A的CPU占用率为三分之一，即为： $10/(10+20)$ 。如果另有一租户C开始执行，对应该值为30，则租户A的CPU占用率为六分之一，即为： $10/(10+20+30)$ 。
内存限额占比 (%)	1%~100%	租户最大占用内存资源的百分比。
并发数	1~2147483647	租户最大可运行查询任务的并发数。该值是对单个FE的限定。例如设置并发数为1，Doris有三个FE节点，则从集群角度看，最大可运行的SQL数为3。
队列长度	1~2147483647	租户最大等待执行的查询任务数。超过并发数的SQL会进入队列等待，当队列满了之后，新提交的查询会被拒绝。
队列等待时长 (毫秒)	1~2147483647	租户等待执行的查询任务最大的等待时长。如果查询等待时间超过该值，那么查询会被拒绝，时间单位为毫秒。
开启内存软隔离	开启、关闭	租户是否可以超量占用内存资源。 <ul style="list-style-type: none"> 如果关闭内存软隔离，系统检测到租户内存使用超出限制后，将立即取消租户组内内存占用最大的若干个任务。 如果开启内存软隔离，若该系统有空闲内存资源时，则该租户在超出内存的限制后可继续使用系统内存；只有当集群资源紧张时，才取消租户组内内存占用最大的若干任务。

步骤6 单击“确定”，租户创建完成。

----结束

用户绑定租户

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 单击左侧集群管理，显示当前集群列表。
- 步骤4** 选择需要操作的集群，单击“集群名称 > 用户管理”进入用户管理页面。
- 步骤5** 选择需要绑定的用户，单击操作列“更多 > 分配租户”，弹出分配租户窗口。
- 步骤6** 选择租户后，单击“确定”，租户分配完成。

📖 说明

- 新建用户默认绑定默认租户normal。
- 一个用户只能绑定一个租户。
- 一个租户可以被多个用户绑定。

----结束

管理租户

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 单击左侧集群管理，显示当前集群列表。
- 步骤4** 选择需要操作的集群，单击“集群名称 > 租户管理”进入租户管理页面。
- 步骤5** 选择需要编辑的租户，单击操作列的“修改”，弹出修改界面，修改参数后，单击“确定”。

📖 说明

租户名称不可修改。

- 步骤6** 如果此租户不可用，也可删除此租户，单击操作列的“删除”，弹出确定窗口，单击“确定”，租户删除。
- 步骤7** 如果想查看多租户的监控信息，单击目标租户操作列“查看监控信息”，可查看租户的监控信息。

----结束

3.7.1.2 租户命令简介

此章节主要介绍SQL基本语法和使用说明。

1. 新增租户。

```
create workload group if not exists test_group properties (  
  "cpu_share"="10",  
  "memory_limit"="30%",  
  "enable_memory_overcommit"="true",  
  "max_concurrency" = "10",  
  "max_queue_size" = "20",  
  "queue_timeout" = "3000"  
);
```

2. 修改租户。

```
alter workload group test_group properties
(cpu_share='20',memory_limit='4%',enable_memory_overcommit='true',max_concurrency='15',max_queue_size='25',queue_timeout='4000');
```
3. 用户绑定租户：通过设置user property将user绑定到workload group，默认为normal。

```
set property for doris_user 'default_workload_group' = 'test_group';
```
4. 查询用户所属租户。

```
show property for user_name like 'default_workload_group';
```
5. 给绑定租户的用户授权。

```
GRANT USAGE_PRIV ON WORKLOAD GROUP 'test_group' TO 'jack';
```
6. 撤销旧租户权限。

```
revoke USAGE_PRIV ON WORKLOAD GROUP 'test_group' from 'jack';
```
7. 查看租户。

```
select * from information_schema.workload_groups;
```
8. 删除租户。

```
mysql> DROP WORKLOAD GROUP if exists 'test_group';
Query OK, 0 rows affected (0.01 sec)
```

3.7.2 配置 Doris 连接数据源

3.7.2.1 Doris 多源数据能力概述

Doris的Hive外表自带create catalog能力，通过连接Hive Metastore，或者兼容Hive Metastore的元数据服务自动获取Hive库表信息，并进行表数据查询，从而避免了传统外部数据目录多需要手动映射以及数据迁移的复杂工程。

多源数据目录（Multi-Catalog）是Doris 1.2.0版本中推出的功能，旨在能够更方便对接外部数据目录，以增强Doris的数据湖分析和联邦数据查询能力。

在之前的Doris版本中，用户数据只有两个层级：Database和Table。当我们需要连接一个外部数据目录时，我们只能在Database或Table层级进行对接。比如通过create external table的方式创建一个外部数据目录中的表的映射，或通过create external database的方式映射一个外部数据目录中的Database。如果外部数据目录中的Database或Table非常多，则需要用户手动进行一一映射，使用体验不佳。

而新的Multi-Catalog功能在原有的元数据层级上，新增一层Catalog，构成Catalog->Database->Table的三层元数据层级。其中，Catalog可以直接对应到外部数据目录。目前支持的外部数据目录包括：

1. Hive
2. JDBC: 对接数据库访问的标准接口（JDBC）来访问各式数据库的数据。

背景

许多客户的Hive表数据存储存储在OBS或HDFS上，需要Doris对接Hive外表，且MRS集群分为安全集群与非安全集群，所以可以使用以下4种方式进行数据查询：

- 认证类型选择SIMPLE，访问存储在HDFS中的Hive数据。
- 认证类型选择KERBEROS，访问存储在HDFS中的Hive数据。
- 认证类型选择SIMPLE，访问存储在OBS中的Hive数据。
- 认证类型选择KERBEROS，访问存储在OBS中的Hive数据。

Kerberos 介绍

Hadoop社区版本提供两种认证方式Kerberos认证（安全模式）和Simple认证（普通模式），在创建集群时，MRS支持配置是否启用Kerberos认证。

在安全模式下MRS集群统一使用Kerberos认证协议进行安全认证。

- 功能
使用Kerberos的系统在设计上采用“客户端/服务器”结构与AES等加密技术，并且能够进行相互认证（即客户端和服务端均可对对方进行身份认证）。可以用于防止窃听、防止replay攻击、保护数据完整性等场合，是一种应用对称密钥体制进行密钥管理的系统。
- 使用前提
需要Kerberos客户端、keytab路径、Kerberos认证用户名称、客户端配置krb5.conf文件。
- 原理结构图
Kerberos原理架构图如下图所示，详细介绍请参见MRS服务的[安全认证原理以及认证机制](#)章节。

图 3-5 原理架构图

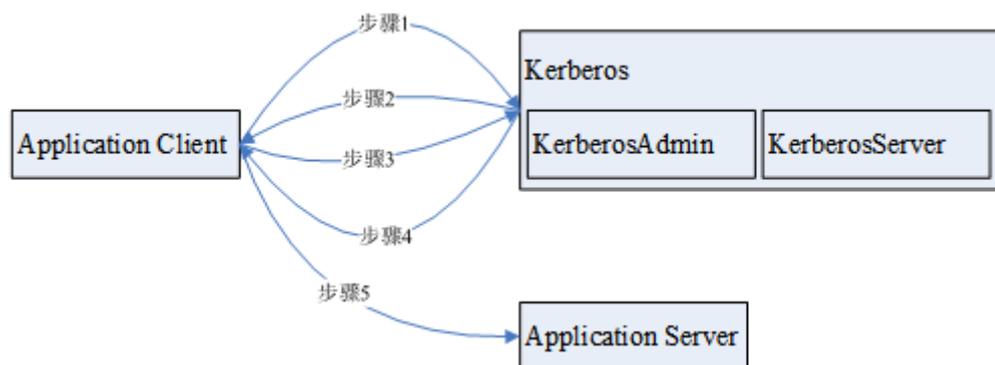


表 3-38 参数说明

参数	说明
Application Client	应用客户端，通常是需要提交任务（或者作业）的应用程序。
Application Server	应用服务端，通常是应用客户端需要访问的应用程序。
Kerberos	提供安全认证的服务。
KerberosAdmin	提供认证用户管理的进程。
KerberosServer	提供认证票据分发的进程。

基础概念

1. Internal Catalog
Doris原有的Database和Table都将归属于Internal Catalog。Internal Catalog是内置的默认Catalog，用户不可修改或删除。

2. External Catalog

可以通过CREATE CATALOG命令创建一个External Catalog。创建后，可以通过SHOW CATALOGS命令查看已创建的Catalog。

3. 切换Catalog

用户登录Doris后，默认进入Internal Catalog，因此默认的使用和之前版本并无差别，可以直接使用SHOW DATABASES，USE DB等命令查看和切换数据库。

用户可以通过SWITCH命令切换Catalog。如：

```
SWITCH internal;  
SWITCH hive_catalog;
```

切换后，可以直接通过SHOW DATABASES，USE DB等命令查看和切换对应Catalog中的Database。Doris会自动通过Catalog中的Database和Table。用户可以像使用Internal Catalog一样，对External Catalog中的数据进行查看和访问。

当前，Doris只支持对 External Catalog中的数据进行只读访问。

4. 删除Catalog

External Catalog中的Database和Table都是只读的。但是可以删除Catalog（Internal Catalog无法删除）。可以通过DROP CATALOG命令删除一个External Catalog。

该操作仅会删除Doris中该Catalog的映射信息，并不会修改或变更任何外部数据目录的内容。

5. Resource

Resource是一组配置的集合。用户可以通过CREATE RESOURCE命令创建一个Resource。之后可以在创建Catalog时使用这个Resource。

一个Resource可以被多个Catalog使用，以复用其中的配置。

3.7.2.2 配置 Doris 对接 Hive 数据源

通过连接Hive Metastore，或者兼容Hive Metastore的元数据服务，Doris可以自动获取Hive库表信息，并进行表数据查询，从而避免了传统外部数据目录多需要手动映射以及数据迁移的复杂工程。

CloudTable目前不支持S3协议，推荐使用OBS协议创建Hive Catalog。

使用前提

- 已[创建MRS集群](#)。
- [创建Doris集群](#)。
- 为了确保网络连通，MRS集群必须与Doris集群的VPC、子网保持一致，且将Doris集群的所有节点IP加入到MRS集群的安全组中。

步骤一：创建 MRS hive 连接

步骤1 登录表格存储服务管理控制台。

步骤2 用户在集群管理界面，选择目标集群进入集群详情页面。

步骤3 单击“连接管理 > 新建连接”，弹出新建连接页面，创建MRS Hive连接。

步骤4 新建连接类型：

- 认证类型选择SIMPLE，OBS按钮关闭状态。

- 认证类型选择KERBEROS，OBS按钮关闭状态。
- 认证类型选择SIMPLE，OBS按钮开启状态。
- 认证类型选择KERBEROS，OBS按钮开启状态。

表 3-39 参数说明

参数	描述
名称	根据连接的数据源类型，用户可以自定义连接名。 说明 必须以字母开头，由字母、数字、中划线组成，不能包含其他的特殊字符，并且长度为4~64个字符。
连接器	Doris支持的外部数据目录组件，目前支持可选：Hive。
Hadoop类型	Doris支持hadoop组件，目前支持的可选：MRS。
集群名称	MRS集群名称。 说明 Doris只能对接一个kerberos集群中的一个用户。
Manager IP	MRS Manager的浮动IP地址，选择连接的集群后，ip地址会自动填充。
认证类型	MRS集群鉴权类型，安全集群可选“KERBEROS”，非安全集群可选“SIMPLE”。选择连接的集群后，安全类型会自动填充。 说明 KERBEROS类型连接只能创建一个。
连接器版本	MRS集群的数据源组件的版本，HIVE支持3版本，可选：3.X。
用户名	MRS集群的用户名，且此用户必须为高权限可以访问Hive、HDFS的底层数据。如果您想创建新用户，请参见 创建MRS集群用户 。
密码	MRS集群mrs_user_name对应的密码。
OBS支持	打开按钮，当前创建的catalog连接的数据表在OBS上。关闭按钮当前创建的catalog连接的数据表在HDFS。

步骤5 配置好参数后，单击测试。如果测试失败请检查用户名和密码是否正确。

步骤6 测试完成后，单击确定按钮，连接设置页面会出现一条新的连接记录。

----结束

步骤二：使用 SQL 命令创建 Hive catalog

步骤1 使用SSH登录工具，通过弹性IP远程登录到Linux弹性云服务器。

具体登录操作步骤请参见弹性云服务器《用户指南》中的“[SSH密码方式登录](#)”。

步骤2 访问Doris集群，参见[使用MySQL客户端连接Doris普通集群](#)。

步骤3 创建Hive catalog表。

- 创建认证类型为SIMPLE的catalog，访问存储在HDFS中的Hive数据。

```
CREATE CATALOG hive_catalog_simple PROPERTIES (
  'type'='hms',
  'hive.metastore.uris'='thrift://192.X.X.X:port,thrift://192.x.x.x:port',
  'hive.metastore.sasl.enabled' = 'false',
  'dfs.nameservices'='hacluster',
  'dfs.ha.namenodes.hacluster'='3,4',
  'dfs.namenode.rpc-address.hacluster.3'='192.x.x.x:port',
  'dfs.namenode.rpc-address.hacluster.4'='192.x.x.x:port',
  'dfs.client.failover.proxy.provider.hacluster'='****',
  'hive.version' = '3.1.0'
);
```

- 创建认证类型为KERBEROS的catalog，访问存储在HDFS中的Hive数据。

```
CREATE CATALOG hive_catalog PROPERTIES (
  'type'='hms',
  'hive.metastore.uris' = 'thrift://192.x.x.x:port,thrift://192.x.x.x:port',
  'hive.metastore.sasl.enabled' = 'true',
  'hive.server2.thrift.sasl.qop'='auth-conf',
  'hive.server2.authentication' = 'KERBEROS',
  'hive.server2.authentication.kerberos.principal' = '****',
  'hive.metastore.kerberos.principal' = '****',
  'dfs.nameservices'='hacluster',
  'dfs.ha.namenodes.hacluster'='3,4',
  'dfs.namenode.rpc-address.hacluster.3'='192.x.x.x:port',
  'dfs.namenode.rpc-address.hacluster.4'='192.x.x.x:port',
  'dfs.client.failover.proxy.provider.hacluster'='****',
  'hadoop.security.authentication'='kerberos',
  'hadoop.kerberos.principal' = '{kinit_result}', -- kinit {USER_NAME} 的结果
  'hive.version' = '3.1.0',
  'fs.defaultFS'='hdfs://hacluster',
  'hadoop.rpc.protection'='privacy'
);
```

- 创建认证类型为SIMPLE的catalog，访问存储在OBS中的Hive数据。

```
CREATE CATALOG hive_catalog_simple_obs PROPERTIES (
  'type'='hms',
  'hive.metastore.uris'='thrift://192.x.x.x:port,thrift://192.x.x.x:port',
  'obs.access_key' = '****',
  'obs.secret_key' = '****',
  'obs.endpoint' = '****',
  'obs.region' = '****',
  'hive.metastore.sasl.enabled' = 'true',
  'hive.version' = '3.1.0'
);
```

- 创建认证类型为KERBEROS的catalog，访问存储在OBS中的Hive数据。

```
CREATE CATALOG hive_catalog_OBS PROPERTIES (
  'type'='hms',
  'hive.metastore.uris' = 'thrift://192.x.x.x:port,thrift://192.x.x.x:port',
  'hive.metastore.sasl.enabled' = 'true',
  'hive.server2.thrift.sasl.qop'='auth-conf',
  'hive.server2.authentication' = 'KERBEROS',
  'hive.server2.authentication.kerberos.principal' = '****',
  'hive.metastore.kerberos.principal' = '****',
  'hadoop.security.authentication'='kerberos',
  'hadoop.kerberos.principal' = 'USER_NAME',
  'hive.version' = '3.1.0',
  'fs.defaultFS'='hdfs://hacluster',
  'hadoop.rpc.protection'='privacy',
  'obs.access_key' = '****',
  'obs.secret_key' = '****',
  'obs.endpoint' = '****',
  'obs.region' = '****'
);
```

相关参数描述以及参数值获取方法。

表 3-40 参数说明

参数	说明
type	对接外部数据类型。
hive.metastore.uris	hive元数据uri，可在hive-site.xml配置文件查看。
hive.metastore.sasl.enabled	可在hive-site.xml配置文件查看。
dfs.nameservices	可在hdfs-site.xml配置文件查看。
dfs.ha.namenodes.hacluster	可在hdfs-site.xml配置文件查看。
dfs.namenode.rpc-address.hacluster.3	可在hdfs-site.xml配置文件查看。 说明 此参数的IP地址可在MRS集群的FusionInsight Manager页面>主页>组件>实例页面查看。
dfs.namenode.rpc-address.hacluster.4	可在hdfs-site.xml配置文件查看。 说明 此参数的IP地址可在MRS集群的FusionInsight Manager页面>主页>组件>实例页面查看。
dfs.client.failover.proxy.provider.hacluster	可在hdfs-site.xml配置文件查看。
hive.version	Hive版本。
hive.server2.thrift.sasl.qop	可在hive-site.xml配置文件查看。
hive.server2.authentication	认证类型，在集群连接页面查看。
hive.server2.authentication.kerberos.principal	可在hive-site.xml配置文件查看。
hive.metastore.kerberos.principal	可在hive-site.xml配置文件查看。

参数	说明
hadoop.kerberos.principal	<p>获取hadoop.kerberos.principal参数值方法：</p> <ul style="list-style-type: none"> - 方法一： <ol style="list-style-type: none"> 1. 登录MRS集群的Manager页面。 2. 单击“系统 > 权限 > 用户”进入用户页面获取用户。 3. 单击“域和互信”，进入域和互信页面获取“本端域”。 4. 参数值为：用户名+本端域。 - 方法二： <ol style="list-style-type: none"> 1. 以root用户登录MRS客户端所在节点。 2. 进入安装客户端路径。 cd /opt/Bigdata/client/ 3. 加载环境变量。 source bigdata_env 4. 执行kinit命令进行用户认证，获取参数值。 kinit 用户名 <div style="border: 1px solid black; padding: 2px; margin-top: 5px;"> <pre>[root@node-master2pKLu client]# kinit ... Password for ...@D55D_76D5_4640_B: ... IE60EAF7.COM</pre> </div>
hive.version	Hive版本。
fs.defaultFS	可在core-site.xml配置文件查看。
hadoop.rpc.protection	可在core-site.xml配置文件查看。
obs.access_key	<p>access key，获取方式请参见获取访问密钥（AK/SK）章节。</p> <p>说明 obs.access_key可以访问到底层数据文件。</p>
obs.secret_key	secret key，获取方式请参见 获取访问密钥（AK/SK） 章节。
obs.endpoint	obs地址，获取方式请参见 获取终端节点 章节。
obs.region	obs region，在obs页面查看。

- a. 以root用户登录MRS客户端所在节点（Master1节点）。
- b. 进入client目录下，显示出MRS的全部组件，目前使用的组件有HDFS和Hive组件。

```
cd /opt/Bigdata/client/
```

📖 说明

HDFS组件中包含了core-site.xml、hdfs-site.xml配置文件，Hive组件中包含了hive-site.xml配置。

- c. 进入组件中的配置文件查看参数。
 - i. HDFS组件查看。
cd ./HDFS/hadoop/etc/
进入hadoop目录。

```
cd hadoop
```

查看hdfs-site.xml配置文件，查找相关参数。

```
cat hdfs-site.xml
```

- ii. 查看Hive组件中的hive-site.xml配置文件，查找相关参数。

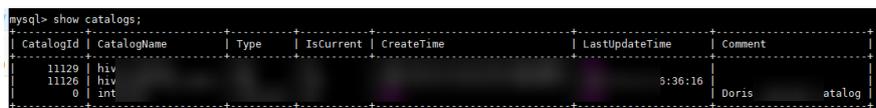
```
cat /opt/Bigdata/client/Hive/config/hive-site.xml
```

----结束

步骤三：查询数据映射表

步骤1 查看Catalogs。

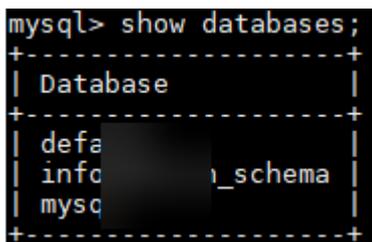
```
show catalogs;
```



CatalogId	CatalogName	Type	IsCurrent	CreateTime	LastUpdateTime	Comment
11129	hiv					
11126	hiv				5:36:16	
0	int					Doris atalog

步骤2 执行以下命令查询Catalog下面的数据库。

```
show databases from catalog名;
```



Database	_schema
defa	
info	
mysq	

步骤3 切换到指定的Catalog。

```
switch Catalog名;
```

步骤4 查看catalog数据库中的指定表。

```
show tables from `catalog名`.`数据库`;
```

指定表。

```
select * from `catalog名`.`数据库名`.`指定表名`;
```

📖 说明

use database;命令只适用于数据表比较少的测试场景，生产集群请谨慎使用。如果数据表比较多，使用此命令会加载所有表信息，容易造成hive-mastore压力剧增影响MRS生产环境。

----结束

常见问题

- 当出现以下报错的时候，需要修改MRS集群的manager页面切换HDFS的主备。

```
mysql> select * from hive_hdfs_test2;
ERROR 1105 (HY000): errCode = 2, detailMessage = get file split failed for table: hive_hdfs_test2, err: org.apache.hadoop.ipc.RemoteException: Operation category READ is not supported in state standby. Visit https://s.apache.org/sbnn-error
at
org.apache.hadoop.hdfs.server.namenode.ha.StandbyState.checkOperation(StandbyState.java:108)
at org.apache.hadoop.hdfs.server.namenode.NameNode
$NameNodeHAContext.checkOperation(NameNode.java:2200)
at org.apache.hadoop.hdfs.server.namenode.FSNamesystem.checkOperation(FSNamesystem.java:
```

- 创建Catalog时可以采用参数file.meta.cache.ttl-second来设置Hive分区文件缓存自动失效时间，也可以将该值设置为0来禁用分区文件缓存，时间单位为：秒。示例如下：

```
CREATE CATALOG hive_catalog PROPERTIES (
  'type'='hms',
  'hive.metastore.uris' = 'thrift://127.x.x.x:port',
  'obs.access_key' = '****',
  'obs.secret_key' = '****',
  'obs.endpoint' = '****',
  'obs.region' = '****'
  'file.meta.cache.ttl-second' = '60'
);
```

3.8 管理 Doris 集群

3.8.1 查看 Doris 集群状态

登录表格存储服务，在左侧导航树，单击集群管理，集群管理列表中会显示CloudTable所有的集群，集群数量较多时，可采用翻页显示，您可以查看任何状态下的集群。

集群管理列表按照集群状态顺序排列，服务中的集群显示在最前端。

表 3-41 集群管理列表参数

参数	参数说明
集群名称	集群的名称，创建集群时设置。
高可用	创建Doris集群节点规格选择小于8U32G，集群默认开启高可用，此处显示“该集群仅能用于测试业务”，如果未开启此处显示“--”。
集群状态	集群当前所处状态，包括创建中、服务中、亚健康、创建失败、冻结。
任务状态	集群任务的状态信息。
引擎类型	Doris
集群版本	集群的版本信息。
企业项目	企业项目管理帮助您将相关的资源（如具有相同使用用途的资源）集中在一起，按企业项目的方式来管理云资源。 说明 <ul style="list-style-type: none"> 当企业业务发生变化，原用户不再拥有企业项目的使用权限时，可将这些用户与该企业项目的关联关系删除。 可进行单个删除和批量删除。 当用户与企业项目的关联关系被删除后，该用户将无法管理该企业项目，如需再次使用，需要重新给该用户关联企业项目。
创建时间	集群创建的时间。
计费模式	集群的计费模式。

参数	参数说明
链接地址（内网）	内网链接地址。
操作	<ul style="list-style-type: none"> 删除：单击“更多 > 删除”，删除集群（按需集群）。 重启：单击“更多 > 重启”，集群重启。 查看监控信息：单击“查看监控信息”，进入“云服务监控 > 表格存储服务”的监控图表页面。 扩容：扩容集群中的计算规格（规格变更）、存储规格（磁盘扩容）、节点数量（节点扩容）。具体操作请参见Doris集群容量调整。 转包周期：单击操作列“更多 > 转包周期”，计费模式转为包年/包月。 退订/释放：单击操作列“更多 > 退订”，释放包周期。 续费：给包周期集群续费，单击“更多 > 续费”，进入续费页面。 查看订单详情：查看当前集群的订单详情。单击“更多 > 查看订单详情”，进入我的订单页面。 开通自动续费：开通当前包周期集群的自动续费。单击“更多 > 开通自动续费”，进入自动续费页面。

表 3-42 按钮说明

按钮	说明
	单击  查看所有项目。
	单击  ，手动刷新集群列表。
	在搜索框中输入集群名称，单击  ，搜索集群。

集群状态

表 3-43 集群状态说明

状态	说明
创建中	表示集群正在创建中。
服务中	当集群创建成功并正常提供服务时，其集群状态为服务中。
亚健康	当超过指定时间监控不到集群状态时，集群状态将显示为“亚健康”。“亚健康”状态的集群需要人工干预让集群恢复正常，比如，可以通过重启集群尝试恢复正常状态。

状态	说明
创建失败	表示集群创建失败。
冻结	续费时所支付的余额已不足，则其状态为“冻结”。 当集群状态为“冻结”时，您需要通过续费保证账户余额不为0才能解冻。 说明 冻结期间集群不可用且关闭集群中所有云主机，解冻后集群自动恢复为“服务中”状态。如果用户没有续费，冻结期限到期后集群会被删除。

任务状态

表 3-44 任务状态说明

状态	说明
删除中	表示集群正在删除中。
重启中	表示集群正在重启中。
修改参数中	表示集群正在修改参数。
节点扩容中	表示集群正在节点扩容。
磁盘扩容中	表示集群正在磁盘扩容。
规格变更中	表示集群正在变更规格。

3.8.2 查看 Doris 集群详情

集群创建完成后，可对集群进行监控和管理。在CloudTable管理控制台单击集群管理，在集群列表中找到所要查看的集群并单击集群名称，进入集群基本信息页面，用户可查看集群信息和集群的网络配置。

表 3-45 集群信息

参数	说明
集群名称	集群的名称，创建集群时设置。
集群ID	集群的ID。
集群状态	集群状态信息。
集群版本	集群的内核版本。
集群访问地址	集群访问的地址，用户可以单击  ，将访问的链接地址复制到剪切板。
计费模式	显示集群的计费模式。

参数	说明
创建时间	显示集群创建的时间。
数据库引擎	用于存储、处理和保护数据的核心服务。
是否开启Broker进程	是否打开Broker进程，用于给客户做数据导入场景。
是否开启https	“是”表示已经开启安全通道，可以下载证书。
集群存储模式	集群存储模式为存算一体。
是否开启接口认证	“是”表示开启接口认证，Doris集群内的rest接口需要账号密码认证，“否”表示未开启接口认证。 说明 当开启https后，显示此字段。
企业项目	企业项目管理帮助您将相关的资源（如具有相同使用用途的资源）集中在一起，按企业项目的方式来管理云资源。 说明 <ul style="list-style-type: none"> 当企业业务发生变化，原用户不再拥有企业项目的使用权限时，可将这些用户与该企业项目的关联关系删除。 可进行单个删除和批量删除。 当用户与企业项目的关联关系被删除后，该用户将无法管理该企业项目，如需再次使用，需要重新给该用户关联企业项目。

表 3-46 网络配置

参数	说明
区域	集群的工作区域，在创建集群时设置。
可用区	显示创建集群时所选择的可用区信息。
虚拟私有云	VPC信息，创建集群时所选。 VPC即虚拟私有云，是通过逻辑方式进行网络隔离，提供安全、隔离的网络环境。
子网	子网信息，创建集群时所选。 通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全。
安全组	显示创建集群时所选择的安全组信息。

表 3-47 FE 节点配置

参数	参数说明
计算规格	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明 支持的计算规格：</p> <ul style="list-style-type: none"> • 4U16G。 • 8U32G。 • 16U64G。 • 32U128G。 • 64U256G。
存储规格	<p>选Doris计算节点的磁盘规格、容量。</p> <p>说明</p> <ul style="list-style-type: none"> • 选Doris计算节点的磁盘规格、容量： <ul style="list-style-type: none"> • 高IO。 • 超高IO。 • 极速SSD。 • 容量范围为200~2000GB/节点。
节点数量	<p>选择集群中的节点个数。</p> <p>支持的节点数范围3个或者5个。</p>

表 3-48 BE 节点配置

参数	参数说明
计算规格	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明 支持的计算规格：</p> <ul style="list-style-type: none"> • 4U16G。 • 8U32G。 • 16U64G。 • 32U128G。 • 64U256G。
存储规格	<p>选Doris计算节点的磁盘规格、容量。</p> <p>说明</p> <ul style="list-style-type: none"> • 支持的存储规格： <ul style="list-style-type: none"> • 高IO。 • 超高IO。 • 极速SSD。 • 容量范围为400~10000GB/节点。

参数	参数说明
节点数量	选择集群中的节点个数。 支持的节点数范围3~100个。

3.8.3 重启 Doris 集群节点

当CloudTable集群节点状态异常时，可通过执行重启节点操作尝试将该节点状态恢复正常。

注意事项

- 重启节点过程中，节点不可用。
- 重启后节点请在业务低谷期进行重启，避免对高峰期业务造成影响。
- 节点重启时，磁盘扩容、节点扩容、规格变更按钮置灰，不可操作。
- 节点重启是进程重启，不是整个节点重启。

节点重启操作步骤

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群管理，进入集群列表。

步骤4 选择需要操作的集群，单击集群名称，进入集群详情页。

步骤5 在集群详情页单击“操作 > 重启”，弹出节点重启窗口。

步骤6 输入“RESTART”或者单击“一键输入”，输入后单击“确定”，节点重启。

节点重启后，“重启”按钮置灰。

步骤7 查看节点重启结果，如果重启成功，则“重启”按钮高亮显示。如果节点重启失败，则任务状态显示“节点重启失败”，“重启”按钮高亮显示，节点可继续重启。

----结束

节点重启状态

表 3-49 重启状态

状态	状态描述
重启	重启按钮高亮显示，一种是节点未重启，一种是节点已经重启。
重启中	集群节点正在重启，且在“集群信息 > 任务状态”显示“节点重启中”。
重启失败	集群节点重启失败，且在“集群信息 > 任务状态”显示“节点重启失败”，重启失败的节点可继续重启。

3.8.4 重启 Doris 集群

当集群处于非均衡或不能正常工作时，可能需要通过重启集群进行恢复。当您修改完配置，例如修改集群安全设置、参数修改相关配置，未立即重启集群的情况下，您可以通过手动重启集群使配置生效。

说明

- 如果集群欠费，可能会导致该功能被限制不可用，请及时充值确保集群可正常使用。
- 如果集群状态亚健康，也会限制该功能不可用，请及时联系技术工程师处理使集群可正常使用。

对系统的影响

- 重启期间集群将无法提供服务。因此，在重启前，请确定集群中没有正在运行的任务，并且所有数据都已经保存。
- 如果集群正在处理业务数据，如导入数据、查询数据，一旦重启集群，有可能导致文件损坏或重启失败。因此，建议停止所有集群任务后，再重启集群。
- 如果重启失败，将有可能导致集群不可用，建议联系技术支持人员进行处理或稍后重试。

操作步骤

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 在集群管理列表中的右上角，输入要搜索的集群名称，单击 。

步骤4 选中需要重启的集群名，在对应的**操作列**单击“更多 > 重启”。

步骤5 在弹出的对话框中，勾选了解重启集群的影响后，单击“确定”，重启集群。

图 3-6 确认重启窗口

 **确认要对集群cloudtable-9781进行重启操作?**

重启集群将会导致业务的中断。

我已了解重启集群带来的后果，确定执行本次操作。

确定

取消

----结束

3.8.5 删除 Doris 集群

如果不再需要集群时，可以删除或退订/释放集群。删除或退订/释放CloudTable集群将会清空所有与该集群相关的资源及数据，且无法恢复，请谨慎操作。

- 按需集群支持直接删除，详情请参见[删除Doris按需集群](#)。
- 包年/包月集群不能直接删除，需要退订（未超期集群）/释放（已超期未续费的集群），详情请参见[退订/释放包周期集群](#)。

删除 Doris 按需集群

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 选中需要删除的集群名，在对应的**操作列**单击“更多 > 删除”。

步骤4 在弹出的对话框中，输入“DELETE”或者单击“一键输入”，输入后单击“确定”，删除集群。

图 3-7 确认删除窗口



----结束

退订/释放包周期集群

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 选中需要删除的集群名，在对应的“操作”列中，单击“更多 > 退订/释放”。

步骤4 在弹出的对话框中，输入“DELETE”或者单击“一键输入”，输入后单击“确定”，“退订/释放”集群。

步骤5 进入“退订资源”页面，确认待退订集群信息，选择退订原因，并确认退订金额和相关费用。

步骤6 勾选“我已确认本次退订的资源已完成数据备份或不再使用，未放入回收站的资源退订后无法恢复。查看回收站说明”，然后单击“退订”。

步骤7 返回控制台查看集群已经退订/释放。

----结束

3.8.6 Doris 按需集群转包周期集群

如果您需长期使用当前集群，为节省开支，可以将按需购买的集群转为包年/包月计费模式（也称为包周期计费模式）。按需转包年/包月的具体操作，请参见本章节内容。

按需转包年包月

- 前提条件：集群计费模式是“按需计费”。
- 按需转包年/包月操作步骤
 - a. 登录CloudTable管理控制台。
 - b. 单击集群管理，显示当前集群列表。
 - c. 在集群列表中，选择指定集群所在行的“操作”列，单击“更多 > 转包周期”。

说明

转包周期操作也可以在集群的详情页面单击“转包周期”，跳转到包年/包月页面。

- d. 跳转到包年/包月页面，选择购买时长，单击“去支付”。
- e. 跳转到支付页面，选择支付方式，单击“确认付款”。
- f. 订单支付成功后，返回集群列表，单击指定集群进入详情页面，查看“计费模式”。

3.9 Doris 集群运维

3.9.1 Doris 集群容量调整

3.9.1.1 Doris 扩容说明

用户需要更多的计算资源或存储资源满足业务需要时，可以在管理控制台进行扩容操作，对已经创建的集群**增加节点数**（节点扩容）或对集群的节点**增加磁盘容量**（纵向扩容）。Doris可以很方便地对节点FE、BE进行节点扩容。一般情况下，可以通过将FE扩展至3个以上实现FE的高可用，且FE节点的扩容过程，不影响当前系统运行。BE节点的扩容过程，不影响当前系统运行以及正在执行的任务，并且不会影响当前系统的性能，数据均衡会自动进行，集群会在几个小时到一天不等的时间内恢复到负载均衡的状态。Doris集群目前不支持纵向扩容，而Doris本身有多种方式可以实现纵向扩容，因此需要对Doris集群增加纵向扩容服务。

说明

新增节点按照当前集群的计费模式进行计费。

注意事项

- 当集群状态处于“服务中”，且没有任何任务操作（如：规格变更、节点扩容、磁盘扩容）时，可以执行扩容操作。
- 请确定计划扩容的计算单元数小于等于用户计算单元的剩余配额，否则系统会提示无法进行扩容操作。
- 节点扩容过程中，不会自动重启集群，保证业务不中断，平滑扩容。纵向扩容过程中，集群会停止运行，直到参数设置完后才会重启集群的各个节点。

- 如果扩容失败，“任务状态”将显示为“单元扩容失败”，集群会自动回滚到扩容前的状态。用户可以再次重试扩容操作。

3.9.1.2 Doris 集群节点扩容

节点扩容增加集群中FE/BE节点的数量，FE/BE节点的扩容过程，不影响当前系统运行。

📖 说明

节点扩容需要注意以下几点：

- 扩容后的FE节点个数必须是奇数，3个或5个。
- 扩容后的BE节点个数为4~100之间。

操作步骤

步骤1 登录CloudTable管理控制台。

步骤2 单击集群管理，显示当前集群列表。

步骤3 在集群列表中，选定集群的操作列，选择“更多 > 节点扩容”，进入扩容页面。

📖 说明

节点扩容操作也可以在集群详情页面单击“节点扩容”，跳转节点扩容页面。

步骤4 选择变配节点为FE/BE。

步骤5 按“+”增加扩容节点。

步骤6 确定费用以及资源配额后，单击确定。

----结束

扩容状态

表 3-50 节点扩容任务状态说明

任务状态	说明
节点扩容中	表示集群正在扩容中。
节点扩容失败	表示集群扩容失败。

查看扩容详情

- 单击确定后，集群任务状态变更为“节点扩容中”，当集群扩容完毕后，集群状态为“服务中”。
- 单击确定后，集群任务状态变更为“节点扩容中”，如果扩容失败，集群状态显示为“服务中”，任务状态显示为“节点扩容失败”。

3.9.1.3 Doris 集群磁盘扩容

磁盘扩容，即纵向扩容，是一种扩容节点内磁盘容量的操作。一般来说，具体的数据存储在BE节点，当BE节点磁盘容量不足时，业务数据将无法存储，因此需要对节点进行纵向扩容。

📖 说明

磁盘扩容过程中，集群业务可能存在抖动。

前提条件

当集群状态处于“服务中”，且没有任何任务操作（如：节点扩容、规格变更、修改参数等）时，可以执行扩容操作。需要注意的是：

- 扩充目标实例的存储容量，单位：GB。
- FE节点的最大扩充后的磁盘容量是2000G。
- BE节点的最大扩充后的磁盘容量是10000G。
- 扩充后的磁盘容量都是100G的倍数。

操作步骤

步骤1 登录CloudTable管理控制台。

步骤2 单击集群管理，显示当前集群列表。

步骤3 在集群列表中，选定集群的操作列，选择“磁盘扩容”，进入扩容页面。如果在英文界面，选定集群的操作列，选择“更多 > 磁盘扩容”，进入扩容页面。

📖 说明

磁盘操作也可以在集群详情页面单击“磁盘扩容”，跳转磁盘扩容页面。

步骤4 选择变配节点为FE/BE。

步骤5 按“+”确定扩容磁盘后的磁盘容量。

步骤6 确定费用以及资源配额后，单击确定。

----结束

磁盘扩容状态

表 3-51 磁盘扩容任务状态说明

状态	说明
磁盘扩容中（任务状态）	表示集群正在扩容中。
服务中（集群状态）	当集群扩容成功并正常提供服务时，其集群状态为服务中。
磁盘扩容失败（任务状态）	表示扩容失败。

查看磁盘扩容信息

单击确定后，**集群任务**状态变更为“磁盘扩容中”，当集群扩容完毕后，**集群状态**更新为“服务中”。

3.9.1.4 Doris 集群规格变更

Doris集群目前不支持计算节点CPU、内存规格升、降配，因此需要对Doris集群增加纵向扩容功能。

使用限制

- 规格变更只支持从小规格变更为大规格，如果要大规格更改为小规格，建议新建小规格集群，采用数据迁移进行切换。
- 一次只支持变更一种类型的节点规格（BE、FE），且变更完成后只生效所选类型的节点规格。
- 如果数据量比较大的情况下，更改节点规格耗时会比较长，因此，建议在业务低峰期扩容节点规格，利于更快完成规格变更。
- 规格变更过程中，整个集群不可用，变更结束前不能再次变更。
- 规格变更需要限制用户操作的频率，短时间内不能多次变更规格。
- 规格变更过程中系统不可用。
- 按需计费形式变更费用指扩容后小时单价。

前提条件

当集群状态处于“服务中”，且没有任何任务操作（如：节点扩容、磁盘扩容、修改参数等）时，可以执行变更操作。

操作步骤

步骤1 登录CloudTable管理控制台。

步骤2 单击集群管理，显示当前集群列表。

步骤3 在集群列表中，选定集群的操作列，选择“更多 > 规格变更”，进入规格变更页面。

说明

规格变更操作也可以在集群详情页面单击“规格变更”，跳转规格变更页面。

步骤4 选择变配节点为FE/BE。

步骤5 按下拉框确定CPU、内存规格。

步骤6 确定费用以及资源配额后，单击确定。

单击确定后，集群状态变更为“亚健康”，任务状态为“规格变更中”，当集群变更完毕后，集群状态更新为“服务中”，任务状态清空。

----结束

规格变更状态

表 3-52 规格变更任务状态说明

状态	说明
规格变更中	表示集群正在变更中。
服务中	当集群变更成功并正常提供服务时，其集群状态为服务中。
规格变更失败	表示集群变更失败。

3.9.2 修改 Doris 参数配置对集群进行性能调优

操作场景

创建完Doris集群后，用户可以在表格存储服务控制台中多次修改集群Doris配置参数，对集群的性能进行调优。

说明

- 修改集群静态参数需要重启集群，可能会导致业务的中断。
- 集群无运行中的任务时，才能修改参数。
- CloudTable Doris集群重启过程中，不允许修改集群参数。

Doris 修改参数步骤

步骤1 登录CloudTable管理控制台。

步骤2 控制台左上角选择区域。

步骤3 在左侧导航栏中，单击“集群管理”。

步骤4 在集群列表中找到所需要的集群，单击集群名称，进入“参数配置”页面。

步骤5 在“参数配置”中单击“参数配置”页签，对Doris参数进行修改。

步骤6 选择修改的节点（FE、BE节点）后，选中需要修改的参数，在“参数值”列单击 。

步骤7 在修改框中输入新值，单击 ，系统提示“参数修改为'xx'成功，请保存本次修改值”，并且修改的参数值后显示红色*。

如果单击 ，则取消修改。

图 3-8 修改参数

max_consumer_num_per_group 4  

步骤8 单击参数列表左上方的“保存修改”，弹出“保存修改”对话框。

步骤9 在“保存修改”对话框中，确认参数修改信息正确后，然后单击“确定”。

图 3-9 保存修改



- 单击确定后，再单击右上角的刷新按钮 ，如果参数表状态显示“已生效”，集群不用重启。如果显示“未生效”，用户需要手动重启集群才能使修改的参数生效。
- 单击参数配置页签左上角的重启按钮，集群重启或者返回集群列表页面重启集群。

说明

静态参数修改完成后，需要手动重启集群，参数才能生效。动态参数修改后生效，不用重启集群。

图 3-10 重启

参数名	参数值	静态参数	参数类型	状态	单位	参数范围	参数默认值	参数描述
max_udfline_load_job_num	150	否	int	已生效	-	(0, 300)	100	该参数限制的用户导入作业的并发数，包括 NEED_SCHEDULED...
use_new_table_scheduler	true	是	boolean	已生效	-	{true/false}	true	是否使用新的副本调度方式
max_breaker_concurrency	10	否	int	未生效	-	(0, 30)	10	breaker scanner 的最大并发数

- 如果修改的参数值不正确，可以在“保存修改”对话框中单击“一键还原”，取消修改参数值。

步骤10 参数修改完成后，可以单击“修改历史”页签查看参数修改的历史。“修改历史”页面中显示如下信息。

表 3-53 参数说明

参数	参数说明
参数名	被修改过的参数名。
旧值	修改前的参数值。
新值	修改后的参数值。
修改时间	记录用户修改参数值的时间。

----结束

Doris 参数修改状态

表 3-54 状态

状态	状态说明
未更改	Doris未更改的参数。
未应用	Doris参数已修改但是未应用。
未生效	Doris静态参数，修改保存后需要重启集群。
已生效	Doris参数修改后已生效： <ul style="list-style-type: none"> 静态参数修改保存后，集群需要重启才能生效。 动态参数修改保存后生效。

FE 节点参数说明

表 3-55 FE 节点参数说明

参数名	参数值	静态参数	参数类型	状态	参数范围	参数默认值	参数描述
max_routine_load_job_num	100	否	int	未更改	(0, 300)	100	该参数限制的例行导入作业的总数，包括 NEED_SCHEDULED, RUNNING, PAUSE这些状态。超过后，不能提交新的作业。
use_new_tablet_scheduler	true	是	boolean	未更改	[true, false]	true	是否启用新的副本调度方式。
max_broker_concurrency	10	否	int	未更改	(5, 30)	10	broker scanner的最大并发数。

参数名	参数值	静态参数	参数类型	状态	参数范围	参数默认值	参数描述
tablet_repair_delay_factor_second	60	否	long	未更改	(0,9223372036854775807)	60	对于不同的调度优先级，我们会延迟不同的时间后开始修复。以防止因为例行重启、升级等过程中，产生大量不必要的副本修复任务。此参数为一个基准系数。对于HIGH优先级，延迟为基准系数 * 1；对于NORMAL优先级，延迟为基准系数 * 2；对于LOW优先级，延迟为基准系数 * 3。即优先级越低，延迟等待时间越长。如果用户想尽快修复副本，可以适当降低该参数。
balance_load_score_threshold	0.1	否	double	未更改	(0,1)	0.1	集群balance百分比的阈值。默认为0.1，即10%。当一个BE节点的load score，不高于或不低于平均load score的10%时，我们认为这个节点是均衡的。如果想让集群负载更加平均，可以适当调低这个参数。
dynamic_partition_check_interval_seconds	600	否	long	未更改	(0,9223372036854775807)	600	动态分区线程的执行频率，默认为600（10分钟），即每10分钟进行一次调度。 单位：s
max_routine_load_task_concurrent_num	5	否	int	未更改	(0,10)	5	该参数限制了一个例行导入作业最大的子任务并发数。建议维持默认值。设置过大，可能导致同时并发的任务数过多，占用集群资源。
dynamic_partition_enable	true	否	boolean	未更改	[true, false]	true	是否开启Doris的动态分区功能。默认为false，即关闭。该参数只影响动态分区表的分区操作，不影响普通表。
query_detail_history_storage_life_time_in_days	7	否	int	未更改	(0,31)	7	该参数表示慢查询记录的保存时间，超过这个时间的记录会被清除。 单位：day

参数名	参数值	静态参数	参数类型	状态	参数范围	参数默认值	参数描述
max_bytes_per_broker_scanner	32 21 22 54 72	否	long	未更改	(0,922 33720 36854 77580 7)	322 122 547 2	broker scanner程序可以在一个broker加载作业中处理的最大字节数。通常，每个BE都有一个broker scanner程序单个BE处理的数据量的最大值，默认为3G。如果单个BE导入的数据量>max_bytes_per_broker_scanner会报错，需要调大参数max_bytes_per_broker_scanner，或者调整导入并发数。具体计算逻辑如下: 本次导入并发数=Math.min(源文件大小/最小处理量、最大并发数、当前BE节点个数) 单位: byte

BE 节点参数说明

表 3-56 BE 节点参数说明

参数名	参数值	静态参数	参数类型	状态	参数范围	参考缺省值	参数描述
max_consumer_num_per_group	3	是	int	未更改	(0, 10)	3	一个数据消费者组中的最大消费者数量，用于routine load。
clone_worker_count	3	是	int	未更改	(0, 10)	3	用于执行克隆任务的线程数，其值影响副本均衡的速度。在磁盘压力不大的情况下，可以通过调整该参数来加快副本均衡。

3.9.3 使用 CES 监控 Doris 集群

3.9.3.1 Doris 集群监控指标说明

功能说明

监控是保持CloudTable服务可靠性、可用性和性能的重要部分，通过监控，用户可以观察CloudTable服务器的运行状态。本章节定义了表格存储服务上报云监控服务的监控指标的命名空间、监控指标列表和维度定义。

命名空间

SYS.CloudTable/Service.CloudTable

FE 节点支持的监控指标

FE节点监控指标如表3-57所示。

表 3-57 FE 节点支持的监控指标

指标名称	显示名称	含义	取值范围	单位	进制	测量对象（维度）	监控周期（原始指标）	命名空间
cmdProcessMem	内存使用率	统计测量对象的内存使用率。	0~100	%	不涉及	CloudTable实例节点	60s	SYS.CloudTable
cmdProcessCPU	CPU使用率	统计测量对象的CPU使用率。	0~100	%	不涉及	CloudTable实例节点	60s	SYS.CloudTable
cmdForUsedStorageRate	已用存储空间比率	统计测量对象所在集群的已用存储空间大小占总配额的比率。	0~100	%	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_image_clean_failed	清理历史元数据镜像文件失败的次数	不应失败，如失败，需人工介入	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_image_clean_success	清理历史元数据镜像文件成功的次数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象(维度)	监控周期(原始指标)	命名空间
doris_fe_image_push_success	将元数据镜像文件推送给其他FE节点的成功次数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_image_write_failed	生成元数据镜像文件失败的次数	不应失败, 如失败, 需人工介入	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_image_write_success	生成元数据镜像文件成功的次数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_max_journal_id	当前FE节点最大元数据日志ID	如果是Master FE, 则是当前写入的最大ID, 如果是非Master FE, 则代表当前回放的元数据日志最大ID; 用于观察多个FE之间的id是否差距过大, 过大则表示元数据同步出现问题。	≥0	不涉及	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_max_tablet_compaction_score	所有BE节点中最大的compaction score值	该值可以观测当前集群最大的compaction score, 以判断是否过高, 如过高则可能出现查询或写入延迟。	≥0	不涉及	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象(维度)	监控周期(原始指标)	命名空间
doris_fe_qps	当前FE每秒查询数量(仅统计查询请求)	QPS	≥0	Count/s	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_query_error	错误查询的累积值	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_query_error_rate	每秒错误查询数	-	≥0	Count/s	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_query_latency_ms_99	查询请求延迟的99分位的查询延迟	-	≥0	ms	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_query_latency_ms_999	查询请求延迟的999分位的查询延迟	-	≥0	ms	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_query_olap_table	查询内部表(OlapTable)的请求个数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_query_total	所有查询请求数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_report_queue_size	BE的各种定期汇报任务在FE端的队列长度	该值反映了汇报任务在Master FE节点上的阻塞程度,数值越大,表示FE处理能力不足。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象(维度)	监控周期(原始指标)	命名空间
doris_fe_request_total	所有通过MySQL端口接收的操作请求(包括查询和其他语句)	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_routine_load_error_rows	集群内所有Routine Load作业的错误行数总和	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_routine_load_receive_bytes	集群内所有Routine Load作业接收的数据量大小	-	≥0	Byte	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_routine_load_rows	集群内所有Routine Load作业接收的数据行数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_rps	当前FE每秒请求数量(包含查询以及其他各类语句)	和 QPS 配合来查看集群处理请求的数量。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_scheduled_tablet_num	Master FE节点正在调度的tablet数量	包括正在修复的副本和正在均衡的副本; 该数值可以反映当前集群,正在迁移的tablet数量; 如果长时间有值,说明集群不稳定。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象(维度)	监控周期(原始指标)	命名空间
doris_fe_tablet_status_count_added	Master FE 节点被调度过的 tablet 数量	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_fe_tablet_status_count_in_scheduled	Master FE 节点被重复调度的 tablet 数量	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_fe_tablet_status_count_not_ready	Master FE 节点未满足调度触发条件的 tablet 数量	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_fe_tablet_status_count_total	Master FE 节点的被检查过的 tablet 数量	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_fe_tablet_status_count_unhealthy	Master FE 节点累积的被检查过的不健康的 tablet 数量	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_fe_txn_counter_begin	提交的事务数量	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_fe_txn_counter_failed	失败的事务数量	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_fe_txn_counter_reject	被拒绝的事务数量	如当前运行事务数大于阈值, 则新的事务会被拒绝。	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象(维度)	监控周期(原始指标)	命名空间
doris_fe_txn_counter_success	成功的事务数量	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_txn_exec_latency_ms_99	99分位的事务执行耗时	-	≥0	ms	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_txn_exec_latency_ms_999	999分位的事务执行耗时	-	≥0	ms	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_txn_publish_latency_ms_99	99分位的事务publish耗时	-	≥0	ms	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_txn_publish_latency_ms_999	999分位的事务publish耗时	-	≥0	ms	不涉及	CloudTable实例节点	60s	SYS.CloudTable
jvm_heap_size_bytes_max	最大堆内存	观测JVM内存使用情况。	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
jvm_heap_size_bytes_committed	已申请的堆内存	观测JVM内存使用情况。	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
jvm_heap_size_bytes_used	已使用的堆内存	观测JVM内存使用情况。	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
jvm_non_heap_size_bytes_committed	已申请的堆外内存	-	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象(维度)	监控周期(原始指标)	命名空间
jvm_non_heap_size_bytes_used	已使用堆外内存	-	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
jvm_old_gc_count	老年代GC次数	观测是否出现长时间的FullGC。 说明 “老年代GC次数”指标仅支持Doris 2.1.6及以下版本。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
jvm_old_gc_time	老年代GC耗时	观测是否出现长时间的FullGC。 说明 “老年代GC耗时”指标仅支持Doris 2.1.6及以下版本。	≥0	ms	不涉及	CloudTable实例节点	60s	SYS.CloudTable
jvm_old_size_bytes_used	老年代内存占用	说明 “老年代内存占用”指标仅支持Doris 2.1.6及以下版本。 -	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
jvm_old_size_bytes_peak_used	老年代内存占用峰值	说明 “老年代内存占用峰值”指标仅支持Doris 2.1.6及以下版本。 -	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
jvm_old_size_bytes_max	老年代内存最大值	说明 “老年代内存最大值”指标仅支持Doris 2.1.6及以下版本。 -	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象(维度)	监控周期(原始指标)	命名空间
jvm_thread_new_count	线程数峰值	观测JVM线程数是否合理。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
jvm_thread_new_count	new状态的线程数	观测JVM线程数是否合理。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
jvm_thread_runnable_count	runnable状态的线程数	观测JVM线程数是否合理。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
jvm_thread_blocked_count	blocked状态的线程数	观测JVM线程数是否合理。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
jvm_thread_waiting_count	waiting状态的线程数	观测JVM线程数是否合理。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
jvm_thread_terminated_count	terminated状态的线程数	观测JVM线程数是否合理。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
jvm_young_gc_count	新生代GC次数	累计值 说明 “新生代GC次数”指标仅支持Doris 2.1.6及以下版本。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
jvm_young_gc_time	新生代GC耗时	累计值 说明 “新生代GC耗时”指标仅支持Doris 2.1.6及以下版本。	≥0	ms	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象(维度)	监控周期(原始指标)	命名空间
jvm_young_size_bytes_used	新生代内存占用	说明 “新生代内存占用”指标仅支持 Doris 2.1.6 及以下版本。 -	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
jvm_young_size_bytes_peak_used	新生代内存占用峰值	说明 “新生代内存占用峰值”指标仅支持 Doris 2.1.6 及以下版本。 -	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
jvm_young_size_bytes_max	新生代内存最大值	说明 “新生代内存最大值”指标仅支持 Doris 2.1.6 及以下版本。 -	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_cache_added_partition	新增的 Partition Cache数量	累计值	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_cache_added_sql	新增的 SQL Cache数量	累计值	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_cache_hit_partition	命中 Partition Cache数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_cache_hit_sql	命中 SQL Cache数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象(维度)	监控周期(原始指标)	命名空间
doris_fe_connection_total	当前FE的MySQL端口连接数	用于监控查询连接数。如果连接数超限,则新的连接将无法接入	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_counter_hit_sql_block_rule	被SQL BLOCK RULE拦截的查询数量	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_edit_log_clean_failed	清理历史元数据日志失败的次数	不应失败,如失败,需人工介入。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_edit_log_clean_success	清理历史元数据日志成功的次数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_edit_log_read	元数据日志读取次数的计数	通过斜率观察元数据读取频率是否正常。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_edit_log_write	元数据日志写入次数的计数	通过斜率观察元数据读取频率是否正常。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_fe_image_push_failed	将元数据镜像文件推送给其他FE节点的失败的次数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable

表 3-58 FE 节点自定义监控指标

指标名称	显示名称	含义	取值范围	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_fe_thrift_rpc_total_{method_name}	doris_fe_thrift_rpc_total_{method_name}	FE thrift接口各个方法接收的RPC请求次数	≥0	CloudTable 实例节点	60s	Service.CloudTable
doris_fe_thrift_rpc_latency_ms_{method_name}	doris_fe_thrift_rpc_latency_ms_{method_name}	FE thrift接口各个方法接收的RPC请求耗时	≥0	CloudTable 实例节点	60s	Service.CloudTable
doris_fe_thread_pool_thrift_server_pool_active_thread_num	doris_fe_thread_pool_thrift_server_pool_active_thread_num	线程池thrift-server-pool正在执行的任务数	≥0	CloudTable 实例节点	60s	Service.CloudTable
doris_fe_thread_pool_thrift_server_pool_active_thread_pct	doris_fe_thread_pool_thrift_server_pool_active_thread_pct	线程池thrift-server-pool正在执行的任务数占最大线程数的百分比	[0, 100]	CloudTable 实例节点	60s	Service.CloudTable
doris_fe_thread_pool_thrift_server_pool_task_in_queue	doris_fe_thread_pool_thrift_server_pool_task_in_queue	线程池thrift-server-pool正在排队等待的任务数	≥0	CloudTable 实例节点	60s	Service.CloudTable
doris_fe_thread_pool_thrift_server_pool_task_rejected	doris_fe_thread_pool_thrift_server_pool_task_rejected	线程池thrift-server-pool拒绝的任务数	≥0	CloudTable 实例节点	60s	Service.CloudTable
doris_fe_thread_pool_mysql_nio_pool_active_thread_num	doris_fe_thread_pool_mysql_nio_pool_active_thread_num	线程池mysql-nio-pool正在执行的任务数	≥0	CloudTable 实例节点	60s	Service.CloudTable

指标名称	显示名称	含义	取值范围	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_fe_thread_pool_mysql_nio_pool_active_thread_pct	doris_fe_thread_pool_mysql_nio_pool_active_thread_pct	线程池mysql-nio-pool正在执行的任务数占最大线程数的百分比	[0, 10]	CloudTable实例节点	60s	Service.CloudTable
doris_fe_thread_pool_mysql_nio_pool_task_in_queue	doris_fe_thread_pool_mysql_nio_pool_task_in_queue	线程池mysql-nio-pool正在排队的任务数	≥0	CloudTable实例节点	60s	Service.CloudTable
doris_fe_thread_pool_mysql_nio_pool_task_rejected	doris_fe_thread_pool_mysql_nio_pool_task_rejected	线程池mysql-nio-pool拒绝的任务数	≥0	CloudTable实例节点	60s	Service.CloudTable
doris_fe_thread_pool_connect_scheduler_pool_active_thread_num	doris_fe_thread_pool_connect_scheduler_pool_active_thread_num	线程池connect-scheduler-pool正在执行的任务数	≥0	CloudTable实例节点	60s	Service.CloudTable
doris_fe_thread_pool_connect_scheduler_pool_active_thread_pct	doris_fe_thread_pool_connect_scheduler_pool_active_thread_pct	线程池connect-scheduler-pool正在执行的任务数占最大线程数的百分比	[0, 100]	CloudTable实例节点	60s	Service.CloudTable
doris_fe_thread_pool_connect_scheduler_pool_task_in_queue	doris_fe_thread_pool_connect_scheduler_pool_task_in_queue	线程池connect-scheduler-pool正在排队的任务数	≥0	CloudTable实例节点	60s	Service.CloudTable
doris_fe_thread_pool_connect_scheduler_pool_task_rejected	doris_fe_thread_pool_connect_scheduler_pool_task_rejected	线程池connect-scheduler-pool拒绝的任务数	≥0	CloudTable实例节点	60s	Service.CloudTable

BE 节点支持的监控指标

BE节点监控指标如表3-59所示。

表 3-59 BE 节点支持的监控指标

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_active_scan_context_count	由外部直接打开的 scanner 的个数	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_add_batch_task_queue_size	接收batch的线程池的队列大小	-	≥0	不涉及	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
cmdForUsedStorageRate	已用存储空间比率	统计测量对象所在集群的已用存储空间大小占总配额的比率。	0~100	%	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_brpc_endpoint_stub_count	已创建的 brpc stub 的数量 (BE)	这些stub用于BE之间的交互	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_brpc_function_endpoint_stub_count	已创建的 brpc stub 的数量 (Remote RPC)	这些stub用于和 Remote RPC之间交互	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_cache_usage_LastestSuccessChannelCache	LRU ChannelCache使用率	LRU DataPage Cache使用率	[0,100]	%	不涉及	CloudTable 实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_cache_usage_ratio_DataPageCache	LRU DataPageCache使用率	-	[0,100]	%	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_cache_usage_ratio_IndexPageCache	LRU IndexPageCache使用率	-	[0,100]	%	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_cache_usage_ratio_SegmentCache	LRU SegmentCache使用率	-	[0,100]	%	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_cache_hit_ratio_DataPageCache	LRU DataPageCache命中率	数据Cache, 直接影响查询效率	[0,100]	%	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_cache_hit_ratio_IndexPageCache	LRU IndexPageCache命中率	索引Cache, 直接影响查询效率	[0,100]	%	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_cache_hit_ratio_LastestSuccessfulChannelCache	LRU ChannelCache命中率	-	[0,100]	%	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_cache_hit_ratio_SegmentCache	LRU SegmentCache命中率	-	[0,100]	%	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_chunk_pool_local_core_alloc_count	ChunkAllocator中,从绑定的core的内存队列中分配内存的次数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_chunk_pool_other_core_alloc_count	ChunkAllocator中,从其他的core的内存队列中分配内存的次数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_chunk_pool_reserved_bytes	ChunkAllocator中预留的内存大小	-	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
doris_be_chunk_pool_system_alloc_cost_ns	SystemAllocator申请内存的耗时	累计值通过斜率可以观测内存分配的耗时	≥0	ns	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_chunk_pool_system_alloc_count	SystemAllocator申请内存的次数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_chunk_pool_system_free_cost_ns	SystemAllocator释放内存的耗时累计值	通过斜率可以观测内存释放的耗时	≥0	ns	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_chunk_pool_system_free_count	SystemAllocator释放内存的次数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_compaction_bytes_total_base	Base Compaction的数据量	累计值	≥0	Byte	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_compaction_bytes_total_cumulative	Cumulative Compaction的数据量	累计值	≥0	Byte	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_compaction_deltas_total_base	Base Compaction处理的rowset个数	累计值	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_compaction_deltas_total_cumulative	Cumulative Compaction处理的rowset个数	累计值	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_compaction_waiting_permits	正在等待Compaction令牌的数量	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_data_stream_receiver_count	数据接收端Receiver的数量	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_fragment_endpoint_count	数据接收端Receiver的数量	同 doris_be_data_stream_receiver_count	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_fragment_request_duration_us	所有fragment instance的执行时间	累计值通过斜率观测 instance 的执行耗时	≥0	us	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_fragment_requests_total	执行过的fragment instance的数量	累计值	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_load_channel_count	当前打开的load channel个数	数值越大, 说明当前正在执行的导入任务越多	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_memory_consumption_tablet_meta	tablet_meta模块的当前总内存开销	-	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
doris_be_memory_consumption_load	load模块的当前总内存开销	-	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
doris_be_memory_allocated_bytes	TcMalloc占用的虚拟内存的大小	-	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
doris_be_memory_pool_bytes_total	所有MemPool当前占用的内存大小	-	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
doris_be_memtable_flush_duration_us	memtable写入磁盘的耗时	累计值通过斜率可以观测写入延迟	≥0	us	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_memtable_flush_total	memtable写入磁盘的个数	累计值通过斜率可以计算写入文件的频率	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_meta_request_duration_read	访问RocksDB中的meta的读取耗时	-	≥0	ms	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_meta_request_duration_write	访问RocksDB中的meta的写入耗时	-	≥0	ms	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_meta_request_total_read	访问 RocksDB 中的 meta 的读取次数	累计值	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_meta_request_total_write	访问 RocksDB 中的 meta 的写入次数	累计值	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_plan_fragment_count	当前已接收的 fragment instance 的数量	观测是否出现 instance 堆积	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_process_fd_num_limit_hard	BE 进程的文件句柄数硬限	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_process_fd_num_limit_soft	BE 进程的文件句柄数软限	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_process_fd_num_used	BE 进程已使用的文件句柄数	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_process_thread_num	BE 进程线程数	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_query_cache_memory_total_byte	Query Cache 占用字节数	-	≥0	Byte	1024(IEC)	CloudTable 实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_query_cache_partition_total_count	当前 Partition Cache缓存个数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_query_cache_sql_total_count	当前SQL Cache缓存个数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_query_scan_bytes	读取数据量的累计值	只统计读取Olap表的数据量	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
doris_be_query_scan_bytes_per_second	读取速率	-	≥0	Byte/s	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable
doris_be_query_scan_rows	读取行数	累计值只统计读取Olap表的数据量, 通过斜率观测查询速率。	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_result_block_queue_count	当前查询结果缓存中的 fragment instance个数	该队列仅用于被外部系统直接读取时使用	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_result_buffer_block_count	当前查询结果缓存中的 query个数	该数值反映当前BE中有多少查询的结果正在等待FE消费	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_routine_load_task_count	当前正在执行的 routine load task 个数	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_rowset_count_generated_and_in_use	自上次启动后,新增的并且正在使用的 rowset id 个数	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_scanner_thread_pool_queue_size	用于 OlapScanner 的线程池的当前排队数量	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_segment_read_segment_read_total	读取的 segment 的个数	累计值	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_segment_read_segment_row_total	读取的 segment 的行数	累计值该数值也包含了被索引过滤的行数	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_send_batch_thread_pool_queue_size	导入时用于发送数据包的线程池的排队个数	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_send_batch_thread_pool_thread_num	导入时用于发送数据包的线程池的线程数	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_small_file_cache_count	当前 BE 缓存的小文件数量	-	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_streaming_load_current_processing	当前正在运行的 stream load 任务数	仅包含 curl 命令发送的任务	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_streaming_load_duration_ms	所有 stream load 任务执行时间的耗时	累计值	≥0	ms	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_streaming_load_requests_total	stream load 任务数	累计值通过斜率可观测任务提交频率。	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_stream_load_pipe_count	当前 stream load 数据管道的个数	包括 stream load 和 routine load 任务	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_stream_load_load_rows	stream load 最终导入的行数	包括 stream load 和 routine load 任务	≥0	Count	不涉及	CloudTable 实例节点	60s	SYS.CloudTable
doris_be_stream_load_receive_bytes	stream load 接收的字节数	包括 stream load 从 http 接收的数据, 以及 routine load 从 kafka 读取的数据。	≥0	Byte	1024(IEC)	CloudTable 实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_tablet_base_max_compaction_score	当前最大的Base Compaction Score	该数值实时变化, 有可能丢失峰值数据; 数值越高, 表示compaction堆积越严重。	≥0	不涉及	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_tablet_cumulative_max_compaction_score	当前最大的Cumulative Compaction Score	-	≥0	不涉及	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_thrift_connections_total_heartbeat	心跳服务的连接数	累计值	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_thrift_connections_total_backend	BE服务的连接数	累计值	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_thrift_current_connections_heartbeat	心跳服务的当前连接数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_thrift_current_connections_backend	BE服务的当前连接数	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_timeout_cancelled_fragment_count	因超时而被取消的fragment instance数量	这个值可能会被重复记录	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_stream_load_txn_request_begin	stream load开始事务数	包括 stream load和 routine load任务	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_stream_load_txn_request_commit	stream load执行成功的事务数	包括 stream load和 routine load任务	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_stream_load_txn_request_rollback	stream load执行失败的事务数	包括 stream load和 routine load任务	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_unused_rowsets_count	当前已废弃的rowset的个数	这些 rowset正常情况下会被定期删除	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_load_bytes	通过tablet sink发送的数量	累计值可观测导入数据量	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_load_rows	通过tablet sink发送的行数	累计值可观测导入数据量	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_fragment_thread_pool_queue_size	当前查询执行线程池等待队列的长度	-	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable
doris_be_compaction_used_permits	Compaction任务已使用的令牌数量	用于反映 Compaction的资源消耗量	≥0	Count	不涉及	CloudTable实例节点	60s	SYS.CloudTable

指标名称	显示名称	含义	取值范围	单位	进制	测量对象 (维度)	监控周期 (原始指标)	命名空间
doris_be_upload_total_byte	rowset数据量累计值	-	≥0	Byte	1024(IEC)	CloudTable实例节点	60s	SYS.CloudTable

表 3-60 BE 节点自定义监控指标

指标名称	显示名称	含义	取值范围	测量对象 (维度)	监控周期 (原始指标)	命名空间
light_work_active_threads	light_work_active_threads	brpc light线程池活跃线程数	≥0	CloudTable实例节点	60s	Service.CloudTable
light_work_pool_queue_size	light_work_pool_queue_size	brpc light线程池队列最大长度, 超过则阻塞提交work	≥0	CloudTable实例节点	60s	Service.CloudTable
fragment_thread_pool_queue_size	fragment_thread_pool_queue_size	当前查询执行线程池等待队列的长度	≥0	CloudTable实例节点	60s	Service.CloudTable
process_thread_num	process_thread_num	BE 进程线程数	≥0	CloudTable实例节点	60s	Service.CloudTable

维度

Key	Value
cluster_id	CloudTable集群ID。
instance_name	CloudTable集群节点名称。

3.9.3.2 设置 Doris 集群告警规则

通过设置 CloudTable Doris的告警规则，用户可以自定义监控目标和通知策略，及时了解Doris运行状态，从而起到预警作用。设置Doris的告警规则包括设置告警规则名称、监控对象、监控指标、告警阈值、监控周期和是否发送通知等参数。本节介绍了设置告警规则的具体方法。

设置 Doris 集群告警规则

步骤1 登录表格存储服务管理控制台。

步骤2 在集群管理列表中的右上角，输入要搜索的集群名称，单击。

步骤3 在集群列表中，找到所需要集群，在对应的“操作”列中，单击“查看监控信息”，系统将跳转到该集群的“云服务监控”页面。

待查看监控信息的集群状态必须为“服务中”。

步骤4 单击“创建告警规则”，进入创建告警规则页面进行添加。

具体参数设置请参考[创建告警规则和通知](#)。

步骤5 配置完成后，单击“立即创建”，完成告警规则的创建。告警规则添加完成后，当监控指标触发设定的阈值时，云监控会在第一时间通过消息通知服务实时告知您资源异常，以免因此造成业务损失。

说明

如果想设置单个指标的告警规则，进入查看监控信息页面：

- 查看监控指标，单击指标图。
- 单击右上角的“+”。
- 进入创建告警规则页面，设置规则。

----结束

设置 Doris 集群自定义监控指标告警规则

步骤1 登录表格存储服务管理控制台。

步骤2 单击“服务列表 > 云监控服务 > 自定义监控 > Service.CloudTable”，进入云监控页面。

步骤3 根据cluster id/instance name/tenantName，查询对应的集群。

步骤4 单击操作列的“创建告警规则”，进入创建告警规则页面，设置参数后，单击“立即创建”。

图 3-11 自定义监控

cluster_id	instance_name	cluster_id	instance_name	tenantName	cluster_id	tenantName
<input type="checkbox"/>	cluster_id	instance_name	cluster_id	instance_name	tenantName	tenantName
<input type="checkbox"/>	cluster_id	instance_name	cluster_id	instance_name	tenantName	tenantName
<input type="checkbox"/>	cluster_id	instance_name	cluster_id	instance_name	tenantName	tenantName
<input type="checkbox"/>	cluster_id	instance_name	cluster_id	instance_name	tenantName	tenantName
<input type="checkbox"/>	cluster_id	instance_name	cluster_id	instance_name	tenantName	tenantName
<input type="checkbox"/>	cluster_id	instance_name	cluster_id	instance_name	tenantName	tenantName

----结束

3.9.3.3 查看 Doris 集群的监控信息

操作场景

云平台提供的云监控服务，可以对Doris集群的运行状态进行日常监控。您可以通过管理控制台，直观的查看Doris的各项监控指标。通过查看集群的监控信息可以快速掌握集群的健康状态并及时获取系统的关键信息。

监控功能

集群监控功能整体分为FE指标监控和BE指标监控两个模块。在每个模块中用户都可调整监控数据的时间范围：方便查看不同时间的历史数据。也可调整监控数据的时间粒度：方便在不同维度层次查看数据。如果想实时查看监控数据，可以开启自动刷新。平台可按1小时、3小时、12小时、24小时、7天的间隔自动刷新监控图表。

如果想看那个指标，可以单击放大按钮查看，也可以导出图片查看。

在 CES 界面查看 Doris 集群监控信息

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 在左侧导航树，单击集群管理。
- 步骤4** 在集群列表中找到需要查看的集群，在对应的操作列中，单击查看监控信息，系统会跳到该集群的“云服务监控”页面。
待查看集群，其状态必须为“服务中”。
- 步骤5** 在集群的“云服务监控”页面中，单击集群ID前的按钮 展开计算单元列表，选择对应节点查看监控信息。
 - **ID**：所监控的实例ID，即CloudTable集群ID。
 - **查看监控指标**：找到所需查看的计算单元，单击“查看监控指标”进入该计算单元的监控指标页面。
- 步骤6** 如果您查看监控页面的指标太多，可以对指标进行设置，设置自己需要查看的指标。
 1. 如果指标太多，可以通过设置监控指标页面对指标进行删除。
 2. 如果页面显示的指标没有您想看的指标，可以在设置监控指标页面进行添加。

3. 选择您在页面中要展示的指标名称，拖动选中指标可以对指标进行排序。

📖 说明

- 如果想实时查看监控数据，可以开启自动刷新。平台可按1小时、3小时、12小时、24小时、7天的间隔自动刷新监控图表。
- 如果想放大单个指标视图，在监控指标视图右上角，单击  可查看监控指标视图详情。

----结束

在集群详情页查看 Doris 集群监控信息

群详情页面主要展示Doris每个节点的CPU使用率、内存使用率、磁盘使用率。

步骤1 登录表格存储服务管理控制台。

步骤2 进入集群管理页面，选择其中一个Doris集群，单击“集群名称 > 详情”

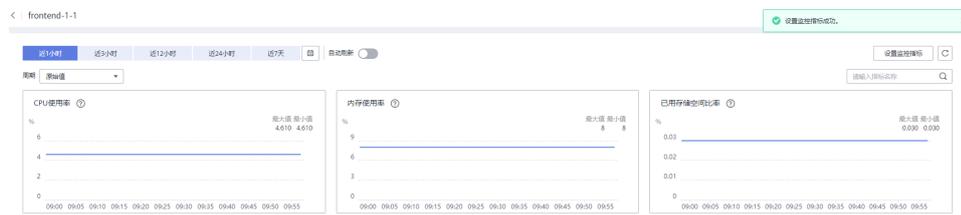
步骤3 进入详情页面，详情页面显示监控页面的指标。

图 3-12 监控页面



步骤4 如要想要查看此节点所有指标信息，请单击详情页面的监控小图标 ，可以跳转CES云服务监控页面查看Doris集群单节点详细监控指标。

图 3-13 监控指标



----结束

3.9.4 Doris 集群日志管理

3.9.4.1 对接 LTS 查看 Doris 集群日志

集群日志服务用于采集集群日志并报送云日志服务（Log Tank Service，以下简称LTS），用户可以在LTS云日志服务查看采集的集群日志或进行日志转储。

📖 说明

Doris集群存放日志路径: /var/log/doris/

- fe.log
- fe.warn.log
- fe.audit.log
- be.INFO
- be.WARNING

开启云日志服务对接步骤

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群，进入集群列表。

步骤4 在集群列表中单击集群名称，进入集群详情页面，左侧导航栏单击“日志服务”，进入日志服务页面。

步骤5 单击页面左上角的 ，开启云日志服务对接。

📖 说明

- 如果首次开启，会弹出创建委托窗口，单击“确定”，进行授权委托。
- 若已开启过LTS云日志服务对接，并授权该服务创建委托，再次开启时则无需授权操作。

----结束

查看集群日志

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群，进入集群列表。

步骤4 在集群列表中单击集群名称，进入集群详情页面，左侧导航栏单击“日志服务”，进入日志服务页面。

步骤5 选择查看日志，单击“操作 > 查看日志”。即可跳转到LTS云日志服务查看日志。

----结束

关闭日志

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群，进入集群列表。

步骤4 在集群列表中单击集群名称，进入集群详情页面，左侧导航栏单击“日志服务”，进入日志服务页面。

步骤5 单击“云日志服务对接”开关，弹出“关闭云日志服务”提示框。

步骤6 确认无误后，单击提示框中的“确认”按钮，即可关闭日志采集。

----结束

3.9.4.2 对接 CTS 查看 Doris 集群审计日志

CloudTable使用云审计服务，可以记录与CloudTable服务相关的操作事件，便于日后的查询、审计和回溯。

CloudTable的以下关键操作事件将记录审计日志，详细内容如表3-61所示。

表 3-61 云审计服务支持的 CloudTable Doris 集群操作事件列表

操作名称	事件名称	资源类型
创建集群	createCloudTableClusterV3	cluster
节点扩容	growCloudTableCluster	cluster
重启集群	rebootCloudTableCluster	cluster
设置存储配额	storageClusterAction	cluster
特性开关	modifyClusterFeatures	cluster
参数配置	modifyClusterSetting	cluster
Doris绑定角色到用户	addAccountRole	cluster
Doris绑定用户到租户	bindAccountWithTenant	cluster
枚举数据库信息	copierListDatabaseInfo	cluster
枚举集群节点信息	copierListNodeInfo	cluster
Doris创建catalog	createCatalogV3	cluster
Doris创建用户	createCloudTableAccount	cluster
创建角色	createRole	cluster
创建租户	createTenante	cluster
Doris创建catalog测试	createTestCatalogV3	cluster
Doris删除Catalog连接	deleteCatalogV3	cluster
删除集群	deleteCloudTableClusterV2	cluster
删除角色	deleteRole	cluster
关闭集群日志	disableLTSAccess	cluster
打开集群日志	enableLTSAccessc	cluster
Doris获取Catalogs	getCatalogs	cluster
获取集群信息	getClusterInfo	cluster

操作名称	事件名称	资源类型
获取数据库信息	getDatabases	cluster
获取角色信息	getRoles	cluster
获取表信息	getTables	cluster
磁盘扩容	growCloudTableDisk	cluster
规格扩容	growCloudTableFlavor	cluster
修改Doris配置参数	modifyClusterSettingV3	cluster
Doris修改角色权限	modifyRolePermission	cluster
重启节点	restartInstance	cluster
Doris修改租户	updateTenant	cluster
重启	REBOOTING	cluster
扩容	GROWING	cluster
删除	DELETING	cluster
包周期集群扩容、规格变更	changeCloudTableCluster	cluster
开启冷热分离	switchHotColdFeature	cluster
创建租户	createTenant	cluster
Doris删除租户	deleteTenant	cluster
查杀sql	killQueryBySqlId	cluster
Doris删除用户	deleteCloudTableAccount	cluster
Doris更新用户	updateCloudTableAccount	cluster

开启云审计服务

使用云审计服务前需要开启云审计服务，开启云审计服务后系统会自动创建一个追踪器，系统记录的所有操作将关联在该追踪器中。目前，一个云账户系统仅支持创建一个追踪器。

步骤1 在CloudTable管理控制台，选择“服务列表 > 管理与部署 > 云审计服务”，进入云审计服务信息页面。

步骤2 开通云审计服务。

如果您是首次使用云审计服务，在追踪器列表中还没有已创建的追踪器，则请参考[《云审计服务快速入门》](#)中的“开通云审计服务”开启云审计服务。

如果您已开通过云审计服务了，开通时系统已为您自动创建了一个管理事件追踪器，管理事件追踪器只能有一个且不可删除。您也可以自行创建数据事件追踪器，详细内容请参见《云审计服务用户指南》中的[创建追踪器](#)。

----结束

关闭审计日志

如果用户想关闭审计日志，需要在云审计服务中停用追踪器。

步骤1 在CloudTable管理控制台，选择“服务列表 > 管理与监管> 云审计服务”，进入云审计服务信息页面。

步骤2 通过停用追踪器，关闭审计日志。如需重新开启审计日志，只要启用追踪器即可。

有关停用/启用追踪器的更多信息，请参考《云审计服务用户指南》中的[启用/停用追踪器](#)。

----结束

查看 CloudTable 的云审计日志

步骤1 登录管理控制台。

步骤2 单击页面上方的“服务列表”，选择“管理与监管 > 云审计服务”，进入云审计服务信息页面。

步骤3 单击左侧导航树的“事件列表”，进入事件列表信息页面。

步骤4 单击事件列表右上方的“筛选”，设置对应的操作事件条件。

当前事件列表支持四个维度的组合查询，详细信息如下：

- “事件来源”、“资源类型”和“筛选类型”。
 - “事件来源”：选择“CloudTable”。
 - “资源类型”：选择“所有资源类型”，或者指定具体的资源类型。
 - “筛选类型”：选择“所有筛选类型”，或者选择以下任一选项。
 - “按事件名称”：选择该选项时，还需选择某个具体的事件名称。
 - “按资源ID”：选择该选项时，还需选择或者手动输入某个具体的资源ID。
 - “按资源名称”：选择该选项时，还需选择或手动输入某个具体的资源名称。
- “操作用户”：在下拉框中选择某一具体的操作用户，此操作用户指用户级别，而非租户级别。
- “事件级别”：可选项为“所有事件级别”、“normal”、“warning”、“incident”，只可选择其中一项。
- “起始时间”、“结束时间”：可通过选择时间段查询操作事件。

步骤5 单击“查询”，查看对应的操作事件。

步骤6 在需要查看的事件左侧，单击  展开该记录的详细信息。

图 3-14 事件



步骤7 在需要查看的事件右侧，单击“查看事件”，弹出一个窗口，显示了该操作事件结构的详细信息。

图 3-15 查看事件

查看事件

```
{
  "service_type": "CloudTable",
  "user": {
    "name": " ",
    "id": "2f210e62179a425c96e5b57f25d1d492",
    "domain": {
      "name": " ",
      "id": "760dfba8429649e7a477445e1033a839"
    }
  },
  "time": "2018/05/29 14:50:16 GMT+08:00",
  "code": 200,
  "resource_type": "cluster",
  "resource_name": "cloudtable ",
  "resource_id": "689ae304-086b-4fa9-984e-16c58e3d75bf",
  "source_ip": " ",
  "trace_name": "rebootCloudTableCluster",
  "trace_type": "ConsoleAction",
  "api_version": "v1.0",
  "record_time": "2018/05/29 14:50:16 GMT+08:00",
  "trace_id": "8b253930-630c-11e8-82ae-286ed488cbe3",
  "trace_status": "normal"
}
```

关于云审计服务事件结构的关键字段详解，请参见《云审计服务用户指南》的[事件结构](#)章节。

----结束

3.10 Doris 常用 SQL 命令

3.10.1 创建数据库

本章节主要介绍Doris创建数据库的SQL基本语法和使用说明。

基本语法

```
CREATE DATABASE [IF NOT EXISTS] db_name
[PROPERTIES ("key"="value", ...)];
```

使用示例

步骤1 使用具有Doris管理权限的用户通过MySQL客户端连接到Doris。

步骤2 执行以下命令创建数据库example_db:

```
create database if not exists example_db;
```

步骤3 执行以下命令查看数据库信息:

```
SHOW DATABASES;
mysql> SHOW DATABASES;
+-----+
| Database      |
+-----+
| example_db    |
| information_schema |
+-----+
2 rows in set (0.00 sec)
```

步骤4 执行以下命令切换到example_db:

```
use example_db;
```

----结束

3.10.2 创建表

本章节主要介绍Doris创建表的SQL基本语法和使用说明。

基本语法

```
CREATE TABLE [IF NOT EXISTS] [database.]table
(
  column_definition_list,
  [index_definition_list]
)
[engine_type]
[keys_type]
[table_comment]
[partition_info]
distribution_desc
[rollup_list]
[properties]
[extra_properties]
```

使用示例

- 创建一个名为**table1**的普通表:

```
CREATE TABLE example_db.table1
(
  k1 TINYINT,
  k2 DECIMAL(10, 2) DEFAULT "10.5",
  k3 CHAR(10) COMMENT "string column",
  k4 INT NOT NULL DEFAULT "1" COMMENT "int column"
)
COMMENT "table comment"
DISTRIBUTED BY HASH(k1) BUCKETS 32;
```

- 创建一个名为**table2**的分区表。使用**event_day**列作为分区列，建立3个分区：
p201706、p201707、p201708，取值为：

- p201706: 范围为[最小值, 2017-07-01)
- p201707: 范围为[2017-07-01, 2017-08-01)
- p201708: 范围为[2017-08-01, 2017-09-01)

每个分区使用**siteid**进行哈希分桶，桶数为10。创建表命令如下：

```
CREATE TABLE table2
(
  event_day DATE,
  siteid INT DEFAULT '10';
```

```
citycode SMALLINT,
username VARCHAR(32) DEFAULT "",
pv BIGINT SUM DEFAULT '0'
)
AGGREGATE KEY(event_day, siteid, citycode, username)
PARTITION BY RANGE(event_day)
(
PARTITION p201706 VALUES LESS THAN ('2017-07-01'),
PARTITION p201707 VALUES LESS THAN ('2017-08-01'),
PARTITION p201708 VALUES LESS THAN ('2017-09-01')
)
DISTRIBUTED BY HASH(siteid) BUCKETS 10
PROPERTIES("replication_num" = "3");
```

📖 说明

- Doris创建表时限制至少指定3副本，以保证高可用。
- 如果是非节点集群，则不指定副本。
- 可以对Table增加上卷表（Rollup）以提高查询性能。
- 表的列的Null属性默认为true，会对查询性能有一定的影响。
- Doris表必须指定分桶列。

- 查看表内容：

- 查看当前数据库中表。

```
SHOW TABLES;
+-----+
| Tables_in_example_db |
+-----+
| table1                |
| table2                |
+-----+
2 rows in set (0.01 sec)
```

- 查看table1表结构。

```
DESC table1;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| siteid | int(11) | Yes  | true | 10      |      |
| citycode | smallint(6) | Yes | true | N/A    |      |
| username | varchar(32) | Yes | true |        |      |
| pv      | bigint(20) | Yes | false | 0      | SUM  |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

- 查看table2表结构。

```
DESC table2;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| event_day | date   | Yes  | true | N/A    |      |
| siteid    | int(11) | Yes  | true | 10     |      |
| citycode  | smallint(6) | Yes | true | N/A    |      |
| username  | varchar(32) | Yes | true |        |      |
| pv        | bigint(20) | Yes | false | 0     | SUM  |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

3.10.3 插入数据

本章节主要介绍Doris插入表数据的SQL基本语法和使用说明。

基本语法

```
INSERT INTO table_name
[ PARTITION (p1, ...) ]
```

```
[ WITH LABEL label]
[ (column [, ...]) ]
[ [ hint [, ...] ] ]
{ VALUES ( { expression | DEFAULT } [, ...] ) [, ...] | query }
```

使用示例

- 创建test表。
CREATE TABLE test
(
c1 TINYINT,
c2 DECIMAL(10, 2) DEFAULT "10.5",
)
COMMENT "table comment"
DISTRIBUTED BY HASH(k1) BUCKETS 32;
- 创建test2。
CREATE TABLE test2 like test;
- 向test表中一次性插入多行数据。
INSERT INTO test VALUES (1, 2), (3, 4);
- 查询test2中是否插入数据。
SELECT * from test2;
- 向test表中导入一个查询语句结果。
INSERT INTO test (c1, c2) SELECT * from test2;

3.10.4 查询数据

本章节主要介绍Doris查询表数据的SQL基本语法和使用说明。

基本语法

```
SELECT
[hint_statement, ...]
[ALL | DISTINCT | DISTINCTROW | ALL EXCEPT ( col_name1 [, col_name2, col_name3, ...] ) ]
select_expr [, select_expr ...]
[FROM table_references
[PARTITION partition_list]
[TABLET tabletid_list]
[TABLESAMPLE sample_value [ROWS | PERCENT]
[REPEATABLE pos_seek]]
[WHERE where_condition]
[GROUP BY [GROUPING SETS | ROLLUP | CUBE] {col_name | expr | position}]
[HAVING where_condition]
[ORDER BY {col_name | expr | position}
[ASC | DESC], ...]
[LIMIT {[offset,] row_count | row_count OFFSET offset}]
[INTO OUTFILE 'file_name']
```

使用示例

- 查询student表中年龄分别为18, 20, 25的学生姓名:
SELECT Name from student where age in (18,20,25);
- 查询student表中年龄最小的三位学生的姓名, 先对age使用order by排序, asc为升序(默认), desc为降序, 然后limit显示3条记录:
SELECT Name from student order by age asc limit 3;

3.10.5 修改表结构

修改表结构时, 针对聚合模型和非聚合模型的修改方式不同; 针对Key列和Value列的修改方式也不同。其中:

- 建表时指定**AGGREGATE KEY**时，为聚合模型；其他场景为非聚合模型。
- 建表语句中的关键字'**unique key**'或'**aggregate key**'或'**duplicate key**'后面的列就是Key列，剩下的就是Value列。

聚合模型示例

聚合列不支持修改聚合类型。

- 在**col1**列后添加**new_col**列（Key列）：

```
ALTER TABLE example_db.my_table ADD COLUMN new_col INT DEFAULT "0" AFTER col1;
```
- 在**col1**后添加**new_col**列（Value列SUM聚合类型）：

```
ALTER TABLE example_db.my_table ADD COLUMN new_col INT SUM DEFAULT "0" AFTER col1;
```
- 修改**col1**列的类型为BIGINT（Key列）：

```
ALTER TABLE example_db.my_table MODIFY COLUMN col1 BIGINT DEFAULT "1";
```
- 修改**col1**列的类型为BIGINT（Value列）：

```
ALTER TABLE example_db.my_table MODIFY COLUMN col1 BIGINT MAX DEFAULT "1";
```
- 删除**col1**列：

```
ALTER TABLE example_db.my_table DROP COLUMN col1;
```

非聚合模型示例

- 在**col1**列后添加**new_col**列（添加Key列）：

```
ALTER TABLE example_db.my_table ADD COLUMN new_col INT KEY DEFAULT "0" AFTER col1;
```
- 在**col1**列后添加**new_col**列（添加Value列）：

```
ALTER TABLE example_db.my_table ADD COLUMN new_col INT DEFAULT "0" AFTER col1;
```
- 修改**col1**列的类型为BIGINT（Key列）：

```
ALTER TABLE example_db.my_table MODIFY COLUMN col1 BIGINT KEY DEFAULT "1";
```
- 修改**col1**列的类型为BIGINT（Value列）：

```
ALTER TABLE example_db.my_table MODIFY COLUMN col1 BIGINT DEFAULT "1";
```
- 删除**col1**列：

```
ALTER TABLE example_db.my_table DROP COLUMN col1;
```

3.10.6 删除表

本章节主要介绍Doris删除表的SQL基本语法和使用说明。

基本语法

```
DROP TABLE [IF EXISTS] [db_name.]table_name [FORCE];
```

使用示例

删除表**my_table**。

```
DROP TABLE IF EXISTS example_db.my_table;
```

4 使用 ClickHouse

4.1 ClickHouse 存算一体表引擎概述

背景介绍

表引擎在ClickHouse中的作用十分关键，不同的表引擎决定了：

- 数据存储和读取的位置。
- 支持哪些查询方式。
- 能否并发式访问数据。
- 能否使用索引。
- 是否可以执行多线程请求。
- 数据复制使用的参数。

其中MergeTree和Distributed是ClickHouse表引擎中最重要，也是最常用的两个引擎，本文将重点进行介绍。

概述

表引擎即表的类型，在云数据库ClickHouse中决定了如何存储和读取数据、是否支持索引、是否支持主备复制等。云数据库ClickHouse支持的表引擎，请参见下表。

说明

MergeTree引擎为单副本，无法保证高可用和数据可靠性，建议只在测试环境中使用。Replicated*MergeTree引擎用于生产环境。

表 4-1 表引擎

系列	描述	表引擎	特点
MergeTree	<ul style="list-style-type: none"> • MergeTree系列引擎适用于高负载任务，支持大数据量的快速写入并进行后续的数据处理，通用程度高且功能强大。 • 该系列引擎的共同特点是支持数据副本、分区、数据采样等特性。 	MergeTree	<ul style="list-style-type: none"> • 基于分区键（partitioning key）的数据分区分块存储。 • 数据索引排序（基于primary key和order by）。 • 支持数据复制（带Replicated前缀的表引擎）。 • 支持数据抽样。 <p>在写入数据时，该系列引擎表会按照分区键将数据分成不同的文件夹，文件夹内每列数据为不同的独立文件，以及创建数据的序列化索引排序记录文件。该结构使得数据读取时能够减少数据检索时的数据量，极大的提高查询效率。</p>
		RelacingMergeTree	用于解决MergeTree表引擎相同主键无法去重的问题，可以删除主键值相同的重复项。
		CollapsingMergeTree	CollapsingMergeTree它通过定义一个sign标记位字段记录数据行的状态。如果sign标记为1，则表示这是一行有效的数据。如果sign标记为-1，则表示这行数据需要被删除。
		VersionedCollapsingMergeTree	在建表语句中新增Version列，用于解决CollapsingMergeTree表引擎乱序写入导致无法正常折叠（删除）的问题。
		SummigMergeTree	用于对主键列进行预先聚合，将所有相同主键的行合并为一行，从而大幅度降低存储空间占用，提升聚合计算性能。
		AggregatingMergeTree	AggregatingMergeTree是预先聚合引擎的一种，用于提升聚合计算的性能。AggregatingMergeTree引擎能够在合并分区时，按照预先定义的条件聚合数据，同时根据预先定义的聚合函数计算数据并通过二进制的格式存入表内。
		GraphiteMergeTree	用于存储Graphite数据并进行汇总，可以减少存储空间，提高Graphite数据的查询效率。

系列	描述	表引擎	特点
Replicated*MergeTree	ClickHouse中的所有MergeTree家族引擎前面加上Replicated就成了支持副本的合并树引擎。	Replicated*MergeTree系列	Replicated系列引擎借助ZooKeeper实现数据的同步，创建Replicated复制表时通过注册到ZooKeeper上的信息实现同一个分片的所有副本数据进行同步。
Distributed	-	Distributed	本身不存储数据，可以在多个服务器上进行分布式查询。

MergeTree

- 建表语法。

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER ClickHouse集群名]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1] [TTL expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2] [TTL expr2],
    ...
    INDEX index_name1 expr1 TYPE type1(...) GRANULARITY value1,
    INDEX index_name2 expr2 TYPE type2(...) GRANULARITY value2
) ENGINE = MergeTree()
ORDER BY expr
[PARTITION BY expr]
[PRIMARY KEY expr]
[SAMPLE BY expr]
[TTL expr [DELETE|TO DISK 'xxx'|TO VOLUME 'xxx'], ...]
[SETTINGS name=value, ...]
```

- 使用示例。

```
CREATE TABLE default.test (name1 DateTime,name2 String,name3 String,name4 String,name5 Date)
ENGINE = MergeTree() PARTITION BY toYYYYMM(name5) ORDER BY (name1, name2) SETTINGS
index_granularity = 8192;
```

示例参数说明：

表 4-2 参数说明

参数	说明
ENGINE = MergeTree()	MergeTree表引擎。
PARTITION BY toYYYYMM(name5)	分区，示例数据将以月份为分区，每个月份一个文件夹。
ORDER BY	排序字段，支持多字段的索引排序，第一个相同的时候按照第二个排序依次类推。
index_granularity = 8192	排序索引的颗粒度，每8192条数据记录一个排序索引值。

📖 说明

如果被查询的数据存在于分区或排序字段中，能极大降低数据查找时间。

ReplacingMergeTree

为了解决MergeTree表引擎相同主键无法去重的问题，云数据库ClickHouse提供了ReplacingMergeTree表引擎，用于删除主键值相同的重复项。

- 建表语句。

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER ClickHouse集群名]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
    ...
) ENGINE = ReplacingMergeTree([ver])
[PARTITION BY expr]
[ORDER BY expr]
[SAMPLE BY expr]
[SETTINGS name=value, ...]
```

CollapsingMergeTree

CollapsingMergeTree表引擎用于消除ReplacingMergeTree表引擎的功能限制。该表引擎要求在建表语句中指定一个标记列Sign，按照Sign的值将行分为两类：Sign=1的行称为状态行，用于新增状态。Sign=-1的行称为取消行，用于删除状态。

- 建表语句。

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER ClickHouse集群名]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
    ...
) ENGINE = CollapsingMergeTree(sign)
[PARTITION BY expr]
[ORDER BY expr]
[SAMPLE BY expr]
[SETTINGS name=value, ...]
```

- 使用示例。

- 示例数据。

例如：我们要计算用户在某个网站上访问了多少页面以及他们在那里的时间。在某个时间点，我们用用户活动的状态写下面的行。

表 4-3 示例数据

UserID	PageViews	Duration	Sign
4324182021466 249494	5	146	1
4324182021466 249494	5	146	-1
4324182021466 249494	6	185	1

- sign— 指定行类型的列名: 1 是一个 “state” 行, -1 是一个 “cancel” 行。
- 建表Test。

```
CREATE TABLE Test(UserID UInt64,PageViews UInt8,Duration UInt8,Sign Int8)ENGINE = CollapsingMergeTree(Sign) ORDER BY UserID;
```

- 插入数据。

- 第一次插入数据。

```
INSERT INTO Test VALUES (4324182021466249494, 5, 146, 1);
```

- 第二次插入数据。

```
INSERT INTO Test VALUES (4324182021466249494, 5, 146, -1),(4324182021466249494, 6, 185, 1);
```

- 查看数据。

```
SELECT * FROM Test;
```

查询结果。

UserID	PageViews	Duration	Sign
4324182021466249494	5	146	-1
4324182021466249494	6	185	1

- 对指定列进行数据聚合。

```
SELECT UserID,sum(PageViews * Sign) AS PageViews,sum(Duration * Sign) AS Duration FROM Test GROUP BY UserID HAVING sum(Sign) > 0;
```

查询结果如下所示。

UserID	PageViews	Duration
4324182021466249494	6	185

- 强制折叠数据，用以下SQL命令。

```
SELECT * FROM Test FINAL;
```

查询结果如下所示。

UserID	PageViews	Duration	Sign
4324182021466249494	6	185	1

VersionedCollapsingMergeTree

为了解决CollapsingMergeTree表引擎乱序写入导致无法正常折叠（删除）问题，云数据库ClickHouse提供了VersionedCollapsingMergeTree表引擎，在建表语句中新增一列Version，用于在乱序情况下记录状态行与取消行的对应关系。后台Compaction时会将主键相同、Version相同、Sign相反的行折叠（删除）。

- 建表语句。

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER ClickHouse集群名]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
    ...
) ENGINE = VersionedCollapsingMergeTree(sign, version)
[PARTITION BY expr]
[ORDER BY expr]
[SAMPLE BY expr]
[SETTINGS name=value, ...]
```

- 使用示例。

- 示例数据。

例如：我们要计算用户在某个网站上访问了多少页面以及他们在那里的时间。在某个时间点，我们用用户活动的状态写下面的行。

表 4-4 示例数据

UserID	PageViews	Duration	Sign	Version
4324182021 466249494	5	146	1	1
4324182021 466249494	5	146	-1	1
4324182021 466249494	6	185	1	2

- sign— 指定行类型的列名: 1 是一个 “state” 行, -1 是一个 “cancel” 行。
- version— 指定对象状态版本的列名。

- 创建表T。

```
CREATE TABLE T(UserID UInt64,PageViews UInt8,Duration UInt8,Sign Int8,Version UInt8)ENGINE = VersionedCollapsingMergeTree(Sign, Version)ORDER BY UserID;
```

- 插入两部分不同的数据。

```
INSERT INTO T VALUES (4324182021466249494, 5, 146, 1, 1);
INSERT INTO T VALUES (4324182021466249494, 5, 146, -1, 1),(4324182021466249494, 6, 185, 1, 2);
```

- 查看数据。

```
SELECT * FROM T;
```

- 对指定列进行数据聚合。

```
SELECT UserID, sum(PageViews * Sign) AS PageViews,sum(Duration * Sign) AS Duration,Version
FROM T GROUP BY UserID, Version HAVING sum(Sign) > 0;
```

查询显示结果如下。

UserID	PageViews	Duration	Version
4324182021466249494	6	185	2

- 强制折叠数据, 用以下SQL命令。

```
SELECT * FROM T FINAL;
```

查询显示结果如下。

UserID	PageViews	Duration	Sign	Version
4324182021466249494	6	185	1	2

SummingMergeTree

SummingMergeTree表引擎用于对主键列进行预先聚合, 将所有相同主键的行合并为一行, 从而大幅度降低存储空间占用, 提升聚合计算性能。

- 建表语句。

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER ClickHouse集群名]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
    ...
) ENGINE = SummingMergeTree([columns])
[PARTITION BY expr]
[ORDER BY expr]
```

```
[SAMPLE BY expr]
[SETTINGS name=value, ...]
```

- 使用示例。

- 创建一个SummingMergeTree表testTable。

```
CREATE TABLE testTable(id UInt32,value UInt32)ENGINE = SummingMergeTree() ORDER BY id;
```

- testTable表中插入数据。

```
INSERT INTO testTable Values(5,9),(5,3),(4,6),(1,2),(2,5),(1,4),(3,8);
```

```
INSERT INTO testTable Values(88,5),(5,5),(3,7),(3,5),(1,6),(2,6),(4,7),(4,6),(43,5),(5,9),(3,6);
```

- 在未合并parts查询所有数据。

```
SELECT * FROM testTable;
```

查询结果。

id	value
1	6
2	5
3	8
4	6
5	12

id	value
1	6
2	6
3	18
4	13
5	14
43	5
88	5

- ClickHouse还没有汇总所有行，需要通过ID进行汇总聚合，需要用到sum和GROUP BY子句。

```
SELECT id, sum(value) FROM testTable GROUP BY id;
```

查询结果。

id	sum(value)
4	19
3	26
88	5
2	11
5	26
1	12
43	5

- 手工执行合并操作。

```
OPTIMIZE TABLE testTable;
```

查询表数据。

```
SELECT * FROM testTable;
```

查询结果。

id	value
1	12
2	11
3	26
4	19
5	26
43	5
88	5

说明

- SummingMergeTree根据ORDER BY排序键作为聚合数据的条件Key。即如果排序key是相同的，则会合并成一条数据，并对指定的合并字段进行聚合。
- 后台执行合并操作时才会进行数据的预先聚合，而合并操作的执行时机无法预测，所以可能存在部分数据已经被预先聚合、部分数据尚未被聚合的情况。因此，在执行聚合计算时，SQL中仍需要使用GROUP BY子句。

AggregatingMergeTree

AggregatingMergeTree表引擎也是预先聚合引擎的一种，用于提升聚合计算的性能。

- 建表语句。

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name [ON CLUSTER ClickHouse集群名]
(
    name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
    name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
    ...
) ENGINE = AggregatingMergeTree()
[PARTITION BY expr]
[ORDER BY expr]
[SAMPLE BY expr]
[TTL expr]
[SETTINGS name=value, ...]
```

- 使用示例。

AggregatingMergeTree无单独参数设置，在分区合并时，在每个数据分区内，会按照ORDER BY聚合，使用何种聚合函数，对哪些列字段计算，则是通过定义AggregateFunction函数类型实现。

- 建表。

```
create table test_table (name1 String,name2 String,name3
AggregateFunction(uniq,String),name4 AggregateFunction(sum,Int),name5 DateTime) ENGINE
= AggregatingMergeTree() PARTITION BY toYYYYMM(name5) ORDER BY (name1,name2)
PRIMARY KEY name1;
```

AggregateFunction类型的数据在写入和查询时需要分别调用*state、*merge函数，*表示定义字段类型时使用的聚合函数。如上示例表test_table定义的名称3、名称4字段分别使用了uniq、sum函数，那么在写入数据时需要调用uniqState、sumState函数，并使用INSERT SELECT语法。

- 插入数据。

```
insert into test_table select '8','test1',uniqState('name1'),sumState(toInt32(100)),'2021-04-30
17:18:00';
insert into test_table select '8','test1',uniqState('name1'),sumState(toInt32(200)),'2021-04-30
17:18:00';
```

- 查询数据。

```
select name1,name2,uniqMerge(name3),sumMerge(name4) from test_table group by
name1,name2;
```

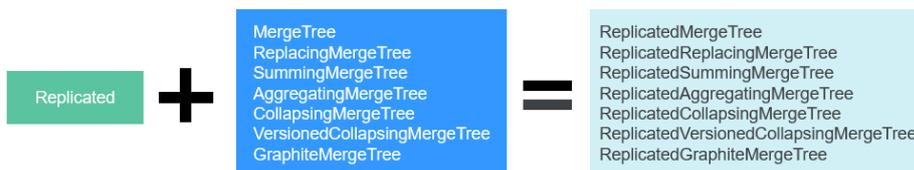
查询结果。

name1	name2	uniqMerge(name3)	sumMerge(name4)
8	test1	1	300

Replicated*MergeTree 引擎

ClickHouse中的所有MergeTree家族引擎前面加上Replicated就成了支持副本的合并树引擎。

图 4-1 合并树引擎图



- Replicated表引擎的创建模板：
ENGINE = Replicated*MergeTree('ZooKeeper存储路径','副本名称', ...)

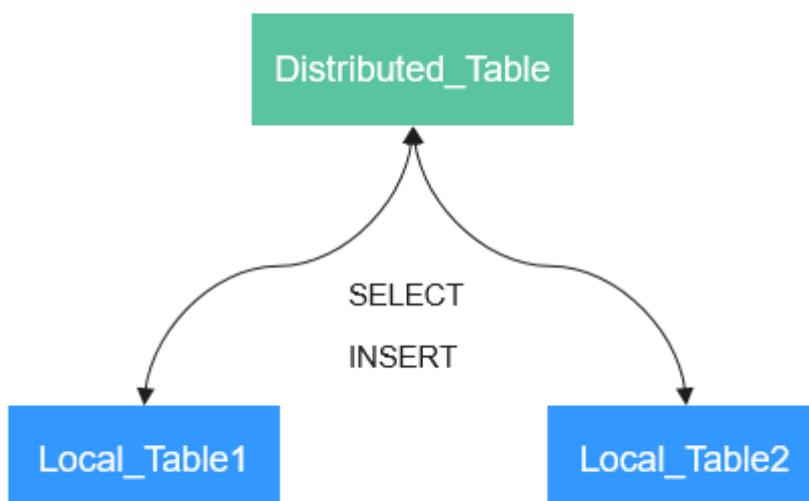
表 4-5 参数表

参数	说明
ZooKeeper存储路径	ZooKeeper中该表相关数据的存储路径，建议规范化，如： /clickhouse/tables/{shard}/数据库名/表名。
副本名称	一般用{replica}即可。

Distributed 表引擎

Distributed表引擎本身不存储任何数据，而是作为数据分片的透明代理，能够自动路由数据到集群中的各个节点，分布式表需要和其他本地数据表一起协同工作。分布式表会将接收到的读写任务分发到各个本地表，而实际上数据的存储在各个节点的本地表中。

图 4-2 Distributed



- Distributed表引擎创建模板：
ENGINE = Distributed(cluster_name, database_name, table_name, [sharding_key])

表 4-6 Distributed 表参数说明

参数	说明
cluster_name	集群名称，在对分布式表执行读写的过程中，使用集群的配置信息查找对应的ClickHouse实例节点。
database_name	数据库名称。
table_name	数据库下对应的本地表名称，用于将分布式表映射到本地表上。
sharding_key	分片键（可选参数），分布式表会按照这个规则，将数据分发到各个本地表中。

- 使用示例。
 - 先创建一个表名为demo的ReplicatedMergeTree本地表。
CREATE TABLE default.demo ON CLUSTER default_cluster(`EventDate` DateTime, `id` UInt64)ENGINE = ReplicatedMergeTree('/clickhouse/tables/{shard}/default/demo', '{replica}') PARTITION BY toYYYYMM(EventDate) ORDER BY id;
 - 基于本地表demo创建表名为demo_all的Distributed表。
CREATE TABLE default.demo_all ON CLUSTER default_cluster(`EventDate` DateTime, `id` UInt64)ENGINE = Distributed(default_cluster, default, demo, rand());
- 分布式表创建规则。
 - 创建Distributed表时需加上on cluster cluster_name，这样建表语句在某一个ClickHouse实例上执行一次即可分发到集群中所有实例上执行。
 - 分布式表通常以本地表加“_all”命名。它与本地表形成一对多的映射关系，之后可以通过分布式表代理操作多张本地表。
 - 分布式表的表结构尽量和本地表的结构一致。如果不一致，在建表时不会报错，但在查询或者插入时可能会抛出异常。

4.2 ClickHouse 使用流程

ClickHouse是面向联机分析处理的列式数据库，其独立于Hadoop大数据体系，最核心的特点是压缩率和极速查询性能。同时支持SQL查询，且查询性能好，特别是基于大宽表的聚合分析查询性能非常优异，比其他分析型数据库速度快一个数量级。

CloudTable ClickHouse集群管理还有以下功能：

ClickHouse集群管理下，用户申请专属集群，按需使用，专属集群是物理隔离且不受其他业务影响，方便用户管理。

ClickHouse集群创建成功后，使用客户端访问集群，请参见以下章节。

- [使用客户端访问集群](#)。
- [ClickHouse入门使用](#)。

图 4-3 ClickHouse 使用流程

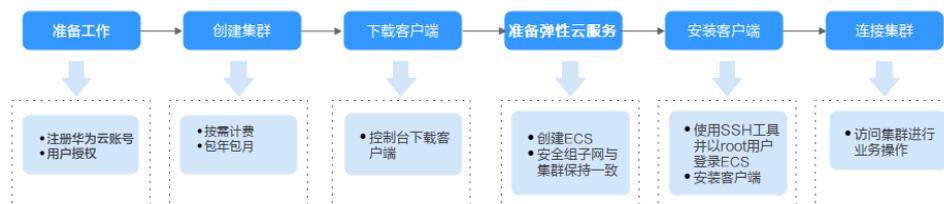


表 4-7 ClickHouse 使用流程

流程	子任务	说明	详细指导
准备工作	创建用户并授权使用 CloudTable	<ul style="list-style-type: none"> 使用 CloudTable ClickHouse 之前，您需要注册华为云账号并进行实名认证，给新账号授权。 将策略授予用户组，并将用户添加至用户组中从而使用户拥有对应的服务权限。 	创建用户并授权使用 CloudTable
创建集群	创建 ClickHouse 集群	使用 ClickHouse 执行任务前，需要创建一个 ClickHouse 集群。	创建 ClickHouse 集群
下载客户端	下载 ClickHouse 客户端	创建集群后，需要下载客户端并安装客户端，安装客户端后，用户才可以借助 SSH 工具连接集群。	ClickHouse 手动安装客户端
准备弹性云服务	-	<p>如果客户端工具的运行环境为 Linux 环境，您需要准备一台和 ClickHouse 集群在相同虚拟私有云的 Linux 弹性云服务器作为客户端主机。</p> <p>如果客户端工具的运行环境为 Windows 环境，您需要准备一台和 ClickHouse 集群在相同虚拟私有云的 Windows 弹性云服务器作为客户端主机。</p>	准备弹性云服务器
安装客户端	-	将下载的客户端放入 ECS，解压安装客户端。	4.4.3-使用客户端连接 ClickHouse 普通集群
连接集群	-	在 ECS 安装好 MySQL 客户端后，可用命令连接集群并进行业务操作。	ClickHouse 手动安装客户端

4.3 创建 ClickHouse 集群

通过表格存储服务可以集中管理集群，如果需要使用CloudTable服务，首要操作就是创建集群，本章节为您介绍如何在表格存储服务管理控制台创建一个新的ClickHouse集群。

ClickHouse集群支持的计费模式有按需付费、包年包月付费。创建集群页面默认采用按需付费方式，即计算单元按实际使用时长计费，计费周期为1小时。这种购买方式比较灵活，可以即开即停，按实际使用时长计费。包年包月可通过预付费，以包年包月模式购买云服务，这种购买方式相对于按需付费提供更大的折扣，对于长期使用者，推荐该计费模式。您可根据业务需要，定制相应计算能力和存储空间的ClickHouse集群。

前提条件

- 创建集群的虚拟私有云、安全组需要和公网ECS保持一致，否则客户端无法访问集群。
- 用户创建集群前，必须配置所使用主机的入方向安全组规则，请参考[安全组规则配置](#)章节，安全组端口请参考[ClickHouse安全规则组表](#)。
- 用户创建集群前安全组规则必须添加ICMP协议，方便管理面通过ping方式查看各节点状态，请参考[安全组规则配置](#)章节。

步骤

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 单击右上角的购买集群按钮，进入“购买集群”页面。
- 步骤4** 参考下面表格配置集群的基本信息。

表 4-8 区域

参数	参数说明
区域	集群的工作区域。 <ul style="list-style-type: none"> • 选择集群节点实际工作区域。 • 有关区域的详细信息，请参见地区和终端节点。
可用区域	选择集群工作区域下关联的可用区。
计费模式	计费模式有按需付费和包年/包月。
订购周期	当计费模式选择包年/包月时，才会出现此选项。客户根据需求选择订购时间。
自动续费	在购买页面勾选此选项，集群到期后会自动续费，不勾选集群到期需要手动续费。

表 4-9 集群和网络配置

参数	参数说明
名称	<p>集群的名称。</p> <p>集群名称在4位到32位之间，必须以字母开头，可以包含字母、数字或者中划线，不能包含其他的特殊字符，不区分大小写。</p>
虚拟私有云	<p>VPC即虚拟私有云，是通过逻辑方式进行网络隔离，提供安全、隔离的网络环境。</p> <p>默认即可。如果没有虚拟私有云，请单击“查看虚拟私有云”进入虚拟私有云，创建一个新的虚拟私有云。</p>
子网	<p>指定虚拟私有云的一个子网。</p> <p>集群使用子网实现与其他网络的隔离，并独享所有网络资源，以提高网络安全。</p>
安全组	<p>安全组用来实现安全组内和安全组间弹性云服务器的访问控制，加强弹性云服务器的安全保护。用户可以在安全组中定义各种访问规则，这些规则可以限定弹性云服务器的哪些端口或者协议可以访问，这些访问规则可以作用在进出虚拟机的网络访问上。当弹性云服务器加入该安全组后，即受到这些访问规则的保护。不属于该安全组的弹性云服务器，是无法同安全组内的弹性云服务器通信的。</p> <p>您可以选择使用已有的安全组，或者单击“查看安全组”创建新的安全组。</p> <p>关于安全组的更多信息，请参见《虚拟私有云用户指南》中的安全组。</p>
数据引擎	选择需要创建的数据组件类型。
集群存储模式	集群存储模式为存算一体。
Clickhouse内核版本	默认即可。
集群高可用	<p>开启集群高可用：在单个分片提供2个副本，默认3个Zookeeper节点。</p> <p>关闭集群高可用：提供单节点clickhouse，仅用于测试，单节点clickhouse提供一个副本、一个分片，不支持节点扩容，该模式不可用于生产环境。</p> <p>说明</p> <ul style="list-style-type: none"> 单节点不支持扩容，不支持数据迁移。 单节点用于非生产环境。 单节点用户需要建立非Replicate表。 <p>建表示例：</p> <pre>CREATE TABLE test_p (`EventDate` DateTime,`CounterID` UInt32,`UserID` UInt32,`ver` UInt16)ENGINE = ReplacingMergeTree()PARTITION BY sipHash64(CounterID) ORDER BY (CounterID, EventDate, intHash32(UserID))SAMPLE BY intHash32(UserID)SETTINGS index_granularity = 8192;</pre>

表 4-10 计算节点参数说明

参数	参数说明
计算规格	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明 支持的计算规格：</p> <ul style="list-style-type: none"> • 8U32G。 • 16U64G。 • 32U128G。 • 32U256G。 • 64U256G。 • 128U512G。
存储规格	<p>选Clickhouse计算节点的磁盘规格、容量。</p> <ul style="list-style-type: none"> • 支持的存储规格： <ul style="list-style-type: none"> - 高IO - 超高IO。 - 极速型SSD。 • 容量范围为100~10000GB/节点之间。
节点数量	选择集群中的节点个数，支持的节点范围为2~40之间。

表 4-11 ZooKeeper 节点参数说明

参数	参数说明
计算规格	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明 支持的计算规格：</p> <ul style="list-style-type: none"> • 4U8G（此规格不能用于生产环境）。 • 4U16G。 • 8U32G。 • 16U64G。 • 32U128G。 <p>ZooKeeper小规格只包含4U8G、4U16G，生产环境建议选择8U32G及以上规格。</p>
存储规格	默认超高IO数据盘，容量100GB/节点。
节点数量	集群中的节点数量，默认为3。

表 4-12 用户名密码参数

参数	参数说明
用户名	管理用户，用于连接集群，默认为admin，不可修改。
密码	<p>设置admin用户密码。</p> <p>密码复杂度要求如下：</p> <ul style="list-style-type: none"> • 8~16个字符。 • 至少包含以下字符中的4种：大写字母，小写字母、数字、特殊字符（\$@!%*?&）。 • 不能与用户名或倒序用户名相同。 • 强弱密码检查。 <p>说明 请妥善保管并定期更新密码，避免安全风险。</p>
确认密码	设置admin用户密码。
是否开启通道加密	<p>通过https端口和安全tcp端口访问clickhouse，保证客户数据通信加密。</p> <ul style="list-style-type: none"> • 创建集群开启安全通道加密选项，后续无法关闭。 • 创建集群选择ZooKeeper节点未开启安全通道加密选项，后续可在集群详情页面开启。
是否开启冷热分离	<p>打开开关，集群支持冷热分离特性，将数据分类存储在不同介质上，有效降低存储成本。关闭开关，集群不支持冷热分离特性。</p>
企业项目	<p>企业项目管理帮助您将相关的资源（如具有相同使用用途的资源）集中在一起，按企业项目的方式来管理云资源。</p> <p>说明</p> <ul style="list-style-type: none"> • 当企业业务发生变化，原用户不再拥有企业项目的使用权限时，可将这些用户与该企业项目的关联关系删除。 • 可进行单个删除和批量删除。 • 当用户与企业项目的关联关系被删除后，该用户将无法管理该企业项目，如需再次使用，需要重新给该用户关联企业项目。

步骤5 配置好参数，单击“立即购买”。

步骤6 进“规格详情”页面，确认集群规格订单信息，单击“提交”，成功提交集群创建任务。

步骤7 单击“返回集群管理列表”，可以查看到集群创建的状态。

集群创建需要时间，请您耐心等待。所创集群的初始状态为“创建中”，创建成功后状态更新为“服务中”。

步骤8 如果创建的集群计费模式为包年/包月，单击提交，成功提交集群创建任务。

单击“去支付”，跳转到购买页面，确认信息后，选择合适的支付方式，确认付款。

返回控制台，查看集群状态，集群创建需要时间，请您耐心等待。所创集群的初始状态为“创建中”，创建成功后状态更新为“服务中”。

---结束

4.4 连接 ClickHouse 集群

4.4.1 准备弹性云服务器

准备弹性云服务

购买弹性云服务器的操作步骤，请参见弹性云服务器[购买弹性云服务器](#)。

购买弹性云服务器需要满足以下要求：

- 弹性云服务器需要与CloudTable集群具有相同的区域、可用区、虚拟私有云和子网。
关于虚拟私有云的更多信息，请参见[虚拟私有云](#)的“用户指南 > 虚拟私有云和子网”。
- 弹性云服务器需要与CloudTable集群具有相同的安全组。
关于安全组的更多信息，请参见《虚拟私有云用户指南》中的[安全组](#)。

📖 说明

在使用跨VPC通信方式访问CloudTable集群时，需要由网络管理员对集群所在VPC，安全组和子网的访问进行授权。

- 购买弹性云服务器时，“弹性公网IP”需选择“现在购买”，或者在ECS创建完成后再为该ECS绑定一个弹性公网IP。
- 访问Linux弹性云服务器，建议使用SSH密码方式登录Linux弹性云服务器。
具体登录操作步骤请参见《弹性云服务器用户指南》中的[登录Linux弹性云服务器](#)。

4.4.2 使用客户端连接 ClickHouse 普通集群

用户在弹性云服务器里可以通过[手动安装客户端](#)的方法，使用SQL访问集群。

📖 说明

创建集群的虚拟私有云、安全组需要和公网ECS保持一致，否则客户端无法访问集群。

ClickHouse 手动安装客户端

步骤1 准备Linux弹性云服务器，具体操作请见[准备弹性云服务器](#)章节。

步骤2 下载客户端。登录表格存储服务管理控制台，在左侧导航树单击“帮助”，然后在页面右侧单击“客户端下载”和“客户端校验文件”，下载客户端安装包和客户端校验文件。

步骤3 安装客户端。

1. 使用SSH登录工具，通过弹性IP远程登录到Linux弹性云服务器。

具体登录操作步骤请参见《弹性云服务器用户指南》中的“[SSH密码方式登录](#)”。

2. 进入SSH工具的根目录。

```
cd /
```

3. 在根目录中新建一个文件夹。

```
mkdir 文件夹名
```

4. 进入新建的目录中。

```
cd /文件夹名/
```

5. 将客户端放在新建的目录下。

6. 解压客户端。

```
tar -zxf 客户端包名
```

7. 解压“客户端校验文件”，与客户端解压在同一个目录下。

- a. 解压“客户端校验文件”。

```
cd <客户端校验文件存放路径>  
tar -xzf Client_sha256.tar.gz
```

- b. 获取客户端校验码。

```
sha256sum ClickHouse_Client_23.3.tar.gz
```

- c. 查看客户端校验文件中的校验码，并与客户端校验码进行对比。如果对比结果一致，代表客户端未被篡改，如果对比结果不一致，则代表客户端篡改。

```
less ClickHouse_Client_23.3.tar.gz.sha256
```

8. 进入clickhouse文件夹加载so。

```
sh install.sh
```

9. 进入bin目录，加权限。

```
cd bin/
```

加权限。

```
chmod 700 clickhouse
```

步骤4 客户端安装后，执行以下命令连接ClickHouse集群，端口请参见[表4-13](#)。

```
./clickhouse client --host 集群内网地址 --port 9000 --user admin --password password
```

- 集群内网地址：集群详情页面中集群访问地址，这里替换成您自己购买的集群的访问地址。
- password为您购买集群时设置的密码，如果有特殊符号，特殊符号前面需要\进行转义，如果密码整体用单引号括起来，则特殊字符不需要转义。

表 4-13 自定义安全规则组

方向	策略	协议端口/范围	类型	目的/源地址	用途
出方向	允许	全部	IPv4/ IPv6	0.0.0.0/0	出方向放行。
入方向	允许	8123		CloudTable ClickHouse集群 所在的安全组。	ClickHouse的Http协议端口。
	允许	9000		ClickHouse的TCP协议端口。	
	允许	8443	ClickHouse的Https协议端口。		

方向	策略	协议端口/范围	类型	目的/源地址	用途
	允许	9440			ClickHouse的安全TCP安全端口。

----结束

ClickHouse 入门使用

1. 创建数据库。
`create database demo;`
2. 使用数据库。
`use demo;`
3. 查看当前使用数据库。
`select currentDatabase();`
4. 创建数据表。
`create table demo_t(uid Int32,name String,age UInt32,gender String)engine = TinyLog;`
5. 查看表结构。
`desc demo_t;`
6. 插入数据。
`insert into demo_t values(1,'Candy','23','M'),(2,'cici','33','F');`
7. 查看表。
`select * from demo_t;`
8. 查看数据库和表。
 - 查看数据库。
`show databases;`
 - 查看表。
`show tables;`
9. 删除数据库和表。
 - 删除表。
`drop table demo_t;`

📖 说明

- 删除数据表前，需确认此数据表是否应用中，以免引起不必要的麻烦。
- 删除数据表后可在24小时内恢复，超过24小时无法恢复。恢复命令如下：
`set allow_experimental_undrop_table_query = 1;`
`UNDROP TABLE 数据表名;`
- 删除数据库。
`drop database demo;`

4.4.3 使用客户端连接 ClickHouse 安全集群

您可以通过创建集群开启加密通道，实现数据加密传输。此章节介绍如何开通 ClickHouse安全通道。

使用限制

- 关闭https会给企业服务带来风险。
- 创建集群开启https选项，后续无法关闭。

- 创建集群未开启https选项，后续无法开启。
- 在详情页面开启安全通道后，集群会重启。
- 安全通道和非安全通道同时开启后无法关闭。

打开安全通道

步骤1 登录表格存储服务管理控制台。

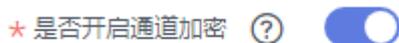
步骤2 控制台左上角选择区域。

步骤3 进入集群管理页面单击右上角的购买集群按钮，进入“购买集群”页面。

步骤4 完成其他配置后，查看安全通道加密按钮是否开启，默认开启。

也可在集群创建成功后在集群详情页打开安全和非安全通道按钮，此时安全通道和非安全通道同时开启。

图 4-4 安全通道



步骤5 配置好参数，单击“立即购买”。

步骤6 进入“规格详情”页面，确认集群规格订单信息，单击“提交”，成功提交集群创建任务，集群创建成功后，进入详情页面查看通道开启状态。

----结束

下载证书

步骤1 集群创建成功后，进入集群详情页下载证书，单击集群信息模块“通道状态”最右侧的“下载证书”。

步骤2 使用SSH登录工具，通过弹性IP远程登录到Linux弹性云服务器。

具体登录操作步骤请参见《弹性云服务器用户指南》中的“[SSH密码方式登录](#)”。

步骤3 配置证书。

自定义证书路径，修改以下配置文件的证书存放路径，保存后将配置文件放在root路径下。

```
<config>
  <secure>true</secure>
  <openSSL>
    <client>
      <caConfig>/etc/ssl/certificate.crt</caConfig>
    </client>
  </openSSL>
</config>
```

- <caConfig>/etc/ssl/certificate.crt</caConfig>指存放证书的路径。
- root指的是配置文件的存放路径。
- 证书下载限流，一分钟只能下载一次。

----结束

使用 ClickHouse 客户端连接集群

步骤1 证书配置成功后，下载客户端。登录表格存储服务管理控制台，在左侧导航树单击“帮助”，然后在页面右侧单击“客户端下载”和“客户端校验文件”，下载客户端安装包和客户端校验文件。

步骤2 安装客户端。

1. 使用SSH登录工具，通过弹性IP远程登录到Linux弹性云服务器。

具体登录操作步骤请参见《弹性云服务器用户指南》中的“[SSH密码方式登录](#)”。

2. 进入SSH工具的根目录。

```
cd /
```

3. 在根目录中新建一个文件夹。

```
mkdir 文件夹名
```

4. 进入新建的目录中。

```
cd /文件夹名/
```

5. 将客户端放在新建的目录下。

6. 解压客户端。

```
tar -zxf 客户端包名
```

7. 解压“客户端校验文件”，与客户端解压在同一个目录下。

a. 解压“客户端校验文件”。

```
cd <客户端校验文件存放路径>  
tar -xzvf Client_sha256.tar.gz
```

b. 获取客户端校验码。

```
sha256sum ClickHouse_Client_23.3.tar.gz
```

c. 查看客户端校验文件中的校验码，并与客户端校验码进行对比。如果对比结果一致，代表客户端未被篡改，如果对比结果不一致，则代表客户端篡改。

```
less ClickHouse_Client_23.3.tar.gz.sha256
```

8. 进入clickhouse文件夹加载so。

```
sh install.sh
```

9. 进入bin目录，加权限。

```
cd bin/
```

加权限。

```
chmod 700 clickhouse
```

步骤3 连接集群。

```
./clickhouse client --host 集群内网地址 --port 9440 --user admin --password password --secure --config-file /root/config.xml
```

----结束

ClickHouse 入门使用

1. 创建数据库。

```
create database demo;
```

2. 使用数据库。

```
use demo;
```

3. 查看当前使用数据库。

```
select currentDatabase();
```

4. 创建数据表。

```
create table demo_t(uid Int32,name String,age UInt32,gender String)engine = TinyLog;
```

5. 查看表结构。
`desc demo_t;`
6. 插入数据。
`insert into demo_t values(1,'Candy','23','M'),(2,'cici','33','F');`
7. 查看表。
`select * from demo_t;`
8. 查看数据库和表。
 - 查看数据库。
`show databases;`
 - 查看表。
`show tables;`
9. 删除数据库和表。
 - 删除表。
`drop table demo_t;`

📖 说明

- 删除数据表前，需确认此数据表是否应用中，以免引起不必要的麻烦。
- 删除数据表后可在24小时内恢复，超过24小时无法恢复。恢复命令如下：
`set allow_experimental_undrop_table_query = 1;`
`UNDROP TABLE 数据表名;`
- 删除数据库。
`drop database demo;`

4.4.4 使用 HTTPS 协议连接 ClickHouse 安全集群

HTTPS是HTTP协议的安全版本，通过在HTTP和TCP之间加入SSL/TLS加密层来保护数据传输安全。您可以通过HTTPS协议连接ClickHouse集群。

使用限制

- 关闭https会给企业服务带来风险。
- 创建集群开启https选项，后续无法关闭。
- 创建集群未开启https选项，后续无法开启。
- 在详情页面开启安全通道后，集群会重启。
- 安全通道和非安全通道同时开启后无法关闭。

打开安全通道

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 进入集群管理页面单击右上角的购买集群按钮，进入“购买集群”页面。
- 步骤4** 完成其他配置后，查看安全通道加密按钮是否开启，默认开启。

也可在集群创建成功后在集群详情页打开安全和非安全通道按钮，此时安全通道和非安全通道同时开启。

图 4-5 安全通道



步骤5 配置好参数，单击“立即购买”。

步骤6 进入“规格详情”页面，确认集群规格订单信息，单击“提交”，成功提交集群创建任务，集群创建成功后，进入详情页面查看通道开启状态。

----结束

下载安全证书并连接 ClickHouse 安全集群

步骤1 ClickHouse安全集群创建成功后，在集群管理页面，单击“集群名称 > 详情”。

步骤2 进入集群详情页下载证书，单击集群信息模块“通道状态”最右侧的“下载证书”。

步骤3 使用SSH登录工具，通过弹性IP远程登录到Linux弹性云服务器。

具体登录操作步骤请参见《弹性云服务器用户指南》中的“[SSH密码方式登录](#)”。

步骤4 自定义证书路径，并修改以下配置文件的证书路径，保存后将配置文件放在root路径下。

- <caConfig>/etc/ssl/certificate.crt</caConfig>指存放证书的路径。
- 证书下载限流，一分钟只能下载一次。

```
<config>
  <secure>true</secure>
  <openSSL>
    <client>
      <caConfig>/etc/ssl/certificate.crt</caConfig>
    </client>
  </openSSL>
</config>
```

步骤5 连接ClickHouse安全集群。

通过curl命令连接集群。

```
echo 'select 1' | curl -H 'X-ClickHouse-User: user' -H 'X-ClickHouse-Key: password' --cacert /clickhouse/client/client/bin/certificate.crt 'https://host:port/?' --data-binary @-
```

表 4-14 参数描述

参数	描述
select 1	执行的sql语句。
user	连接集群的用户名。
password	创集群时创建的密码。
/clickhouse/client/client/bin/certificate.crt	指证书存放路径。
host、port	host指的是内网地址，port指的是https协议端口。

----结束

4.5 配置 ClickHouse 用户权限

ClickHouse用户权限管理实现了对集群中各个ClickHouse节点上用户、角色、权限的统一管理。通过在控制台创建角色、创建用户、用户绑定角色方式，实现用户权限控制。

注意事项

- 删除操作不可逆，即使删除后立即重新添加同名角色，也可能存在权限和以前不一致的情况。所以，删除角色前需确认此角色是否应用中。
- 删除用户前需确认此账户是否再用。删除操作不可逆，即使删除后立即重新添加同名账户，也可能存在权限和以前不一致的情况。
- admin、root和default用户是ClickHouse集群中默认的用户，不能删除。
- 只有“用户管理”中的用户可以修改密码，root和默认用户的密码不可修改。
- 不支持在集群的单节点上创建用户和角色。
- 给角色授予某张表的权限之后，删除表时不会清除已经授予角色的该表的权限，重新创建同名表后角色会继承原有该表的权限。如果需要，可以手动清除已经授予角色的表的权限。
- 给用户A授予表格B的操作权限，当用户A删除数据表B时需要先手动删除授予表格B的操作权限，否则用户C创建的同名表格，用户A可操作同名表，且有表格B同等的操作权限。
- 组件不支持查看已有账户的密码，如果忘记密码可以通过重置密码功能设置新密码。

前提条件

- 已创建ClickHouse集群，集群运行正常。
- 已安装ClickHouse客户端。

步骤一：创建 ClickHouse 角色

步骤1 登录CloudTable控制台。

步骤2 在“集群管理”列表中选择目标集群，单击“集群名称 > 角色管理”，进入角色管理页面。

步骤3 单击“新建角色”，弹出新建角色页面，配置表中相关参数。

表 4-15 角色权限说明

参数	说明
角色名	角色名称以字母开头，长度范围1~64。
全局权限	全局权限指拥有所有数据库和表的权限，包括查询、插入、更改、创建、删除权限。

参数	说明
各级权限	<p>数据库和表，指的是集群中所创建的数据库和表。</p> <ul style="list-style-type: none"> • 数据库有“高危权限”和普通权限；表只有“普通权限”。 • 权限类别 <ul style="list-style-type: none"> - 高危权限：创建表、删除表、创建视图、删除视图。 - 普通权限：查询、插入、更改。

步骤4 参数配置完成后，单击“确认”。

----结束

步骤二：创建 ClickHouse 用户

步骤1 角色创建完成后，单击“用户管理”，进入用户管理页面。

步骤2 单击“新建用户”，弹出“新建用户”页面，配置用户名、密码。

表 4-16 用户参数说明

参数	说明
用户名	新建用户名必须以字母开头，长度范围1~64。
密码	<p>新建用户的密码。</p> <p>说明 设置密码规则：</p> <ul style="list-style-type: none"> • 8~16个字符。 • 至少包含以下字符中的4种：大写字母、小写字母、数字、特殊字符（\$@!%*?&）。 • 不能与用户名或倒序的用户名相同。
确认密码	确认密码。

步骤3 参数配置完成后，单击“确认”。

----结束

步骤三：ClickHouse 用户绑定角色

例如：角色A拥有查询、插入、更改、创建、删除。用户绑定角色A后拥有角色A权限。

步骤1 角色、用户创建完成后，在“用户管理”页面，选择目标用户，单击操作列“更多 > 分配角色”。

步骤2 弹出“分配角色”页面，选择角色。单击“确定”。

步骤3 单击操作列“查看权限”，用户拥有角色权限。

步骤4 使用新建的用户连接集群。

非安全集群连接命令，详细操作请参见[使用客户端连接ClickHouse普通集群](#)。

```
./clickhouse client --host 集群内网地址 --port 9000 --user admin --password password
```

安全集群连接命令，详细操作请参见[使用ClickHouse客户端连接集群](#)。

```
./clickhouse client --host 集群内网地址 --port 9440 --user admin --password password --secure --config-file /root/config.xml
```

步骤5 验证：在命令窗口中执行查询、插入、更改、创建、删除权限。

- 如果可以执行查询、插入、更改、创建、删除权限，说明绑定角色成功。
- 如果无法执行查询、插入、更改、创建、删除权限，请检查角色是否配置权限、用户是否绑定角色。以上两种情况无问题请联系技术人员处理。

---结束

管理用户权限

- 管理角色
 - 删除角色：单击操作列“删除”，弹出删除的页面，输入“DELETE”或者单击“一键输入”，输入后单击“确定”，删除角色。
 - 修改角色权限：单击操作列“修改”，弹出“修改角色”页面，根据用户需求选择权限，单击“确定”。
 - 查看角色权限：单击操作列“查看权限”，弹出“查看权限”页面，查看角色当下拥有使用数据库和表的权限。
- 管理用户
 - 查看用户权限：单击操作列“查看权限”，弹出“查看权限”页面，查看用户当下拥有使用数据库和表的权限。
 - 删除用户：单击操作列“删除”，弹出删除的页面，单击“确定”，删除用户。
 - 修改用户密码：单击操作列“更多 > 修改密码”，弹出“修改密码”页面，修改密码后单击“确定”。

用户权限常用命令

- 存算一体用户权限使用命令
 - 创建角色，role_name：指新建角色名称。
CREATE role IF NOT EXISTS 'role_name' ON CLUSTER default_cluster;
 - 给角色赋予权限。
 - 授予全部创建（库、表）权限。
GRANT CREATE ON *.* TO role1 ON CLUSTER default_cluster;
 - 授予test_db库下创建表权限。
GRANT CREATE TABLE ON test_db.* TO role1;
 - 授予全部删除（库、表）权限。
GRANT DROP ON test_db.* TO role1;
 - 删除角色。
DROP ROLE 'role_name' ON CLUSTER default_cluster;
 - 创建用户。
CREATE USER IF NOT EXISTS name ON CLUSTER default_cluster IDENTIFIED WITH sha256_password BY 'password';

- e. 指定IP。
`CREATE USER name HOST IP '127.0.0.x' IDENTIFIED WITH sha256_password BY 'password';`
- f. 修改用户密码。
`ALTER USER IF EXISTS name ON CLUSTER default_cluster IDENTIFIED WITH sha256_password BY 'password';`
- g. 给用户分配角色。
`GRANT role1, role2 TO new_user ON CLUSTER default_cluster;`
- h. 撤销分配角色。
`REVOKE role1 FROM user ON CLUSTER default_cluster;`
- i. 删除用户。
`DROP USER IF EXISTS 'name1' ON CLUSTER default_cluster;`
- j. 查询用户的权限和角色。
 - 查询用户权限。
`show grants for all;`
 - 查询角色权限。
`show grants for role1;`

4.6 导入导出数据至 ClickHouse 集群

4.6.1 数据导入导出

本章节主要介绍使用ClickHouse客户端导入导出文件数据的基本语法和使用说明。

CSV 格式数据导入导出

- CSV格式数据导入。
 - 非安全集群
`cat csv_ssl | ./clickhouse client --host 192.168.x.x --port port --user admin --password password --database test010 --query="INSERT INTO test145 FORMAT CSV"`
 - 安全集群
`cat csv_no_ssl | ./clickhouse client --host 192.168.x.x --port port --user admin --password password --config-file ./config.xml --database test010 --query="INSERT INTO test146 FORMAT CSV"`
- 1. host: 主机名/ClickHouse实例IP地址。
- 2. port: 端口号（在集群详情页面查看）。
- 3. user: 创建集群时创建的用户名。
- 4. database: 数据库名。
- 5. password: 创建集群时，创建的密码。
- 6. INSERT INTO: 后面跟数据表。
- 7. cat文件路径: 文件存放的路径，路径自定义。
- 8. config-file ./config.xml: 指定配置文件，请参见[使用客户端连接ClickHouse安全集群](#)章节。
- CSV格式数据导出。
 - 非安全集群
`./clickhouse client --host 192.168.x.x --port port --user admin --password Password --database test010 -m --query="select * from test139 FORMAT CSV" > ./csv_no_ssl`
 - 安全集群

```
./clickhouse client --host 192.168.x.x --port port --user admin --password password --config-file ./config.xml --database test010 -m --query="select * from test139 FORMAT CSV" > ./csv_no_ssl
```

1. host: 主机名/ClickHouse实例IP地址。
2. port: 端口号（在集群详情页面查看）。
3. user: 创建集群时创建的用户名。
4. database: 数据库名。
5. password: 创建集群时，创建的密码。
6. SELECT * FROM: 后面跟数据表。
7. ./csv_no_ssl: 指文件存放路径，存放路径自定义。
8. config-file ./config.xml: 指定配置文件，请参见[使用客户端连接ClickHouse安全集群](#)章节。

parquet 格式数据导入导出

- parquet格式数据导入。

- 非安全集群

```
cat parquet_no_ssl.parquet | ./clickhouse client --host 192.168.x.x --port port --user admin --password password --database test010 --query="INSERT INTO test145 FORMAT Parquet"
```

- 安全集群

```
cat parquet_no_ssl.parquet | ./clickhouse client --host 192.168.x.x --port port --user admin --password password --config-file ./config.xml --database test010 --query="INSERT INTO test146 FORMAT Parquet"
```

1. parquet_no_ssl.parquet: 表示格式文件存放路径，路径自定义。
2. host: 主机名/ClickHouse实例IP地址。
3. port: 端口号（在集群详情页面查看）。
4. user: 创建集群时创建的用户名。
5. database: 数据库名。
6. password: 创建集群时，创建的密码。
7. INSERT INTO: 后面跟数据表。
8. config-file ./config.xml: 指定配置文件，请参见[使用客户端连接ClickHouse安全集群](#)章节。

- parquet格式数据导出。

- 非安全集群

```
./clickhouse client --host 192.168.x.x --port port --user admin --password password --database test010 -m --query="select * from test139 FORMAT Parquet" > ./parquet_no_ssl.parquet
```

- 安全集群

```
./clickhouse client --host 192.168.x.x --port port --user admin --password password --config-file ./config.xml --database test010 -m --query="select * from test139 FORMAT Parquet" > ./parquet_ssl.parquet
```

1. host: 主机名/ClickHouse实例IP地址。
2. port: 端口号（在集群详情页面查看）。
3. user: 创建集群时创建的用户名。
4. database: 数据库名。
5. password: 创建集群时，创建的密码。
6. select * from: 后面跟数据表。

7. `./parquet_no_ssl.parquet`: 代表parquet格式文件导出路径，路径自定义。
8. `config-file ./config.xml`: 指定配置文件，请参见[使用客户端连接ClickHouse安全集群](#)章节。

ORC 格式数据导入导出

- ORC格式数据导入。
 - 非安全集群


```
cat orc_no_ssl.orc | ./clickhouse client --host 192.168.x.x --port port --user admin --password password --database test010 --query="INSERT INTO test143 FORMAT ORC"
```
 - 安全集群


```
cat orc_no_ssl.orc | ./clickhouse client --host 192.168.x.x --port port --user admin --password password --config-file ./config.xml --database test010 --query="INSERT INTO test144 FORMAT ORC"
```
- 1. `cat orc_no_ssl.orc`: orc格式文件存放路径，路径自定义。
- 2. `host`: 主机名/ClickHouse实例IP地址。
- 3. `port`: 端口号（在集群详情页面查看）。
- 4. `user`: 创建集群时创建的用户名。
- 5. `database`: 数据库名。
- 6. `password`: 创建集群时，创建的密码。
- 7. `INSERT INTO`: 后面跟数据表。
- 8. `config-file ./config.xml`: 指定配置文件，请参见[使用客户端连接ClickHouse安全集群](#)章节。
- ORC格式数据导出。
 - 安全集群


```
./clickhouse client --host 192.168.x.x --port port --user admin --password password --config-file ./config.xml --database test010 -m --query="select * from test139 FORMAT ORC" > ./orc_ssl.orc
```
 - 非安全集群


```
./clickhouse client --host 192.168.x.x --port port --user admin --password password --database test010 -m --query="select * from test139 FORMAT ORC" > ./orc_no_ssl.orc
```
- 1. `host`: 主机名/ClickHouse实例IP地址。
- 2. `port`: 端口号（在集群详情页面查看）。
- 3. `user`: 创建集群时创建的用户名。
- 4. `database`: 数据库名。
- 5. `password`: 创建集群时，创建的密码。
- 6. `config-file ./config.xml`: 指定配置文件，请参见[使用客户端连接ClickHouse安全集群](#)章节。
- 7. `select * from`: 后面跟数据表。
- 8. `/opt/student.orc`: 导出的ORC格式文件路径，路径自定义。

JSON 格式数据导入导出

- JSON格式数据导入。
 - 非安全集群


```
cat ./jsonnossl.json | ./clickhouse client --host 192.168.x.x --port port --user admin --password password --database test010 --query="INSERT INTO test141 FORMAT JSON"
```
 - 安全集群

```
cat ./jsonssl.json | ./clickhouse client --host 192.168.x.x --port port --user admin --password password --config-file ./config.xml --database test010 --query="INSERT INTO test142 FORMAT JSON"
```

1. cat文件路径：导入文件路径，路径自定义。
 2. host：主机名/ClickHouse实例IP地址。
 3. port：端口号（在集群详情页面查看）。
 4. user：创建集群时创建的用户名。
 5. database：数据库名。
 6. password：创建集群时，创建的密码。
 7. INSERT INTO：后面跟数据表。
 8. config-file ./config.xml：指定配置文件，请参见[使用客户端连接ClickHouse安全集群](#)章节。
- JSON格式数据导出。
 - 安全集群

```
./clickhouse client --host 192.168.x.x --port port --user admin --password password --database test010 -m --query="select * from test139 FORMAT JSON" > ./jsonnossl.json
```
 - 非安全集群

```
./clickhouse client --host 192.168.x.x --port port --user admin --password password --config-file ./config.xml --database test010 -m --query="select * from test139 FORMAT JSON" > ./jsonssl.json
```
1. host：主机名/ClickHouse实例IP地址。
 2. port：端口号（在集群详情页面查看）。
 3. user：创建集群时创建的用户名。
 4. database：数据库名。
 5. password：创建集群时，创建的密码。
 6. SELECT * FROM：后面跟数据表。
 7. ./jsonssl.json：文件导出路径，路径自定义。
 8. config-file ./config.xml：指定配置文件，请参见[使用客户端连接ClickHouse安全集群](#)章节。

4.6.2 ClickHouse 访问 RDS MySQL 服务

ClickHouse面向OLAP场景提供高效的数据分析能力，支持通过MySQL等数据库引擎将远程数据库服务器中的表映射到ClickHouse集群中，后续可以在ClickHouse中进行数据分析。以下操作通过ClickHouse集群和RDS服务下的MySQL数据库实例对接进行举例说明。

前提条件

- 已提前准备好对接的RDS数据库实例及数据库用户名、密码。详细操作可以参考[创建和连接RDS数据库实例](#)。
- 已成功创建ClickHouse集群且集群状态正常。

约束限制

- RDS数据库实例和ClickHouse集群在相同的VPC和子网内。
- 在进行数据同步操作时需要评估对源数据库和目标数据库性能的影响，同时建议您在业务低峰期执行数据同步。

- 当前ClickHouse支持和RDS服务下的MySQL、PostgreSQL实例进行对接，不支持对接SQL Server实例。

ClickHouse 通过 MySQL 引擎对接 RDS 服务

MySQL引擎用于将远程的MySQL服务器中的表映射到ClickHouse中，并允许您对表进行INSERT和SELECT查询，以方便您在ClickHouse与MySQL之间进行数据交换。

- MySQL引擎使用语法：

```
CREATE DATABASE [IF NOT EXISTS] db_name [ON CLUSTER cluster]
ENGINE = MySQL('host:port', ['database' | database], 'user', 'password')
```

表 4-17 MySQL 数据库引擎参数说明

参数	描述
hostport	RDS服务MySQL数据库实例IP地址和端口。
database	RDS服务MySQL数据库名。
user	RDS服务MySQL数据库用户名。
password	RDS服务MySQL数据库用户密码。

MySQL引擎使用示例：

- a. 连接到RDS服务的MySQL数据库。详细操作可以参考[RDS服务MySQL实例连接](#)。
- b. 在MySQL数据库上创建表，并插入数据。
- c. 使用客户端命令连接ClickHouse。

非安全集群连接命令

```
./clickhouse client --host 集群内网地址 --port 9000 --user admin --password password
```

安全集群连接命令，详细操作请参见[使用客户端连接ClickHouse安全集群](#)章节。

```
./clickhouse client --host 集群内网地址 --port 9440 --user admin --password password --secure
--config-file /root/config.xml
```

📖 说明

集群内网地址：集群详情页面中集群访问地址，这里替换成您自己购买的集群的访问地址。

- d. 在ClickHouse中创建MySQL引擎的数据库，创建成功后自动与MySQL服务器交换数据。

```
CREATE DATABASE mysql_db ENGINE = MySQL('RDS服务MySQL数据库实例IP地址:MySQL数据库实例端口', 'MySQL数据库名', 'MySQL数据库用户名', 'MySQL数据库用户名密码');
```

- e. 切换到新建的数据库mysql_db，并查询表数据。

```
USE mysql_db;
```

在ClickHouse中查询MySQL数据库表数据。

```
SELECT * FROM mysql_table;
```

int_id	float
1	2

新增插入数据后也可以正常进行查询。

```
INSERT INTO mysql_table VALUES (3,4);
```

```
SELECT * FROM mysql_table;
```

int_id	float
1	2
3	4

4.7 ClickHouse 企业级能力增强

4.7.1 查看 ClickHouse 慢查询语句

慢SQL页面中包含“运行中的慢查询”和“已完成的慢查询”两个标签页，默认展示当前集群中“运行中的慢查询”耗时500ms以上的慢SQL。对于正在运行中的SQL，提供了“结束查询”功能，您可手动选择是否结束正在运行的SQL。

运行中的慢 SQL 查询操作步骤

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 在集群列表中找到所要查看的集群并单击“集群名称”，进入详情页。
- 步骤3** 在左侧导航树，单击“慢查询管理”，进入运行中慢查询页面。
- 步骤4** 可通过对慢查询运行时长、时间范围、用户、远程IP以及SQL语句的调整来实现数据的筛选以及结束运行中的SQL的命令。

表 4-18 运行中慢 sql 查询参数说明

参数	参数说明
查询编号	查询每一个SQL命令的ID。
执行开始时间	SQL语句执行开始时间。
执行结束时间	SQL语句执行结束时间。
数据库	使用的数据库。
SQL语句	执行的SQL语句。
运行时长 (ms)	SQL执行时长。
状态	SQL运行的状态。
用户	提交SQL的用户。
远程IP	用户提交SQL使用的客户端地址。
操作	进行停止查询等操作。

----结束

已完成的慢 SQL 查询操作步骤

- 步骤1** 登录表格存储服务管理控制台。

- 步骤2** 在集群列表中找到所要查看的集群并单击“集群名称”，进入详情页。
- 步骤3** 在左侧导航树，单击“慢查询管理”，进入已完成的慢查询页面。
- 步骤4** 可通过对慢查询运行时长、时间范围、状态、用户、远程IP以及SQL语句的调整来实现数据的筛选。

表 4-19 已完成慢 sql 查询参数说明

参数	参数说明
查询编号	查询每一个SQL命令的ID。
执行开始时间	SQL语句执行开始时间。
执行结束时间	SQL语句执行结束时间。
数据库	使用的数据库。
SQL语句	执行的SQL语句。
运行时长 (ms)	SQL执行时长。
状态	SQL运行的状态。
用户	提交SQL的用户。
远程IP	用户提交SQL使用的客户端地址。

----结束

慢 SQL 查询状态说明

SQL语句的运行状态为“已完成”、“运行中”、“已取消”中的一种。当运行状态为“运行中”时，对应的操作列中会出现“结束查询”操作。

表 4-20 状态说明

状态	描述
已完成	表示SQL语句已经执行完毕。
运行中	表示SQL语句正在运行中，可手动选择是否结束运行中的SQL语句。
已取消	表示SQL语句已被取消执行。

4.7.2 ClickHouse 数据迁移

本章节为您介绍数据迁移能力。

使用场景

ClickHouse集群节点扩容后，需要使用数据迁移对数据进行迁移。

注意事项

- 存在分区大于50GB的表不支持页面操作迁移，需要手工迁移。
- 迁移表需要非空，且设置唯一分区字段。空表或多字段分区在创建任务时无法选择。
- 冷热分离表涉及热盘数据和冷盘数据迁移，如果对冷数据查询频次不高或者冷数据查询性能要求不高，可以无需迁移。后续新扩容节点的热数据盘使用率会随着业务数据写入，慢慢接近旧节点的热数据盘使用率。如果需要频繁查询冷数据，且对冷数据查询有性能要求，可以迁移冷热分离表，迁移过程中冷数据迁移速率慢。
- 每个数据表只能绑定一个任务。一个集群仅支持一个执行中的任务。
- 本地表的数据库必须为atomic（默认）或ordinary，且表类型为Mergetree家族系列引擎的表（包括非复制表和复制表，不支持物化视图表）。
- 本地表副本关系和cluster一致，有分布式表作为分片之间的关系。
- 数据迁移过程中原表默认为只读状态。
- 数据迁移的时候数据首先会保存在临时表中，执行的时候用迁移的数据表替换原表，该过程中可能读取到错误的数，切换时间为秒级。
- 数据迁移过程中可能由于集群问题导致此过程暂停，根据报错修复集群继续执行任务。
- 数据迁移的时候，源节点和重分布节点必须存在相同表，这样才可以进行数据迁移。
- 单节点不支持数据迁移。

新建数据迁移任务自动迁移

前提：存在分区数据表数据小于50GB。

- 步骤1** 登录表格存储服务管理控制台。
- 步骤2** 控制台左上角选择区域。
- 步骤3** 进入集群管理页面，选择集群进入集群详情页面。
- 步骤4** 单击数据迁移进入数据迁移管理页面。

表 4-21 数据迁移参数说明

参数	说明
任务ID/名称	新建迁移任务ID/名称。
逻辑集群	选取的逻辑集群名称。
源节点	数据存放的节点。
重分布节点	数据分布的节点。
状态/进度	数据分布的状态/进度。 状态：初始化、运行中、已完成。
创建时间	创建数据任务的时间。

参数	说明
执行开始时间	启动数据任务的时间。
更新时间	修改数据任务的时间。
操作	<ul style="list-style-type: none"> ● 启动：启动数据任务。 ● 修改：修改任务信息。 ● 取消：取消迁移任务。 ● 详情：任务详情页面。 ● 删除：删除任务。

步骤5 单击左上角“新建任务”。

1. 用户自定义任务名称（以字母开头）。
2. 选择逻辑集群。
3. 选择迁移百分比。
4. 选择源节点。
5. 选择重分布节点。
6. 选择迁移的数据表。

步骤6 选择后单击确定，完成新建任务。

步骤7 单击操作列“启动”，任务启动。

----结束

手动迁移数据

前提条件：存在分区数据表数据大于50GB。

- 步骤一：创分布式表和本地表
 - a. 使用SSH工具访问目标集群。请参见[ClickHouse手动安装客户端](#)。
 - b. 进入命令窗口创建数据库。

```
CREATE DATABASE IF NOT EXISTS test_fetch on cluster default_cluster;
```
 - c. 使用数据库。

```
USE test_fetch;
```
 - d. 创建本地表。

```
CREATE TABLE IF NOT EXISTS test_fetch.example_table on cluster default_cluster (d DateTime, a Int) ENGINE = ReplicatedMergeTree('/clickhouse/tables/{shard}/test_fetch/example_table', '{replica}') PARTITION BY toYYYYMM(d) ORDER BY d;
```
 - e. 创建分布式表。

```
CREATE TABLE IF NOT EXISTS test_fetch.example_table_dist ON CLUSTER default_cluster (d DateTime, a Int) ENGINE = Distributed('default_cluster', 'test_fetch', 'example_table', rand());
```
 - f. 在分片1分别给分布式表和本地表插入数据。此时各个分片都包含202306 202307 202308的数据，仅分片1包含202309 2023010的数据。

```
INSERT INTO test_fetch.example_table_dist select '2023-06-20',rand64() from numbers(1000);
INSERT INTO test_fetch.example_table_dist select '2023-07-20',rand64() from numbers(1000);
INSERT INTO test_fetch.example_table_dist select '2023-08-20',rand64() from numbers(1000);
INSERT INTO test_fetch.example_table select '2023-09-20',rand64() from numbers(1000);
INSERT INTO test_fetch.example_table select '2023-10-20',rand64() from numbers(1000);
```

- 步骤二：fetch partition进行分片间数据迁移（将数据从分片1迁移到分片2）

- a. 选择迁移的分片和表数据分区。

```
SELECT
    partition AS `分区名称`,
    formatReadableSize(sum(data_uncompressed_bytes)) AS `未压缩大小`,
    formatReadableSize(sum(data_compressed_bytes)) AS `压缩后大小`,
    sum(rows) AS `总行数`
FROM system.parts
WHERE (active = 1) AND (database = 'test_fetch') AND (table = 'example_table')
GROUP BY partition
ORDER BY partition DESC;
```

运行结果展示。

分区名称	未压缩大小	压缩后大小	总行数
202308	3.81 KiB	1.98 KiB	488
202307	4.12 KiB	2.14 KiB	527
202306	3.88 KiB	2.02 KiB	496

- b. 在目标分片的任意节点使用fetch partition获取分区数据，然后attach分区获取源分片中202309的分区并拉取当前分片。

```
ALTER TABLE test_fetch.example_table FETCH PARTITION '202309' FROM '/clickhouse/tables/1/test_fetch/example_table/';
```

'/clickhouse/tables/1/test_fetch/example_table/' 参数中的1代表分片1，可以根据实际调整。

将源分片获取的数据加载到当前分片。

```
ALTER TABLE test_fetch.example_table ATTACH PARTITION '202309';
```

📖 说明

迁移执行时，目标分区节点如果提示detached分区已存在，可以查询detached_parts确认分区是否存在。如果已存在且为冗余数据或脏数据，可以删除。如需保留，可以跳过此分区，迁移其他分区。

```
SELECT * FROM system.detached_parts WHERE table = 'example_table' AND partition_id = '202309';
ALTER TABLE test_fetch.example_table DROP DETACHED PARTITION '202309' SETTINGS allow_drop_detached = 1;
```

- 步骤三：验证结果。

- a. 数据行数、数据内容采样检查。

```
SELECT count() FROM test_fetch.example_table WHERE toYYYYMM(d) = '202309';
SELECT * FROM test_fetch.example_table WHERE toYYYYMM(d) = '202309' order by a limit 10;
```

结果展示：

```
cloudtable-18d8-zp-server-3-1 :) SELECT count() FROM test_fetch.example_table WHERE toYYYYMM(d) = '202309';
SELECT count()
FROM test_fetch.example_table
WHERE toYYYYMM(d) = '202309'
Query id: 12d08040-9797-4dfc-86e7-163b7a82d758
Connecting to cloudtabl-server-3-1-xL80nhhI.cloudtable. .cloud.com:9000.
Connected to ClickHouse server version . revision 54469.
ClickHouse client version is older than ClickHouse server. It may lack support for new features.
count()
┌───┐
│ 2000 │
└───┘
1 row in set. Elapsed: 0.008 sec.
cloudtable-18d8-zp-server-3-1 :) SELECT * FROM test_fetch.example_table WHERE toYYYYMM(d) = '202309' order by a limit 10;
SELECT *
FROM test_fetch.example_table
WHERE toYYYYMM(d) = '202309'
ORDER BY a ASC
LIMIT 10
Query id: c5e7a23a-0cc2-472d-8072-abfca2ae9e83
┌───┬───┐
│ d │ a │
├───┴───┘
│ 2023-09-20 00:00:00 │ -2146912088 │
│ 2023-09-20 00:00:00 │ -2145071775 │
│ 2023-09-20 00:00:00 │ -2139916528 │
│ 2023-09-20 00:00:00 │ -2135846661 │
│ 2023-09-20 00:00:00 │ -2133011096 │
│ 2023-09-20 00:00:00 │ -2130777014 │
│ 2023-09-20 00:00:00 │ -2127945789 │
│ 2023-09-20 00:00:00 │ -2126303638 │
│ 2023-09-20 00:00:00 │ -2125606392 │
│ 2023-09-20 00:00:00 │ -2122100790 │
└───┴───┘
10 rows in set. Elapsed: 0.004 sec. Processed 2.00 thousand rows, 16.00 KB (494.60 thousand rows/s., 3.96 MB/s.)
```

b. 去重数据，删除源分片对应分区。

i. 方法一：连接源节点分片删除数据分区。

```
ALTER TABLE test_fetch.example_table DROP PARTITION '202309';
```

ii. 方法二：detach数据。

```
ALTER TABLE test_fetch.example_table DETACH PARTITION '202309';
```

📖 说明

如果提示分区数据大于50G无法删除，执行设置以下参数，再次操作删除。

```
set max_table_size_to_drop=0;
set max_partition_size_to_drop=0;
```

i. 删除源分片分区后，查询源分片中的分布式表、目标分片中本地表。

```
SELECT count() FROM test_fetch.example_table_dist WHERE toYYYYMM(d) = '202309';
SELECT count() FROM test_fetch.example_table WHERE toYYYYMM(d) = '202309';
```

修改数据迁移任务

步骤1 登录表格存储服务管理控制台。

步骤2 进入集群管理页面，选择集群进入集群详情页面。

步骤3 单击数据迁移进入数据迁移管理页面。

步骤4 单击操作列“修改”，进入修改任务页面。

步骤5 修改参数后，单击确定。

----结束

查看任务详情

步骤1 登录表格存储服务管理控制台。

步骤2 进入集群管理页面，选择集群进入集群详情页面。

步骤3 单击数据迁移进入数据迁移管理页面。

步骤4 单击操作列“修改”，进入任务详情页面。

步骤5 查看任务相关信息。

----结束

删除任务

步骤1 登录表格存储服务管理控制台。

步骤2 进入集群管理页面，选择集群进入集群详情页面。

步骤3 单击数据迁移进入数据迁移管理页面。

步骤4 单击操作列“删除”，弹出删除窗口，单击确定，删除任务。

----结束

4.8 管理 ClickHouse 集群

4.8.1 查看 ClickHouse 集群状态

登录表格存储服务，在左侧导航树单击集群管理，集群列表中会显示所有的集群，集群数量比较多时，可以采用翻页显示，您可以查看任何状态下的集群。

表 4-22 集群管理列表参数

参数	参数说明
集群名称	集群的名称，创建集群时设置。
高可用	创建ClickHouse集群开启“集群高可用”，此处显示“该集群仅能用于测试业务”，如果未开启此处显示“--”。
集群状态	集群当前所处状态，包括创建中、服务中、亚健康、创建失败、冻结。
任务状态	集群任务的状态信息。
引擎类型	ClickHouse
集群版本	集群的版本信息。
企业项目	企业项目管理帮助您将相关的资源（如具有相同使用用途的资源）集中在一起，按企业项目的方式来管理云资源。 说明 <ul style="list-style-type: none"> 当企业业务发生变化，原用户不再拥有企业项目的使用权限时，可将这些用户与该企业项目的关联关系删除。 可进行单个删除和批量删除。 当用户与企业项目的关联关系被删除后，该用户将无法管理该企业项目，如需再次使用，需要重新给该用户关联企业项目。
创建时间	集群创建的时间。
计费模式	集群的计费模式，有按需和转包周期。

参数	参数说明
操作	<ul style="list-style-type: none"> 查看监控信息：单击“查看监控信息”，进入“云服务监控 > 表格存储服务”的监控图表页面。 扩容：扩容集群中的计算规格（规格变更）、存储规格（磁盘扩容）、节点数量（节点扩容）。具体操作请参见ClickHouse 集群容量调整。 开启冷热分离：单击开启冷热分离，开启冷热分离功能。 删除：单击删除，删除一个集群。 重启：单击重启，集群重启。 转包周期：集群的计费模式转为包年/包月。 退订/释放：单击操作列“更多 > 退订/释放”，释放包周期。

表 4-23 按钮说明

按钮	说明
	单击  查看所有项目。
	单击  ，手动刷新集群列表。
	在搜索框中输入集群名称，单击  ，搜索集群。

集群状态

表 4-24 集群状态说明

状态	说明
创建中	表示集群正在创建中。
服务中	当集群创建成功并正常提供服务时，其集群状态为服务中。
亚健康	当超过指定时间监控不到集群状态时，集群状态将显示为“亚健康”。“亚健康”状态的集群需要人工干预让集群恢复正常，比如，可以通过重启集群尝试恢复正常状态。
创建失败	表示集群创建失败。
冻结	<p>续费时所支付的余额已不足，则其状态为“冻结”。</p> <p>当集群状态为“冻结”时，您需要通过续费保证账户余额不为0才能解冻。</p> <p>说明 冻结期间集群不可用且关闭集群中所有云主机，解冻后集群自动恢复为“服务中”状态。如果用户没有续费，冻结期限到期后集群会被删除。</p>

任务状态

表 4-25 任务状态说明

状态	说明
删除中	表示集群正在删除中。
重启中	表示集群正在重启中。
扩容中	表示集群正在扩容中。
开启冷热分离中	表示集群正在开启冷热分离。

4.8.2 查看 ClickHouse 集群详情

集群创建完成后，可对集群进行监控和管理。在CloudTable管理控制台单击集群管理，在集群列表中找到所要查看的集群并单击集群名称，进入集群基本信息页面，用户可查看集群信息和集群的网络配置。

表 4-26 集群信息

参数	说明
集群名称	集群的名称，创建集群时设置。
集群ID	集群的ID。
集群状态	集群状态信息。
集群版本	集群的内核版本。
集群访问地址	集群访问的地址，用户可以单击  ，将访问的链接地址复制到剪切板。
集群访问地址端口	集群访问的地址端口： <ul style="list-style-type: none"> • Http端口：8123。 • TCP端口：9000。 • Https端口：8443。 • 安全TCP端口：9440。
计费模式	显示集群的计费模式。 <ul style="list-style-type: none"> • 按需计费。 • 包年包月。
创建时间	显示集群创建的时间。

参数	说明
企业项目	<p>企业项目管理帮助您将相关的资源（如具有相同使用用途的资源）集中在一起，按企业项目的方式来管理云资源。</p> <p>说明</p> <ul style="list-style-type: none"> 当企业业务发生变化，原用户不再拥有企业项目的使用权限时，可将这些用户与该企业项目的关联关系删除。 可进行单个删除和批量删除。 当用户与企业项目的关联关系被删除后，该用户将无法管理该企业项目，如需再次使用，需要重新给该用户关联企业项目。
数据库引擎	用于存储、处理和保护数据的核心服务。
通道状态	表示安全通道是否打开。
同时开启安全通道和非安全通道	如果安全通道开启，此按钮不显示，如果安全通道未开启，显示此按钮。
是否开启冷热分离	“否”代表未开启冷热分离，“是”代表开启冷热分离。
存储资源名称	存储冷热分离数据资源名称“hot_to_cold”。
冷热分存策略	如果节点总存储量超过设置的值，则优先迁移早期数据至冷存储。
冷热存储已使用容量	冷存储数据的存储容量。
客户端使用	ClickHouse组件目前有新旧两种客户端。如果客户使用新的客户端，客户端使用显示“新”；如果客户端使用旧客户端，客户端使用显示“旧”。
集群存储模式	存算一体

表 4-27 网络配置

参数	说明
区域	集群的工作区域，在创建集群时设置。
可用区	显示创建集群时所选择的可用区信息。
虚拟私有云	<p>VPC信息，创建集群时所选。</p> <p>VPC即虚拟私有云，是通过逻辑方式进行网络隔离，提供安全、隔离的网络环境。</p>
子网	<p>子网信息，创建集群时所选。</p> <p>通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全。</p>
安全组	显示创建集群时所选择的安全组信息。

表 4-28 计算节点参数说明

参数	参数说明
计算规格	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明 支持的计算规格：</p> <ul style="list-style-type: none"> • 8U32G。 • 16U64G。 • 32U128G。 • 32U256G。 • 64U256G。 • 128U512G。
存储规格	<p>选Clickhouse计算节点的磁盘规格、容量。</p> <ul style="list-style-type: none"> • 支持的存储规格： <ul style="list-style-type: none"> - 高IO - 超高IO。 - 极速型SSD。 • 容量范围为100~10000GB/节点之间。
节点数量	选择集群中的节点个数，支持的节点范围为2~40之间。

表 4-29 ZooKeeper 节点参数说明

参数	参数说明
计算规格	<p>请根据业务需求合理选择计算规格。在计算规格列表中展示了每一种计算规格单个节点的CPU、内存和建议使用场景。</p> <p>说明 支持的计算规格：</p> <ul style="list-style-type: none"> • 4U8G（此规格不能用于生产环境）。 • 4U16G。 • 8U32G。 • 16U64G。 • 32U128G。 <p>ZooKeeper小规格只包含4U8G、4U16G，生产环境建议选择8U32G及以上规格。</p>
存储规格	默认超高IO数据盘，容量100GB/节点。
节点数量	集群中的节点数量，默认为3。

4.8.3 重启 ClickHouse 集群节点

当CloudTable集群节点状态异常时，可通过执行重启节点操作尝试将该节点状态恢复正常。

注意事项

- 重启节点过程中，节点不可用。
- 重启后节点请在业务低谷期进行重启，避免对高峰期业务造成影响。
- 节点重启时，磁盘扩容、节点扩容、规格变更按钮置灰，不可操作。
- 节点重启是进程重启，不是整个节点重启。

节点重启操作步骤

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群管理，进入集群列表。

步骤4 选择需要操作的集群，单击集群名称，进入集群详情页。

步骤5 在集群详情页单击“操作 > 重启”，弹出节点重启窗口。

步骤6 输入“RESTART”或者单击“一键输入”，输入后单击“确定”，节点重启。

节点重启后，“重启”按钮置灰。

步骤7 查看节点重启结果，如果重启成功，则“重启”按钮高亮显示。如果节点重启失败，则任务状态显示“节点重启失败”，“重启”按钮高亮显示，节点可继续重启。

----结束

节点重启状态

表 4-30 重启状态

状态	状态描述
重启	重启按钮高亮显示，一种是节点未重启，一种是节点已经重启。
重启中	集群节点正在重启，且在“集群信息 > 任务状态”显示“节点重启中”。
重启失败	集群节点重启失败，且在“集群信息 > 任务状态”显示“节点重启失败”，重启失败的节点可继续重启。

4.8.4 重启 ClickHouse 集群

当集群处于非均衡或不能正常工作时，可能需要通过重启集群进行恢复。

📖 说明

- 如果集群欠费，可能会导致该功能被限制不可用，请及时充值确保集群可正常使用。
- 如果集群状态亚健康，也会限制该功能不可用，请及时联系技术工程师处理使集群可正常使用。

对系统的影响

- 重启期间集群将无法提供服务。因此，在重启前，请确定集群中没有正在运行的任务，并且所有数据都已经保存。
- 如果集群正在处理业务数据，如导入数据、查询数据，一旦重启集群，有可能导致文件损坏或重启失败。因此，建议停止所有集群任务后，再重启集群。
- 如果重启失败，将有可能导致集群不可用，建议联系技术支持人员进行处理或稍后重试。

操作步骤

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 进入集群管理页面，在集群管理列表中的右上角，输入要搜索的集群名称，单击 。

步骤4 选中需要重启的集群名，在对应的**操作列**单击“更多 > 重启”。

步骤5 在弹出的对话框中，勾选了解重启集群的影响后，单击“确定”，重启集群。

图 4-6 确认重启窗口



----结束

4.8.5 删除 ClickHouse 集群

如果不再需要集群时，可以删除集群。删除CloudTable ClickHouse集群将会清空所有与该集群相关的资源及数据，且无法恢复，请谨慎操作。

- 按需集群支持直接删除，详情请参见[删除ClickHouse按需集群](#)。
- 包年/包月集群不能直接删除，需要退订（未超期集群）/释放（已超期未续费的集群），详情请参见[退订/释放包周期集群](#)。

删除 ClickHouse 按需集群

- 步骤1 登录表格存储服务管理控制台。
- 步骤2 控制台左上角选择区域。
- 步骤3 选中需要删除的集群名，在对应的操作列单击“更多 > 删除”。
- 步骤4 在弹出的对话框中，输入“DELETE”或者单击“一键输入”，输入后单击“确定”，删除集群。

图 4-7 确认删除窗口



----结束

退订/释放包周期集群

- 步骤1 登录表格存储服务管理控制台。
- 步骤2 控制台左上角选择区域。
- 步骤3 选中需要删除的集群名，在对应的“操作”列中，单击“更多 > 退订/释放”。
- 步骤4 在弹出的对话框中，输入“DELETE”或者单击“一键输入”，输入后单击“确定”，“退订/释放”集群。
- 步骤5 进入“退订资源”页面，确认待退订集群信息，选择退订原因，并确认退订金额和相关费用。
- 步骤6 勾选“我已确认本次退订的资源已完成数据备份或不再使用，未放入回收站的资源退订后无法恢复。查看回收站说明”，然后单击“退订”。
- 步骤7 返回控制台查看集群已经退订/释放。

----结束

4.8.6 ClickHouse 按需集群转包周期集群

如果您需长期使用当前集群，为节省开支，可以将按需购买的集群转为包年/包月计费模式（也称为包周期计费模式）。按需转包年/包月的具体操作，请参见本章节内容。

按需转包年包月

- 前提条件：集群计费模式是“按需计费”。

- 按需转包年/包月操作步骤
 - a. 登录CloudTable管理控制台。
 - b. 单击集群管理，显示当前集群列表。
 - c. 在集群列表中，选择指定集群所在行的“操作”列，单击“更多>转包周期”。

说明

转包周期操作也可以在集群的详情页面单击“转包周期”，跳转到包年/包月页面。

- d. 跳转到包年/包月页面，选择购买时长，单击“去支付”。
- e. 跳转到支付页面，选择支付方式，单击“确认付款”。
- f. 订单支付成功后，返回集群列表，单击指定集群进入详情页面，查看“计费模式”。

4.9 ClickHouse 集群运维

4.9.1 ClickHouse 集群容量调整

4.9.1.1 ClickHouse 集群节点扩容

ClickHouse节点扩容增加集群中计算节点的数量，计算节点的扩容过程，不影响当前系统运行。

使用限制

- 余额不足账户无法扩容。
- 配额不足无法扩容。
- zookeeper节点不支持节点扩容。

操作步骤

步骤1 登录CloudTable管理控制台。

步骤2 单击集群管理，显示当前集群列表。

步骤3 在集群列表中，选定集群的操作列，单击“更多 > 节点扩容”，进入扩容页面。

说明

节点扩容操作也可以在集群详情页面单击“节点扩容”，跳转节点扩容页面。

步骤4 按“+”增加扩容节点。

步骤5 确定费用以及资源配额后，单击确定。

----结束

ClickHouse 节点扩容状态

表 4-31 节点扩容任务状态说明

任务状态	说明
节点扩容中	表示集群正在扩容中。
节点扩容失败	表示集群扩容失败。

查看扩容详情

- 单击确定后，集群任务状态变更为“节点扩容中”，当集群扩容完毕后，集群状态为“服务中”。
- 单击确定后，集群任务状态变更为“节点扩容中”，如果扩容失败，集群状态显示为“服务中”，任务状态显示为“节点扩容失败”。

4.9.1.2 ClickHouse 集群磁盘扩容

磁盘扩容，即纵向扩容，是一种扩容节点内磁盘容量的操作。一般来说，具体的数据存储的计算节点，当计算节点磁盘容量不足时，业务数据将无法存储，因此需要对节点进行纵向扩容。

说明

- 磁盘扩容过程中，集群业务可能存在抖动。
- 当集群状态处于“服务中”，且没有任何任务操作（如：节点扩容、规格变更等）时，可以执行扩容操作。

磁盘扩容操作步骤

步骤1 登录CloudTable管理控制台。

步骤2 单击集群管理，显示当前集群列表。

步骤3 选择操作集群，单击操作列的“磁盘扩容”，进入扩容页面。

说明

磁盘操作也可以在集群详情页面单击“磁盘扩容”，跳转磁盘扩容页面。

步骤4 选择变配节点。

步骤5 按“+”确定扩容磁盘后的磁盘容量。

步骤6 确定费用以及资源配额后，单击确定。

----结束

磁盘扩容状态

表 4-32 磁盘扩容任务状态说明

状态	说明
磁盘扩容中（任务状态）	表示集群正在扩容中。
服务中（集群状态）	当集群扩容成功并正常提供服务时，其集群状态为服务中。
磁盘扩容失败（任务状态）	表示扩容失败。

4.9.1.3 ClickHouse 集群规格变更

前提条件

当集群状态处于“服务中”，且没有任何任务操作（如：节点扩容、磁盘扩容等）时，可以执行变更操作。

📖 说明

- 规格变更只支持从小规格变更为大规格，如果要大规格更改为小规格，建议新建小规格集群，采用数据迁移进行切换。
- 一次只支持变更一种类型的节点规格（计算节点、ZooKeeper节点），且变更完成后只生效所选类型的节点规格。
- 规格变更过程中，整个集群不可用，变更结束前不能再次变更。
- 规格变更过程中系统不可用。

操作步骤

步骤1 登录CloudTable管理控制台。

步骤2 单击集群管理，显示当前集群列表。

步骤3 在集群列表中，选定集群的操作列，选择“更多 > 规格变更”，进入规格变更页面。

图 4-8 规格变更图

规格变更

1. 规格变更过程中系统不可用
2. 按需计费形式变更费用指本次变更新增费用

集群ID/名称	1523cde8-7[REDACTED]6d704cac92/cl...
地域/可用区	[REDACTED]
计费模式	按需
计算节点规格	4 vCPUs 16 GB
计算节点存储规格	高IO 500 GB
计算节点数量	2
Zookeeper节点规格	4 vCPUs 8 GB
ZooKeeper节点存储规格	超高IO 100 GB
ZooKeeper节点数量	3
变配节点	<input checked="" type="radio"/> 计算节点 <input type="radio"/> ZooKeeper节点
* 计算节点规格	<div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">16U64G</div> 16*CU₄ (1个CU₄ = 1核4GB)
整体费用 (?)	--
变更费用 (?)	--

确定
取消

📖 说明

规格变更操作也可以在集群详情页面单击“规格变更”，跳转规格变更页面。

步骤4 选择变配节点。

步骤5 按下拉框确定CPU、内存规格。

步骤6 确定费用以及资源配额后，单击确定。

单击确定后，集群状态变更为“亚健康”，任务状态为“规格变更中”，当集群变更完毕后，集群状态更新为“服务中”，任务状态清空。

----结束

规格变更状态

表 4-33 规格变更任务状态说明

状态	说明
规格变更中	表示集群正在变更中。
服务中	当集群变更成功并正常提供服务时，其集群状态为服务中。
规格变更失败	表示集群变更失败。

4.9.2 使用 CES 监控 ClickHouse 集群

4.9.2.1 ClickHouse 集群监控指标说明

功能说明

监控是保持CloudTable服务可靠性、可用性和性能的重要部分，通过监控，用户可以观察CloudTable服务器的运行状态。本章节定义了表格存储服务上报云监控服务的监控指标的命名空间、监控指标列表和维度定义。

命名空间

SYS.CloudTable

ClickHouse 监控指标

表 4-34 监控指标

指标名称	显示名	含义	取值范围	指标单位	进制	测量对象（维度）	监控周期
cmdProcessCPU	CPU使用率	统计测量对象的CPU使用率。	0~100	%	不涉及	Cloud Table实例节点	60s
memory_usage	内存使用量	每个节点的内存使用量。	[0, 节点内存容量]	MiB	1024(IEC)	Cloud Table实例节点	60s
cmdProcessMem	内存使用率	统计测量对象的内存使用率。	0~100	%	不涉及	Cloud Table实例节点	60s

指标名称	显示名	含义	取值范围	指标单位	进制	测量对象 (维度)	监控周期
disk_usage	磁盘使用量	每个节点的磁盘使用量。	[0, 节点磁盘容量]	MiB	1024(IEC)	Cloud Table 实例节点	60s
cmdForUsedStorageRate	已用存储空间比率	统计测量对象所在集群的已用存储空间大小占总配额的比率。	>=0	%	不涉及	Cloud Table 实例节点	60s
network_throughput_inbound_rate	网络吞吐流入速率	每个节点每秒的网络流入数据量。	>=0	KiB/s	1024(IEC)	Cloud Table 实例节点	60s
network_throughput_outgoing_rate	网络吞吐流出速率	每个节点每秒的网络流出数据量。	>=0	KiB/s	1024(IEC)	Cloud Table 实例节点	60s
number_of_mutations	Mutation个数	每个节点正在运行的修改任务数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_run_merges	运行Merge个数	正在运行的合并任务个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_merges_thread	Merge中线程个数	正在合并中的线程数量。	[0, cpu核数]	Count	不涉及	Cloud Table 实例节点	60s
number_of_zookeeper_requests	ZK请求数	当前时间节点请求ZK的请求数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_zk_session	ZK的session个数	当前时间节点链接ZK的session个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s

指标名称	显示名	含义	取值范围	指标单位	进制	测量对象 (维度)	监控周期
number_of_zookeeper_watches	ZK watch 个数	当前时间节点链接ZK watch个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_zookeeper_nodes	Zookeeper中节点数	Zookeeper中创建临时的节点数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_parts	part数量	part数量。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_threads_holding_read_locks	持有读锁的线程数	持有读锁的线程数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_threads_waiting_to_be_read	等待读的线程数	等待读的线程数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_threads_waiting_to_be_written	等待写的线程数	等待写的线程数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_threads_holding_write_locks	持有写锁的线程数	持有写锁的线程数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
maximum_number_of_active_blocks	最大活跃分块数	clickhouse实例最大的活跃分块个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_consistency_checkdata_blocks	一致性校验数据分块数	clickhouse实例一致性校验数据分块个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
extracting_the_number_of_data_chunks_from_the_replica	正在从副本中提取数据分块数	clickhouse实例正在从副本中提取的数据分块个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s

指标名称	显示名	含义	取值范围	指标单位	进制	测量对象 (维度)	监控周期
number_of_data_chunks_being_sent_to_the_replica	正在发送到副本的数据分块数	clickhouse实例正在发送到副本的数据分块个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_readonly_replicas	只读副本个数	此指标反映了clickhouse实例中只读副本数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_inserted_into_all_tables	插入行数	clickhouse实例中所有表插入行数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_open_read_files	读文件打开数	clickhouse实例上的读文件打开数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_open_write_files	写文件打开数	clickhouse实例上的写文件打开数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
total_file_opens	文件打开总数	clickhouse实例上的文件打开数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_read_file_descriptors	文件描述符读个数	文件描述符读个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_written_file_descriptors	文件描述符写个数	文件描述符写个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
instance_process_running_duration	实例进程运行时长	clickhouse实例进程运行时长。	>=0	s	不涉及	Cloud Table 实例节点	60s

指标名称	显示名	含义	取值范围	指标单位	进制	测量对象 (维度)	监控周期
number_of_threads_running_in_global_thread_pool	全局线程池中运行任务的线程数	clickhouse实例上的全局线程池中运行任务的线程数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_threads_in_global_thread_pool	全局线程池中线程数	clickhouse实例上的全局线程池中线程数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_threads_in_local_thread_pool	本地线程池中的线程数	clickhouse实例上的本地线程池中的线程数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_threads_waiting_locked	Context中等待加锁的线程数	clickhouse实例上的context中等待加锁的线程数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_threads_running_in_local_thread_pool	本地线程池中运行任务的线程数	clickhouse实例上的本地线程池中运行任务的线程数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_tcp_connections	TCP连接数	TCP协议连接个数。	[0,4096]	Count	不涉及	Cloud Table 实例节点	60s
number_of_http_connections	HTTP连接数	HTTP协议连接个数。	[0,4096]	Count	不涉及	Cloud Table 实例节点	60s
number_of_databases	数据库数量	每个节点数据库数量。	[0, 5000]	Count	不涉及	Cloud Table 实例节点	60s
number_of_tables	表数量	每个节点本地表数量。	[0, 5000]	Count	不涉及	Cloud Table 实例节点	60s

指标名称	显示名	含义	取值范围	指标单位	进制	测量对象(维度)	监控周期
write_size_per_second	每秒写入大小	每个节点单位时间写入的数据量。	>=0	Byte、KiB、MiB、GiB、TiB、PiB	1024(IEC)	Cloud Table 实例节点	60s
number_of_running_queries	运行 Query 个数	每个节点正在运行的查询个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_query_threads	Query 线程数	每个节点正在运行的查询线程数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_distributed_ddls	分布式 DDL 个数	分布式 DDL 语句的个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_distributed_table_files	分布式表文件个数	向分布式表写数据时待插入数据的文件个数。	>=0	Count	不涉及	Cloud Table 实例节点	60s
number_of_concurrency	ClickHouse 各节点并发度	当前 ClickHouse 各节点的并发度。	[0,100]	Count	不涉及	Cloud Table 实例节点	60s
zookeeper_quota_percentage	ZooKeeper 数量配额百分比	ClickHouse 服务在 ZooKeeper 上目录的数量配额使用百分比。	0~100	%	不涉及	Cloud Table 实例节点	60s
zookeeper_capacity_quota_percentage	ZooKeeper 容量配额百分比	ClickHouse 服务在 ZooKeeper 上目录的容量配额使用百分比。	0~100	%	不涉及	Cloud Table 实例节点	60s

指标名称	显示名	含义	取值范围	指标单位	进制	测量对象 (维度)	监控周期
zookeeper_process_is_alive	zookeeper进程存活	zookeeper心跳检查, 1表示正常, 0表示异常。	[0, 1]	Count	不涉及	Cloud Table实例节点	60s
number_of_global_sessions	全局session个数	当前时间全局的session个数。	>=0	Count	不涉及	Cloud Table实例节点	60s
number_of_rejected_connections	拒绝链接个数	单位时间内zookeeper拒绝链接的个数。	>=0	Count	不涉及	Cloud Table实例节点	60s
number_of_request_submission_queues	请求提交队列个数	当前时间请求提交队列个数。	>=0	Count	不涉及	Cloud Table实例节点	60s
waiting_time_of_the_preprocessing_queue	预处理队列等待时间	单位时间内预处理队列等待时间。	>=0	s	不涉及	Cloud Table实例节点	60s
number_of_zk_watches	ZK_watch个数	当前时间ZK_watch个数。	>=0	Count	不涉及	Cloud Table实例节点	60s
zk_heapmemory_usage	ZooKeeper使用的堆内存大小	ZooKeeper使用的堆内存大小。	>=0	Byte、KiB、MiB、GiB、TiB、PiB	1024(IEC)	Cloud Table实例节点	60s
zk_directmemory_usage	ZooKeeper使用的直接内存	ZooKeeper使用的直接内存。	>=0	Byte、KiB、MiB、GiB、TiB、PiB	1024(IEC)	Cloud Table实例节点	60s
zk_heapmemory_usage_ratio	ZooKeeper堆内存使用率	ZooKeeper堆内存使用百分比统计。	0~100	%	不涉及	Cloud Table实例节点	60s

指标名称	显示名	含义	取值范围	指标单位	进制	测量对象 (维度)	监控周期
zk_directmemory_usage_ratio	ZooKeeper直接内存使用率	ZooKeeper直接内存使用百分比统计。	0~100	%	不涉及	Cloud Table实例节点	60s
cluster_number_of_sent_packets	zookeeper发包个数	zookeeper节点发送的packet个数。	>=0	Count	不涉及	Cloud Table实例节点	60s
cluster_number_of_received_packets	zookeeper收包个数	zookeeper节点接收的packet个数。	>=0	Count	不涉及	Cloud Table实例节点	60s
cluster_preprocessing_time	预处理时间	单位时间内预处理时间。	>=0	s	不涉及	Cloud Table实例节点	60s
rows_written_per_second	每秒写入行数	每个节点单位时间写入的行数。	>=0	Count	不涉及	Cloud Table实例节点	60s
number_of_failed_queries	失败Query个数	失败的query语句个数。	>=0	Count	不涉及	Cloud Table实例节点	60s
number_of_failed_insert_queries	失败Insert Query个数	失败的插入语句个数。	>=0	Count	不涉及	Cloud Table实例节点	60s
number_of_failed_select_query_messages	失败Select Query个数	失败的select语句个数。	>=0	Count	不涉及	Cloud Table实例节点	60s
number_of_delayed_inserts	delayed inserts个数	每个节点延迟插入个数。	>=0	Count	不涉及	Cloud Table实例节点	60s
number_of_rows	字段数量	每个节点mergetree引擎字段数量。	>=0	Count	不涉及	Cloud Table实例节点	60s

指标名称	显示名	含义	取值范围	指标单位	进制	测量对象(维度)	监控周期
disk_throughput_read_rate	磁盘吞吐读速率	磁盘读取吞吐速率。	>=0	Byte/s 、 KiB/s 、 MiB/s 、 GiB/s 、 TiB/s 、 PiB/s	1024(IEC)	Cloud Table实例节点	60s
disk_throughput_write_rate	磁盘吞吐写速率	磁盘写入吞吐速率。	>=0	Byte/s 、 KiB/s 、 MiB/s 、 GiB/s 、 TiB/s 、 PiB/s	1024(IEC)	Cloud Table实例节点	60s

维度

Key	Value
cluster_id	CloudTable集群ID。
instance_name	CloudTable集群节点名称。

4.9.2.2 设置 ClickHouse 集群监控规则

通过设置CloudTable ClickHouse的告警规则，用户可以自定义监控目标和通知策略，及时了解ClickHouse运行状态，从而起到预警作用。设置ClickHouse的告警规则包括设置告警规则名称、监控对象、监控指标、告警阈值、监控周期和是否发送通知等参数。本节介绍了设置告警规则的具体方法。

设置 ClickHouse 集群告警规则

步骤1 登录表格存储服务管理控制台。

步骤2 在集群管理列表中的右上角，输入要搜索的集群名称，单击 。

步骤3 在集群列表中，找到所需要集群，在对应的“操作”列中，单击“查看监控信息”，系统将跳转到该集群的“云服务监控”页面。

待查看监控信息的集群状态必须为“服务中”。

步骤4 单击“创建告警规则”，进入创建告警规则页面进行添加。

具体参数设置请参考[创建告警规则和通知](#)。

步骤5 配置完成后，单击“立即创建”，完成告警规则的创建。告警规则添加完成后，当监控指标触发设定的阈值时，云监控会在第一时间通过消息通知服务实时告知您资源异常，以免因此造成业务损失。

📖 说明

如果想设置单个指标的告警规则，进入查看监控信息页面：

- 查看监控指标，单击指标图。
- 单击右上角的“+”。
- 进入创建告警规则页面，设置规则。

----结束

4.9.2.3 查看 ClickHouse 集群监控信息

操作场景

云平台提供的云监控服务，可以对ClickHouse集群的运行状态进行日常监控。您可以通过管理控制台，直观的查看ClickHouse的各项监控指标。通过查看集群的监控信息可以快速掌握集群的健康状态并及时获取系统的关键信息。

监控功能

集群监控功能整体分为计算节点指标监控和ZooKeeper指标监控两个模块。在每个模块中用户都可调整监控数据的时间范围：方便查看不同时间的历史数据。也可调整监控数据的时间粒度：方便在不同维度层次查看数据。如果想实时查看监控数据，可以开启自动刷新。平台可按1小时、3小时、12小时、24小时、7天的间隔自动刷新监控图表。

如果想看那个指标，可以单击放大按钮查看，也可以导出图片查看。

在 CES 界面查看 ClickHouse 集群监控信息

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 进入集群管理页面，在集群列表中找到需要查看的集群，在对应的操作列中，单击查看监控信息，系统会跳到该集群的“云服务监控”页面。

待查看集群，其状态必须为“服务中”。

步骤4 在集群的“云服务监控”页面中，单击集群ID前的按钮  展开计算单元列表，选择对应节点查看监控信息。

步骤5 如果您查看监控页面的指标太多，可以对指标进行设置，设置自己需要查看的指标。

1. 如果指标太多，可以通过设置监控指标页面对指标进行删除。
2. 如果页面显示的指标没有您想看的指标，可以在设置监控指标页面进行添加。
3. 选择您在页面中要展示的指标名称，拖动选中指标可以对指标进行排序。

📖 说明

- 如果想实时查看监控数据，可以开启自动刷新。平台可按1小时、3小时、12小时、24小时、7天的间隔自动刷新监控图表。
- 如果想放大单个指标视图，在监控指标视图右上角，单击  可查看监控指标视图详情。

----结束

在集群详情页查看 ClickHouse 集群监控信息

集群详情页面主要展示每个节点的CPU使用率、内存使用率、磁盘使用率、磁盘使用量、磁盘总量。

步骤1 登录表格存储服务管理控制台。

步骤2 进入集群管理页面，选择目标ClickHouse集群，单击“集群名称 > 详情”。

步骤3 进入详情页面，详情页面显示监控页面的指标。

图 4-9 ClickHouse 监控页面



步骤4 如要想要查看此节点所有指标信息，请单击详情页面的监控小图标 ，可以跳转CES云服务监控页面查看ClickHouse集群单节点详细监控指标。

----结束

4.9.3 ClickHouse 集群日志管理

4.9.3.1 对接 LTS 查看 ClickHouse 集群日志

集群日志服务用于采集集群日志并报送云日志服务（Log Tank Service，以下简称 LTS），用户可以在LTS云日志服务查看采集的集群日志或进行日志转储。

📖 说明

当前支持的日志类型：

ClickHouse集群存放日志路径：/var/log/clickhouse/

- clickhouse-server.log
- clickhouse-server.err.log
- clickhouse-server-audit.log

开启云日志服务对接步骤

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群，进入集群列表。

步骤4 在集群列表中单击集群名称，进入集群详情页面，左侧导航栏单击“日志服务”，进入日志服务页面。

步骤5 单击页面左上角的 ，开启云日志服务对接。

📖 说明

- 如果首次开启，会弹出创建委托窗口，单击“确定”，进行授权委托。
- 若已开启过LTS云日志服务对接，并授权该服务创建委托，再次开启时则无需授权操作。

----结束

查看集群日志

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群，进入集群列表。

步骤4 在集群列表中单击集群名称，进入集群详情页面，左侧导航栏单击“日志服务”，进入日志服务页面。

步骤5 选择查看日志，单击“操作 > 查看日志”。即可跳转到LTS云日志服务查看日志。

----结束

关闭日志

步骤1 登录表格存储服务管理控制台。

步骤2 控制台左上角选择区域。

步骤3 单击集群，进入集群列表。

步骤4 在集群列表中单击集群名称，进入集群详情页面，左侧导航栏单击“日志服务”，进入日志服务页面。

步骤5 单击“云日志服务对接”开关，弹出“关闭云日志服务”提示框。

步骤6 确认无误后，单击提示框中的“确认”按钮，即可关闭日志采集。

---结束

4.9.3.2 对接 CTS 查看 ClickHouse 集群审计日志

CloudTable使用云审计服务，可以记录与CloudTable服务相关的操作事件，便于日后的查询、审计和回溯。在开启了云审计服务后，系统开始记录CloudTable服务的操作日志。云审计服务管理控制台保存最近7天的操作记录。

CloudTable的以下关键操作事件将记录审计日志，详细内容如表4-35所示。

表 4-35 云审计服务支持的 CloudTable 操作事件列表

操作名称	事件名称	资源类型
创建集群	createCloudTableClusterV3	cluster
节点扩容	growCloudTableCluster	cluster
重启集群	rebootCloudTableCluster	cluster
设置存储配额	storageClusterAction	cluster
特性开关	modifyClusterFeatures	cluster
创建数据迁移任务	copierCreateTask	cluster
枚举数据库信息	copierListDatabaseInfo	cluster
枚举集群节点信息	copierListNodeInfo	cluster
数据迁移任务详情	copierTaskDetail	cluster
ClickHouse创建用户	createCloudTableAccount	cluster
创建角色	createRole	cluster
删除集群	deleteCloudTableClusterV2	cluster
删除角色	deleteRole	cluster
关闭集群日志	disableLTSAccess	cluster
打开集群日志	enableLTSAccessc	cluster
获取集群信息	getClusterInfo	cluster
获取数据库信息	getDatabases	cluster
获取角色信息	getRoles	cluster
获取表信息	getTables	cluster
磁盘扩容	growCloudTableDisk	cluster
规格扩容	growCloudTableFlavor	cluster
重启节点	restartInstance	cluster

操作名称	事件名称	资源类型
重启	REBOOTING	cluster
扩容	GROWING	cluster
删除	DELETING	cluster
包周期集群扩容、规格变更	changeCloudTableCluster	cluster
ClickHouse同时开启安全和非安全通道	enableBothSslAndNone	cluster
开启冷热分离	switchHotColdFeature	cluster
查杀sql	killQueryBySqlId	cluster
ClickHouse删除用户	deleteCloudTableAccount	cluster
ClickHouse更新用户	updateCloudTableAccount	cluster

开启云审计服务

使用云审计服务前需要开启云审计服务，开启云审计服务后系统会自动创建一个追踪器，系统记录的所有操作将关联在该追踪器中。目前，一个云账户系统仅支持创建一个追踪器。

步骤1 在CloudTable管理控制台，选择“服务列表 > 管理与部署 > 云审计服务”，进入云审计服务信息页面。

步骤2 开通云审计服务。

如果您是首次使用云审计服务，在追踪器列表中还没有已创建的追踪器，则请参考《[云审计服务快速入门](#)》中的“[开通云审计服务](#)”开启云审计服务。

如果您已开通过云审计服务了，开通时系统已为您自动创建了一个管理事件追踪器，管理事件追踪器只能有一个且不可删除。您也可以自行创建数据事件追踪器，详细内容请参见《[云审计服务用户指南](#)》中的[创建追踪器](#)。

----结束

关闭审计日志

如果用户想关闭审计日志，需要在云审计服务中停用追踪器。

步骤1 在CloudTable管理控制台，选择“服务列表 > 管理与监管 > 云审计服务”，进入云审计服务信息页面。

步骤2 通过停用追踪器，关闭审计日志。如需重新开启审计日志，只要启用追踪器即可。

有关停用/启用追踪器的更多信息，请参考《[云审计服务用户指南](#)》中的[启用/停用追踪器](#)。

----结束

查看 CloudTable 的云审计日志

步骤1 登录管理控制台。

步骤2 单击页面上方的“服务列表”，选择“管理与监管 > 云审计服务”，进入云审计服务信息页面。

步骤3 单击左侧导航树的“事件列表”，进入事件列表信息页面。

步骤4 单击事件列表右上方的“筛选”，设置对应的操作事件条件。

当前事件列表支持四个维度的组合查询，详细信息如下：

- “事件来源”、“资源类型”和“筛选类型”。
 - “事件来源”：选择“CloudTable”。
 - “资源类型”：选择“所有资源类型”，或者指定具体的资源类型。
 - “筛选类型”：选择“所有筛选类型”，或者选择以下任一选项。
 - “按事件名称”：选择该选项时，还需选择某个具体的事件名称。
 - “按资源ID”：选择该选项时，还需选择或者手动输入某个具体的资源ID。
 - “按资源名称”：选择该选项时，还需选择或手动输入某个具体的资源名称。
- “操作用户”：在下拉框中选择某一具体的操作用户，此操作用户指用户级别，而非租户级别。
- “事件级别”：可选项为“所有事件级别”、“normal”、“warning”、“incident”，只可选择其中一项。
- “起始时间”、“结束时间”：可通过选择时间段查询操作事件。

步骤5 单击“查询”，查看对应的操作事件。

步骤6 在需要查看的事件左侧，单击  展开该记录的详细信息。

图 4-10 事件

事件名称	资源类型	云服务	资源ID	资源名称	事件级别	操作用户	操作时间	操作
login	user	IAM	dsd(2a91cbad...		normal		2025/05/27 09:58:42 GMT+08:00	查看详情
<pre> trace_id 1d099662-4ebd0 code 302 trace_name login resource_type user trace_rating normal message [{"login":{"mode":"pass","device":"phone","status":"on"}}] source_ip 1.13.25 domain_id 08aa4c01688f7b7c7c95508960 trace_type ConsoleAction service_type IAM event_type global project_id 08c4f1a0105611 read_only false resource_id 05a84352e1c8a0240c446 </pre>								

步骤7 在需要查看的事件右侧，单击“查看事件”，弹出一个窗口，显示了该操作事件结构的详细信息。

图 4-11 查看事件

查看事件

```
{
  "service_type": "CloudTable",
  "user": {
    "name": " ",
    "id": "2f210e62179a425c96e5b57f25d1d492",
    "domain": {
      "name": " ",
      "id": "760dfba8429649e7a477445e1033a839"
    }
  },
  "time": "2018/05/29 14:50:16 GMT+08:00",
  "code": 200,
  "resource_type": "cluster",
  "resource_name": "cloudtable ",
  "resource_id": "689ae304-086b-4fa9-984e-16c58e3d75bf",
  "source_ip": " ",
  "trace_name": "rebootCloudTableCluster",
  "trace_type": "ConsoleAction",
  "api_version": "v1.0",
  "record_time": "2018/05/29 14:50:16 GMT+08:00",
  "trace_id": "8b253930-630c-11e8-82ae-286ed488cbe3",
  "trace_status": "normal"
}
```

关于云审计服务事件结构的关键字段详解，请参见《云审计服务用户指南》的[事件结构](#)章节。

----结束

4.10 ClickHouse 存算一体常用 SQL 命令

4.10.1 数据类型

此章节描述ClickHouse的数据类型。

ClickHouse不支持JSON以及Object('json')数据类型。

数据类型表

表 4-36 数据类型表

分类	关键字	数据类型	描述
整数类型	Int8	Int8	取值范围：【 -128, 127 】
	Int16	Int16	取值范围：【 -32768, 32767 】
	Int32	Int32	取值范围：【 -2147483648, 2147483647 】
	Int64	Int64	取值范围：【 -9223372036854775808, 9223372036854775807 】
浮点类型	Float32	单精度浮点数	同C语言Float类型，单精度浮点数在机内占4个字节，用32位二进制描述。

分类	关键字	数据类型	描述
	Float64	双精度浮点数	同C语言Double类型，双精度浮点数在机内占8个字节，用64位二进制描述。
Decimal类型	Decimal	Decimal	<p>有符号的定点数，可在加、减和乘法运算过程中保持精度。支持几种写法：</p> <ul style="list-style-type: none"> • Decimal(P, S) • Decimal32(S) • Decimal64(S) • Decimal128(S) <p>说明</p> <p>P: 精度，有效范围: [1:38]，决定可以有多少个十进制数字（包括分数）。</p> <p>S: 规模，有效范围: [0: P]，决定数字的小数部分中包含的小数位数。</p>
字符串类型	String	字符串	字符串可以是任意长度的。它可以包含任意的字节集，包含空字节。因此，字符串类型可以代替其他DBMSs中的VARCHAR、BLOB、CLOB等类型。
	FixedString	固定字符串	<p>当数据的长度恰好为N个字节时，FixedString类型是高效的。在其他情况下，这可能会降低效率。可以有效存储在FixedString类型的列中的值的示例：</p> <ul style="list-style-type: none"> • 二进制表示的IP地址（IPv6使用FixedString（16）） • 语言代码（ru_RU, en_US ...） • 货币代码（USD, RUB ...） • 二进制表示的哈希值（MD5使用FixedString（16），SHA256使用FixedString（32））
时间日期类型	Date	日期	用两个字节存储，表示从1970-01-01（无符号）到当前的日期值。日期中没有存储时区信息。

分类	关键字	数据类型	描述
	DateTime	时间戳	用四个字节（无符号的）存储 Unix 时间戳。允许存储与日期类型相同的范围内的值。最小值为 1970-01-01 00:00:00。时间戳类型值精确到秒（没有闰秒）。时区使用启动客户端或服务器时的系统时区。
	DateTime64	DateTime64	此类型允许以日期（date）加时间（time）的形式来存储一个时刻的时间值。
布尔型	Boolean	Boolean	ClickHouse 没有单独的类型来存储布尔值。可以使用 UInt8 类型，取值限制为 0 或 1。
数组类型	Array	Array	Array(T)，由 T 类型元素组成的数组。T 可以是任意类型，包含数组类型。但不推荐使用多维数组，ClickHouse 对多维数组的支持有限。例如，不能在 MergeTree 表中存储多维数组。
元组类型	Tuple	Tuple	Tuple(T1, T2, ...)，元组，其中每个元素都有单独的类型，不能在表中存储元组（除了内存表）。它们可以用于临时列分组。在查询中，IN 表达式和带特定参数的 lambda 函数可以用来对临时列进行分组。
Domains 数据类型	Domains	Domains	Domains 类型是特定实现的类型： <ul style="list-style-type: none"> IPv4 是与 UInt32 类型保持二进制兼容的 Domains 类型，用于存储 IPv4 地址的值。它提供了更为紧凑的二进制存储的同时支持识别可读性更加友好的输入输出格式。 IPv6 是与 FixedString(16) 类型保持二进制兼容的 Domain 类型，用于存储 IPv6 地址的值。它提供了更为紧凑的二进制存储的同时支持识别可读性更加友好的输入输出格式。
枚举类型	Enum8	Enum8	取值范围：【 -128, 127 】 Enum 保存 'string' = integer 的对应关系，例如：Enum8('hello' = 1, 'world' = 2)

分类	关键字	数据类型	描述
	Enum16	Enum16	取值范围：【 -32768, 32767 】
可为空	Nullable	Nullable	<p>除非在ClickHouse服务器配置中另有说明，否则NULL是任何Nullable类型的默认值。Nullable类型字段不能包含在表索引中。</p> <p>可以与TypeName的正常值存放一起。例如，Nullable(Int8) 类型的列可以存储Int8类型值，而没有值的行将存储NULL。</p>
嵌套类型	nested	nested	<p>嵌套的数据结构就像单元格内的表格。嵌套数据结构的参数（列名和类型）的指定方式与CREATE TABLE查询中的指定方式相同。每个表行都可以对应于嵌套数据结构中的任意数量的行。</p> <p>示例：Nested(Name1 Type1, Name2 Type2, ...)</p>

4.10.2 CREATE DATABASE

本章节介绍创建数据库的基本用法。

CREATE DATABASE

```
CREATE DATABASE [IF NOT EXISTS] db_name [ON CLUSTER ClickHouse集群名];
```

表 4-37 参数说明

参数	说明
db_name	数据库
IF NOT EXISTS	如果CREATE语句中存在IF NOT EXISTS关键字，则当数据库已经存在时，该语句不会创建数据库，且不会返回任何错误。
ON CLUSTER ClickHouse集群名	用于指定集群名称。

📖 说明

集群名信息可以使用以下语句的cluster字段获取：

```
select cluster,shard_num,replica_num,host_name from system.clusters;
```

使用示例

- 创建数据库demo。
create database demo ON CLUSTER default_cluster;

- 查看新建的数据库。
host-172-16-30-9 :) show databases;
SHOW DATABASES
Query id: ced1af23-0286-40cc-9c7a-ccbca41178d8

name
INFORMATION_SCHEMA
default
demo
information_schema
system

5 rows in set. Elapsed: 0.002 sec.

4.10.3 CREATE TABLE

此章节介绍如何创建表。

创建本地表

```
CREATE TABLE [IF NOT EXISTS] [database_name.]table_name [ON CLUSTER ClickHouse集群名]
(
  name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
  name2[type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
  ...
) ENGINE = engine_name()
[PARTITION BY expr_list]
[ORDER BY expr_list]
```

表 4-38 参数说明

参数	说明
database_name	数据库的名称，默认为当前选择的数据库。
table_name	本地表名。
ON CLUSTER <i>ClickHouse 集群名</i>	在每一个节点上都创建一个本地表，固定为ON CLUSTER <i>ClickHouse 集群名</i> 。
name1,name2	列名。
ENGINE = engine_name()	表引擎类型。 双副本版集群建表时，需要使用MergeTree系列引擎中支持数据复制的Replicated*引擎，否则副本之间不进行数据复制，导致数据查询结果不一致。使用该引擎建表时，参数填写方式如下。 <ul style="list-style-type: none"> ReplicatedMergeTree('/clickhouse/tables/{database}/{table}/{shard}', '{replica}'), 固定配置，无需修改。 ReplicatedMergeTree(), 等同于 ReplicatedMergeTree('/clickhouse/tables/{database}/{table}/{shard}', '{replica}')。
ORDER BY expr_list	排序键，必填项，可以是一组列的元组或任意表达式。
[PARTITION BY expr_list]	分区键。一般按照日期分区，也可以使用其他字段或字段表达式。

示例：

- 创建数据库。请参见[CREATE DATABASE](#)。

- 使用数据库。

```
use demo;
```

- 创建名为demo.test表。

```
CREATE TABLE demo.test ON CLUSTER default_cluster('EventDate` DateTime, `id` UInt64)ENGINE =
ReplicatedMergeTree('/clickhouse/tables/{shard}/default/test', '{replica}') PARTITION BY
toYYYYMM(EventDate) ORDER BY id;
```

复制表结构创建表

可以通过复制表结构创建与源表具有相同结构的表。语法：

```
CREATE TABLE [IF NOT EXISTS] [db.]table_name2 ON CLUSTER ClickHouse集群名
AS [db.]table_name1 [ENGINE = engine_name];
```

表 4-39 参数说明

参数	说明
db	数据库的名称，默认为当前选择的数据库。
table_name1	被复制表结构的源表。
table_name2	新创建的表。
ON CLUSTER ClickHouse集群名	在每一个节点上都创建一个表，固定为ON CLUSTER ClickHouse集群名。
[ENGINE = engine_name]	表引擎类型。如果没有指定表引擎，默认与被复制表结构的 表相同。

示例：

- 创建数据库。

```
create database demo;
```

- 使用数据库。

```
use demo;
```

- 创建数据表。

```
create table demo_t(uid Int32,name String,age UInt32,gender String)engine = TinyLog;
```

- 复制表结构。

```
create table demo_t2 as demo_t;
```

- [查看表结构](#)。

SELECT 语句创建

使用指定的表引擎创建一个与SELECT子句的结果具有相同结构的表，并使用SELECT子句的结果进行填充。

```
CREATE TABLE [IF NOT EXISTS] [database_name.]table_name ENGINE = engine_name AS SELECT ...
```

表 4-40 参数说明

参数	说明
database_name	数据库的名称，默认为当前选择的数据库。
table_name	通过SELECT语句创建的表。
ENGINE = engine_name()	表的引擎类型。
SELECT ...	SELECT子句。

示例:

- 创建表。
CREATE TABLE default.demo1 ON CLUSTER default_cluster(`EventDate` DateTime, `id` UInt64)ENGINE = ReplicatedMergeTree('/clickhouse/tables/{shard}/default/demo1', '{replica}') PARTITION BY toYYYYMM(EventDate) ORDER BY id;
- 通过SELECT语句创建表。
create table t3 ON CLUSTER default_cluster ENGINE =MergeTree() order by EventDate as select * from default.demo1;
- 查询demo1和t3表结构。
desc demo1;

查询结果显示，两张表结构一样。

```
cloudtable-wlr-click-20230730-06-server-1-1 :) desc demo1;
```

```
DESCRIBE TABLE demo1
```

```
Query id: 712f6b91-668d-4f70-b160-aac8e52f63a4
```

name	type	default_type	default_expression	comment
EventDate	DateTime			
id	UInt64			

```
2 rows in set. Elapsed: 0.001 sec.
```

```
cloudtable-wlr-click-20230730-06-server-1-1 :) desc t3;
```

```
DESCRIBE TABLE t3
```

```
Query id: 11b67532-26f0-49c5-b36d-439d45c279bf
```

name	type	default_type	default_expression	comment
EventDate	DateTime			
id	UInt64			

```
2 rows in set. Elapsed: 0.001 sec.
```

4.10.4 DESC 查询表结构

本章节主要介绍ClickHouse查询表结构的SQL基本语法和使用说明。

基本语法

```
DESC|DESCRIBE TABLE [database_name.]table [INTO OUTFILE filename] [FORMAT format]
```

接[复制表结构创建表](#)示例，查询demo_t和demo_2表结构：

```
cloudtable-wlr-click-20230730-06-server-1-1 :) desc demo_t;
```

```
DESCRIBE TABLE demo_t
```

```
Query id: 27a38d90-9459-430f-962e-881817789fc9
```

name	type	default_type	default_expression	comment	codec_expression
EventDate	DateTime				
id	UInt64				

```

tll_expression┌
uid          | Int32 |
name         | String|
age          | UInt32|
gender       | String|
└───────────┘

4 rows in set. Elapsed: 0.001 sec.
cloudtable-wlr-click-20230730-06-server-1-1 :) desc demo_t2;
DESCRIBE TABLE demo_t2
Query id: 60054fe3-794c-410a-be13-cd0b204a9129
┌name┐┌type┐┌default_type┐┌default_expression┐┌comment┐┌codec_expression┐
tll_expression┌
uid          | Int32 |
name         | String|
age          | UInt32|
gender       | String|
└───────────┘

4 rows in set. Elapsed: 0.001 sec.

```

4.10.5 CREATE VIEW

本章节介绍如何在ClickHouse中创建普通视图。

创建视图

```
CREATE VIEW [IF NOT EXISTS] [db.]view_name [ON CLUSTER ClickHouse集群名] AS SELECT ...
```

表 4-41 参数说明

参数	说明
db	数据库的名称，默认为当前选择的数据库。
view_name	视图名。
[ON CLUSTER ClickHouse集群名]	在每一个节点上都创建一个视图，固定为ON CLUSTER ClickHouse集群名。
SELECT ...	SELECT子句。当数据写入视图中SELECT子句所指定的源表时，插入的数据会通过SELECT子句查询进行转换并将最终结果插入到视图中。

示例：

1. 创建源表。
create table DB.table1 ON CLUSTER default_cluster (id Int16,name String) ENGINE = MergeTree()
ORDER BY (id);
2. 创建视图。
CREATE VIEW test_view ON CLUSTER default_cluster AS SELECT * FROM DB.table1;
3. 插入数据到源表中。
insert into DB.table1 values(1,'X'),(2,'Y'),(3,'Z');
4. 查询视图。
SELECT * FROM test_view;
5. 删除视图。
drop table test_view ON CLUSTER default_cluster;

 说明

- 如果建表语句中包含了“ON CLUSTER ClickHouse集群名”，删除表命令：
drop table 表名 ON CLUSTER default_cluster;
- 如果建表语句不包含“ON CLUSTER ClickHouse集群名”，删除表命令：
drop table 表名;

4.10.6 CREATE MATERIALIZED VIEW

本章节介绍如何在ClickHouse中创建物化视图。

创建物化视图

```
CREATE MATERIALIZED VIEW [IF NOT EXISTS] [db.]Materialized_name [TO[db.]name] [ON
CLUSTER ClickHouse集群名]
ENGINE = engine_name()
ORDER BY expr
[POPULATE]
AS SELECT ...
```

表 4-42 参数说明

参数	说明
db	数据库的名称，默认为当前选择的数据库。
Materialized_name	物化视图名。
TO[db.]name	将物化视图的数据写入到新表中。
[ON CLUSTER ClickHouse集群名]	在每一个节点上都创建一个物化视图，固定为ON CLUSTER ClickHouse集群名。
ENGINE = engine_name()	表引擎类型。
[POPULATE]	POPULATE关键字。如果创建物化视图时指定了POPULATE关键字，则在创建时将SELECT子句所指定的源表数据插入到物化视图中。不指定POPULATE关键字时，物化视图只会包含在物化视图创建后新写入源表的数据。 说明 一般不推荐使用POPULATE关键字，因为在物化视图创建期间写入源表的数据将不会写入物化视图中。
SELECT ...	SELECT子句。当数据写入物化视图中SELECT子句所指定的源表时，插入的数据会通过SELECT子句查询进行转换并将最终结果插入到物化视图中。 说明 SELECT查询可以包含DISTINCT、GROUP BY、ORDER BY和LIMIT等，但是相应的转换是在每个插入数据块上独立执行的。

示例：

1. 创建源表。
`create table DB.table1 ON CLUSTER default_cluster (id Int16,name String) ENGINE = MergeTree() ORDER BY (id);`
2. 插入数据。
`insert into DB.table1 values(1,'X'),(2,'Y'),(3,'Z');`
3. 创建基于源表的物化视图。
`CREATE MATERIALIZED VIEW demo_view ON CLUSTER default_cluster ENGINE = MergeTree() ORDER BY (id) AS SELECT * FROM DB.table1;`
4. 查询物化视图。
`SELECT * FROM demo_view;`

📖 说明

查询数据为空，说明未指定POPULATE关键字时，查询不到物化视图创建前写入源表的数据。

5. DB.table1表中插入数据。
`insert into demo_view values(4,'x'),(5,'y'),(6,'z');`
6. 查询物化视图。
`SELECT * FROM demo_view;`

查询结果。

id	name
4	x
5	y
6	z

4.10.7 INSERT INTO

本章节介绍如何插入数据。

基本语法

- 标准格式插入数据。
`INSERT INTO [db.]table [(c1, c2, c3)] VALUES (v11, v12, v13), (v21, v22, v23), ...`

📖 说明

对于存在于表结构中但不存在于插入列表中的列，它们将会按照如下方式填充数据：

- 如果存在DEFAULT表达式，根据DEFAULT表达式计算被填充的值。
- 如果没有定义DEFAULT表达式，则填充零或空字符串。

接[复制表结构创建表](#)示例，插入数据：

```
insert into demo_t values(1,'Candy','23','M'),(2,'cici','33','F');
```

- 使用SELECT的结果写入。
`INSERT INTO [db.]table [(c1, c2, c3)] SELECT ...`

📖 说明

写入的列与SELECT的列的对应关系是使用位置来进行对应的，它们在SELECT表达式与INSERT中的名称可以是不同的。需要对它们进行对应的类型转换。

除了VALUES格式之外，其他格式中的数据都不允许出现诸如now(), 1+2等表达式。VALUES格式允许您有限度的使用这些表达式，但是不建议您这么做，因为执行这些表达式很低效。

4.10.8 SELECT

描述如何使用SELECT语句查询数据。

基本语法

```
SELECT [DISTINCT] expr_list
[FROM [database_name.]table | (subquery) | table_function] [FINAL]
[SAMPLE sample_coeff]
[ARRAY JOIN ...]
[GLOBAL] [ANY|ALL|ASOF] [INNER|LEFT|RIGHT|FULL|CROSS] [OUTER|SEMI|ANTI] JOIN (subquery)|table
(ON <expr_list>)|(USING <column_list>)
[PREWHERE expr]
[WHERE expr]
[GROUP BY expr_list] [WITH TOTALS]
[HAVING expr]
[ORDER BY expr_list] [WITH FILL] [FROM expr] [TO expr] [STEP expr]
[LIMIT [offset_value, ]n BY columns]
[LIMIT [n, ]m] [WITH TIES]
[UNION ALL ...]
[INTO OUTFILE filename]
[FORMAT format]
```

示例:

- 查看ClickHouse集群信息。
select * from system.clusters;
- 显示当前节点设置的宏。
select * from system.macros;
- 查看数据库容量。
select sum(rows) as "总行数", formatReadableSize(sum(data_uncompressed_bytes)) as "原始大小",
formatReadableSize(sum(data_compressed_bytes)) as "压缩大小",
round(sum(data_compressed_bytes) / sum(data_uncompressed_bytes) * 100, 0) "压缩率" from
system.parts;
- 查询test表容量。where条件根据实际情况添加修改。
select sum(rows) as "总行数", formatReadableSize(sum(data_uncompressed_bytes)) as "原始大小",
formatReadableSize(sum(data_compressed_bytes)) as "压缩大小",
round(sum(data_compressed_bytes) / sum(data_uncompressed_bytes) * 100, 0) "压缩率" from
system.parts where table in ('test') and partition like '2020-11-%' group by table;

4.10.9 ALTER TABLE 修改表结构

本章节主要介绍ClickHouse修改表结构的SQL基本语法和使用说明。

基本语法

```
ALTER TABLE [database_name].name [ON CLUSTER ClickHouse集群名] ADD|DROP|CLEAR|COMMENT|
MODIFY COLUMN ...
```

📖 说明

ALTER仅支持 *MergeTree , Merge以及Distributed等引擎表。

示例:

1. 创建表DB_table1。
CREATE TABLE DB_table1 ON CLUSTER default_cluster(Year UInt16,Quarter UInt8,Month
UInt8,DayofMonth UInt8,DayOfWeek UInt8,FlightDate Date,FlightNum String,Div5WheelsOff
String,Div5TailNum String)ENGINE = MergeTree() PARTITION BY toYYYYMM(FlightDate) PRIMARY
KEY (intHash32(FlightDate)) ORDER BY (intHash32(FlightDate),FlightNum) SAMPLE BY
intHash32(FlightDate) SETTINGS index_granularity= 8192;
2. 给DB_table1增加列test。
ALTER TABLE DB_table1 ADD COLUMN test String DEFAULT 'defaultvalue';
查表。
desc DB_tables;
3. 修改表DB_table1列Year类型为UInt8。

```
ALTER TABLE DB_table1 MODIFY COLUMN Year UInt8;
```

查表结构。

```
desc DB_tables;
```

4. 删除表DB_table1列test。

```
ALTER TABLE DB_table1 DROP COLUMN test;
```

查表。

```
desc DB_tables;
```

5. 修改表DB_table1列Month为Month_test。

```
ALTER TABLE DB_table1 RENAME COLUMN Month to Month_test;
```

查表。

```
desc DB_tables;
```

4.10.10 DROP 删除表

此章节主要介绍ClickHouse删除表的SQL基本语法和使用说明。

基本语法

```
DROP [TEMPORARY] TABLE [IF EXISTS] [database_name.]name [ON CLUSTER cluster] [SYNC]
```

示例：

删除表t1。

```
drop table t1 SYNC;
```

说明

- 在删除复制表时，因为复制表需要在Zookeeper上建立一个路径，存放相关数据。ClickHouse默认的库引擎是原子数据库引擎，删除Atomic数据库中的表后，它不会立即删除，而是会在24小时后删除。在删除表时，加上SYNC字段，即可解决该问题，例如：

```
drop table t1 SYNC;
```
- 删除本地表和分布式表，则不会出现该问题，可不带SYNC字段，例如：

```
drop table t1;
```
- 如果建表语句中包含了“ON CLUSTER ClickHouse集群名”，删除表命令：

```
drop table 表名 ON CLUSTER default_cluster;
```
- 如果建表语句不包含“ON CLUSTER ClickHouse集群名”，删除表命令：

```
drop table 表名;
```
- 删除数据表前，需确认此数据表是否应用中，以免引起不必要的麻烦。删除数据表后可在24小时内恢复，超过24小时无法恢复。恢复命令如下：

```
set allow_experimental_undrop_table_query = 1;  
UNDROP TABLE 数据表名;
```

4.10.11 SHOW 显示数据库和表信息

此章节主要介绍ClickHouse显示数据库和表信息的SQL基本语法和使用说明。

基本语法

```
show databases;  
show tables;
```

示例：

- 查询数据库。

```
show databases;
```
- 查询表信息。

```
show tables;
```