

ALIBABA CLOUD

阿里云

云数据库ClickHouse

云数据库ClickHouse公共云文档合集

文档版本：20220712

 阿里云

法律声明

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

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

通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置>网络>设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
Courier字体	命令或代码。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	<code>bae log list --instanceid Instance_ID</code>
[] 或者 [a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ } 或者 {a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

目录

1.动态与公告	08
1.1. 版本发布记录	08
2.产品简介	14
2.1. 什么是云数据库ClickHouse	14
2.2. 产品优势	15
2.3. 内核版本功能对比	16
2.4. 典型应用	17
2.5. 基本概念	18
2.6. 使用限制	19
3.产品计费	21
3.1. 计费方式	21
3.2. 计费概述	21
3.3. 规格及定价	23
3.3.1. 社区兼容版计算节点费用	23
3.3.2. 社区兼容版存储空间费用	23
3.3.3. 社区兼容版备份空间费用	24
3.3.4. 云原生版费用	25
3.4. 变更配置计费说明	26
3.5. 到期或欠费的影响	26
3.6. 续费	26
3.7. 查看消费明细	28
3.8. 依赖服务计费说明	28
4.快速入门	30
4.1. 总体流程	30
4.2. 准备权限	30
4.3. 新建集群	31

4.4. 创建账号	33
4.5. 设置白名单	34
4.6. 连接集群	35
4.7. 创建数据库	37
4.8. 创建表	38
4.9. 导入数据	43
5.数据迁移同步	46
5.1. 从Kafka同步数据	46
5.2. 从Flink导入	48
5.2.1. 从JDBC connector导入	48
5.2.2. 从Flink SQL导入	55
5.3. 从大数据导入/同步	58
5.3.1. 从DataWorks同步	58
5.3.2. 从MaxCompute导入	64
5.3.2.1. 从MaxCompute外表导入 (19.15.2.2)	64
5.3.2.2. 从MaxCompute外表导入 (20.3及以上)	67
5.4. 从RDS MySQL导入/同步	72
5.4.1. 从RDS MySQL导入	72
5.4.2. 从RDS MySQL同步	75
5.5. 从SLS同步	78
5.6. 从OSS导入	80
5.7. 访问OSS数据	86
5.8. 从自建ClickHouse迁移上云	90
5.9. 云数据库ClickHouse实例间的数据迁移	92
5.10. 从本地数据导入	95
5.10.1. 从clickhouse-client导入	95
5.10.2. 从Spark导入	99
6.用户指南	102

6.1. 新建集群	102
6.1.1. 新建社区兼容版集群	102
6.1.2. 新建云原生版集群	104
6.2. 变配和扩缩容	105
6.2.1. 社区兼容版变配和扩缩容	105
6.2.2. 云原生版变配	107
6.3. 社区兼容版重启实例	108
6.4. 云原生版重启计算组	109
6.5. 开通MySQL协议	109
6.6. 开通HTTPS协议	110
6.7. 申请外网地址	111
6.8. 获取集群的VPC网段IP	111
6.9. 连接数据库	112
6.9.1. DMS连接ClickHouse	112
6.9.2. ClickHouse命令行工具连接	114
6.9.3. 通过JDBC方式连接ClickHouse	115
6.9.4. 通过MySQL协议连接ClickHouse	117
6.9.5. 通过HTTPS协议连接ClickHouse	118
6.10. 账号与安全	120
6.10.1. 账号管理	120
6.10.2. 云盘加密	122
6.11. 备份与恢复	123
6.11.1. 数据备份	123
6.11.2. 数据恢复	125
6.12. 监控与报警	126
6.12.1. 集群监控	127
6.12.2. 报警规则设置	130
6.13. 查询管理	133

6.14. RDS MySQL分析实例	135
6.15. 操作审计日志	139
6.16. 冷热数据分层存储	144
6.17. 数据字典	148
7.SDK参考	151
7.1. SDK使用说明	151
8.常见问题	154
8.1. 常见问题	154
8.2. 如何处理ClickHouse超时问题	166
9.性能测试指南	169
10.管理规范	180
10.1. 安全白皮书	180
11.相关协议	182
11.1. 云数据库服务协议	182
11.2. 服务等级协议	182

1. 动态与公告

1.1. 版本发布记录

本文介绍云数据库ClickHouse的产品功能动态。

 说明 ClickHouse开源版的产品功能动态，请参见[Changelog](#)。

2022年7月

功能名称	描述	相关文档
新增存储类型	新特性 社区兼容版集群支持ESSD PL0云盘存储。	社区兼容版存储空间费用

2022年5月

功能名称	描述	相关文档
监控告警	<ul style="list-style-type: none">新特性 新增监控项：<ul style="list-style-type: none">Inactive Data Part个数TCP连接数运行Init Query个数HTTP连接数MaterializeMySQL个数失败Insert Query个数Kafka外表个数失败Select Query个数MaterializeMySQL同步失败个数Kafka外表消费发生错误个数优化 增强实例可用性和异常监控，提升运维稳定性。	集群监控

2022年4月

功能名称	描述	相关文档
备份恢复	新特性 2021年12月01日之前创建的20.8版本和21.8版本的集群支持数据备份和数据恢复。	<ul style="list-style-type: none">数据备份数据恢复

功能名称	描述	相关文档
新规格发布	新特性 社区兼容版集群新增80核384 GB规格。	无

2022年3月

功能名称	描述	相关文档
云原生版集群	<p>新特性</p> <p>云数据库ClickHouse推出云原生版集群。具体特性如下。</p> <ul style="list-style-type: none"> 云原生版集群为存储计算分离架构。 支持存储与计算独立变配。 变配效率是社区兼容版集群的10倍以上。 <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 5px; margin-top: 10px;"> <p>? 说明 云数据库ClickHouse云原生版目前处于公测阶段，仅支持在华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域创建云原生版集群。您可以点击申请试用。</p> </div>	<ul style="list-style-type: none"> 什么是云数据库ClickHouse 内核版本功能对比 云原生版快速入门 新建云原生版集群 云原生版变配

2022年1月

功能名称	描述	相关文档
HTTPS协议	<p>新特性</p> <p>支持HTTPS协议实现数据的加密传输。</p>	<ul style="list-style-type: none"> 开通HTTPS协议 通过HTTPS协议连接ClickHouse
MaterializedMySQL引擎升级	<ul style="list-style-type: none"> 新特性 兼容Enum脏数据。 优化 <ul style="list-style-type: none"> 优化全量同步，同步过程不锁表，从而减少业务影响。 优化DDL同步，同步出现的异常信息会写入系统表中。 	无
默认开启TCP的KeepAlive	<p>新特性</p> <p>默认开启TCP的KeepAlive，保证连接的可靠性。</p>	无

2021年12月

功能名称	描述	相关文档
MaterializedMySQL引擎升级	新特性 MaterializedMySQL引擎支持Decimal类型。	无

2021年11月

功能名称	描述	相关文档
21.8版本	<ul style="list-style-type: none"> 新特性 <ul style="list-style-type: none"> 支持更多SQL和JSON标准。 新增MaterializedPostgreSQL表引擎和数据库引擎，允许进行库和表的同步复制。 新增 <code>leftPad()</code>、<code>rightPad()</code>、<code>leftPadUTF8()</code> 和 <code>rightPadUTF8()</code> 函数。 新增 <code>bin</code> 和 <code>unbin</code> 函数。 在ADD INDEX命令中，新增 <code>FIRST</code> 关键字，将索引添加到索引列表的开头。 在 <code>MapAdd</code> 和 <code>MapSubtract</code> 函数中，新增Map和UInt128、Int128、UInt256、Int256类型。 支持 <code>DISTINCT ON (columns)</code> 表达式。 优化 <ul style="list-style-type: none"> 降低从表中读取多列数据的短查询延迟。 	无

2021年10月

功能名称	描述	相关文档
Binlog预留功能	MaterializeMySQL引擎支持通过配置参数 <code>enable_binlog_reserved</code> 开启Binlog预留功能，避免Binlog清理导致的无法正常同步数据的问题，从而提升MaterializeMySQL引擎的稳定性。	MaterializeMySQL引擎

2021年08月

功能名称	描述	相关文档
重启实例	云数据库ClickHouse支持重启实例。遇到实例不可用或者需要重启解决的问题时，用户可以自助完成重启。	社区兼容版重启实例
启用MySQL协议	云数据库ClickHouse支持启用MySQL协议。用户通过MySQL客户端也能访问云数据库ClickHouse。	开通MySQL协议
监控优化	提升页面流畅度，优化监控体验。	无

功能名称	描述	相关文档
RDS MySQL分析实例	云数据库ClickHouse支持通过RDS MySQL分析实例将数据同步到云数据库ClickHouse中，进行企业数据的集中分析。	RDS MySQL分析实例

2021年07月

功能名称	描述	相关文档
云实例迁移	控制台上发布云实例迁移功能，实现相同内核版本实例之间或低版本实例向高版本实例的数据迁移。	云数据库ClickHouse实例间的数据迁移

2021年05月

功能名称	描述	相关文档
支持选择扩容方式	水平扩容，新增一种扩容方式——简单扩容。用户进行水平扩容时，可以选择： <ul style="list-style-type: none"> • 迁移扩容 • 简单扩容 	水平扩容/水平缩容
MaterializedMySQL升级	<ul style="list-style-type: none"> • 完善对于JSON、TIME、YEAR数据类型的支持。 • 修复了key关键字、增量建表的bug。 • 支持两个配置参数： <ul style="list-style-type: none"> ◦ 支持skip_error_count ◦ 支持alter include_tables 	MaterializeMySQL引擎
水平扩缩容性能优化	优化clickhouse-copier，加快扩缩容效率。	社区兼容版变配和扩缩容

2021年03月

功能名称	描述	相关文档
备份恢复	发布备份恢复功能，避免数据意外丢失，提高数据安全性。	数据备份

2021年02月

功能名称	描述	相关文档
冷热数据分层	支持自动冷热数据迁移，保证性能前提下，存储成本降低80%以上。	冷热数据分层存储
支持ESSD PL2，ESSD PL3云盘	支持ESSD PL2，ESSD PL3云盘存储，IO吞吐效率提升数倍。	无

功能名称	描述	相关文档
云盘加密	基于块存储对整个数据盘进行加密，即使数据备份泄露也无法解密，保护数据安全。	云盘加密

2020年11月

功能名称	描述	相关文档
华北5（呼和浩特）和中国香港地区正式开服	无	无
日志投递链路优化发布	支持SLS日志数据投递到ClickHouse进行实时日志分析。	从SLS同步

2020年10月

功能名称	描述	相关文档
华北3（张家口）、西南（成都）、日本（东京）地区正式开服	无	无
发布控制台账号管理、数据库和数据字典权限管理功能	用户登录控制台进行新账号的创建和密码修改；支持对新用户和已有用户进行数据库和数据字典的权限修改。	账号管理
全局参数修改	用户登录产品控制台可直接完成全局参数修改和生效重启。	config.xml参数修改

2020年08月

功能名称	描述	相关文档
数据均衡扩容	保证增加集群节点时数据分布均匀，避免数据倾斜问题。	无
请求查询管理	请求查询管理功能，支持慢SQL定位和运行请求中断。	查询管理
新增4core 16G规格	新增4Core16G规格，提高单节点处理能力。	计费概述
数据字典	支持控制台配置数据字典，当业务中有常量数据或者业务数据可以划分为业务事实表和业务维度表，通过数据字典来替代维度表，直接查询字典获得维度数据，增加查询效率。	数据字典
资源队列	解决开源版中单个大查询消耗资源过多导致实例宕机问题。	ClickHouse资源队列

2020年07月

功能名称	描述	相关文档
接入Quick BI	支持在Quick BI中添加ClickHouse数据源并通过Quick BI构建BI系统。	Quick BI

2020年06月

功能名称	描述	相关文档
20.3版本	发布内核20.3.10.75版，修复多处内核bug，增强稳定性。	无
支持MySQL数据迁移和实时同步导入	支持MySQL数据迁移和同步导入，实现业务数据实时加速分析，实现业务实时交互式运营分析。	从RDS MySQL导入。
发布在线日志投递分析功能	企业用户只需要在云数据库ClickHouse中创建投递任务，将SLS中的用户行为日志数据在线投递到ClickHouse，即可基于云数据库ClickHouse在线进行多维日志数据分析。	从SLS同步

2020年05月

功能名称	描述	相关文档
接入数据管理服务DMS	云数据库ClickHouse接入数据管理服务DMS，支持在线库、表管理，在线数据查询，在线SQL任务编排调度。	无

2020年02月

功能名称	描述	相关文档
增加新规格	在此前发布的C8规格基础之上，新增了C4、C24、C64、C104四个新规格，同时开通了HTTP端口。	计费概述

2. 产品简介

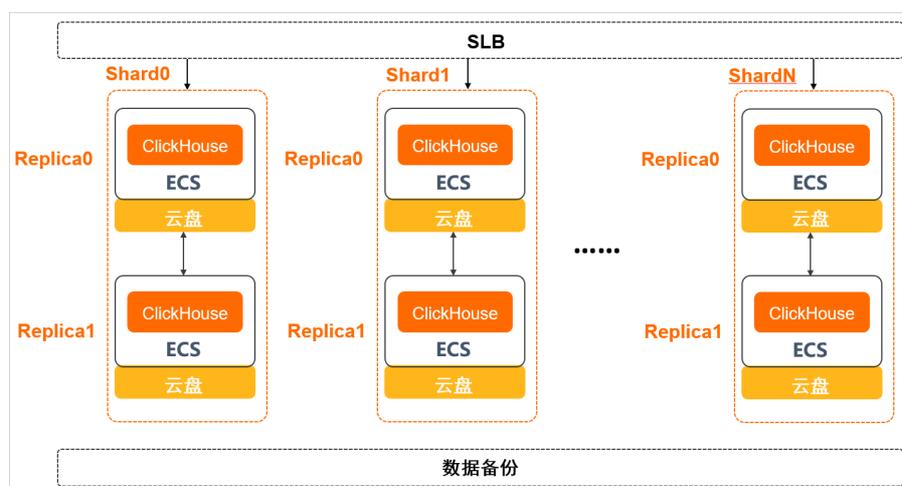
2.1. 什么是云数据库ClickHouse

云数据库ClickHouse是开源列式数据库管理系统ClickHouse在阿里云上的托管服务，用户可以在阿里云上便捷地购买云资源，搭建自己的ClickHouse集群。

产品架构

云数据库ClickHouse分为社区兼容版和云原生版，在开源版本基础上提升了稳定性、安全性和可运维性。

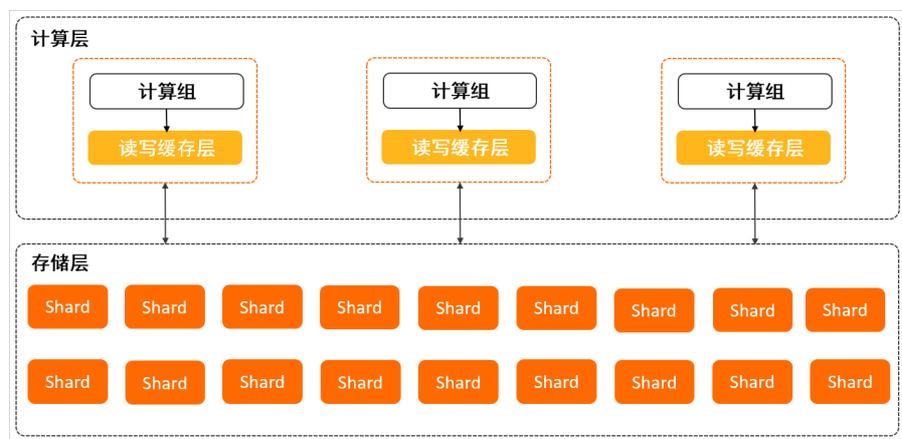
社区兼容版架构



社区兼容版计算与存储耦合，具体特点如下。

- 节点和节点完全对等，每一个节点都可以承载查询请求和写入请求，以及后台数据的计算和操作。
- 每个云数据库ClickHouse社区兼容版集群包含1个或多个分片（Shard），每个分片内部包含1个或多个副本（Replica）。
- 所有节点都部署在阿里云弹性计算服务器ECS之上，底层采用高可靠的云盘作为持久化存储介质。

云原生版架构



云原生版计算与存储分离，整体资源分为计算层和存储层，具体特点如下。

- 计算层包含计算组和读写缓存层，计算组支持变配，读写缓存层用于加速请求处理。

- 存储层解耦计算层，用于数据的持久化存储，数据按照逻辑分片（Shard）分布，根据用户需求按量收费，可以动态调整，不用提前设置存储空间。

② 说明 云数据库ClickHouse云原生版目前处于公测阶段，仅支持在华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域创建云原生版集群。您可以点击[申请试用](#)。

云原生版的更多信息，具体请参见[ClickHouse云原生版产品解析](#)。

产品特性

云数据库ClickHouse是面向联机分析处理的列式数据库，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。

主要特性如下。

- 数据压缩比高。
- 多核并行计算。
- 向量化计算引擎。
- 支持嵌套数据结构。
- 支持稀疏索引。
- 支持数据Insert和Update。

ClickHouse官网地址，请参见<https://clickhouse.yandex/>。

ClickHouse中文文档链接，请参见<https://clickhouse.yandex/docs/zh/>。

增强特性

云数据库ClickHouse主要在企业级特性方面做了如下增强。

- 增强安全性：增加用户管理、白名单等功能。
- 可视化运维：版本升级、故障切换等。
- 可扩展性：多规格选配、弹性扩容。
- 数据生态：打通云上各个数据源，方便数据同步和迁移。

参考资料

更多产品介绍，请参见[产品详情页](#)。

产品定价信息，请参见[云数据库ClickHouse购买页](#)。

2.2. 产品优势

本文介绍了云数据库ClickHouse的产品优势和特点。

核心能力

核心优势	描述
高可靠存储引擎	基于阿里飞天分布式系统的云盘存储，充分利用云平台存储可靠性特性。

核心优势	描述
扩展性强	提供数据库扩容管理模块，能便捷地按需对集群进行扩容，无需手动进行数据搬迁。
数据闭环流通	与云上日志系统、数据库、数据应用工具无缝集成，能便捷地进行数据同步，降低数据迁移工作量。
专业服务能力	阿里云提供专业的数据库专家，给用户的技术支持和服务。

产品特性

特性	功能介绍	详细描述
架构灵活	支持单节点、单副本、多节点、多副本多种架构	<p>节点数量：最低支持创建一个节点的集群，也支持创建若干个节点的集群。</p> <p>动态升降配：支持节点规格升降配，节点数可扩缩容，后台自动进行数据迁移，无需手动搬迁数据。</p> <p>副本数量：支持创建单副本集群，也支持创建多副本集群。</p>
数据安全	访问白名单和一键恢复，多层网络安全防护	<p>VPC私有网络：实例部署在利用OverLay技术在物理网络基础上构建的专有VPC虚拟网络上，在TCP层直接进行网络隔离保护。</p> <p>DDoS防护：在网络入口实时监测，当发现超大流量攻击时，对源IP进行清洗，清洗无效情况下可以直接将恶意IP拉进黑洞。</p> <p>IP白名单配置：支持配置访问IP白名单规则，直接从访问源进行风险控制。</p>
便捷运维	专业监控和数据库管理平台	<p>监控平台：提供CPU使用率、IOPS、连接数、磁盘空间等实例信息实时监控及报警，随时随地了解实例动态。</p> <p>可视化管理平台：管理控制平台对实例重启等操作可便捷的进行一键式操作。</p> <p>可视化DMS平台：专业的DMS数据管理平台，提供可视化的数据管理、数据查询工作台。</p> <p>数据库内核版本管理：提供数据库内核版本升级管理操作模块，降低内核版本管理成本。</p>

如需使用，请前往[购买页](#)开通集群[云数据库ClickHouse购买页](#)。

2.3. 内核版本功能对比

本文介绍云数据库ClickHouse不同内核版本的重要功能差异。

功能	社区兼容版				云原生版	文档链接
	19.15	20.3	20.8	21.8	21.8	
资源队列	☐	✔ [Ⓜ]	☐	☐	☐	ClickHouse资源队列
二级索引	☐	✔ [Ⓜ]	✔ [Ⓜ]	☐	☐	ClickHouse二级索引
备份恢复	☐	✔ [Ⓜ]	☐	✔ [Ⓜ]	☐	数据备份
冷热数据分层	☐	☐	✔ [Ⓜ]	✔ [Ⓜ]	☐	冷热数据分层存储
MaterializedMySQL引擎 (MySQL实时同步)	☐	☐	✔ [Ⓜ]	✔ [Ⓜ]	☐	MaterializeMySQL引擎
云数据库ClickHouse实例间的数据迁移 ❓ 说明 此功能只对目的实例的版本有限制。	☐	☐	✔ [Ⓜ]	✔ [Ⓜ]	☐	云数据库ClickHouse实例间的数据迁移
MaxCompute外表引擎	✔ [Ⓜ]	✔ [Ⓜ]	✔ [Ⓜ]	✔ [Ⓜ]	☐	<ul style="list-style-type: none"> • 从MaxCompute外表导入 (19.15.2.2) • 从MaxCompute外表导入 (20.3及以上)
OSS外表引擎	✔ [Ⓜ]	✔ [Ⓜ]	✔ [Ⓜ]	✔ [Ⓜ]	☐	从OSS导入

❓ 说明 云数据库ClickHouse云原生版目前处于公测阶段，仅支持在华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域创建云原生版集群。您可以点击[申请试用](#)。

2.4. 典型应用

云数据库ClickHouse应用场景介绍。

交互式报表

基于ClickHouse和BI工具构建实时运营监控报表

利用ClickHouse构建实时交互式报表，实时分析订单、收入、用户数等核心业务指标；构建用户来源分析系统，跟踪各渠道PV、UV来源。

海量数据实时多维查询

在数亿至数百亿记录规模大宽表，数百以上维度自由查询，响应时间通常在100毫秒以内。让业务人员能持续探索式查询分析，无需中断分析思路，便于深挖业务价值，具有非常好的查询体验。

用户画像分析

随着数据时代的发展，各行各业数据平台的体量越来越大，用户个性化运营的诉求也越来越突出，用户标签系统，做为个性化千人千面运营的基础服务，应运而生。如今，几乎所有行业（如互联网、游戏、教育等）都有实时精准营销的需求。通过系统生成用户画像，在营销时通过条件组合筛选用户，快速提取目标群体。

基于ClickHouse构建用户特征行为分析系统

利用ClickHouse对人群标签数据进行实时筛选并进行群体画像统计；自定义条件对海量明细日志记录进行过滤，分析用户行为。

用户分群统计

构建用户特征大宽表，任意选择用户属性标签数据和筛选条件，进行人群特征统计分析。

访客来源分析展示

通过批量离线计算对用户访问日志中的用户行为进行关联，生成用户行为路径大宽表同步到ClickHouse，基于ClickHouse构建交互式访客来源探索分析可视化系统。

2.5. 基本概念

本文将向您介绍云数据库ClickHouse产品中涉及的一些基本名词概念，以便于您更好地理解云数据库ClickHouse产品。

地域（Region）

购买云数据库ClickHouse的服务器所处地理位置。您需要在购买云数据库ClickHouse服务时指定Region，Region一旦指定后就不允许更改。

可用区（Zone）

同一地域下，电力、网络隔离的物理区域，可用区之间内网互通，可用区内网络延时更小。

ClickHouse集群（Cluster）

在物理构成上，ClickHouse集群是由多个ClickHouse Server实例组成的分布式数据库。这些ClickHouse Server根据购买规格的不同而可能包含1个或多个副本（Replica）、1个或多个分片（Shard）。在逻辑构成上，一个ClickHouse集群可以包含多个数据库（Database）对象。

副本配置（Edition）

ClickHouse集群包含如下副本。

- 双副本版：每个节点包含两个副本，某个副本服务不可用的时候，同一分片的另一个副本还可以继续服务。
- 单副本版：每个节点只有1个副本，该副本服务不可用时，会导致整个集群不可用，需要等待此副本完全恢复服务状态，集群才能继续提供稳定服务。

② 说明

- 双副本版的资源以及对应的购买成本都是单副本版的2倍。
- 由于底层云盘提供高可靠保证，即便是单副本版也能确保数据不会丢失。

分片（Shard）

在超大规模海量数据处理场景下，单台服务器的存储、计算资源会成为瓶颈。为了进一步提高效率，云数据库ClickHouse将海量数据分散存储到多台服务器上，每台服务器只存储和处理海量数据的一部分，在这种架构下，每台服务器被称为一个分片（Shard）。

副本（Replica）

为了在异常情况下保证数据的安全性和服务的高可用性，云数据库ClickHouse提供了副本机制，将单台服务器的数据冗余存储在2台或多台服务器上。

计算组

计算组由一个或者多个ECS资源构成，是云原生版集群的最小可见计算资源粒度。

数据库（Database）

数据库是云数据库ClickHouse集群中的最高级别对象，内部包含表（Table）、列（Column）、视图（View）、函数、数据类型等。

表（Table）

表是数据的组织形式，由多行、多列构成。

云数据库ClickHouse的表从数据分布上，可以分为本地表和分布式表两种类型。

类型	说明	区别
本地表（Local Table）	数据只会存储在当前写入的节点上，不会被分散到多台服务器上。	<ul style="list-style-type: none">本地表的写入和查询，受限于一台服务器的存储、计算资源，不具备横向拓展能力。分布式表的写入和查询，可以利用多台服务器的存储、计算资源，具有较好的横向拓展能力。
分布式表（Distributed Table）	本地表的集合，它将多个本地表抽象为一张统一的表，对外提供写入、查询功能。当写入分布式表时，数据会被自动分发到集合中的各个本地表中；当查询分布式表时，集合中的各个本地表都会被分别查询，并且把最终结果汇总后返回。	

云数据库ClickHouse的表从存储引擎上，可以分为单机表、复制表两种类型。

类型	说明	区别
单机表（Non-Replicated Table）	数据只会存储在当前服务器上，不会被复制到其他服务器，即只有一个副本。	<ul style="list-style-type: none">单机表在异常情况下无法保证服务高可用。复制表在至少有一个正常副本的情况下，能够对外提供服务。
复制表（Replicated Table）	数据会被自动复制到多台服务器上，形成多个副本。	

2.6. 使用限制

本文介绍云数据库ClickHouse在使用上的限制。

功能限制

- 高可用集群必须用复制表引擎。
- 默认所有Server都会自动组成名字为default的集群，DDL需要使用 `ON CLUSTER default` 语句在所有

Server上执行。

- 不支持用户自行配置remote_servers。
- 不支持File、URL表引擎。
- 不支持File、URL表函数。
- 不支持用户自定义profile。

对象命名规范限制

对象	命名规则	限制
数据库名	以小写字母开头，可包含字母、数字以及下划线（_），但不能包含连续两个及以上的下划线（_），长度不超过64个字符。	数据库名不能是system，system是内置数据库。
表名	以字母或下划线（_）开头，可包含字母、数字以及下划线（_），长度为1到127个字符。	表名不包含引号、感叹号（!）和空格。表名不能是SQL保留关键字。
列名	以字母或下划线（_）开头，可包含字母、数字以及下划线（_），长度为1到127个字符。	列名不包含引号、感叹号（!）和空格。列名不能是SQL保留关键字。
账号名	以小写字母开头，小写字母或数字结尾，可包含小写字母、数字以及下划线（_），长度为2到16个字符。	-
密码限制	包含大写字母、小写字母、数字以及特殊字符（!）、（@）、（#）、（\$）、（%）、（^）、（&）、（*）（()）、（_）、（+）、（-）、（=），每个密码至少包含其中三项（大写字母、小写字母、数字以及特殊字符），长度为8到32个字符。	-

购买及配置集群限制

描述	限制
每个阿里云账号可购买集群数量的最大值	30
每个集群可购买节点组的最大值	24
每个集群可创建数据库账号的最大值	500
每天集群扩容、缩容、变配的总次数	12
每个集群可创建数据库的最大值	256

3. 产品计费

3.1. 计费方式

本文介绍云数据库ClickHouse的两种计费方式以及如何变更计费方式。

产品的具体价格，请参见[计费概述](#)。

计费方式

云数据库ClickHouse支持包年包月和按量付费两种计费方式。

计费方式	描述
包年包月	<p>也称为预付费，即在购买集群时支付费用。适合长期需求，价格比按量付费更实惠，且购买时长越长，折扣越多。包年包月集群到期后前7天处于锁定状态，无法被访问。到期后第8天，集群的计算资源和存储数据被自动释放，无需手动释放。</p> <p>若您在合同期内退订包年包月的集群，具体请参见退款规则及退款流程。</p> <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 5px;"><p> 注意：包年包月集群无法变更为按量付费集群。</p></div>
按量付费	<p>也称为后付费，即每小时生成一个收费订单，并按照生成订单时的集群规格从阿里云账号扣费。适合短期需求，例如测试阶段，用完可立即删除集群，节省费用。待资源用量稳定后，可将按量付费变更为包年包月套餐。</p>

按量付费转包年包月

1. 登录[云数据库ClickHouse控制台](#)。
2. 在集群列表页面，单击目标集群操作列的[按量转包月](#)。
3. 在[按量付费转包年包月](#)页面，设置购买时长，选中《[ClickHouse包年包月服务协议](#)》。
4. 单击[立即购买](#)根据提示完成支付。

3.2. 计费概述

通过阅读本文，您可以快速了解云数据库ClickHouse的计费项及计费规则。

计费项

 **说明** 云数据库ClickHouse云原生版目前处于公测阶段，仅支持在华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域创建云原生版集群。您可以点击[申请试用](#)。

云数据库ClickHouse包含如下计费项。

集群类型	计费项	计费规则
	计算节点	集群计算节点的资源费用。

集群类型	计费项	计费规则
社区兼容版	存储空间	<p>集群存储空间的资源费用。云数据库ClickHouse提供热数据盘和冷数据盘两种存储类型。</p> <ul style="list-style-type: none"> 热数据盘：创建集群时所选的ESSD云盘或高效云盘。 冷数据盘：需要先开通冷热分层存储功能。
	备份空间	<p>集群的备份费用。云数据库ClickHouse提供免费备份空间，超出免费备份空间的部分依据备份空间单价按量按小时进行收费。</p> <p>备份空间费用=(备份空间使用量-免费备份空间)×备份空间单价×使用时长。</p>
云原生版	计算组	<p>集群计算组的资源费用。</p> <p>计算组费用=计算组规格单价×使用时长。</p>
	缓存空间	<p>集群每个计算组的缓存空间资源费用。</p> <p>缓存空间费用=缓存单价×缓存空间×使用时长。</p>
	存储空间	<p>集群存储的资源费用，按小时出帐单，并推送使用量和计量数据，无需提前购买存储资源。</p> <p>存储费用=存储单价×存储空间×使用时长。</p>

折扣优惠方案

方案分类	方案名称	优惠条件描述	限制条件	活动周期
通用优惠	包1年85折	长期优惠，针对所有用户，新购、续费、升级均支持。	不限	长期
	包2年7折			
	包3年5折			
首购优惠	新购首月10元	针对第一次预付费购买集群的用户，第一个月计费10元。	限购1个集群，配置如下： <ul style="list-style-type: none"> 副本配置：单副本版。 规格：4核16 GB或8核32 GB。 节点数量：1。 存储空间：100 GB。 	长期
	首购转包月7折	针对第一次预付费购买集群的用户，续费、升级月付7折。		
	首购转包年6折	针对第一次预付费购买集群的用户，续费、升级1年付6折。		

相关文档

- [社区兼容版计算节点费用](#)
- [社区兼容版存储空间费用](#)
- [社区兼容版备份空间费用](#)
- [云原生版费用](#)

3.3. 规格及定价

3.3.1. 社区兼容版计算节点费用

本文介绍云数据库ClickHouse社区兼容版集群计算节点的费用。

计算节点费用

 说明 本文的费用仅供参考，实际费用请以控制台为准。

副本配置	节点规格	按量付费 (元/小时)	包年包月 (元/月)
双副本版	4核16 GB	2.8823	1402.0000
	8核32 GB	5.6823	2746.8000
	16核64 GB	11.2810	5434.8000
	32核128 GB	22.4823	10810.8000
	64核256 GB	51.4063	22906.80
	80核384 GB	61.6823	29626.8000
	104核384 GB	72.8823	35002.8000
单副本版	4核16 GB	1.4412	701.0000
	8核32 GB	2.8412	1373.4000
	16核64 GB	5.6412	2717.4000
	32核128 GB	11.2412	5405.4000
	64核256 GB	24.6812	11856.6000
	80核384 GB	30.8412	14813.4000
	104核384 GB	36.4412	17501.4000

3.3.2. 社区兼容版存储空间费用

云数据库ClickHouse提供热数据盘和冷数据盘两种存储类型，本文介绍不同存储类型的费用。

 说明 本文的费用仅供参考，实际费用请以控制台为准。

背景知识

- 热数据盘：创建集群时所选的ESSD云盘或高效云盘，存储访问频次较高的数据，满足高性能的访问需求。

- 冷数据盘：需要先开通冷热分层存储功能，存储访问频次较低的数据，满足高性价比的存储需求。

 说明 如何开通冷热分层存储功能，请参见[开通冷热分层存储功能](#)。

存储空间费用

双副本版

存储类型	分类	按量付费（元/GB/小时）	包年包月（元/GB/月）
热数据盘	ESSD PL0云盘	0.0014	1
	ESSD PL1云盘	0.0028	2
	ESSD PL2云盘	0.0084	4
	ESSD PL3云盘	0.0168	8
	高效云盘	0.00098	0.7
冷数据盘	无	0.0002	不支持包年包月

单副本版

存储类型	分类	按量付费（元/GB/小时）	包年包月（元/GB/月）
热数据盘	ESSD PL0云盘	0.0007	0.5
	ESSD PL1云盘	0.0014	1
	ESSD PL2云盘	0.0042	2
	ESSD PL3云盘	0.0084	4
	高效云盘	0.00049	0.35
冷数据盘	无	0.0002	不支持包年包月

3.3.3. 社区兼容版备份空间费用

本文介绍云数据库ClickHouse社区兼容版集群的备份空间费用。

 说明 本文的费用仅供参考，实际费用请以控制台为准。

备份空间费用

- 计费规则：云数据库ClickHouse提供了免费备份空间，免费备份空间=100%×集群存储空间。超出免费备份空间后，超出部分依据备份空间单价按量按小时进行收费。
- 计费公式：备份空间费用=（备份空间使用量-免费备份空间）×备份空间单价×使用时长。
- 备份空间单价：0.0002元/GB/小时。

- 示例：
 - 备份空间使用量不超过免费备份空间：
若免费备份空间为100 GB，备份空间使用量为50 GB，则本次备份数据免费。
 - 备份空间使用量超过免费备份空间：
以购买华东1（杭州）社区兼容版集群为例，免费备份空间为50 GB，备份空间使用量为100 GB，使用时长为1小时，则本次备份空间费用=（100-50）x0.0002x1=0.01元。

3.3.4. 云原生版费用

云数据库ClickHouse的云原生版集群收取计算组、缓存空间和存储空间费用，本文介绍不同计费项的费用。

注意事项

- 云数据库ClickHouse云原生版目前处于公测阶段，仅支持在华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域创建云原生版集群。您可以点击[申请试用](#)。
- 本文的费用仅供参考，实际费用请以控制台为准。

计算组费用

- 计费公式：计算组费用=计算组规格单价×计算组核数×使用时长。
- 计算组规格单价：
 - 按量付费：0.37元/核/小时
 - 包年包月：175.00元/核/月
- 示例：
 - 购买按量付费云原生版集群，使用时长为1小时，计算组规格为8核32 GB，则计算组费用=0.37x8x1=2.96元。
 - 购买包年包月云原生版集群，使用时长为1个月，计算组规格为8核32 GB，则计算组费用=175.00x8x1=1400.00元。

缓存空间费用

- 计费公式：缓存空间费用=缓存空间单价×缓存空间×使用时长。
- 缓存空间单价：
 - 按量付费：0.00210元/GB/小时
 - 包年包月：1.00元/GB/月
- 示例：
 - 购买按量付费云原生版集群，使用时长为1小时，缓存空间为200 GB，则缓存空间费用=0.00210x200x1=0.42元。
 - 购买包年包月云原生版集群，使用时长为1个月，缓存空间为200 GB，则缓存空间费用=1.00x200x1=200.00元。

存储空间费用

存储空间费用无需提前支付，在购买云原生版集群后，根据实际使用的存储空间量，按小时出帐单。

- 计费公式：存储空间费用=存储空间单价×存储空间×使用时长。
- 存储空间单价：0.00032元/GB/小时。

- 示例：

购买云原生版集群后，使用时长为1小时，实际使用的存储空间为100 GB，则存储空间费用 = $0.00032 \times 100 \times 1 = 0.032$ 元。

3.4. 变更配置计费说明

云数据库ClickHouse扩容、缩容集群时需遵循以下规则。

计费方式	变配说明
包年包月	<ul style="list-style-type: none"> • 合同期内变配 <p>扩容集群所需的费用 = (集群扩容后每天的价格 - 集群扩容前每天的价格) * 集群扩容当天到服务到期日的剩余天数。</p> <ul style="list-style-type: none"> ◦ 若扩容当天至服务到期日的剩余天数小于365天，则扩容后集群每天的价格为包月价格。 ◦ 若扩容当天至服务到期日的剩余天数大于等于365天，则扩容后集群每天的价格为包年价格。 • 到期时续费并变配 <p>按照新集群配置和包年或包月的时间进行计费。</p>
按量付费	集群变配后，计费规则不变，仍然是每小时扣费一次，按扣费时的集群配置计费。

3.5. 到期或欠费的影响

本文向您介绍集群到期或欠费后将集群中的资源产生哪些影响。

集群类型	集群状态
包年包月集群	<ul style="list-style-type: none"> • 集群到期后，集群处于锁定状态，无法被访问。 • 集群锁定后第16天，集群的计算资源被释放，数据不再保留。
按量付费集群	<ul style="list-style-type: none"> • 阿里云账号欠费后，该账号下所有按量付费集群将会变成欠费状态。 <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 5px; margin: 5px 0;"> <p>? 说明 如果未启用延期免停，欠费后集群将进入锁定状态。如已开启延期免停，集群会延期停服，具体延期情况请参见用户中心。</p> </div> <ul style="list-style-type: none"> • 欠费后第1天到第15天，集群处于锁定状态，无法被访问。 • 集群锁定后第16天，集群的计算资源被释放，数据不再保留。

以上所有动作都会通过短信、邮件和站内信的方式提醒您。如何配置提醒方式请参见[消息接收管理](#)。

3.6. 续费

续费是指延长包年包月集群的使用时间，续费的费用和新购集群费用相同。按量付费集群没有到期时间，无需续费，您只需保证阿里云账号的余额充足即可。

 **说明** 建议您开通自动续费，以免忘记续费而导致业务中断。

注意事项

- 若您在自动扣款日期前进行了手动续费，则系统将在下一次到期前进行自动续费。
- 自动续费功能于次日生效。若您的集群于次日到期，为避免业务中断，请手动续费。

创建集群时开通自动续费

您可在[购买包年包月集群](#)时，勾选到期自动续费。

 **说明** 按量付费集群没有到期时间，不涉及续费操作。开通自动续费后，系统将根据您选择的续费周期进行自动续费。例如，如果您选择了3个月的续费周期，则每次自动续费时会缴纳3个月的费用。

云数据库ClickHouse控制台续费

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在集群列表中，单击目标集群右侧的续费。
4. 在续费页面，设置购买时长。时间越长，折扣越多。
5. 勾选《ClickHouse包年包月服务协议》，单击立即支付，完成支付即可。

续费管理控制台手动续费

在续费管理控制台，可以实现对单个或多个集群开通手动续费、自动续费等功能。

1. 登录[云数据库ClickHouse控制台](#)。
2. 在控制台右上方，选择费用 > 续费管理。



<input checked="" type="checkbox"/>	产品	实例ID/实例名称	地域	实例状态	计费时	付费方式	开始/结束时间	操作
<input checked="" type="checkbox"/>	ClickHouse包年包月	实例ID/实例名称	华东1 (杭州)	运行中	31天	包年包月	2021-10-14 09:53:02 2021-11-15 00:00:00	续费 开通自动续费 不续费

 **说明** 包年包月集群开通自动续费后，系统将自动以选择的续费时长进行续费。例如，选择3个月的续费时长，则每次自动续费时会缴纳3个月的费用。

○ 单击手动续费页签

在目标实例右侧单击对应操作项，可实现对单个实例的续费、开通自动续费和续费操作。

勾选多个目标实例，在页面下方单击对应操作项，可实现对多个实例的批量续费、开通自动续费、设置为不续费和导出续费清单操作。

○ 单击自动续费页签

在目标实例右侧单击对应操作项，可实现对单个实例的续费、修改自动续费、不续费和恢复手动续费操作。

勾选多个目标实例，在页面下方单击对应操作项，可实现对多个实例的批量续费、修改自动续费、恢复为手动续费、设置为不续费和导出续费清单操作。

○ 单击到期不续费页签

在目标实例右侧单击对应操作项，可实现对单个实例的续费、开通自动续费和恢复手动续费操作。

勾选多个目标实例，在页面下方单击对应操作项，可实现对多个实例的批量续费、开通自动续费、恢复手动续费和导出续费清单操作。

3.7. 查看消费明细

您可以通过阿里云管理控制台查看云数据库ClickHouse的收费明细。

操作步骤

1. 登录[阿里云管理控制台](#)。
2. 在页面右上方，选择费用 > 费用账单。
3. 在费用账单页面，可以通过以下方式查看消费详情。
 - **账单总览**：查看最近12个月的云产品消费概览，即每个产品一条消费汇总信息。
 - **账单**：按照订单号/账单号查看最近12个月的云产品消费流水信息，包括每一笔订单和每个计费周期的账单消费信息。
 - **账单明细**：查看最近12个月的云产品详细消费信息，例如商品的计费项、用量、价格以及资源包抵扣等。

3.8. 依赖服务计费说明

云数据库ClickHouse部分功能的开通和使用需要依赖其他云服务，例如ARMS（Application Real-Time Monitoring Service）、CLB（Classic Load Balancer）和ENI（Elastic Network Interface）。本文介绍所有依赖服务及其计费说明。

说明

- 2021年9月30日之后创建的云数据库ClickHouse集群，其部分功能的开通和使用需要依赖于其他云服务。在此之前创建的云数据库ClickHouse集群不依赖于其他云服务。
- 云数据库ClickHouse集群释放后，对应的依赖服务会自动停止计费。

依赖服务	支持功能	计费说明
ARMS	监控告警	<ul style="list-style-type: none"> ● 监控：依赖于Prometheus监控，开通Prometheus监控服务后，您可以在云数据库ClickHouse控制台中免费查看15天的监控数据，超过15天自动清除。 ● 告警：依赖于ARMS告警管理服务，ARMS会根据开通的服务类型进行计费。详细的计费规则，请参见ARMS告警管理计费规则。
CLB	开通VPC网络或申请外网	<ul style="list-style-type: none"> ● 开通VPC网络：依赖于CLB，CLB会按照规格1：简约型I（slb.s1.small）进行计费，计费项为规格费。 ● 申请外网：依赖于CLB，CLB会按照规格1：简约型I（slb.s1.small）进行计费，计费项为实例费、流量费和规格费。 <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>说明 CLB的计费规则，请参见CLB收费策略。</p> </div>

依赖服务	支持功能	计费说明
ENI	通过ENI开通网络	免费，自动开通。

4.快速入门

4.1. 总体流程

快速入门介绍购买并使用云数据库ClickHouse集群的完整过程，旨在帮助新用户快速使用云数据库ClickHouse。

本教程以购买并使用云原生版集群为例，结束后，您将得到一个数据表ontime_local，并可以使用该表进行查询分析。

 **说明** 云数据库ClickHouse云原生版目前处于公测阶段，仅支持在华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域创建云原生版集群。您可以点击[申请试用](#)。

在本教程中，您将进行以下操作。



- 准备权限**：RAM用户按照本教程使用云数据库ClickHouse前，需要具备AliyunClickHouseFullAccess权限，介绍如何使用阿里云账号为RAM用户授权。
- 新建集群**：以新建包年包月的云原生版集群为例，介绍如何新建云数据库ClickHouse集群。
- 创建账号**：以创建权限类型为超级账户的账号为例，介绍如何创建云数据库ClickHouse集群的数据库账号。
- 设置白名单**：介绍如何设置云数据库ClickHouse集群的白名单，以允许外部设备访问。
- 连接集群**：以通过数据管理服务DMS（Data Management Service）连接云数据库ClickHouse集群为例，介绍如何连接云数据库ClickHouse集群。
- 创建数据库**：介绍如何使用DMS为云数据库ClickHouse集群创建数据库。
- 创建表**：以创建与On Time数据集类型对应的表为例，介绍如何使用DMS在已创建的数据库中创建本地表和分布式表。
- 导入数据**：以导入On Time数据集至已创建的分布式表为例，介绍如何使用客户端clickhouse-client工具导入数据到云数据库ClickHouse。

4.2. 准备权限

RAM用户按照快速入门教程在云数据库ClickHouse控制台新建集群、创建账号等，需要具备AliyunClickHouseFullAccess权限。本文介绍如何使用阿里云账号为RAM用户授权。如果您使用主账号登录使用云数据库ClickHouse，可跳过本文档。

前提条件

已注册阿里云账号。若尚未注册，请前往[阿里云官网](#)进行注册。

操作步骤

- 使用阿里云账号登录[RAM控制台](#)。
- 在左侧导航栏，选择身份管理 > 用户。
- 在用户页面，单击目标RAM用户操作列的添加权限。
- 在添加权限面板，为RAM用户添加权限，参数说明如下。

参数	说明	示例
授权范围	<ul style="list-style-type: none">◦ 整个云账号：权限在当前阿里云账号内生效。◦ 指定资源组：权限在指定的资源组内生效。	整个云账号
授权主体	需要授权的RAM用户，系统会自动填入当前的RAM用户，您也可以添加其他RAM用户。	test@cores.onalilyun.com
选择权限	<p>权限策略分为系统策略和自定义策略。</p> <ul style="list-style-type: none">◦ 系统策略：阿里云提供多种具有不同管理目的的默认权限策略。云数据库ClickHouse使用的系统策略如下。<ul style="list-style-type: none">■ AliyunClickHouseFullAccess：管理云数据库ClickHouse的权限，包括对云数据库ClickHouse中所有资源的所有操作权限。■ AliyunClickHouseReadOnlyAccess：云数据库ClickHouse资源只读访问的权限，支持查看云数据库ClickHouse集群列表，查看数据库账号等权限。◦ 自定义策略：需要您精准地设计权限策略，适用于熟悉阿里云各种云服务API以及具有精细化控制需求的用户。 <p> 说明 每次最多绑定5条策略，如需绑定更多策略，请分次操作。</p>	AliyunClickHouseFullAccess

5. 单击**确定**。
6. 单击**完成**。

下一步

新建集群

4.3. 新建集群

本文介绍如何新建云数据库ClickHouse云原生版集群。

操作步骤

 **说明** 云数据库ClickHouse云原生版目前处于公测阶段，仅支持在华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域创建云原生版集群。您可以点击[申请试用](#)。

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在**集群列表**页面，选择**云原生版实例列表**。
4. 单击**新建集群**。
5. 选择**商品类型**，本文以创建**包年包月**集群为例。

商品类型	适用场景
包年包月	预付费，即在购买集群时支付费用。适合长期需求，价格比按量付费更实惠，且购买时长越长，折扣越多。
按量付费	后付费，即每小时生成一个收费订单，并按照生成订单时的集群规格从阿里云账号扣费。适合短期需求，例如测试阶段，用完可立即删除集群，节省费用。待资源用量稳定后，可将按量付费变更为包年包月套餐。

6. 根据页面提示，配置如下参数。

参数	说明	示例
地域和可用区	集群所在的地理位置，购买后无法更换地域。建议根据业务所在的地理位置就近选择地域，从而提升访问速度。	中国，华东1（杭州）
可用区	同一地域内，电力和网络互相独立的物理区域，同一地域不同可用区之间没有实质性区别。	杭州可用区H
网络类型	目前仅支持专有网络，也称为VPC（Virtual Private Cloud）。VPC是一种隔离的网络环境，安全性较高。	专有网络
专有网络（VPC）	专有网络VPC。如果没有可用的VPC，请前往 专有网络管理控制台 创建。	vpc-bp175iuvg8nxqraf2****
专有网络交换机	选择专有网络下的交换机。如果没有可用的交换机，请前往 专有网络管理控制台 创建。	vsw-bp1gz31twhlo0sa5****
集群类型	云数据库ClickHouse支持的集群类型。	云原生版
内核版本	对应的开源社区版本号。	21.8
副本配置	云原生版仅支持单副本版。	单副本版
实例CPU	单个计算组的CPU。	8核
实例内存大小	单个计算组的内存大小。	32 GB
实例缓存类型	存储的云盘类型，ESSD云盘的应用场景如下。 <ul style="list-style-type: none"> 大型OLTP数据库：如MySQL、PostgreSQL、Oracle、SQL Server等关系型数据库。 NoSQL数据库：如MongoDB、HBase、Cassandra等非关系型数据库。 ElasticSearch分布式日志：ELK（Elasticsearch、Logstash和Kibana）日志分析等。 	ESSD PL1（云盘）
实例缓存大小	单个计算组的存储空间。	4800 GB
购买时长	商品类型选择为包年包月时，需配置购买时长。	1个月

参数	说明	示例
到期自动续费	商品类型选择为包年包月时，可根据需要勾选到期是否自动续费。如果创建集群时，您没有勾选到期自动续费，可以在续费管理控制台进行自动或手动续费。更多信息，请参见 续费 。	勾选到期自动续费

7. 单击**立即购买**。
8. 在**确认订单**页面，查看配置详情，并勾选**服务协议**，单击**去支付**。
9. 在**支付**页面，确认订单信息，单击**订购**。

支付成功后，您可以在**集群列表**页面单击**云原生版实例列表**，选择集群所属地域，查看新建的云数据库ClickHouse云原生版集群。

下一步

创建账号

4.4. 创建账号

要使用云数据库ClickHouse云原生版集群，首先需要有一个数据库账号。通过云数据库ClickHouse控制台，您可以创建不同权限的账号。本文以创建权限类型为超级账户的账号为例，介绍创建账号的完整流程。

使用限制

单个云数据库ClickHouse集群支持创建500个数据库账号。

前提条件

已新建云数据库ClickHouse云原生版集群，且集群的状态为运行中。如何新建，请参见[新建集群](#)。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在**集群列表**页面，选择**云原生版实例列表**，单击目标集群ID。
4. 在左侧导航栏单击**账号管理**。
5. 单击右上角的**创建账号**。
6. 根据页面提示，配置如下参数。本文以创建数据库账号test为例。

参数	描述	示例
数据库账号	数据库账号，设置规则如下。 <ul style="list-style-type: none">名称唯一。由小写字母、数字或下划线组成。以小写字母开头，以小写字母或数字结尾。长度为2~64个字符。	test

参数	描述	示例
密码	数据库账号的密码，设置规则如下。 <ul style="list-style-type: none"> 由大写字母、小写字母、数字和特殊字符中的至少三种组成。 特殊字符为!@#\$\$%^&*()_+== 长度为8~32个字符。 	123456Aa
确认密码	再次输入相同的密码。	123456Aa
备注说明	数据库账号的描述，设置规则如下。 <ul style="list-style-type: none"> 不能以http://或https://开头。 以中文、英文字母开头。 长度为2~256个字符。 	test
权限类型	数据库账号的权限类型。	超级账户

7. 单击**确定**。

创建成功后，您可以在**账号管理**页面查看创建的数据库账号。

下一步

设置白名单

4.5. 设置白名单

为保障云数据库ClickHouse数据库的安全稳定，系统默认禁止所有IP地址访问云数据库ClickHouse集群。在开始使用云数据库ClickHouse集群前，您需要将客户端的IP地址或IP地址段添加到云数据库ClickHouse的白名单中。本文介绍如何设置白名单。

注意事项

- 设置白名单可以让云数据库ClickHouse集群得到高级别的访问安全保护，建议您定期维护白名单。
- 设置白名单不会影响云数据库ClickHouse集群的正常运行。
- 为了确保数据安全，云数据库ClickHouse禁止设置白名单为0.0.0.0或0.0.0.0/0。
- 默认白名单分组（default）不能删除，只能修改和清空。
- 默认白名单只包含127.0.0.1，禁止所有IP地址访问云数据库ClickHouse集群。
- 请勿修改或删除系统自动生成的分组，避免影响相关产品的使用。例如ali_dms_group为DMS（Data Management Service）产品的IP地址白名单分组。

前提条件

已新建云数据库ClickHouse集群，且集群的状态为运行中。详细信息，请参见[新建集群](#)。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在**集群列表**页面，选择云原生版实例列表，单击目标集群ID。

4. 在左侧导航栏单击**数据安全**。
5. 单击**添加白名单分组**。
6. 根据页面提示，配置如下参数。

参数	描述	示例
分组名称	白名单分组名称，设置规则如下。 <ul style="list-style-type: none"> ◦ 由小写字母、数字或下划线组成。 ◦ 以小写字母开头，以小写字母或数字结尾。 ◦ 长度为2~32个字符。 	test
组内白名单	白名单，设置规则如下。 <ul style="list-style-type: none"> ◦ IP地址形式，例如192.168.0.1表示允许192.168.0.1的IP地址访问云数据库ClickHouse。 ◦ IP地址段形式，例如192.168.0.0/24表示允许从192.168.0.1到192.168.0.255的IP地址访问云数据库ClickHouse。 <div style="background-color: #e6f2ff; padding: 10px; margin-top: 10px;"> <p>说明</p> <ul style="list-style-type: none"> ◦ 若您需要添加多个IP地址或IP地址段，请用英文逗号隔开。 ◦ 设置为127.0.0.1代表禁止所有IP地址访问云数据库ClickHouse集群。 ◦ 为了确保数据安全，禁止设置白名单为0.0.0.0或0.0.0.0/0。 </div>	192.168.xx.xx

快速入门教程以通过DMS创建数据库和表并使用clickhouse-client导入数据为示例，所以本教程中，需将DMS和clickhouse-client所在服务器的IP地址添加到云数据库ClickHouse集群的白名单中。

说明 新建云数据库ClickHouse集群时，系统会自动为云数据库ClickHouse集群添加一个名为ali_dms_group的白名单分组，并将DMS服务器的IP地址加入该分组中，若自动添加失败需手动添加。DMS服务器不同地域的IP地址列表，具体请参见[DMS IP地址列表](#)。

7. 单击**确定**。
添加成功后，您可以在**数据安全**页面查看添加的白名单分组。

下一步

[连接集群](#)

4.6. 连接集群

云数据库ClickHouse支持通过DMS（Data Management Service）、ClickHouse命令行工具或其它客户端（DataGrip、DBeaver）连接云数据库ClickHouse集群。本文介绍如何通过DMS连接云数据库ClickHouse集群。

背景信息

DMS是阿里云提供的图形化数据管理工具，可用于管理关系型数据库和NoSQL数据库，支持数据管理、SQL操作、数据方案（数据导入/导出、数据库克隆等）、性能与优化、安全审计等功能。您可以使用DMS实现易用的数据库管理，在线进行云数据库ClickHouse库、表创建及在线数据查询，让数据更安全、管理更高效、数据价值更清晰。

前提条件

已完成快速入门的如下步骤。

- [新建集群](#)
- [创建账号](#)
- [设置白名单](#)

 **说明** 新建云数据库ClickHouse集群时，系统会自动为云数据库ClickHouse云原生版集群添加一个名为ali_dms_group的白名单分组，并将DMS服务器的IP地址加入该分组中，若自动添加失败需手动添加。DMS服务器不同地域的IP地址列表，具体请参见[DMS IP地址列表](#)。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在集群列表页面，选择云原生版实例列表，单击目标集群ID。
4. 在集群信息页面，单击集群计算组操作列的登录。
5. 根据页面提示，配置如下参数。

参数	说明	示例
数据库类型	默认为ClickHouse，云数据库ClickHouse集群的类型，无需修改。	ClickHouse
实例地区	默认为当前云数据库ClickHouse集群所在的地域，无需修改。  说明 若您需要登录其他地域下的云数据库ClickHouse集群，从下拉列表中选择目标集群所在的地域即可。	华东1（杭州）
实例ID	默认为当前云数据库ClickHouse集群的集群ID，无需修改。  说明 若您需要登录其他云数据库ClickHouse集群，从下拉列表中选择目标集群ID即可。	cc-bp16qwvp7hy8i*** *
数据库账号	数据库的登录账号。	test
数据库密码	数据库账号对应的密码。	123456Aa

参数	说明	示例
记住密码	勾选 记住密码 ，下次登录此云数据库ClickHouse集群时，无需重复输入账号密码即可自动登录。	勾选记住密码

② 说明

- 首次通过DMS登录云数据库ClickHouse集群时，管控模式默认为**自由操作**。登录成功后，您可以通过编辑实例功能来修改管控模式。更多信息，请参见[编辑实例](#)和[管控模式](#)。
- 配置完登录参数后，您可以单击左下角**测试连接**。如果测试连接失败，请按照报错提示检查录入的集群信息。

6. 单击**登录**。

下一步

[创建数据库](#)

相关文档

如果您需要使用其他方式连接云数据库ClickHouse云原生版集群，请参见如下文档。

- [ClickHouse命令行工具连接](#)
- [通过JDBC方式连接ClickHouse](#)
- [通过MySQL协议连接ClickHouse](#)

4.7. 创建数据库

在进行数据写入和查询前，您需要先创建数据库。本文介绍如何使用DMS（Data Management Service）创建数据库。

使用限制

每个云数据库ClickHouse集群最多可以创建256个数据库。

前提条件

已完成快速入门的如下步骤。

- [新建集群](#)
- [创建账号](#)
- [设置白名单](#)

② **说明** 新建云数据库ClickHouse集群时，系统会自动为云数据库ClickHouse集群添加一个名为**dms**的白名单分组，并将DMS服务器的IP地址加入该分组中，若自动添加失败需手动添加。DMS服务器不同地域的IP地址列表，具体请参见[DMS IP地址列表](#)。

- [连接集群](#)

操作步骤

1. 在[数据管理DMS控制台](#)的SQL Console页面中，输入CREATE DATABASE语句创建数据库。

- 语法：

```
CREATE DATABASE [IF NOT EXISTS] db [ON CLUSTER default]
```

- 参数说明：

`db_name`：数据库名。以小写字母开头，可包含字母、数字以及下划线（_），但不能包含连续两个及以上的下划线（_），长度不超过64个字符。

 **说明** 数据库名不能是system，system是内置数据库。更多参数信息，请参见[Create DataBase](#)。

- 示例：

```
create database if not exists clickhouse_demo ON CLUSTER default;
```

- 单击左上角的执行，数据库创建成功。

下一步

[创建表](#)

4.8. 创建表

本文介绍如何使用DMS（Data Management Service）在云数据库ClickHouse中创建表。

前提条件

已完成快速入门的如下步骤。

- [新建集群](#)
- [创建账号](#)
- [设置白名单](#)
- [连接集群](#)
- [创建数据库](#)

操作步骤

- 在[数据管理DMS控制台](#)的SQL Console页面中，输入 `CREATE TABLE` 语句创建本地表。

- 语法：

```
CREATE TABLE [IF NOT EXISTS] [db.]local_table_name ON CLUSTER cluster
(
  name1 [type1] [DEFAULT|MATERIALIZED|ALIAS expr1],
  name2 [type2] [DEFAULT|MATERIALIZED|ALIAS expr2],
  ...
  INDEX index_name1 expr1 TYPE type1(...) GRANULARITY value1,
  INDEX index_name2 expr2 TYPE type2(...) GRANULARITY value2
) ENGINE = engine_name()
[PARTITION BY expr]
[ORDER BY expr]
[PRIMARY KEY expr]
[SAMPLE BY expr]
[SETTINGS name=value, ...];
```

● 参数说明：

参数	说明
db	指定数据库的名称，默认为当前选择的数据库。
local_table_name	本地表名。
cluster	指定集群的名称，固定为default。
name1,name2	列名。
type1,type2	列类型。
engine_name	表引擎类型，具体请参见 表引擎 。 ❓ 说明 单副本版为MergeTree，双副本版为ReplicatedMergeTree。
PARTITION BY	指定分区键。

参数	说明
ORDER BY	指定排序键。
PRIMARY KEY	指定主键。
SAMPLE BY	抽样表达式。如果要使用抽样表达式，主键中必须包含这个表达式。
SETTINGS	影响性能的其他参数。

? 说明

更多参数说明，请参见[Create Table](#)。

● 示例：

```
CREATE TABLE clickhouse_demo.ontime_local ON CLUSTER default
(
  `Year` UInt16,
  `Quarter` UInt8,
  `Month` UInt8,
  `DayOfMonth` UInt8,
  `DayOfWeek` UInt8,
  `FlightDate` Date,
  `Reporting_Airline` String,
  `DOT_ID_Reporting_Airline` Int32,
  `IATA_CODE_Reporting_Airline` String,
  `Tail_Number` String,
  `Flight_Number_Reporting_Airline` String,
  `OriginAirportID` Int32,
  `OriginAirportSeqID` Int32,
  `OriginCityMarketID` Int32,
  `Origin` FixedString(5),
  `OriginCityName` String,
  `OriginState` FixedString(2),
  `OriginStateFips` String,
  `OriginStateName` String,
  `OriginWac` Int32,
  `DestAirportID` Int32,
  `DestAirportSeqID` Int32,
  `DestCityMarketID` Int32,
  `Dest` FixedString(5),
  `DestCityName` String,
  `DestState` FixedString(2),
  `DestStateFips` String.
```

```

`DestStateName`      String,
`DestWac`            Int32,
`CRSDepTime`        Int32,
`DepTime`           Int32,
`DepDelay`          Int32,
`DepDelayMinutes`  Int32,
`DepDel15`          Int32,
`DepartureDelayGroups` String,
`DepTimeBlk`        String,
`TaxiOut`           Int32,
`WheelsOff`         Int32,
`WheelsOn`          Int32,
`TaxiIn`            Int32,
`CRSArrTime`        Int32,
`ArrTime`           Int32,
`ArrDelay`          Int32,
`ArrDelayMinutes`  Int32,
`ArrDel15`          Int32,
`ArrivalDelayGroups` Int32,
`ArrTimeBlk`        String,
`Cancelled`          UInt8,
`CancellationCode`  FixedString(1),
`Diverted`           UInt8,
`CRSElapsedTime`    Int32,
`ActualElapsedTime` Int32,
`AirTime`           Nullable(Int32),
`Flights`           Int32,
`Distance`          Int32,
`DistanceGroup`     UInt8,
`CarrierDelay`      Int32,
`WeatherDelay`      Int32,
`NASDelay`          Int32,
`SecurityDelay`     Int32,
`LateAircraftDelay` Int32,
`FirstDepTime`      String,
`TotalAddGTime`     String,
`LongestAddGTime`   String,
`DivAirportLandings` String,
`DivReachedDest`    String,
`DivActualElapsedTime` String,
`DivArrDelay`       String,
`DivDistance`       String,
`Div1Airport`        String,
`Div1AirportID`     Int32,
`Div1AirportSeqID`  Int32,
`Div1WheelsOn`       String,
`Div1TotalGTime`    String,
`Div1LongestGTime`  String,
`Div1WheelsOff`     String,
`Div1TailNum`        String,
`Div2Airport`        String,
`Div2AirportID`     Int32,
`Div2AirportSeqID`  Int32,
`Div2WheelsOn`       String,
```

```

`Div2TotalGTime`      String,
`Div2LongestGTime`   String,
`Div2WheelsOff`      String,
`Div2TailNum`        String,
`Div3Airport`         String,
`Div3AirportID`      Int32,
`Div3AirportSeqID`   Int32,
`Div3WheelsOn`       String,
`Div3TotalGTime`     String,
`Div3LongestGTime`   String,
`Div3WheelsOff`      String,
`Div3TailNum`        String,
`Div4Airport`         String,
`Div4AirportID`      Int32,
`Div4AirportSeqID`   Int32,
`Div4WheelsOn`       String,
`Div4TotalGTime`     String,
`Div4LongestGTime`   String,
`Div4WheelsOff`      String,
`Div4TailNum`        String,
`Div5Airport`         String,
`Div5AirportID`      Int32,
`Div5AirportSeqID`   Int32,
`Div5WheelsOn`       String,
`Div5TotalGTime`     String,
`Div5LongestGTime`   String,
`Div5WheelsOff`      String,
`Div5TailNum`        String
)ENGINE = ReplicatedMergeTree()
PARTITION BY toYYYYMM(FlightDate)
PRIMARY KEY (intHash32(FlightDate))
ORDER BY (intHash32(FlightDate))
SAMPLE BY intHash32(FlightDate)
SETTINGS index_granularity= 8192 ;

```

2.创建分布式表，方便数据的写入和查询。

- 语法：

```

CREATE TABLE [db.]d_table_name ON CLUSTER cluster
AS db.local_table_name ENGINE = Distributed(cluster, db, local_table_name [, sharding_key]
)

```

- 参数说明：

参数	说明
db	指定数据库的名称，默认为当前选择的数据库。

参数	说明
<code>d_table_name</code>	分布式表名。
<code>cluster</code>	指定集群的名称，固定为default。
<code>local_table_name</code>	已创建的本地表名。
<code>sharding_key</code>	分片表达式。

? 说明

更多参数说明，请参见[Create Table](#)。

● 示例：

```
CREATE TABLE ontime_local_distributed ON CLUSTER default
AS clickhouse_demo.ontime_local
ENGINE = Distributed(default, clickhouse_demo, ontime_local, rand());
```

下一步

[导入数据](#)

4.9. 导入数据

本文以导入On Time数据集至clickhouse_demo数据库的分布式表ontime_local_distributed为例，介绍如何使用客户端clickhouse-client工具导入数据到云数据库ClickHouse。

前提条件

1. 已完成快速入门系列的如下步骤。
 - i. [新建集群](#)
 - ii. [创建账号](#)
 - iii. [设置白名单](#)

? **说明** 您需要将clickhouse-client所在服务器的IP地址添加到云数据库ClickHouse云原生版集群的白名单中。

- iv. [连接集群](#)

v. 创建数据库

vi. 创建表

2. 已安装与云数据库ClickHouse云原生版集群版本对应的clickhouse-client工具。下载链接，请参见[clickhouse-client](#)。

操作步骤

1. 单击[On Time Data](#)下载On Time数据集。
2. 解压下载的On Time数据集。

```
unzip ontime-data.zip;
```

3. 连接云数据库ClickHouse集群并导入数据到云数据库ClickHouse。

在clickhouse-client客户端的安装目录下执行如下命令。

```
./clickhouse-client --host=<host> --port=<port> --user=<user> --password=<password> --query="INSERT INTO <ClickHouse_table> FORMAT CSVWithNames" < ontime-data.csv;
```

参数说明如下。

参数	说明
host	外网地址或VPC地址，您可以在 集群信息 页面，单击 集群计算组操作列的查看连接查看 。 如果clickhouse-client所在服务器与云数据库ClickHouse集群在同一VPC内，您可以使用VPC地址。否则，请使用外网地址。
port	TCP端口号，您可以在 集群信息 页面，单击 集群计算组操作列的查看连接查看 。
user	您通过云数据库ClickHouse控制台创建的数据库账号。
password	数据库账号对应的密码。
ClickHouse_table	要导入数据的云数据库ClickHouse表。

示例语句如下。

```
./clickhouse-client --host=vw-uf632p11n08is****.public.clickhouse.ads.aliyuncs.com --port=9000 --user=test --password=123456Aa --query="INSERT INTO clickhouse_demo.ontime_local_distributed FORMAT CSVWithNames" < ontime-data.csv;
```

4. 查询数据，验证数据是否导入成功。

```
SELECT OriginCityName, count(*) AS flights
FROM ontime_local_distributed
GROUP BY OriginCityName
ORDER BY flights DESC
LIMIT 10;
```

查询结果如下。

OriginCityName	flights
Chicago, IL	24114
Atlanta, GA	22001
Dallas/Fort Worth, TX	17340
Los Angeles, CA	14494
Denver, CO	14170
New York, NY	14075
Washington, DC	11985
Houston, TX	11483
San Francisco, CA	11259
St. Louis, MO	10721

5. 数据迁移同步

5.1. 从Kafka同步数据

本文介绍如何从消息队列Kafka版中同步数据至云数据库ClickHouse。

注意事项

- 消息队列Kafka版为付费服务。计费详情，请参见[计费概述](#)。
- 请确保数据源消息队列Kafka版实例和目的云数据库ClickHouse集群在同一地域并使用相同的VPC。

前提条件

已创建消息队列Kafka版的Topic和Group。如何创建，请参见[创建资源](#)。

操作步骤

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在集群列表页面，选择默认实例列表或云原生版本实例列表，单击目标集群ID。
4. 如果您的集群是社区兼容版，请单击右上方导航栏的登录数据库。
如果您的集群是云原生版，请单击集群计算组操作列的登录。
5. 在登录实例页面，输入数据库账号和密码，单击登录。
6. 创建Kafka消费表，示例语句如下。

```
CREATE TABLE default.kafka_src_table ON CLUSTER default
(
  //定义表结构的字段
  id Int32,
  name String
) ENGINE = Kafka()
SETTINGS
  kafka_broker_list = 'alikaafka-post-cn-tl32i5sc****-1-vpc.alikaafka.aliyuncs.com:9092,alikaafka-post-cn-tl32i5sc****-2-vpc.alikaafka.aliyuncs.com:9092',
  kafka_topic_list = 'test',
  kafka_group_name = 'test',
  kafka_format = 'CSV',
  kafka_num_consumers = 1,
  kafka_max_block_size = 65536,
  kafka_skip_broken_messages = 0,
  kafka_auto_offset_reset = 'latest';
```

🔍 说明

- Kafka消费表不能直接作为结果表使用。
- Kafka消费表只是用来消费Kafka数据，没有真正的存储数据。

常用参数说明如下。

名称	是否必选	说明
kafka_broker_list	是	以英文逗号分隔的Kafka的接入点地址列表。如何查看接入点，请参见 查看接入点 。
kafka_topic_list	是	以英文逗号分隔的Topic名称列表。
kafka_group_name	是	Kafka的消费组名称。更多信息，请参见 创建Group 。
kafka_format	是	云数据库ClickHouse支持处理的消息体格式。 <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 5px;"> <p>? 说明 云数据库ClickHouse支持的消息体格式，具体请参见Formats for Input and Output Data。</p> </div>
kafka_row_delimiter	否	行分隔符，用于分割不同的数据行。默认为“\n”，您也可以根据数据写入的实际分割格式进行设置。
kafka_num_consumers	否	单个表的消费者数量，默认值为1。 <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 5px;"> <p>? 说明</p> <ul style="list-style-type: none"> ◦ 一个消费者的吞吐量不足时，需要指定更多的消费者。 ◦ 消费者的总数不应超过Topic中的分区数，因为每个分区只能分配一个消费者。 </div>
kafka_max_block_size	否	Kafka消息的最大批次大小，单位：Byte，默认值为65536 Byte。
kafka_skip_broken_messages	否	kafka消息解析器对于脏数据的容忍度，默认值为0。如果 <code>kafka_skip_broken_messages=N</code> ，则引擎将跳过N条无法解析的Kafka消息（一条消息等于一行数据）。
kafka_commit_every_batch	否	执行Kafka commit的频率，取值说明如下。 <ul style="list-style-type: none"> ◦ 0：完全写入一整个Block数据块的数据后才执行commit。 ◦ 1：每写完一个Batch批次的数据就执行一次commit。
kafka_auto_offset_reset	否	从哪个offset开始读取Kafka数据，取值说明如下。 <ul style="list-style-type: none"> ◦ earliest：从最早的offset开始读取Kafka数据。 ◦ latest：从最晚的offset开始读取Kafka数据。 <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 5px; margin-top: 5px;"> <p>? 说明 21.8版本的云数据库ClickHouse集群不支持该参数。</p> </div>

? **说明** 更多参数说明，请参见[Kafka](#)。

7. 创建云数据库ClickHouse目的表。

i. 创建本地表。

```
create table default.kafka_table_local ON CLUSTER default (  
  id Int32,  
  name String  
) ENGINE = MergeTree()  
  ORDER BY (id);
```

ii. 创建分布式表。

 **说明** 如果您只需要同步数据至本地表，可跳过此步骤。

```
CREATE TABLE kafka_table_distributed ON CLUSTER default AS default.kafka_table_loca  
l  
ENGINE = Distributed(default, default, kafka_table_local, id);
```

8. 创建VIEW将Kafka消费表消费到的数据同步到云数据库ClickHouse目的表。

如果您同步的目的表是本地表，请将分布式表名更换为本地表名，再进行同步。

```
CREATE MATERIALIZED VIEW consumer ON CLUSTER default TO kafka_table_distributed AS SELE  
CT * FROM kafka_src_table;
```

9. 在消息队列Kafka版的Topic端发送消息。

本文以发送消息 `1,a` 和 `2,b` 为例。

- i. 登录[消息队列kafka控制台](#)。
- ii. 在实例列表页面，单击目标实例名称。
- iii. 在Topic管理页面，单击目标Topic操作列的更多 > 体验发送消息。
- iv. 在快速体验消息收发页面，输入发送的消息内容。
- v. 单击确定。

10. 查询云数据库ClickHouse目的表，查询语句如下。

```
select * from kafka_table_distributed;
```

 **说明** 如果您同步的目的表是本地表，请将查询语句中的分布式表名更换为本地表名，再进行查询。

查询结果如下。

id	name
1	a
2	b

5.2. 从Flink导入

5.2.1. 从JDBC connector导入

本节主要介绍如何利用ClickHouse JDBC connector，使用不同版本的Flink写入数据到ClickHouse中。

背景信息

Flink在1.11.0版本对其JDBC connector进行了一次较大的重构：

- 重构之前（1.10.1及之前版本），包名为flink-jdbc。
- 重构之后（1.11.0及之后版本），包名为flink-connector-jdbc。

二者对Flink中以不同方式写入ClickHouse Sink的支持情况如下：

API名称	flink-jdbc	flink-connector-jdbc
DataStream	不支持	支持
Table API (Legacy)	支持	不支持
Table API (DDL)	不支持	不支持

flink-connector-jdbc完全移除了对Table API (Legacy) 的支持，只能通过DDL的方式调用Table API。但是，Table DDL方式硬编码了其所支持的JDBC Driver，不支持ClickHouse。

下面，我们依次以Flink 1.10.1 + flink-jdbc 以及Flink 1.11.0 + flink-connector-jdbc 为例，介绍Flink写入ClickHouse的方法。

Flink 1.10.1 + flink-jdbc

Flink 1.10.1及之前版本需要采用flink-jdbc+Table API的方式写入数据到ClickHouse。本节我们使用Maven及Flink 1.10.1版本进行示例。

1. 用`mvn archetype:generate`命令创建项目，生成过程中根据提示输入group-id和artifact-id等。

```
$ mvn archetype:generate \  
  -DarchetypeGroupId=org.apache.flink \  
  -DarchetypeArtifactId=flink-quickstart-scala \  
  -DarchetypeVersion=1.10.1
```

2. 编辑pom.xml中的 `<dependencies />` 小节添加依赖。

```
//添加Flink Table API相关的依赖
<dependency>
  <groupId>org.apache.flink</groupId>
  <artifactId>flink-table-planner-blink_${scala.binary.version}</artifactId>
  <version>${flink.version}</version>
  <scope>provided</scope>
</dependency>
<dependency>
  <groupId>org.apache.flink</groupId>
  <artifactId>flink-table-api-scala-bridge_${scala.binary.version}</artifactI
d>
  <version>${flink.version}</version>
</dependency>
<dependency>
  <groupId>org.apache.flink</groupId>
  <artifactId>flink-table-common</artifactId>
  <version>${flink.version}</version>
  <scope>provided</scope>
</dependency>
//添加Flink JDBC以及Clickhouse JDBC Driver相关的依赖
<dependency>
  <groupId>org.apache.flink</groupId>
  <artifactId>flink-jdbc_${scala.binary.version}</artifactId>
  <version>${flink.version}</version>
</dependency>
<dependency>
  <groupId>ru.yandex.clickhouse</groupId>
  <artifactId>clickhouse-jdbc</artifactId>
  <version>0.2.4</version>
</dependency>
```

3. 创建数据写入程序文件。

示例程序使用 `CsvTableSource` 读入 CSV 文件产生Table Source，使用 `JDBCAppendTableSink` 将数据写入到ClickHouse Sink中。

🔍 说明

- 由于ClickHouse单次插入的延迟比较高，我们需要设置 `BatchSize` 来批量插入数据，提高性能。
- 在JDBCAppendTableSink的实现中，若最后一批数据的数目不足 `BatchSize`，则不会插入剩余数据。

```
package org.myorg.example
import org.apache.flink.streaming.api.scala._
import org.apache.flink.table.sources._
import org.apache.flink.table.api.scala.StreamTableEnvironment
import org.apache.flink.table.api._
import org.apache.flink.types.Row
import org.apache.flink.table.api.{
  TableEnvironment,
  TableSchema,
```

```
Types,
ValidationException
}
import org.apache.flink.api.java.io.jdbc.JDBCAppendTableSink
import org.apache.flink.api.common.typeinfo.TypeInformation
object StreamingJob {
def main(args: Array[String]) {
val SourceCsvPath =
"/<your-path-to-test-csv>/source.csv"
val CkJdbcUrl =
"jdbc:clickhouse://<clickhouse-host>:<port>/<database>"
val CKUsername = "<your-username>"
val CkPassword = "<your-password>"
val BatchSize = 500 // 设置您的batch size
val env = StreamExecutionEnvironment.getExecutionEnvironment
val tEnv = StreamTableEnvironment.create(env)
val csvTableSource = CsvTableSource
.builder()
.path(SourceCsvPath)
.ignoreFirstLine()
.fieldDelimiter(",")
.field("name", Types.STRING)
.field("age", Types.LONG)
.field("sex", Types.STRING)
.field("grade", Types.LONG)
.field("rate", Types.FLOAT)
.build()
tEnv.registerTableSource("source", csvTableSource)
val resultTable = tEnv.scan("source").select("name, grade, rate")
val insertIntoCkSql =
"""
| INSERT INTO sink_table (
|   name, grade, rate
| ) VALUES (
|   ?, ?, ?
| )
"""
.stripMargin
//将数据写入 ClickHouse Sink
val sink = JDBCAppendTableSink
.builder()
.setDrivername("ru.yandex.clickhouse.ClickHouseDriver")
.setDBUrl(CkJdbcUrl)
.setUsername(CKUsername)
.setPassword(CkPassword)
.setQuery(insertIntoCkSql)
.setBatchSize(BatchSize)
.setParameterTypes(Types.STRING, Types.LONG, Types.FLOAT)
.build()
tEnv.registerTableSink(
"sink",
Array("name", "grade", "rate"),
Array(Types.STRING, Types.LONG, Types.FLOAT),
sink
)
}
```

```
tEnv.insertInto(resultTable, "sink")
env.execute("Flink Table API to ClickHouse Example")
}
}
```

参数说明：

- `SourceCsvPath` : 源CSV文件路径。
- `CkJdbcUrl` : 目标ClickHouse集群地址。
- `CkUsername` : 目标ClickHouse集群用户名。
- `CkPassword` : 目标ClickHouse集群对应密码。

4. 编译运行。

```
$ mvn clean package
$ ${FLINK_HOME}/bin/flink run target/example-0.1.jar
```

Flink 1.11.0 + flink-connector-jdbc

Flink 1.11.0及之后版本需要采用flink-connector-jdbc+DataStream的方式写入数据到ClickHouse。本节我们使用Maven及Flink 1.11.0版本进行示例。

1. 用`mvn archetype:generate`命令创建项目，生成过程中会提示输入group-id和artifact-id等。

```
$ mvn archetype:generate \
  -DarchetypeGroupId=org.apache.flink \
  -DarchetypeArtifactId=flink-quickstart-scala \
  -DarchetypeVersion=1.11.0
```

2. 编辑pom.xml中的 `<dependencies />` 小节添加依赖。

```
//添加Flink Table API相关的依赖
<dependency>
  <groupId>org.apache.flink</groupId>
  <artifactId>flink-table-planner-blink_${scala.binary.version}</artifactId>
  <version>${flink.version}</version>
  <scope>provided</scope>
</dependency>
<dependency>
  <groupId>org.apache.flink</groupId>
  <artifactId>flink-table-api-scala-bridge_${scala.binary.version}</artifactI
d>
  <version>${flink.version}</version>
</dependency>
<dependency>
  <groupId>org.apache.flink</groupId>
  <artifactId>flink-table-common</artifactId>
  <version>${flink.version}</version>
  <scope>provided</scope>
</dependency>
//添加Flink JDBC Connector以及Clickhouse JDBC Driver相关的依赖
<dependency>
  <groupId>org.apache.flink</groupId>
  <artifactId>flink-connector-jdbc_${scala.binary.version}</artifactId>
  <version>${flink.version}</version>
</dependency>
<dependency>
  <groupId>ru.yandex.clickhouse</groupId>
  <artifactId>clickhouse-jdbc</artifactId>
  <version>0.2.4</version>
</dependency>
```

3. 创建数据写入程序文件。

示例程序使用 `CsvTableSource` 读入CSV文件产生Table Source，通过 `TableEnvironment.toAppendStream` 将Table转换为DataStream。使用 `JdbcSink` 将数据写入到ClickHouse中。

② 说明

- 由于ClickHouse单次插入的延迟比较高，我们需要设置 `BatchSize` 来批量插入数据，提高性能。
- 当前版本的flink-connector-jdbc，使用Scala API调用JdbcSink时会出现lambda函数的序列化问题。我们只能采用手动实现interface的方式来传入相关JDBC Statement build函数（`class CkSinkBuilder`）。

```
class CkSinkBuilder extends JdbcStatementBuilder[(String, Long, Float)] {
  def accept(ps: PreparedStatement, v: (String, Long, Float)): Unit = {
    ps.setString(1, v._1)
    ps.setLong(2, v._2)
    ps.setFloat(3, v._3)
  }
}
```

```
package org.mvora.example
```

```
import org.apache.flink.streaming.api.scala._
import org.apache.flink.table.sources._
import org.apache.flink.table.api.bridge.scala.StreamTableEnvironment
import org.apache.flink.table.api._
import org.apache.flink.types.Row
import org.apache.flink.table.api.{
  TableEnvironment,
  TableSchema,
  Types,
  ValidationException
}
import org.apache.flink.connector.jdbc._
import java.sql.PreparedStatement
//手动实现interface的方式来传入相关JDBC Statement build函数
class CkSinkBuilder extends JdbcStatementBuilder[(String, Long, Float)] {
  def accept(ps: PreparedStatement, v: (String, Long, Float)): Unit = {
    ps.setString(1, v._1)
    ps.setLong(2, v._2)
    ps.setFloat(3, v._3)
  }
}
object StreamingJob {
  def main(args: Array[String]) {
    val SourceCsvPath =
      "</your-path-to-test-csv>/source.csv"
    val CkJdbcUrl = "jdbc:clickhouse://<clickhouse-host>:<port>/<database>"
    val CkUsername = "<your-username>"
    val CkPassword = "<your-password>"
    val BatchSize = 500 // 设置您的 batch size
    val env = StreamExecutionEnvironment.getExecutionEnvironment
    val tEnv = StreamTableEnvironment.create(env)
    val csvTableSource = CsvTableSource
      .builder()
      .path(SourceCsvPath)
      .ignoreFirstLine()
      .fieldDelimiter(",")
      .field("name", Types.STRING)
      .field("age", Types.LONG)
      .field("sex", Types.STRING)
      .field("grade", Types.LONG)
      .field("rate", Types.FLOAT)
      .build()
    tEnv.registerTableSource("source", csvTableSource)
    val resultTable = tEnv.scan("source").select("name, grade, rate")
    //将Table转换为DataStream
    val resultDataStream =
      tEnv.toAppendStream[(String, Long, Float)](resultTable)
    val insertIntoCkSql =
      """
      | INSERT INTO sink_table (
      |   name, grade, rate
      | ) VALUES (
      |   ?, ?, ?
      | )
      """
  }
}
```

```
"".stripMargin
//将数据写入ClickHouse JDBC Sink
resultDataStream.addSink(
  JdbcSink.sink[(String, Long, Float)](
    insertIntoCkSql,
    new CkSinkBuilder,
    new JdbcExecutionOptions.Builder().withBatchSize(BatchSize).build(),
    new JdbcConnectionOptions.JdbcConnectionOptionsBuilder()
      .withDriverName("ru.yandex.clickhouse.ClickHouseDriver")
      .withUrl(CkJdbcUrl)
      .withUsername(CkUsername)
      .withPassword(CkPassword)
      .build()
  )
)
env.execute("Flink DataStream to ClickHouse Example")
}
```

参数说明：

- `SourceCsvPath` : 源CSV文件路径。
- `CkJdbcUrl` : 目标ClickHouse集群地址。
- `CkUsername` : 目标ClickHouse集群用户名。
- `CkPassword` : 目标ClickHouse集群对应密码。

4. 编译运行。

```
$ mvn clean package
$ ${FLINK_HOME}/bin/flink run target/example-0.1.jar
```

5.2.2. 从Flink SQL导入

本文介绍如何使用Flink SQL方式将Flink中的数据写入云数据库ClickHouse。

使用限制

只有Flink计算引擎VVR 3.0.2及以上版本支持使用Flink SQL写入云数据库ClickHouse。

前提条件

- 已在云数据库ClickHouse中创建表。更多信息，请参见[创建表](#)。
- 已在云数据库ClickHouse中设置白名单。更多信息，请参见[设置白名单](#)。
- 已开通Flink全托管。更多信息，请参见[开通Flink全托管](#)。

操作步骤

1. 登录Flink全托管开发控制台，新建作业。
 - i. 登录[实时计算管理控制台](#)。
 - ii. 在Flink全托管页签，单击目标工作空间操作列下的控制台。
 - iii. 在左侧导航栏，单击作业开发。

iv. 单击新建。

v. 在新建文件对话框，填写作业配置信息。

作业参数	说明
文件名称	<p>作业的名称。</p> <p> 说明 作业名称在当前项目中必须保持唯一。</p>
文件类型	<p>流作业和批作业均支持以下文件类型：</p> <ul style="list-style-type: none"> ■ SQL ■ JAR ■ PYTHON <p> 说明 实时计算引擎VVR 3.0.1及以上版本支持批作业。</p>
部署目标	<p>选择作业需要部署的集群，支持以下两种集群模式：</p> <ul style="list-style-type: none"> ■ Per-Job集群（默认）：适用于占用资源比较大或持续稳定运行的作业。因为作业之间资源隔离，每个作业都需要一个独立的JM，小任务JM的资源利用率较低。 ■ Session集群：适用于占用资源比较小或任务启停比较频繁的作业。因为多个作业可以复用相同的JM，可以提高JM资源利用率。 <p> 说明 如果您需要开启SQL Preview功能，必须选择Session集群，且已将其设置为SQL Previews集群，详情请参见作业调试和配置开发测试环境（Session集群）。</p>
存储位置	<p>指定该作业的代码文件所属的文件夹。</p> <p>您还可以在现有文件夹右侧，单击  图标，新建子文件夹。</p>

vi. 单击确认。

2. 在作业开发页面，编写并执行代码。

- i. 创建源表、结果表，并将源表数据插入到结果表中。

```
--创建源表sls_test_single_local。
CREATE TEMPORARY TABLE sls_test_single_local (
  id INT,
  name VARCHAR,
  age BIGINT,
  rate FLOAT
) WITH (
  'connector' = 'datagen',
  'rows-per-second' = '50'
);
--创建结果表clickhouse_output。
CREATE TEMPORARY TABLE clickhouse_output (
  id INT,
  name VARCHAR,
  age BIGINT,
  rate FLOAT
) WITH (
  'connector' = 'clickhouse',
  'url' = 'jdbc:clickhouse://demo.aliyuncs.com:8123',
  'userName' = 'test',
  'password' = '280226Ck',
  'tableName' = 'sls_test_single_local',
);
--将源表数据插入到结果表。
INSERT INTO clickhouse_output
SELECT
  id,
  name,
  age,
  rate
FROM sls_test_single_local;
```

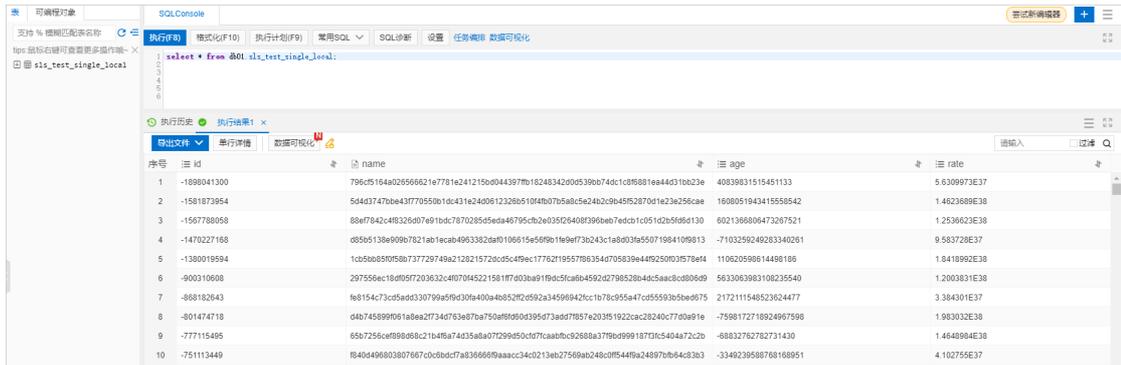
 **说明** 创建结果表的语法说明，请参见[ClickHouse结果表](#)。

- ii. 单击保存。
 - iii. 单击验证。
 - iv. 单击上线。
 - v. 在弹出的**上线确认**窗口中单击**确认**，将作业发布至生产环境。
 - vi. 上线成功后，在弹出的**上线成功**，请前往**运维查看详情**中单击**运维**。
3. 在**作业运维**页面，启动作业。
 - i. 单击右上方**启动**。
 - ii. 在弹出的**作业启动配置**页面，单击**确认启动**。
 4. 在云数据库ClickHouse中查询表。
 - i. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
 - ii. 在页面左上角，选择目标集群所在的地域。
 - iii. 在**集群列表**页面，选择**默认实例列表**或**云原生版本实例列表**，单击目标集群ID。

- iv. 如果您的集群是社区兼容版，请单击右上方导航栏的登录数据库。
如果您的集群是云原生版，请单击集群计算组操作列的登录。
- v. 在登录实例页面，输入数据库账号和密码，单击登录。
- vi. 输入查询语句并单击执行（F8）。本文使用下面的示例语句。

```
select * from db01.sls_test_single_local;
```

查询结果如下。



序号	id	name	age	rate
1	-1898041300	796c5164a026566521e7781e241215b0d044397fb182483420d05396b740c1c8f6881ea44310623e	40839631515451133	5.0309973E37
2	-1581873954	5d4437470bb43f70550b10c431e240061232605104f0b7b5a805e24b2c9b45f52870d1e23e256cae	1608051943415558542	1.4623688E38
3	-1587788058	88e7842c4f8326d07e91bdc787028545eaa46795cb2e0352640f3906eb7edcb1c051d2b505e130	6021368806473267521	1.2539623E38
4	-1470227168	d85d5138e899b7821ab1ecab49633826af0106915e569b1e9ef73b243c1a8d03fa55071984109813	-7103259249283340261	9.583728E37
5	-1380019594	1c058b85f0f5b737729749a2128215726c054f9ec1776219557863546705639a449250930578e4	110620598614498106	1.8418902E38
6	-900310608	29755ec18d0f97203632c4f0704522158187d03ea91f9dc51ca6b4592d2796528a44c5aac8c08069	563306398310823540	1.2003031E38
7	-868182643	fa8154c73c05add33079a59d30fa40094b852f2c592a34599942fcc1b78c955a47c0d5593b05be0675	2172111548523624477	3.384301E37
8	-801474718	d4b745899f0e1a8ea2f34c76387ba750af8f06d0395073ad07857e203351922cac28240c770d91e	-7598172718924967598	1.983032E38
9	-777115495	6567256cef898d68c21b4f6a7435a8a071299450c167caab0c92888a37f9d999187f3c5404472c2b	-68832762782731430	1.4848994E38
10	-751113449	0b40498030807667c0c5bdc7f8b366669aaccc34c0213eb27569a0248c0f5449a24897b64c3263	-3348239580768168951	4.102755E37

5.3. 从大数据导入/同步

5.3.1. 从DataWorks同步

本文以MaxCompute为例介绍如何将数据通过DataWorks同步至云数据库ClickHouse。

背景信息

您可以通过DataWorks，将支持的各种数据源数据离线同步至云数据库ClickHouse。离线同步支持的数据源类型，具体请参见[支持的数据源与读写插件](#)。

注意事项

- 云数据库ClickHouse仅支持独享数据集成资源组。DataWorks独享数据集成资源组为付费服务，计费详情，请参见[独享数据集成资源组计费说明：包年包月](#)。
- 如果您需要对已同步过的表再次同步，请首先登录[云数据库ClickHouse控制台](#)，使用 `truncate table <表名>`；清空云数据库ClickHouse表中已同步的数据。

前提条件

- 已创建工作空间，引擎服务选择为MaxCompute。如何创建，请参见[创建工作空间](#)。
- 已创建并配置独享数据集成资源组。如何创建和配置，请参见[新增和使用独享数据集成资源组](#)。
- 如果RAM用户使用DataWorks，需要阿里云主账号为RAM用户配置合适的成员角色。如何配置，请参见[添加工作空间成员和角色](#)。

操作步骤

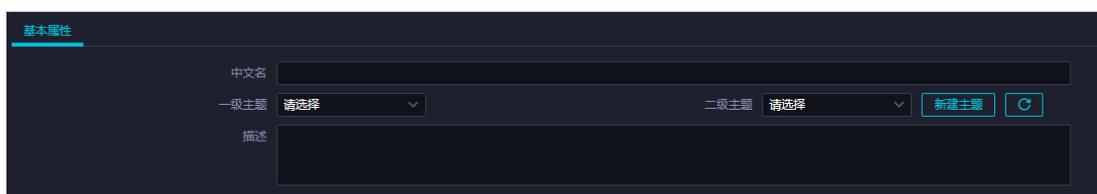
1. 配置数据源。

您需要为MaxCompute和云数据库ClickHouse分别配置数据源。

 **说明** 如何配置数据源，请参见[配置MaxCompute数据源](#)和[配置ClickHouse数据源](#)。

2. 创建MaxCompute表。

- i. 登录DataWorks控制台。
- ii. 在左侧导航栏，单击工作空间列表。
- iii. 选择工作空间所在地域后，单击相应工作空间后的进入数据开发。
- iv. 在数据开发页面，鼠标悬停至 **+新建** 图标，单击MaxCompute > 表。
- v. 在新建表对话框中，输入表名，本文以odptabletest1表名为例。
- vi. 单击新建。
- vii. 在基本属性区域，配置各项参数。



参数含义如下。

名称	描述
中文名	MaxCompute表的中文名称。
一级主题	新建表所处的一级目标文件夹名称。  说明 一级、二级主题仅仅是DataWorks上文件夹的摆放形式，目的是为了您能更好地管理您的表。
二级主题	新建表所处的二级目标文件夹名称。
新建主题	单击 新建主题 ，进入 主题管理 页面，您可以在该页面创建一级主题、二级主题。 新建主题后，单击  图标，即可同步新建的主题。
描述	对新建表进行简单描述。

- viii. 单击工具栏中的DDL模式。
- ix. 在DDL模式对话框中，输入如下建表语句，单击生成表结构。

```
CREATE TABLE IF NOT EXISTS odptabletest1
(
  v1 TINYINT,
  v2 SMALLINT
);
```

- x. 在**确认操作**对话框中，单击**确认**。
- xi. 单击工具栏的**提交到生产环境**。

- xii. 在弹出的提交生产确认对话框中，勾选我已悉知风险，确认提交。
- xiii. 单击确认。

3. 写入MaxCompute表数据。

- i. 在数据开发页面，单击左侧导航栏的临时查询。
- ii. 鼠标悬停至 **+新建** 图标，单击新建 > ODPS SQL。
- iii. 在新建节点对话框中，输入节点名称，并选择目标文件夹。
- iv. 单击提交。
- v. 在节点的编辑页面，输入如下语句，为MaxCompute表写入数据。

```
insert into odptabletest1 values (1,"a"),(2,"b"),(3,"c"),(4,"d");
```

- vi. 单击工具栏中的  图标。
- vii. 在MaxCompute计算成本估计窗口，单击运行。

4. 创建云数据库ClickHouse表。

- i. 使用阿里云账号登录 [云数据库ClickHouse控制台](#)。
- ii. 在页面左上角，选择目标集群所在的地域。
- iii. 在集群列表页面，选择默认实例列表或云原生版本实例列表。
- iv. 单击目标集群ID。
- v. 如果您的集群是社区兼容版，请单击右上方导航栏的登录数据库。
如果您的集群是云原生版，请单击集群计算组操作列的登录。
- vi. 在登录实例页面，输入数据库账号和密码，单击登录。
- vii. 输入建表语句并单击执行（F8）。建表语句如下。

```
create table default.dataworktest ON CLUSTER default (  
v1 Int,  
v2 String  
) ENGINE = MergeTree ORDER BY v1;
```

 **说明** 云数据库ClickHouse表的结构类型需与MaxCompute对应。

5. 创建业务流程。

如果您已有业务流程，可以忽略该步骤。

- i. 在数据开发页面，单击左侧导航栏的数据开发。
- ii. 鼠标悬停至 **+新建** 图标，选择业务流程。
- iii. 在新建业务流程对话框，输入业务名称。

 **注意** 业务名称必须由大小写字母、中文、数字、下划线（_）以及小数点（.）组成，且不能超过128个字符。

- iv. 单击新建。

6. 创建离线同步节点。

- i. 展开业务流程，右键单击数据集成。
- ii. 单击新建 > 离线同步。
- iii. 在新建节点对话框中，输入节点名称，并选择目标文件夹。

 **注意** 节点名称必须由大小写字母、中文、数字、下划线（_）以及小数点（.）组成，且不能超过128个字符。

iv. 单击提交。

7. 配置数据来源和数据去向。

- i. **配置数据来源：**数据来源为DataWorks支持的任一数据源，本文以MaxCompute为例。



参数含义如下。

参数	描述
数据源	数据源的类型和名称。
生产项目名	默认不可以修改。
表	需要同步的表。
分区信息	如果您每日增量数据限定在对应日期的分区中，可以使用分区进行每日增量。例如，配置分区v17的值为\${bizdate}。  说明 DataWorks不支持对MaxCompute分区表进行字段映射，您需要单独配置分区字段。

 **说明** 参数配置的更多信息，请参见[MaxCompute Reader](#)。

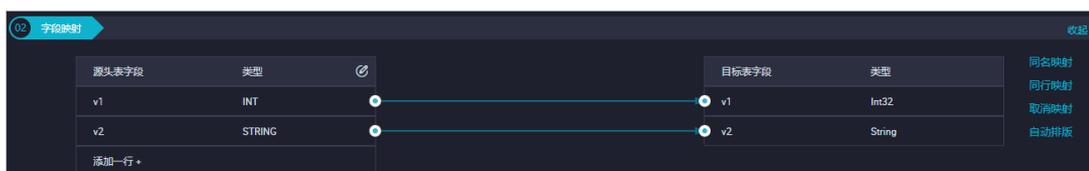
ii. 配置数据去向：数据源的类型选择为ClickHouse。



参数含义如下。

参数	描述
数据源	数据源的类型与名称，其中，数据源的类型需选择ClickHouse。
表	需要同步的表。
主键或者唯一键冲突处理	固定选择为insert into（插入）。
导入前准备语句	执行数据同步任务之前率先执行的SQL语句。
导入后完成语句	执行数据同步任务之后执行的SQL语句。
批量插入字节大小	批量插入数据的最大字节数。
批量条数	批量插入数据的条数。

iii. （可选）字段映射：您可以选择字段的映射关系，左侧的源头表字段和右侧的目标表字段为一一对应关系。



说明 参数说明，具体请参见[字段映射](#)。

iv. (可选) 通道控制：配置作业速率上限和脏数据检查规则。



说明 参数说明，具体请参见通道控制。

8. 配置数据集成资源组。

单击右侧的数据集成资源组配置，选择已经创建好的独享数据集成资源组。



9. 保存并运行数据同步任务。

- i. 单击工具栏中的 图标，保存数据同步任务。
- ii. 单击工具栏中的 图标，运行数据同步任务。



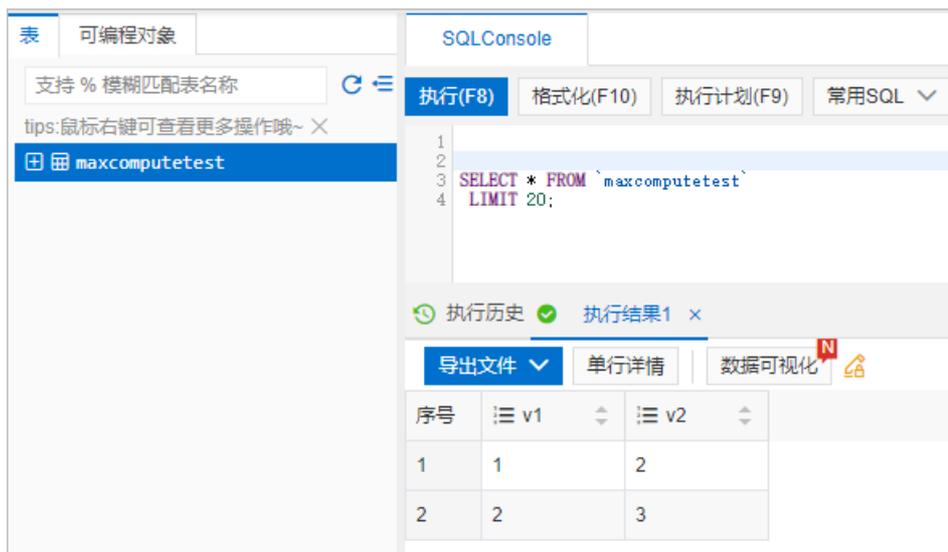
10. 查询云数据库ClickHouse表。

- i. 登录云数据库ClickHouse控制台。

- ii. 在集群列表页面，选择默认实例列表或云原生版本实例列表，单击目标集群ID。
- iii. 如果您的集群是社区兼容版，请单击右上方导航栏的登录数据库。
如果您的集群是云原生版，请单击集群计算组操作列的登录。
- iv. 在登录实例页面，输入数据库账号和密码，单击登录。
- v. 输入查询语句并单击执行（F8）。本文使用下面的示例语句。

```
select * from dataworktest;
```

查询结果如下。



5.3.2. 从MaxCompute导入

5.3.2.1. 从MaxCompute外表导入（19.15.2.2）

本节主要介绍如何在云数据库ClickHouse社区兼容版集群（内核版本为19.15.2.2）中使用阿里云MaxCompute外表方式导入数据。

数据导入

1. 登录ClickHouse目标数据库，创建外表表。

语法如下：

```
CREATE TABLE <table_name> [on cluster default]
(
  'col_name' col_type, [col_name col_type]
)
ENGINE = MaxCompute('<tunnel-endpoint>', '<project-name>', '<table-name>', '<partition-spec>', '<access-key-id>', '<access-key-secret>');
```

参数名	描述
table_name	自定义外部表名。

参数名	描述
'col_name' col_type	列名，列类型。 建表时，ClickHouse的列类型，应该与MaxCompute列类型具有对应关系，请参见 类型映射 。
ENGINE = MaxCompute	表示该表是外部表，使用的存储引擎是MaxCompute。
tunnel-endpoint	MaxCompute的EndPoint（域名节点）。 <div style="border: 1px solid #ADD8E6; padding: 5px;"> <p> 说明 为了确保ClickHouse后端节点与您的MaxCompute服务之间网络通畅，<i>tunnel-endpoint</i> 必须是VPC网络类型的endpoint，且MaxCompute必须与您的ClickHouse实例在同一地域，比如都在上海Region。查看MaxCompute tunnel endpoint地址请参见Endpoint。</p> </div>
project-name	MaxCompute中数据源所在的项目名。 <div style="border: 1px solid #ADD8E6; padding: 5px;"> <p> 说明 必须完全与MaxCompute中的名称保持一致。</p> </div>
table-name	MaxCompute中数据源所在的表名。 <div style="border: 1px solid #ADD8E6; padding: 5px;"> <p> 说明 必须完全与MaxCompute中的名称保持一致。</p> </div>
partition-spec	希望访问的MaxCompute分区。 需要指定特定分区，在多个分区表达式之间不能有空格，且不要包含单引号，比如正确的示例为： <i>sale_date=2020-09-01,region=beijing</i> 。
access-key-id	您在访问MaxCompute上的数据源时所持有的AccessKey ID。 <div style="border: 1px solid #ADD8E6; padding: 5px;"> <p> 说明 必须对MaxCompute中目标数据相应的项目、表有读取权限。</p> </div>
access-key-secret	您在访问MaxCompute上的数据源时所持有的AccessKey Secret。

示例

```
CREATE TABLE default.odps_src_tbl on cluster default
(
  `shop_name` String,
  `customer_id` String,
  `total_price` Float64
)
ENGINE = MaxCompute('http://dt.cn-shanghai.maxcompute.aliyun-inc.com', 'test_project'
, 'test_tbl', 'sale_date=2020-09-01,region=beijing', '<your-access-key-id>', '<your-a
ccess-key-secret>');
```

2. 创建目标表。

```
CREATE TABLE default.odps_dst_tbl_local on cluster default
(
  `shop_name` String,
  `customer_id` String,
  `total_price` Float64
)
ENGINE= MergeTree()
ORDER BY customer_id;
CREATE TABLE default.odps_dst_tbl_dist on cluster default
as default.odps_dst_tbl_local
ENGINE= Distributed(default, default, odps_dst_tbl_local, rand());
```

3. 将外部表中数据导入目标表。

```
insert into odps_dst_tbl_dist select * from odps_src_tbl;
```

类型映射

MaxCompute类型	ClickHouse类型
boolean	UInt8
tinyint	UInt8, Int8
smallint	UInt16, Int16
int	UInt32, Int32
bigint	UInt64, Int64
float	Float32
double	Float64
char	String
varchar	String
binary	String
string	String

MaxCompute类型	ClickHouse类型
date	UInt16
datetime	UInt32
UUID	暂不支持
INTERVAL	暂不支持
Decimal	暂不支持
Timestamp	暂不支持
复合数据类型	暂不支持

5.3.2.2. 从MaxCompute外表导入（20.3及以上）

本文介绍如何将数据通过阿里云MaxCompute外表方式导入至云数据库ClickHouse。

注意事项

- 云原生版集群不支持通过阿里云MaxCompute外表方式导入至云数据库ClickHouse。
- 云数据库ClickHouse社区兼容版集群的内核版本为20.3及以上。
- MaxCompute服务为付费服务。计费详情，请参见[MaxCompute计费方式](#)。
- 为了确保云数据库ClickHouse与MaxCompute服务之间网络通畅，您需要采用阿里云VPC网络访问方式，且MaxCompute必须与您的云数据库ClickHouse集群在同一地域。

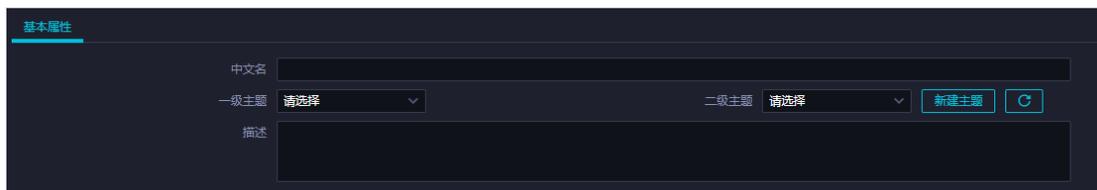
前提条件

- 已开通MaxCompute服务。如何开通，请参见[开通MaxCompute](#)。
- 已创建MaxCompute项目，且与云数据库ClickHouse集群在同一地域。如何创建，请参见[创建MaxCompute项目](#)。
- 如果RAM用户使用MaxCompute，需要阿里云主账号为RAM用户授权。如何授权，请参见[准备RAM用户](#)。

操作步骤

1. 创建MaxCompute外表。
 - i. 登录[MaxCompute控制台](#)。
 - ii. 在[项目管理](#)页签，单击相应项目后的[数据开发](#)。
 - iii. 在[数据开发](#)页面，鼠标悬停至 **+新建** 图标，单击MaxCompute > 表。
 - iv. 在新建表对话框中，输入表名，本文以maxcompute表名为例。
 - v. 单击新建。

vi. 在基本属性区域，配置各项参数。



参数含义如下。

名称	描述
中文名	MaxCompute表的中文名称。
一级主题	新建表所处的一级目标文件夹名称。 <div style="border: 1px solid #ccc; background-color: #e0f2f1; padding: 5px; border-radius: 5px;"> <p>? 说明 一级、二级主题仅仅是DataWorks上文件夹的摆放形式，目的是为了您能更好地管理您的表。</p> </div>
二级主题	新建表所处的二级目标文件夹名称。
新建主题	单击新建主题，进入主题管理页面，您可以在该页面创建一级主题、二级主题。 新建主题后，单击  图标，即可同步新建的主题。
描述	对新建表进行简单描述。

vii. 单击工具栏中的DDL模式。

viii. 在DDL模式对话框中，输入如下建表语句，单击生成表结构。

```
CREATE TABLE IF NOT EXISTS maxcompute
(
  v1 INT,
  v2 INT
)
PARTITIONED BY
(
  v3 STRING
);
```

ix. 在确认操作对话框中，单击确认。

x. 单击工具栏的提交到生产环境。

xi. 在弹出的提交生产确认对话框中，勾选我已悉知风险，确认提交。

xii. 单击确认。

2. 写入MaxCompute外表数据。

i. 在数据开发页面，单击左侧导航栏的临时查询。

ii. 鼠标悬停至图标，单击新建 > ODPS SQL。

iii. 在新建节点对话框中，输入节点名称，并选择目标文件夹。

- iv. 单击提交。
- v. 在节点的编辑页面，输入如下语句，为MaxCompute外表写入数据。

```
insert into maxcompute PARTITION (v3='2021') values (1,2), (2,3);
```

- vi. 单击工具栏中的图标。
- vii. 在参数窗口，选择公共调度资源组，单击确定。
- viii. 在MaxCompute计算成本估计窗口，确认预估费用，单击运行。

3. 创建云数据库ClickHouse表。

- i. 登录[云数据库ClickHouse控制台](#)。
- ii. 在[集群列表](#)页面，单击目标集群ID。
- iii. 单击右上方导航栏的[登录数据库](#)。
- iv. 输入建表语句并单击执行（F8）。

建表语法如下。

```
CREATE TABLE <table_name> [on cluster default]
(
  'col_name1' col_type1,
  'col_name2' col_type2,
  ...
)
ENGINE = MaxCompute('<tunnel-endpoint>', '<project-name>', '<table-name>', '<partition-spec>', '<access-key-id>', '<access-key-secret>', <read-thread-num>);
```

 **说明** 如果您使用的表引擎为MaxComputeRaw，请更改建表语法中的 `tunnel-endpoint` 为 `odps-endpoint`，即VPC网络Endpoint。如何查看VPC网络Endpoint，请参见[Endpoint](#)。

参数说明如下。

参数名	描述
<code>table_name</code>	表名。
<code>col_name1, col_name2</code>	列名。
<code>col_type1, col_type2</code>	列类型。  说明 云数据库ClickHouse表的结构类型需与MaxCompute对应。具体映射关系，请参见 数据类型映射 。

参数名	描述
tunnel-endpoint	<p>Tunnel服务的连接地址Tunnel Endpoint。</p> <p>为了确保云数据库ClickHouse与MaxCompute服务之间网络通畅，您需要采用阿里云VPC网络访问方式，且MaxCompute必须与您的云数据库ClickHouse集群在同一地域。</p> <p> 说明 如何查看VPC网络的 tunnel-endpoint ，请参见Endpoint。</p>
project-name	MaxCompute项目名称。
table-name	MaxCompute外表名。
partition-spec	MaxCompute分区。
access-key-id	访问MaxCompute项目的AccessKey ID。
access-key-secret	访问MaxCompute项目的AccessKey Secret。
read-thread-num (可选)	读取MaxCompute表时单个分片的并发数，默认值是1。

建表语句如下。

```
create table default.maxcomputetest ON CLUSTER default (
  v1 Int32,
  v2 Int32
) ENGINE = MaxCompute('http://dt.cn-hangzhou.maxcompute.aliyun-inc.com', 'ckfw', '
maxcompute', 'v3=2021', 'LTAI5tDVcUKu2CbcBwhr****', 'WAcroHBcL7IVOK8FIHzNJy91Lc****
');
```

4. 查询云数据库ClickHouse表。

- i. 登录[云数据库ClickHouse控制台](#)。
- ii. 在[集群列表](#)页面，单击目标集群ID。
- iii. 单击右上方导航栏的[登录数据库](#)。

iv. 输入查询语句并单击执行（F8）。

```
select * from maxcomputetest;
```

查询结果如下。



数据类型映射

MaxCompute类型	ClickHouse类型
Boolean	UInt8
Tinyint	UInt8, Int8
Smallint	UInt16, Int16
Int	UInt32, Int32
Bigint	UInt64, Int64
Float	Float32
Double	Float64
Char	String
Varchar	String
Binary	String
String	String
Date	Date
Datetime	Datetime
UUID	暂不支持
INTERVAL	暂不支持

MaxCompute类型	ClickHouse类型
Decimal	String
Timestamp	暂不支持
Map	<p>Nested(Nullable(keyType), Nullable(valueType))。目前只支持一层非嵌套的Map结构。</p> <p>例如MaxCompute中map<string, string>对应云数据库ClickHouse中的类型是：Nested(k Nullable(String), v Nullable(String))。</p> <p> 说明 Nested内部的类型必须是Nullable，否则建表会报错。</p>
Array	<p>Array(Nullable(Type))。目前只支持一层非嵌套的Array结构。</p> <p>例如MaxCompute中array<int>对应云数据库ClickHouse中的类型是：Array(Nullable(Int32))。</p> <p> 说明 Array内部的类型必须是Nullable，否则建表会报错。</p>
Struct	<p>Tuple(Nullable(Type))。目前只支持一层非嵌套的Struct结构。</p> <p>例如MaxCompute中struct<x:int, y:string>对应云数据库ClickHouse中的类型是：Tuple(Nullable(Int32), Nullable(String))。</p> <p> 说明 Tuple内部的类型必须是Nullable，否则建表会报错。</p>

5.4. 从RDS MySQL导入/同步

5.4.1. 从RDS MySQL导入

本文介绍如何从阿里云关系型数据库RDS（Relational Database Service）MySQL中导入数据至云数据库ClickHouse。

注意事项

- RDS MySQL为付费服务。计费详情，请参见[收费项、计费方式与价格](#)。
- 请确保数据源RDS MySQL实例和目的云数据库ClickHouse集群在同一地域并使用相同的VPC。

前提条件

- 已创建云数据库ClickHouse集群，如何创建，请参见[新建集群](#)。
- 已创建RDS MySQL实例，如何创建，请参见[创建RDS MySQL实例](#)。
- 已为RDS MySQL实例创建数据库和账号，如何创建，请参见[创建数据库和账号](#)。
- 已将云数据库ClickHouse集群的IP地址添加到RDS MySQL的白名单中，如何添加，请参见[设置IP白名单](#)。

 **说明** 您可以通过 `select * from system.clusters;` 命令查看云数据库ClickHouse集群的IP地址。

操作步骤

1. 创建RDS MySQL表并写入数据

- i. 登录[RDS管理控制台](#)。
- ii. 在实例列表页面，单击目标实例ID。
- iii. 单击右上方的登录数据库。
- iv. 在登录实例页面，输入数据库账号和密码，单击登录。
- v. 执行建表语句，以下示例为在数据库testdb中创建表mysql_test_table。

```
CREATE TABLE testdb.mysql_test_table (  
  v1 Int NOT NULL,  
  v2 Int DEFAULT NULL,  
  v3 Float DEFAULT NULL  
) ENGINE=InnoDB;
```

- vi. 向RDS MySQL表写入数据。

```
insert into testdb.mysql_test_table values (4,4,4.0), (1,1,2.0);
```

2. 创建云数据库ClickHouse表

- i. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
- ii. 在页面左上角，选择目标集群所在的地域。
- iii. 在集群列表页面，选择默认实例列表或云原生版本实例列表，单击目标集群ID。
- iv. 如果您的集群是社区兼容版，请单击右上方导航栏的登录数据库。
如果您的集群是云原生版，请单击集群计算组操作列的登录。
- v. 在登录实例页面，输入数据库账号和密码，单击登录。
- vi. 创建本地表。

```
create table default.clickhouse_test_table ON CLUSTER default (  
  v1 Int32,  
  v2 Nullable(Int32),  
  v3 Nullable(Float32)  
) ENGINE = MergeTree ORDER BY v1;
```

说明

- 云数据库ClickHouse表的结构类型需与RDS MySQL表对应。具体映射关系，请参见[数据类型映射](#)。
- RDS MySQL建表语句中未指定NOT NULL的列，值可以为NULL。同时，云数据库ClickHouse建表语句中的对应列使用Nullable进行标识。

vii. 创建分布式表。

说明 如果您只需要导入RDS MySQL数据至本地表，可跳过此步骤。

```
CREATE TABLE clickhouse_test_table_distributed ON CLUSTER default
AS clickhouse_test_table
ENGINE = Distributed(default, default, clickhouse_test_table, rand());
```

3. 导入RDS MySQL数据至云数据库ClickHouse

语法如下。

```
insert into <ClickHouse表名> select * from mysql('<RDS MySQL连接地址>:<端口>', '<RDS MySQL数据库>', '<RDS MySQL表名>', '<RDS MySQL数据库账号>', '<RDS MySQL数据库账号的密码>')
```

说明 如何获取RDS MySQL的连接地址和端口，请参见[通过客户端、命令行连接RDS MySQL实例](#)。

示例如下。

```
insert into clickhouse_test_table_distributed select * from mysql('rm-bp16t9h3999xb****.mysql.rds.aliyuncs.com:3306', 'testdb', 'mysql_test_table', 'test', '123456Aa');
```

4. 查询导入到云数据库ClickHouse的数据

```
select * from clickhouse_test_table_distributed;
```

说明 如果您导入的是本地表，请将查询语句中的分布式表名更换为本地表名，再进行查询。

查询结果如下。

```
┌─v1─┬─v1─┬─v3─┐
│ 4   │ 4   │ 4.0 │
├───┬───┬───┤
│ 1   │ 1   │ 2.0 │
└───┬───┬───┘
```

数据类型映射

RDS MySQL类型	ClickHouse类型
Unsigned tinyint	UInt8
Tinyint	Int8
Unsigned smallint	UInt16
Smallint	Int16
Unsigned int, Unsigned mediumint	UInt32
Int, Mediumint	Int32

RDS MySQL类型	ClickHouse类型
Unsigned bigint	UInt64
Bigint	Int64
Float	Float32
Double	Float64
Date	Date
Datetime, Timestamp	DateTime
Binary	FixedString
其他	String

5.4.2. 从RDS MySQL同步

云数据库ClickHouse支持通过创建RDS MySQL分析实例将RDS MySQL数据同步至云数据库ClickHouse。

背景信息

RDS MySQL分析实例通过在云数据库ClickHouse中自动创建MaterializedMySQL引擎库，将RDS MySQL数据同步到云数据库ClickHouse中，利用其超高的分析性能解决多维聚合实时分析问题。

 **说明** 关于MaterializeMySQL引擎，请参见[MaterializeMySQL引擎](#)。

使用限制

- 云数据库ClickHouse的云原生版集群不支持RDS MySQL分析实例。
- 目标实例只支持20.8及其以上版本的云数据库ClickHouse社区兼容版集群。
- 用户具有RDS MySQL的Binlog访问权限以及RDS MySQL数据库的RELOAD、REPLICATION SLAVE、REPLICATION CLIENT和SELECT权限。

 **说明** 您可以通过 `GRANT RELOAD, REPLICATION SLAVE, REPLICATION CLIENT, SELECT ON *.* TO 'your-user-name';` 命令获取RDS MySQL数据库的相关权限。

- RDS MySQL实例版本需满足以下条件才可以创建RDS MySQL分析实例：
 - RDS MySQL 8.0三节点企业版
 - RDS MySQL 8.0高可用版（SSD云盘、ESSD云盘或本地SSD盘）
 - RDS MySQL 5.7三节点企业版
 - RDS MySQL 5.7高可用版（SSD云盘、ESSD云盘或本地SSD盘）
 - RDS MySQL 5.6
- RDS MySQL源实例和云数据库ClickHouse目标实例必须在同一地域、同一VPC网络。
- 同一集群的同一数据库不能重复配置同步任务。
- 没有主键的表无法同步。

- 云数据库ClickHouse的 `DateTime` 类型数据的时间范围为 `[1970-01-01 00:00:00,2106-02-07 06:28:15]`，如果RDS MySQL中的时间超出当前范围，会导致同步到云数据库ClickHouse的时间值不正确。

操作步骤

1. 创建同步任务。

- i. 登录云数据库ClickHouse控制台。

 说明 您也可以直接登录云数据库RDS控制台，参考步骤v及以后的操作步骤。

- ii. 在集群列表页面，选择默认实例列表，单击目标集群ID。
- iii. 在左侧导航栏单击MySQL实时同步。
- iv. 如果是首次创建同步任务，单击提示语句中的RDS for MySQL，跳转至云数据库RDS控制台。



 说明 如果您已经创建了同步任务，请登录云数据库RDS控制台创建同步任务。

- v. 在实例列表页面，单击目标实例ID。

vi. 在基本信息页面的实例分布区域，单击分析实例后的云ClickHouse分析实例。



2. 如果是首次创建分析实例，需要进行授权。具体步骤如下。

说明 若您之前已完成授权操作，单击云ClickHouse分析实例直接跳转至MySQL到ClickHouse分析实例数据同步配置页面。

- i. 在弹出的创建服务关联角色窗口，单击AliyunServiceRoleForClickHouse。
 - ii. 单击确认。
3. 配置数据同步信息。
- i. 在MySQL到ClickHouse分析实例数据同步配置页面，选择ClickHouse实例，并分别填写MySQL实例和ClickHouse实例对应的用户名和密码。



- ii. 单击测试连接以进行下一步，进行实例连通性校验。
 - 测试连接成功，进行下一步。
 - 测试连接失败，进行异常提示。请根据提示信息进行排查。

iii. 在源库对象框中选中待同步的表。

说明

- 默认会选中忽略不支持同步的表结构，建议保持选中状态。目前不支持同步无主键的表。
- 不能只选中同步库，只有选中要同步的表才能配置成功。

iv. 单击  图标将其同步至目标实例配置框。



v. 单击下一步：保存任务并启动同步。

- 配置成功，进行下一步。
- 配置失败，进行异常提示。请按照提示信息进行排查。

vi. (可选) 配置成功后，数据同步任务启动，单击点击查看同步任务。

说明 如果同步任务时出现异常提示：task inner error，请及时提交工单。

5.5. 从SLS同步

本文介绍如何使用日志投递功能从日志服务（SLS）同步数据到云数据库ClickHouse。

注意事项

- 云数据库ClickHouse的云原生版集群和在2021年12月份之后创建的社区兼容版集群不支持日志投递功能。
- 云数据库ClickHouse目前不支持数组类型的数据。支持的数据类型，具体请参见[数据类型](#)。
- SLS服务为付费服务。计费详情，请参见[计费概述](#)。
- 请确保SLS的Project与您的云数据库ClickHouse集群在同一地域。

前提条件

- 已开通SLS服务。如何开通，请参见[开通日志服务](#)。
- 已在云数据库ClickHouse集群所在地域创建SLS的Project和Logstore，并接入了同步数据。如何创建和接入数据，请参见[快速入门](#)。

- 如果使用RAM用户投递SLS日志到云数据库ClickHouse，需要阿里云主账号为RAM用户授予AliyunLogReadOnlyAccess权限。如何授权，请参见[创建RAM用户及授权](#)。
- 已创建与SLS数据结构和类型对应的云数据库ClickHouse表。如何创建，请参见[创建表](#)。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在集群列表页面，选择默认实例列表，单击目标集群ID。
3. 在左侧导航选择日志投递，单击新建SLS日志投递。
4. 在SLS数据投递页面，配置各项参数。参数含义如下。

参数	说明
投递名称	投递任务的名称。
投递描述	投递任务的描述。
SLS名称	已创建的Project名称。
SLS日志库	已创建的Logstore名称。
集群名称	云数据库ClickHouse的集群ID。
数据库名称	数据库。
表名	数据库中的表名。
账号名称	数据库账号。
账号密码	数据库账号的密码。
字段映射	日志字段和云数据库ClickHouse表字段的映射。
投递开始时间	日志写入SLS的时间。例如，选择时间为 <code>2021-12-31 08:00</code> ，即把 <code>2021-12-31 08:00</code> 之后的日志数据投递到云数据库ClickHouse。

参数	说明
是否过滤脏数据	<p>是否对脏数据进行过滤。建议打开开关，如果关闭，遇到无法解析的脏数据，会导致投递任务失败。</p> <p>说明</p> <p>具体的脏数据记录，您可以单击投递任务操作列的监控查看。</p>

5. 单击**确定**。

6. 成功配置SLS数据投递后，您可以在**日志投递**页面查看投递任务。

任务状态从**待同步**变更为**同步中**后，即成功开启SLS日志投递。

任务名称	状态	SLS项目	SLS日志库	ClickHouse库	ClickHouse表	创建时间	操作
SLS-test	同步中	ck-test	ck-test1	default	sls_test	2021-12-28 22:56:36	监控 修改 停止 删除

每页显示 30 | 50 | 100

< 上一页 1 下一页 >

说明

您可以对投递任务进行**监控、修改、停止和删除**操作。

5.6. 从OSS导入

本文介绍如何将数据从阿里云对象存储OSS（Object Storage Service）导入至云数据库ClickHouse。

注意事项

- 云数据库ClickHouse的云原生版集群不支持从OSS导入数据。
- OSS服务为付费服务。计费详情，请参见[计费概述](#)。
- 请确保OSS存储空间（Bucket）与您的云数据库ClickHouse集群在同一地域，并使用阿里云VPC网络访问OSS。

前提条件

- 已开通OSS服务。如何开通，请参见[开通OSS服务](#)。
- 已在云数据库ClickHouse集群所在地域创建存储空间（Bucket）。如何创建，请参见[创建存储空间](#)。
- 访问OSS的账号已具备对OSS对象的读权限。如何设置权限，请参见[访问控制概述](#)。

操作步骤

1. 准备测试数据

将如下示例数据，存储为ceshi.csv。

```
1,yang,32,shanghai,http://example.com
2,wang,22,beijing,http://example.com
3,xiao,23,shenzhen,http://example.com
4,jess,45,hangzhou,http://example.com
5,jack,14,shanghai,http://example.com
6,tomy,25,hangzhou,http://example.com
7,lucy,45,shanghai,http://example.com
8,tengyin,26,shanghai,http://example.com
9,wangli,27,shenzhen,http://example.com
10,xiaohua,37,shanghai,http://example.com
```

2. 上传数据至OSS

- i. 登录[OSS管理控制台](#)。
- i. 单击左侧导航栏的[Bucket列表](#)。
- ii. 单击目标Bucket名称。
- iii. 在[文件管理](#)页签，单击[上传文件](#)。
- iv. 在[上传文件](#)页面，上传示例数据集ceshi.csv。

 **说明** 详细参数配置，请参见[上传文件](#)。

- v. 单击[上传文件](#)。

3. 创建云数据库ClickHouse表

- i. 登录[云数据库ClickHouse控制台](#)。
- ii. 在[集群列表](#)页面，选择默认实例列表，单击目标集群ID。
- iii. 单击右上方导航栏的[登录数据库](#)。
- iv. 在[登录实例](#)页面，输入数据库账号和密码，单击[登录](#)。

v. 创建本地表。

 **说明** 云数据库ClickHouse表的结构类型需与OSS数据对应。

- 如果您的集群是单副本版，请输入如下建表语句。

```
create table oss_test_tbl_local on cluster default
(
  id UInt8,
  user_name String,
  age UInt16,
  city String,
  access_url String
)
engine = MergeTree()
order by id;
```

- 如果您的集群是双副本版，请输入如下建表语句。

```
create table oss_test_tbl_local on cluster default
(
  id UInt8,
  user_name String,
  age UInt16,
  city String,
  access_url String
)
engine = ReplicatedMergeTree('/clickhouse/tables/{database}/{table}/{shard}', '{r
eplica}')
order by id;
```

vi. 创建分布式表。

 **说明** 如果您只需要导入OSS数据至本地表，可跳过此步骤。

```
create table oss_test_tbl_distributed on cluster default
(
  id UInt8,
  user_name String,
  age UInt16,
  city String,
  access_url String
)
engine = Distributed(default, default, oss_test_tbl_local, rand());
```

4. 导入OSS数据至云数据库ClickHouse

- 方法一：通过表函数导入OSS数据

导入语法如下。

```
insert into oss_test_tbl_distributed select * from oss('<oss-endpoint>', '<access-key
-id>', '<access-key-secret>', '<oss-file-path>', '<file-format-name>', '<column-defin
itions>');
```

CSV文件默认列分隔符为 `,`，如果您导入的CSV文件为其他列分隔符，需要在导入语句中通过 `format_csv_delimiter` 指定其他列分隔符。例如CSV文件列与列之间使用竖线分割，则导入语法如下。

```
insert into oss_test_tbl_distributed select * from oss('<oss-endpoint>', '<access-key-id>', '<access-key-secret>', '<oss-file-path>', '<file-format-name>', '<column-definitions>') settings format_csv_delimiter='|';
```

参数说明如下。

参数名	描述
<code>oss-endpoint</code>	ECS的VPC网络访问（内网）的Endpoint。如何获取，请参见 访问域名和数据中心 。 说明 请确保OSS存储空间（Bucket）与您的云数据库ClickHouse集群在同一地域。
<code>access-key-id</code>	访问OSS数据的AccessKey ID。
<code>access-key-secret</code>	访问OSS数据的AccessKey Secret。
<code>oss-file-path</code>	文件的存储路径。一般格式为 <code>oss://<bucket-name>/<path-to-file></code> 。 说明 <code>oss-file-path</code> 参数支持通过通配符进行模糊匹配。更多信息，请参见 通配符模糊匹配OSS的存储路径 。
<code>file-format-name</code>	文件的格式，本文为CSV。
<code>column-definitions</code>	OSS数据的列名、列类型。

示例如下。

```
insert into oss_test_tbl_distributed select * from oss('oss-cn-shanghai-internal.aliyuncs.com', 'LTAI5tDVcUKu2CbcBwhr****', 'WAcroHBcL7IVOK8FIHzNJy91Lc****', 'oss://testBucketName/ceshi.csv', 'CSV', 'id UInt8, user_name String, age UInt16, city String, access_url String');
```

- 方法二：通过表引擎导入OSS数据
 - a. 创建OSS外表。
建表语法如下。

```
CREATE TABLE <table_name> [on cluster default]
(
  'col_name1' col_type1,
  'col_name2' col_type2,
  ...
)
ENGINE = OSS('<oss-endpoint>', '<access-key-id>', '<access-key-secret>', '<oss-file-path>', '<file-format-name>');
```

参数说明如下。

参数名	描述
<code>table_name</code>	表名。
<code>col_name1, col_name2</code>	列名。
<code>col_type1, col_type2</code>	列类型。 说明 OSS外表的结构类型需与OSS数据对应。
<code>oss-endpoint</code>	ECS的VPC网络访问（内网）的Endpoint。如何获取，请参见 访问域名和数据中心 。 说明 请确保OSS存储空间（Bucket）与您的云数据库ClickHouse集群在同一地域。
<code>access-key-id</code>	访问OSS数据的AccessKey ID。
<code>access-key-secret</code>	访问OSS数据的AccessKey Secret。
<code>oss-file-path</code>	文件的存储路径。一般格式为 <code>oss://<bucket-name>/<path-to-file></code> 。 说明 <code>oss-file-path</code> 参数支持通过通配符进行模糊匹配。更多信息，请参见 通配符模糊匹配OSS的存储路径 。
<code>file-format-name</code>	文件的格式，本文为CSV。

示例如下。

```
CREATE TABLE oss_test_tbl on cluster default
(
  id UInt8,
  user_name String,
  age UInt16,
  city String,
  access_url String
)
ENGINE = OSS('oss-cn-shanghai-internal.aliyuncs.com', 'LTAI5tDVcUKu2CbcBwhr****',
'WAcroHBcL7IVOK8FIHzNJy91Lc****', 'oss://testBucketName/ceshi.csv', 'CSV');
```

b. 将OSS外表数据导入至云数据库ClickHouse。

```
insert into oss_test_tbl_distributed select * from oss_test_tbl;
```

CSV文件默认列分隔符为 `,`，如果您导入的CSV文件为其他列分隔符，需要在导入语句中通过 `format_csv_delimiter` 指定其他列分隔符。例如CSV文件列与列之间使用竖线分割，则命令示例如下。

```
insert into oss_test_tbl_distributed select * from oss_test_tbl settings format_csv_delimiter='|';
```

5. 查询云数据库ClickHouse表

执行查询语句。

```
select * from oss_test_tbl_distributed;
```

 **说明** 如果您导入的是本地表，请将查询语句中的分布式表名更换为本地表名，再进行查询。

查询结果如下。

序号	id	user_name	age	city	access_url
1	1	yang	32	shanghai	http://example.com
2	2	wang	22	beijing	http://example.com
3	3	xiao	23	shenzhen	http://example.com
4	4	jess	45	hangzhou	http://example.com
5	5	jack	14	shanghai	http://example.com
6	6	tomy	25	hangzhou	http://example.com
7	7	lucy	45	shanghai	http://example.com
8	8	tengyin	26	shanghai	http://example.com
9	9	wangli	27	shenzhen	http://example.com
10	10	xiaohua	37	shanghai	http://example.com

通配符模糊匹配OSS的存储路径

OSS中通常存在多个具备相同命名规则的小文件，为了简化对于小文件的分析，`oss-file-path` 参数支持通过如下通配符进行模糊匹配。

- `*` : 匹配任意文件名、目录名。如 `/dir/*` 匹配 `/dir` 下的所有文件。
- `{x, y, z}` : 匹配大括号中的任意一个值。如 `file_{x,y,z}` 匹配 `file_x` 、 `file_y` 或 `file_z` 。
- `{num1..num2}` : 匹配[num1,num2]展开后的任意一个值。如 `file_{1..3}` 匹配 `file_1` 、 `file_2` 或 `file_3` 。
- `?` : 匹配任意一个单字符。如 `file_?` 匹配 `file_a` 、 `file_b` 、 `file_c` 等等。

示例

上传文件的目录结构如下。

```
oss://testBucketName/  
  doc-data/  
    oss-import/  
      small_files/  
        access_log_csv_1.txt  
        access_log_csv_2.txt  
        access_log_csv_3.txt
```

`oss-file-path` 参数的部分参考取值如下。

- `oss://testBucketName/doc-data/oss-import/small_files/*`
- `oss://testBucketName/doc-data/oss-import/small_files/access*`
- `oss://testBucketName/doc-data/oss-import/small_files/access_log_csv_{1,2,3}.txt`
- `oss://testBucketName/doc-data/oss-import/*/access_log_csv_{1,2,3}.txt`
- `oss://testBucketName/doc-data/oss-import/**`
- `oss://testBucketName/doc-data/oss-import/*/access_log_csv_{1..3}.txt`
- `oss://testBucketName/doc-data/oss-import/*/access_log_csv_?.txt`

5.7. 访问OSS数据

本文介绍如何通过云数据库ClickHouse查询阿里云对象存储OSS（Object Storage Service）数据以及如何写入数据至OSS。

注意事项

- 云数据库ClickHouse的云原生版集群不支持访问OSS数据。
- 云数据库ClickHouse支持访问多种格式的OSS文件，除不支持Protobuf和CapnProto外，其他均支持，具体请参见[ClickHouse支持的文件格式](#)。
- 请确认OSS存储空间（Bucket）与您的云数据库ClickHouse集群在同一地域，并使用阿里云VPC网络访问OSS。
- 通过云数据库ClickHouse写入数据至OSS，会覆盖原来的OSS文件数据。

前提条件

- 已开通OSS服务。如何开通，请参见[开通OSS服务](#)。
- 已在云数据库ClickHouse所在地域创建存储空间（Bucket）。如何创建，请参见[创建存储空间](#)。
- 已上传文件至OSS。如何上传，请参见[上传文件](#)。

 说明 本文以 `1,tick,32,shanghai,http://example.com` 为示例数据，存储为ceshi.csv。

- 访问OSS的账号已具备对OSS对象的读写权限。如何设置权限，请参见[访问控制概述](#)。

操作步骤

1. 登录云数据库ClickHouse数据库

- i. 登录[云数据库ClickHouse控制台](#)。
- ii. 在[集群列表](#)页面，选择默认实例列表，单击目标集群ID。
- iii. 单击右上方导航栏的登录数据库。
- iv. 在[登录实例](#)页面，输入数据库账号和密码，单击登录。

2. 通过云数据库ClickHouse查询OSS数据

云数据库ClickHouse提供了表引擎和表函数两种方法查询OSS数据。

○ 方法一：通过表引擎查询OSS数据

 说明 如果您需要写入数据至OSS，请使用此方法先创建外表，再查询数据。

a. 创建OSS外表。

建表语法如下。

```
CREATE TABLE <table_name> [on cluster default]
(
  'col_name1' col_type1,
  'col_name2' col_type2,
  ...
)
ENGINE = OSS('<oss-endpoint>', '<access-key-id>', '<access-key-secret>', '<oss-file-path>', '<file-format-name>');
```

参数说明如下。

参数名	描述
<code>table_name</code>	表名。
<code>col_name1, col_name2</code>	列名。
<code>col_type1, col_type2</code>	列类型。  说明 OSS外表的结构类型需与OSS数据对应。

参数名	描述
<code>oss-endpoint</code>	ECS的VPC网络访问（内网）的Endpoint。如何获取，请参见 访问域名和数据中心 。 说明 请确保OSS存储空间（Bucket）与您的云数据库ClickHouse集群在同一地域。
<code>access-key-id</code>	访问OSS数据的AccessKey ID。
<code>access-key-secret</code>	访问OSS数据的AccessKey Secret。
<code>oss-file-path</code>	文件的存储路径。一般格式为 <code>oss://<bucket-name>/<path-to-file></code> 。 说明 <code>oss-file-path</code> 参数支持通过通配符进行模糊匹配。更多信息，请参见 通配符模糊匹配OSS的存储路径 。
<code>file-format-name</code>	文件的格式，本文为CSV。

示例如下。

```
CREATE TABLE oss_test_tbl on cluster default
(
  id UInt8,
  user_name String,
  age UInt16,
  city String,
  access_url String
)
ENGINE = OSS('oss-cn-shanghai-internal.aliyuncs.com', 'LTAI5tDVcUKu2CbcBwhr****',
'WAcroHBcL7IVOK8FIHzNJy91Lc****', 'oss://testBucketName/ceshi.csv', 'CSV');
```

b. 查询外表数据，示例如下。

```
select * from oss_test_tbl;
```

查询结果如下。

id	user_name	age	city	access_url
1	tick	32	shanghai	http://example.com

o 方法二：通过表函数查询OSS数据

查询语法如下。

```
SELECT * FROM oss('<oss-endpoint>', '<access-key-id>', '<access-key-secret>', '<oss-file-path>', '<file-format-name>', '<column-definitions>');
```

参数说明如下。

参数名	描述
<code>oss-endpoint</code>	ECS的VPC网络访问（内网）的Endpoint。如何获取，请参见 访问域名和数据中心 。 ❓ 说明 请确保OSS存储空间（Bucket）与您的云数据库ClickHouse集群在同一地域。
<code>access-key-id</code>	访问OSS数据的AccessKey ID。
<code>access-key-secret</code>	访问OSS数据的AccessKey Secret。
<code>oss-file-path</code>	文件的存储路径。一般格式为 <code>oss://<bucket-name>/<path-to-file></code> 。 ❓ 说明 <code>oss-file-path</code> 参数支持通过通配符进行模糊匹配。更多信息，请参见 通配符模糊匹配OSS的存储路径 。
<code>file-format-name</code>	文件的格式，本文为CSV。
<code>column-definitions</code>	OSS数据的列名、列类型。

示例如下。

```
SELECT * FROM oss('oss-cn-shanghai-internal.aliyuncs.com', 'LTAI5tDVcUKu2CbcBwhr****', 'WAcroHBcL7IVOK8FIHzNjy91Lc****', 'oss://testBucketName/ceshi.csv', 'CSV', 'id UInt8, user_name String, age UInt16, city String, access_url String');
```

查询结果如下。

id	user_name	age	city	access_url
1	tick	32	shanghai	http://example.com

3. 通过云数据库ClickHouse写入数据至OSS

❓ 说明 请确认您已经参照[步骤2](#)创建了OSS外表 `oss_test_tbl`。

```
insert into oss_test_tbl values(11, 'tick', 25, 'shanghai', 'http://example.com');
```

4. 查询OSS外表数据

```
select * from oss_test_tbl;
```

查询结果如下。

id	user_name	age	city	access_url
11	tick	25	shanghai	http://example.com

5.8. 从自建ClickHouse迁移上云

本文主要介绍如何从自建的ClickHouse将数据库表及数据迁移到云数据库ClickHouse上。

前提条件

出于安全考虑，云数据库ClickHouse无法直接访问公网，只能访问同一VPC内的其他服务器。所以数据迁移前，请首先确认源实例与目标实例云数据库ClickHouse之间是否网络通畅。

- 如果源实例是阿里云自建的ClickHouse实例，目标实例是云数据库ClickHouse，且在同一个VPC内，直接使用remote函数进行迁移。
- 如果源实例是阿里云自建的ClickHouse实例，目标实例是云数据库ClickHouse，但是属于不同VPC，首先需要将两个VPC网络连通，详细操作请参见[连接VPC](#)。网络连通完成后，使用remote函数进行数据迁移。
- 如果源实例不在阿里云上，例如在线下自建IDC内，首先需要将线下IDC与阿里云VPC连通，详细操作请参见[连接本地IDC](#)。网络连通完成后，使用remote函数进行数据迁移。
- 如果无法进行网络连通操作，建议在源实例将数据导出为文件，然后通过clickhouse-client将文件导入到目标实例云数据库ClickHouse。
- 如果无法进行网络连通操作，但是已经有了Spark、Flink等基础设施，也可以尝试编写Spark、Flink job将源实例数据读出，然后写入目标实例云数据库ClickHouse。

元数据的迁移

ClickHouse元数据的迁移，主要指进行建表DDL迁移。

如需安装clickhouse-client工具，请安装与目标实例云数据库ClickHouse版本一致的clickhouse-client工具，下载链接请参见[clickhouse-client](#)，如何安装和使用请参见[社区文档](#)。

步骤一：查看源实例的database列表

```
clickhouse-client --host="<old host>" --port="<old port>" --user="<old user name>" --password="<old password>" --query="SHOW databases" > database.list
```

参数解释如下：

参数	描述
old host	源实例的地址。
old port	源实例的端口。
old user name	登录源实例的账号，拥有DML读写和设置权限，允许DDL权限。
old password	上述账号对应的密码。

 说明 system是系统数据库，不需要迁移，可以直接过滤掉。

步骤二：查看源实例的table列表

```
clickhouse-client --host="<old host>" --port="<old port>" --user="<old user name>" --password="<old password>" --query="SHOW tables from <database_name>" > table.list
```

<database_name>表示希望查看的database。其他参数描述可参见“步骤一：查看源实例的database列表”。

您可以通过系统表直接查询所有database、table名称。

```
select distinct database, name from system.tables where database != 'system';
```

说明 查询到的表中，如果有以inner开头的表，则它们是物化视图的内部表示，不需要迁移，可以直接过滤掉。

步骤三：导出源实例的建表DDL

```
clickhouse-client --host="<old host>" --port="<old port>" --user="<old user name>" --password="<old password>" --query="SHOW CREATE TABLE <database_name>.<table_name>" > table.sql
```

步骤四：将建表DDL导入到目标实例云数据库ClickHouse

```
clickhouse-client --host="<new host>" --port="<new port>" --user="<new user name>" --password="<new password>" < table.sql
```

参数	描述
new host	目标实例云数据库ClickHouse的地址。
new port	目标实例云数据库ClickHouse的端口。
new user name	登录目标实例云数据库ClickHouse的账号，拥有DML读写和设置权限，允许DDL权限。
new password	上述账号对应的密码。

通过remote函数进行数据迁移

1. 将目标实例云数据库ClickHouse的网段加入到源实例的白名单中。

可以在云数据库ClickHouse控制台查看目标实例云数据库ClickHouse的VPC网络，并且将该VPC的整个网段都加入到源实例的白名单中。如果希望控制白名单的网段范围，避免潜在安全问题，那么可以通过如下SQL查询到目标实例云数据库ClickHouse的后台Server IP，只将这几个IP加入源实例的白名单中。

```
select * from system.clusters;
```

2. 在目标实例云数据库ClickHouse中，通过如下SQL进行数据迁移。

```
insert into <new_database>.<new_table> select * from remote('old_endpoint', <old_database>.<old_table>, '<username>', '<password>');
```

20.8版本优先使用remoteRaw函数进行数据迁移，如果失败可以申请小版本升级。

```
insert into <new_database>.<new_table> select * from remoteRaw('old_endpoint', <old_database>.<old_table>, '<username>', '<password>');
```

参数	描述
new_database	目标实例云数据库ClickHouse中的数据库名。
new_table	目标实例云数据库ClickHouse中的表名。
old_endpoint	源实例的endpoint。此处要填写的是VPC内endpoint，而不是公网endpoint。
old_database	源实例的数据库名。
old_table	源实例的表名。
username	源实例的账号。
password	源实例的密码。

通过文件导出导入方式进行数据迁移

通过文件，将数据从源实例数据库导出到目标实例云数据库ClickHouse中。

通过CSV文件导出导入

1. 将数据从源实例数据库导出为CSV格式文件。

```
clickhouse-client --host="<old host>" --port="<oldport>" --user="<old user name>" -
--password="<old password>" --query="select * from <database_name>.<table_name> FOR
MAT CSV" > table.csv
```

2. 导入CSV文件到目标实例云数据库ClickHouse。

```
clickhouse-client --host="<new host>" --port="<new port>" --user="<new user name>"
--password="<new password>" --query="insert into <database_name>.<table_name> FORM
AT CSV" < table.csv
```

通过Linux pipe管道进行流式导出导入

```
clickhouse-client --host="<old host>" --port="<old port>" --user="<user name>" --passwo
rd="<password>" --query="select * from <database_name>.<table_name> FORMAT CSV" |
clickhouse-client --host="<new host>" --port="<new port>" --user="<user name>" --passwo
rd="<password>" --query="INSERT INTO <database_name>.<table_name> FORMAT CSV"
```

5.9. 云数据库ClickHouse实例间的数据迁移

云数据库ClickHouse提供实例之间的数据迁移，由目的实例发起。您可以通过该功能实现相同内核版本实例之间或低版本实例向高版本实例的数据迁移。本文介绍如何在控制台上使用数据迁移功能。

注意事项

- 云数据库ClickHouse的云原生版实例不支持进行实例之间的数据迁移。
- 当前支持目的实例版本为20.8及以上，源实例版本不做限制。

- 源实例和目的实例的副本需一致。
- 源实例和目的实例需在同一个VPC或者已打通云企业网。
- 源实例和目的实例的冷热分层状态需一致。
- 迁移发起时，源实例和目的实例不能有正在运行的管控任务（包含扩容，升降配等）。
- 目的实例的存储空间需大于等于源实例的存储空间的1.2倍。
- 源实例的所有本地表都具有唯一的分布式表。
- 云实例数据迁移仅支持全量迁移。
- 非MergeTree类型（如外表、Log表等）仅进行结构迁移，不进行数据迁移。

前提条件

- 将源实例的IP地址添加到目的实例的白名单中。
- 将目的实例的IP地址添加到源实例的白名单中。

 **说明** 您可以通过 `select * from system.clusters;` 查看云数据库ClickHouse实例的IP地址。如何设置白名单，请参见[设置白名单](#)。

迁移影响

源实例和目的实例在迁移过程中以及迁移完成后受到的影响如下。

实例类型	迁移过程	迁移完成
源实例	可读，不可写。	支持读写。
目的实例	不支持读写。	支持读写。

操作步骤

1. 登录[ClickHouse控制台](#)。
2. 在[集群列表](#)页面，选择默认实例列表，单击目的集群ID。
3. 单击左侧导航栏的实例迁移。
4. 在实例迁移页面，单击创建迁移任务。



 **说明** 如果当前实例没有迁移记录，则显示没有数据。

5. 进入到任务配置环节，选择源实例ID，并分别填写源实例和目的实例的数据库账号和数据库密码。

The screenshot shows a configuration page for data migration with three steps: 1. Select source and target instances, 2. Migration content, and 3. Pre-check and start synchronization. The 'Source Instance Information' section includes fields for 'Instance ID', 'Database Account', and 'Database Password'. The 'Target Instance Information' section includes fields for 'Instance ID', 'Database Account', and 'Database Password'. A 'Test Connection to Proceed' button is visible at the bottom right.

注意 同地域实例之间可以进行迁移，不同地域之间不能进行数据迁移。

6. 单击**测试连接以进行下一步**，进行实例连通性校验。

- 测试连接成功，进行下一步。
- 测试连接失败。
 - 如果提示源实例或者目的实例账号密码认证失败，请重新填写**数据库账号**或**数据库密码**。
 - 如果提示源实例和目的实例网络连通性错误，请检查源实例是否和目的实例在同一个VPC网络或者已经打通云企业网。

说明 如何打通云企业网请参见[同账号VPC互连](#)。

7. 测试连接成功后，进入迁移内容确认环节。单击**下一步：预监测并启动同步**。

The screenshot shows the 'Migration Content' step of the migration process. A central message box states: '数据迁移内容包含：集群、库、表、数据字典、物化视图、用户权限和集群配置迁移' (Data migration content includes: clusters, databases, tables, data dictionaries, materialized views, user permissions, and cluster configuration migration). At the bottom, there are two buttons: '上一步：返回配置源实例和目标实例' (Previous step: Return to configure source instance and target instance) and '下一步：预检测并启动同步' (Next step: Pre-check and start synchronization).

说明 源实例可迁移内容包含：集群、库、表、数据字典、物化视图、用户权限和集群配置迁移。

8. 迁移任务配置完成后，后台进行迁移链路预监测和任务启动。

- 校验成功，单击**完成**。



- 校验失败，您需要按照提示信息进行操作，重新进行数据迁移。检测项目及要求如下。

检测项目	检测要求
实例状态检测	迁移发起时，源实例和目的实例不能有正在运行的管控任务（包含扩容，升降配等）。如果当前源实例和目的实例有管控任务正在运行，则不能发起迁移任务。
副本一致性检测	保持源实例和目的实例的副本一致。副本不一致，会导致参数缺失，无法进行副本的配置。
存储空间检测	迁移同步进行前，进行存储空间校验。保证目的实例的存储空间大于等于源实例的存储空间空间的1.2倍。
本地表和分布式表检测	如果存在本地表没有创建分布式表或者分布式表不唯一，则校验失败。请删除多余的分布式表。

9. 迁移任务完成后您可以在实例迁移页面查看迁移任务。

相关文档

如何从自建的ClickHouse将数据迁移到云数据库ClickHouse上，请参见[从自建ClickHouse迁移上云](#)。

5.10. 从本地数据导入

5.10.1. 从clickhouse-client导入

本文介绍如何通过clickhouse-client工具将本地文件的数据导入至云数据库ClickHouse。

注意事项

支持导入到云数据库ClickHouse的常见文件格式为TabSeparated、TabSeparatedWithNames、TabSeparatedWithNamesAndTypes、CSV和CSVWithNames。更多支持的文件格式，请参见[文件格式及说明](#)。

前提条件

- 已将本地机器的IP地址添加到云数据库ClickHouse的白名单中。
- 已安装与云数据库ClickHouse集群版本对应的clickhouse-client工具。下载链接，请参见[clickhouse-client](#)。

操作步骤

1. 准备测试数据

在clickhouse-client客户端的安装目录下，创建文件ceshi.csv并写入如下数据。

```
1,yang,32,shanghai,http://example.com
2,wang,22,beijing,http://example.com
3,xiao,23,shenzhen,http://example.com
4,jess,45,hangzhou,http://example.com
5,jack,14,shanghai,http://example.com
6,tomy,25,hangzhou,http://example.com
7,lucy,45,shanghai,http://example.com
8,tengyin,26,shanghai,http://example.com
9,wangli,27,shenzhen,http://example.com
10,xiaohua,37,shanghai,http://example.com
```

2. 创建云数据库ClickHouse表

- i. 登录[云数据库ClickHouse控制台](#)。
- ii. 在页面左上角，选择目标集群所在的地域。
- iii. 在[集群列表](#)页面，选择默认实例列表或云原生版本实例列表，单击目标集群ID。
- iv. 如果您的集群是社区兼容版，请单击右上方导航栏的[登录数据库](#)。
如果您的集群是云原生版，请单击集群计算组操作列的[登录](#)。
- v. 在[登录实例](#)页面，输入数据库账号和密码，单击[登录](#)。

vi. 创建本地表。

? 说明 云数据库ClickHouse表的结构类型需与本地文件数据对应。

- 如果您的集群是单副本版，请输入如下建表语句。

```
create table test_tbl_local on cluster default
(
  id UInt8,
  user_name String,
  age UInt16,
  city String,
  access_url String
)
engine = MergeTree()
order by id;
```

- 如果您的集群是双副本版，请输入如下建表语句。

```
create table test_tbl_local on cluster default
(
  id UInt8,
  user_name String,
  age UInt16,
  city String,
  access_url String
)
engine = ReplicatedMergeTree('/clickhouse/tables/{database}/{table}/{shard}', '{r
eplica}')
order by id;
```

vii. 创建分布式表。

? 说明 如果您只需要导入本地文件数据至本地表，可跳过此步骤。

```
create table test_tbl_distributed on cluster default
(
  id UInt8,
  user_name String,
  age UInt16,
  city String,
  access_url String
)
engine = Distributed(default, default, test_tbl_local, rand());
```

3. 导入本地文件数据

在clickhouse-client客户端的安装目录下执行如下命令。

```
cat <本地文件名> | ./clickhouse-client --host=<数据库连接地址> --port=<TCP端口号> --user=<
数据库账号> --password=<数据库账号的密码> --query="INSERT INTO <ClickHouse表名> FORMAT <本
地文件格式>";
```

- ② 说明 外网导入数据速度较慢，如果数据量较大或者测试导入性能，您可以采用如下方法。
- 内网导入：购买与云数据库ClickHouse集群在同一个VPC网络的ECS实例，从ECS通过VPC地址连接云数据库ClickHouse集群并导入数据。
 - 多进程导入：将原始文件等分为多份，启动多个client多进程并发导入。

云数据库ClickHouse支持的常用文件格式如下。

常用文件格式	说明
TabSeparated	数据按行写入文本文件中，列与列之间通过Tab分割。需要对Tab、换行符、反斜线进行转义处理，分别对应为： <code>\t</code> 、 <code>\n</code> 、 <code>\\</code> 。NULL值使用 <code>\N</code> 进行表示。
TabSeparatedWithNames	与TabSeparated类似，区别是第一行是列名。解析时，第一行会被忽略，因此不能够使用第一行列名来决定列的位置。
TabSeparatedWithNamesAndTypes	与TabSeparated类似，区别是第一行是列名，第二行是类型。解析时，第一行、第二行都会被忽略。
CSV	数据按行写入文件中。字符串类型使用英文双引号括起来，字符串本身的双引号用两个双引号进行转义，数值类型不使用双引号。默认列分隔符为 <code>,</code> ，您可以通过 <code>--format_csv_delimiter</code> 指定其他列分隔符，例如列与列之间使用竖线分割，则命令示例如下。 <pre>cat data.csv ./clickhouse-client --format_csv_delimiter=" " --host=cc-bp1631724nkf8****.clickhouse.ads.aliyuncs.com -- port=3306 --user=test --password=123456Aa --query="INSERT INTO test_tbl_distributed FORMAT CSV";</pre>
CSVWithNames	与CSV类似，区别是第一行表示列名。解析时，第一行会被忽略，因此不能够使用第一行列名来决定列的位置。

- ② 说明 云数据库ClickHouse支持的其他文件格式，详情请参见[文件格式及说明](#)。

4. 查询云数据库ClickHouse表

执行查询语句。

```
select * from test_tbl_distributed;
```

- ② 说明 如果您导入的是本地表，请将查询语句中的分布式表名更换为本地表名，再进行查询。

查询结果如下。

序号	id	user_name	age	city	access_url
1	1	yang	32	shanghai	http://example.com
2	2	wang	22	beijing	http://example.com
3	3	xiao	23	shenzhen	http://example.com
4	4	jess	45	hangzhou	http://example.com
5	5	jack	14	shanghai	http://example.com
6	6	tomy	25	hangzhou	http://example.com
7	7	lucy	45	shanghai	http://example.com
8	8	tengyin	26	shanghai	http://example.com
9	9	wangli	27	shenzhen	http://example.com
10	10	xiaohua	37	shanghai	http://example.com

5.10.2. 从Spark导入

本文介绍如何通过Spark程序导入数据至云数据库ClickHouse。

前提条件

- 已将本地机器的IP地址添加到云数据库ClickHouse的白名单中。如何添加，请参见[设置白名单](#)。
- 已创建与导入数据结构类型对应的云数据库ClickHouse表。如何创建，请参见[创建表](#)。

操作步骤

1. 准备Spark程序目录结构。

```
find .
.
./build.sbt
./src
./src/main
./src/main/scala
./src/main/scala/com
./src/main/scala/com/spark
./src/main/scala/com/spark/test
./src/main/scala/com/spark/test/WriteToCk.scala
```

2. 编辑build.sbt配置文件并添加依赖。

```
name := "Simple Project"
version := "1.0"
scalaVersion := "2.12.10"
libraryDependencies += "org.apache.spark" %% "spark-sql" % "3.0.0"
libraryDependencies += "ru.yandex.clickhouse" % "clickhouse-jdbc" % "0.2.4"
```

3. 创建WriteToCk.scala数据并写入程序文件。

```
package com.spark.test
import java.util
import java.util.Properties
import org.apache.spark.sql.execution.datasources.jdbc.JDBCOptions
import org.apache.spark.SparkConf
import org.apache.spark.sql.{SaveMode, SparkSession}
import org.apache.spark.storage.StorageLevel
object WriteToCk {
  val properties = new Properties()
  properties.put("driver", "ru.yandex.clickhouse.ClickHouseDriver")
  properties.put("user", "<yourUserName>")
  properties.put("password", "<yourPassword>")
  properties.put("batchsize", "100000")
  properties.put("socket_timeout", "300000")
  properties.put("numPartitions", "8")
  properties.put("rewriteBatchedStatements", "true")
  val url = "jdbc:clickhouse://<yourUrl>:8123/default"
  val table = "<yourTableName>"
  def main(args: Array[String]): Unit = {
    val sc = new SparkConf()
    sc.set("spark.driver.memory", "1G")
    sc.set("spark.driver.cores", "4")
    sc.set("spark.executor.memory", "1G")
    sc.set("spark.executor.cores", "2")
    val session = SparkSession.builder().master("local[*]").config(sc).appName("write-to-ck").getOrCreate()
    val df = session.read.format("csv")
      .option("header", "true")
      .option("sep", ",")
      .option("inferSchema", "true")
      .load("<yourFilePath>")
      .selectExpr(
        "colName1",
        "colName2",
        "colName3",
        ...
      )
      .persist(StorageLevel.MEMORY_ONLY_SER_2)
    println(s"read done")
    df.write.mode(SaveMode.Append).option(JDBCOptions.JDBC_BATCH_INSERT_SIZE, 100000).jdbc(url, table, properties)
    println(s"write done")
    df.unpersist(true)
  }
}
```

参数说明如下。

参数	说明
<code>yourUserName</code>	云数据库ClickHouse中创建的数据库账号。
<code>yourPassword</code>	数据库账号的密码。

参数	说明
<code>yourUrl</code>	数据库连接地址。
<code>yourTableName</code>	云数据库ClickHouse中创建的表名。
<code>yourFilePath</code>	导入数据文件的存储路径，包含文件地址和文件名。
<code>colName1, colName2, colName3</code>	云数据库ClickHouse表的列名。

4. 编译打包。

```
sbt package
```

5. 运行。

```
${SPARK_HOME}/bin/spark-submit --class "com.spark.test.WriteToCk" --master local[4] -  
-conf "spark.driver.extraClassPath=${HOME}/.m2/repository/ru/yandex/clickhouse/clickhou  
se-jdbc/0.2.4/clickhouse-jdbc-0.2.4.jar" --conf "spark.executor.extraClassPath=${HOME}/  
.m2/repository/ru/yandex/clickhouse/clickhouse-jdbc/0.2.4/clickhouse-jdbc-0.2.4.jar" ta  
rget/scala-2.12/simple-project_2.12-1.0.jar
```

6. 用户指南

6.1. 新建集群

6.1.1. 新建社区兼容版集群

本文介绍如何新建云数据库ClickHouse社区兼容版集群。

操作步骤

1. 使用阿里云账号登录 [云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在集群列表页面，选择默认实例列表。
4. 单击新建集群。
5. 选择商品类型，本文以创建包年包月集群为例。

商品类型	适用场景
包年包月	预付费，即在购买集群时支付费用。适合长期需求，价格比按量付费更实惠，且购买时长越长，折扣越多。
按量付费	后付费，即每小时生成一个收费订单，并按照生成订单时的集群规格从阿里云账号扣费。适合短期需求，例如测试阶段，用完可立即删除集群，节省费用。待资源用量稳定后，可将按量付费变更为包年包月套餐。

6. 根据页面提示，配置如下参数。

参数	说明	示例
地域和可用区	集群所在的地理位置，购买后无法更换地域。建议根据业务所在的地理位置就近选择地域，从而提升访问速度。	中国，华东1（杭州）
可用区	同一地域内，电力和网络互相独立的物理区域，同一地域不同可用区之间没有实质性区别。	杭州可用区H
网络类型	目前仅支持专有网络，也称为VPC（Virtual Private Cloud）。VPC是一种隔离的网络环境，安全性较高。	专有网络
专有网络（VPC）	专有网络VPC。如果没有可用的VPC，请前往 专有网络管理控制台 创建。	vpc-bp175iuvg8nxqraf2****
专有网络交换机	选择专有网络下的交换机。如果没有可用的交换机，请前往 专有网络管理控制台 创建。	vsw-bp1gzt31twhlo0sa5****
集群类型	云数据库ClickHouse的集群类型分为社区兼容版和云原生版。	社区兼容版
内核版本	对应的开源社区版本号。	20.8

参数	说明	示例
副本配置	<ul style="list-style-type: none"> 单副本版：一个节点只有一个副本，该副本服务不可用时，会导致整个集群不可用，需要等待副本完全恢复服务状态，集群才能继续提供稳定服务。 双副本版：一个节点包含两个副本，某个副本服务不可用的时候，同一分片的另一个副本还可以继续服务。 <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 10px; margin-top: 10px;"> <p>? 说明</p> <ul style="list-style-type: none"> 双副本版的资源以及对应的购买成本都是单副本版的2倍。 由于底层云盘提供高可靠保证，即便是单副本版也能确保数据不会丢失。 </div>	双副本版
规格	单个节点的CPU和内存。	4核16 GB
节点数量	节点的数量。	1
存储类型	<p>存储的云盘类型，应用场景如下。</p> <ul style="list-style-type: none"> ESSD云盘 <ul style="list-style-type: none"> 大型OLTP数据库：如MySQL、PostgreSQL、Oracle、SQL Server等关系型数据库。 NoSQL数据库：如MongoDB、HBase、Cassandra等非关系型数据库。 ElasticSearch分布式日志：ELK（Elasticsearch、Logstash和Kibana）日志分析等。 高效云盘 <ul style="list-style-type: none"> 开发与测试业务。 系统盘。 	ESSD PL1云盘（推荐）
加密类型	是否开启云盘加密功能。开启后可以基于块存储对整个数据盘进行加密，即使备份数据泄露也无法解密。更多信息，请参见 云盘加密 。	云盘加密
服务关联角色	加密类型选择为云盘加密时，需关联服务角色，允许访问其它云产品等服务。	已创建
加密密钥	加密类型选择为云盘加密时，需配置加密密钥。如果您在当前区域还未创建KMS加密密钥，请前往 密钥管理服务控制台 创建。	0d2470df-da7b-4786-b981-9a164dae****
存储空间	单个节点的存储空间。总存储空间=节点数量*单个节点的存储空间。	100 GB
购买时长	商品类型选择为包年包月时，需配置购买时长。	1个月

参数	说明	示例
到期自动续费	商品类型选择为包年包月时，可根据需要勾选到期是否自动续费。如果创建集群时，您没有勾选到期自动续费，可以在续费管理控制台进行自动或手动续费。更多信息，请参见 续费 。	勾选到期自动续费

7. 单击**立即购买**。

8. 在**确认订单**页面，查看配置详情，并勾选**服务协议**，单击**立即开通**。

开通成功后，您可以在**集群列表**页面，选择集群所属地域，查看新建的云数据库ClickHouse社区兼容版集群。

6.1.2. 新建云原生版集群

本文介绍如何新建云数据库ClickHouse云原生版集群。

操作步骤

 **说明** 云数据库ClickHouse云原生版目前处于公测阶段，仅支持在华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域创建云原生版集群。您可以点击[申请试用](#)。

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在**集群列表**页面，选择**云原生版实例列表**。
4. 单击**新建集群**。
5. 选择**商品类型**，本文以创建**包年包月**集群为例。

商品类型	适用场景
包年包月	预付费，即在购买集群时支付费用。适合长期需求，价格比按量付费更实惠，且购买时长越长，折扣越多。
按量付费	后付费，即每小时生成一个收费订单，并按照生成订单时的集群规格从阿里云账号扣费。适合短期需求，例如测试阶段，用完可立即删除集群，节省费用。待资源用量稳定后，可将按量付费变更为包年包月套餐。

6. 根据页面提示，配置如下参数。

参数	说明	示例
地域和可用区	集群所在的地理位置，购买后无法更换地域。建议根据业务所在的地理位置就近选择地域，从而提升访问速度。	中国，华东1（杭州）
可用区	同一地域内，电力和网络互相独立的物理区域，同一地域不同可用区之间没有实质性区别。	杭州可用区H
网络类型	目前仅支持专有网络，也称为VPC（Virtual Private Cloud）。VPC是一种隔离的网络环境，安全性较高。	专有网络

参数	说明	示例
专有网络 (VPC)	专有网络VPC。如果没有可用的VPC，请前往 专有网络管理控制台 创建。	vpc-bp175iuvg8nxqraf2****
专有网络交换机	选择专有网络下的交换机。如果没有可用的交换机，请前往 专有网络管理控制台 创建。	vsw-bp1gzt31twhlo0sa5****
集群类型	云数据库ClickHouse支持的集群类型。	云原生版
内核版本	对应的开源社区版本号。	21.8
副本配置	云原生版仅支持 单副本版 。	单副本版
实例CPU	单个计算组的CPU。	8核
实例内存大小	单个计算组的内存大小。	32 GB
实例缓存类型	存储的云盘类型，ESSD云盘的应用场景如下。 <ul style="list-style-type: none">大型OLTP数据库：如MySQL、PostgreSQL、Oracle、SQL Server等关系型数据库。NoSQL数据库：如MongoDB、HBase、Cassandra等非关系型数据库。ElasticSearch分布式日志：ELK (Elasticsearch、Logstash和Kibana) 日志分析等。	ESSD PL1 (云盘)
实例缓存大小	单个计算组的存储空间。	4800 GB
购买时长	商品类型选择为 包年包月 时，需配置购买时长。	1个月
到期自动续费	商品类型选择为 包年包月 时，可根据需要勾选到期是否自动续费。如果创建集群时，您没有勾选 到期自动续费 ，可以在续费管理控制台进行自动或手动续费。更多信息，请参见 续费 。	勾选到期自动续费

- 单击**立即购买**。
- 在**确认订单**页面，查看配置详情，并勾选**服务协议**，单击**去支付**。
- 在**支付**页面，确认订单信息，单击**订购**。

支付成功后，您可以在**集群列表**页面单击**云原生版实例列表**，选择集群所属地域，查看新建的云数据库ClickHouse云原生版集群。

6.2. 变配和扩缩容

6.2.1. 社区兼容版变配和扩缩容

如果云数据库ClickHouse社区兼容版集群当前的配置无法满足需求，您可以对集群配置进行变更，ClickHouse社区兼容版集群配置变更包括垂直升配、垂直降配、水平扩容和水平缩容。

前提条件

- 目标集群没有正在进行的配置变更。
- 您的阿里云账号中没有未支付的续费订单。

背景信息

ClickHouse社区兼容版集群配置变更包括：

- 垂直升配/垂直降配：节点规格升、降配，或扩展存储空间，纵向扩展集群计算能力或扩展存储空间。请参见[垂直升配/垂直降配](#)。

② 说明

- 升降配流程总耗时约10-15分钟，期间会对实例做重启操作1次，会出现一次秒级闪断，建议在业务低峰期进行。
- 当集群性能指标不满足业务需求时，请优先选择升配。

- 水平扩容/水平缩容：增加或减少节点数量，横向扩展集群的计算能力。请参见[水平扩容/水平缩容](#)。

② 说明 水平扩容/水平缩容都是基于原有数据进行复制同步的，只更改了节点数量。

变更集群配置后，费用将会发生变化，更多信息，请参见[变更配置计费说明](#)。

垂直升配/垂直降配

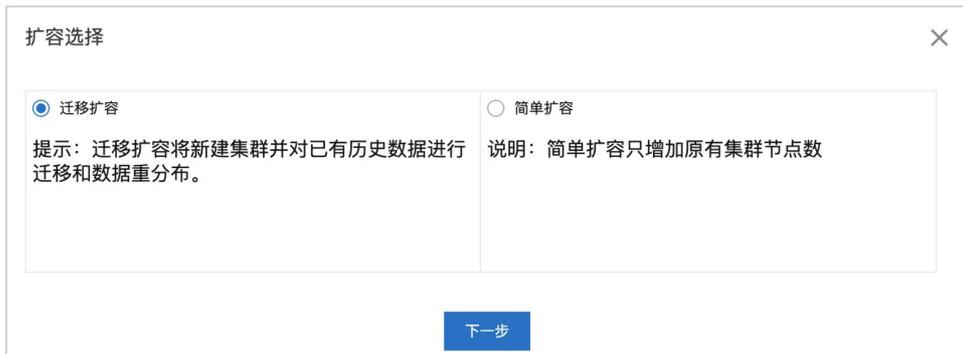
1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在[集群列表](#)页面，选择默认实例列表，单击目标集群ID。
4. 单击目标集群右侧操作列中的更多>垂直升配或垂直降配。
5. 根据您的需求选择对应的节点规格。

🔔 注意 变更规格和存储空间配置需分开进行，无法一次性同时修改两项。

6. 单击立即购买，根据提示完成支付。配置变更后约5-10分钟开始生效。

水平扩容/水平缩容

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在[集群列表](#)页面，选择默认实例列表，单击目标集群ID。
4. 单击目标集群右侧操作列中的水平扩容或更多 > 水平缩容。
5. 水平扩容需要选择扩容方式，选择完成后单击下一步。水平缩容会跳过该步骤，直接进行下一步状态检测。



○ 迁移扩容

创建新的实例集群，并在新集群上增加节点，将原有集群的数据复制到新集群上。在数据写入新集群的过程中，基于原有集群的分片键和分片方式，在新集群进行数据分片写入，保证扩容后集群各个分片之间的负载均衡。

说明 迁移扩容创建的新实例集群会继承原集群ID。

○ 简单扩容

直接在原有的集群上增加节点，并在新增节点上创建分布式表和本地表，扩容后新写入的数据按照原有的分布策略进行写入。

6. 在弹出的扩容/缩容检测窗口，查看检测状态。简单扩容会跳过检测，直接进入扩容/缩容页面。

○ 检测成功：单击下一步。

注意

- 水平扩容/缩容期间，集群将处于可读不可写状态。
- MergeTree类型引擎表扩容，会将原有历史数据迁移写入新的集群中并自动完成重分布。
- 非MergeTree类型（如外表，Log表等）仅进行表结构移，不进行数据迁移。物化视图仅进行结构迁移，不进行数据迁移。
- 扩容后内部节点IP会发生变化，如果依赖节点IP数据写入和访问，需重新获取集群的VPC网段IP。

○ 检测失败：根据页面提示进行相应修改，修改完成后单击重新检测，检测成功后单击下一步。

检测失败异常内容有以下两种：

- 缺失唯一的分布式表：表示本地表没有创建分布式表，需要再创建一个对应的分布式表。
- 对应的分布式表不唯一：表示本地表有不止一个分布式表，请删除多余的分布式表，仅保留一个即可。

7. 在扩容/缩容页面，根据您的需求修改节点数量。

8. 单击立即购买，根据提示完成支付，扩容耗时跟数据量有关。

6.2.2. 云原生版变配

如果云原生版集群及其计算组当前的规格和缓存大小无法满足需求，您可以通过计算组变配实现云原生版集群的变配。

前提条件

- 目标计算组和集群的状态为运行中。
- 您的阿里云账号中没有未支付的续费订单。

注意事项

- 云数据库ClickHouse云原生版目前处于公测阶段，仅支持在华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域创建云原生版集群。您可以点击[申请试用](#)。
- 变配过程中可能会产生缓存数据迁移，造成短暂的业务停写。如果变配评估显示的时间过长，建议在业务低峰期进行变配。
- 变更计算组配置后，配置费用会发生变化。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在集群列表页面，选择云原生版实例列表。
4. 单击目标集群ID。
5. 在集群信息页面，单击目标计算组操作列的变配。
6. 在弹出的变配窗口，修改目标配置规格和缓存大小。
7. 单击点击进入变配评估。

 **说明** 如果评估失败，请根据页面提示进行相应修改。

8. 在弹出的计算组变配窗口，查看配置费用，单击下一步：确定启动变配。



启动变配后，集群状态变为变配中，计算组状态变为资源变配中并重启。待状态全部变为运行中，即完成变配。

6.3. 社区兼容版重启实例

云数据库ClickHouse社区兼容版支持重启实例，遇到实例不可用或者需要重启解决的问题时，您可以自主完成重启。本文介绍如何在控制台上重启实例。

注意事项

- 只有状态为运行中的实例才能进行重启。
- 重启实例会造成30秒左右的连接中断，建议在业务低峰期重启。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。

2. 在页面左上角，选择目标集群所在的地域。
3. 在集群列表页面，选择默认实例列表，单击目标集群ID。
4. 单击右上方导航栏的**重启实例**。
5. 在弹出的**重启实例**对话框中单击**确定**。
6. 在弹出的**手机验证**对话框中，单击**获取验证码**给阿里云账号绑定的手机号发送信息。输入获取的验证码，单击**确定**。
重启过程中，实例状态变为**重启中**。待状态变为**运行中**，即完成重启。

相关API

API	描述
RestartInstance	调用RestartInstance接口重启云数据库ClickHouse社区兼容版集群。

6.4. 云原生版重启计算组

云数据库ClickHouse云原生版不支持重启实例，但是可以对计算组进行重启。遇到计算组不可用或者需要重启解决的问题时，您可以自主完成重启。本文介绍如何在控制台上重启计算组。

注意事项

- 云数据库ClickHouse云原生版目前处于公测阶段，仅支持在华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域创建云原生版集群。您可以点击[申请试用](#)。
- 只有状态为运行中的计算组才能进行重启。
- 重启计算组会造成30秒左右的连接中断，当前计算组短暂不可用，建议在业务低峰期重启。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在集群列表页面，选择云原生版实例列表，单击目标集群ID。
4. 单击集群计算组操作列的**重启**。
5. 在弹出的警告对话框中单击**确定**。
6. 在弹出的**手机验证**对话框中，单击**获取验证码**给阿里云账号绑定的手机号发送信息。输入获取的验证码，单击**确定**。
重启过程中，计算组状态变为**重启中**。待状态变为**运行中**，即完成重启。

6.5. 开通MySQL协议

本文介绍如何开通MySQL协议。开通后，您可以通过MySQL协议连接云数据库ClickHouse集群。

注意事项

- 云数据库ClickHouse的社区兼容版集群仅20.8及以上版本支持MySQL协议。
- 2021年12月份之后创建的20.8及以上版本的社区兼容版集群，会自动开通MySQL协议。在此之前创建的20.8及以上版本的社区兼容版集群需手动开通。
- 云数据库ClickHouse的云原生版集群计算组支持MySQL协议，且自动开通。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在[集群列表](#)页面，选择默认实例列表，单击目标集群ID。
4. 在[集群信息](#)页面，单击启用MySQL协议。
5. 在弹出的确认开启MySQL协议窗口中，单击确定。
启用MySQL协议后，集群状态会变更为创建网络中，该状态将持续一分钟左右，请耐心等待集群状态变更为运行中，即成功开通MySQL协议。

相关文档

[通过MySQL协议连接ClickHouse](#)

6.6. 开通HTTPS协议

您可以通过HTTPS协议启用SSL（Secure Sockets Layer）加密，实现数据的加密传输。本文介绍如何开通HTTPS协议。开通后，您可以通过HTTPS协议连接云数据库ClickHouse集群。

背景信息

SSL是Netscape公司所提出的安全保密协议，在客户端和Web服务器之间构造安全通道来进行数据传输，采用RC4、MD5、RSA等加密算法实现安全通讯。国际互联网工程任务组（IETF）对SSL 3.0进行了标准化，标准化后更名为安全传输层协议（TLS）。由于SSL这一术语更为常用，因此启用SSL加密实际是指启用TLS加密。

云数据库ClickHouse支持SSL 2.0、SSL3.0、TLS1.0、TLS1.1和TLS1.2。

使用限制

- 云数据库ClickHouse的社区兼容版集群仅20.8及以上版本支持HTTPS协议。
- 云数据库ClickHouse的云原生版集群不支持HTTPS协议。

注意事项

- 开通HTTPS协议会重启云数据库ClickHouse集群，请谨慎操作。
- 通过HTTPS协议连接云数据库ClickHouse集群会增加网络连接响应时间。
- 通过HTTPS协议连接云数据库ClickHouse集群会显著增加CPU使用率，建议您在使用外网且有加密需求时，使用HTTPS协议访问云数据库ClickHouse。使用VPC网络相对安全，一般无需使用HTTPS协议访问云数据库ClickHouse。
- 使用外网地址或VPC地址访问同一个云数据库ClickHouse集群时，SSL CA证书是相同的，下载后免费使用，有效期至2031年12月25日。
- 外网地址和VPC地址均支持HTTPS协议，HTTPS端口号固定为8443。
- 开通HTTPS协议后，不影响使用其他协议连接云数据库ClickHouse集群。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在[集群列表](#)页面，选择默认实例列表，单击目标集群ID。
3. 在[集群信息](#)页面，单击点击开通HTTPS协议支持。
4. 在弹出的提示窗口中，单击确定。

开通HTTPS协议后，集群状态会变更为重启中，该状态将持续一分钟左右，请耐心等待集群状态变更为运行中，即成功开通HTTPS协议。

5. 单击下载CA证书，下载SSL CA证书。

下载的文件为ClickHouse-CA-Chain.pem，用于其他系统或应用中导入CA证书。

相关文档

[通过HTTPS协议连接ClickHouse](#)

6.7. 申请外网地址

云数据库ClickHouse支持VPC地址和外网地址两种地址类型，默认提供VPC地址供您内部访问云数据库ClickHouse，如果需要外网访问，请申请外网地址。本文介绍如何通过云数据库ClickHouse控制台申请外网地址。

背景信息

如果VPC网络之外的客户端要访问云数据库ClickHouse，您需要给集群申请一个外网地址，并通过外网地址连接云数据库ClickHouse集群。

操作步骤

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在集群列表页面，选择默认实例列表或云原生版实例列表，单击目标集群ID。
4. 如果您的集群是社区兼容版，请在集群信息页面，单击[申请外网](#)。
如果您的集群是云原生版，请在集群信息页面，单击集群计算组操作列的[查看连接](#)，再单击[申请外网](#)。
5. 在[申请外网](#)提示框中单击[确定](#)，成功生成外网地址。

 **说明** 外网地址生成后，如果需要使用外网地址访问云数据库ClickHouse，您还需要将客户端的IP地址或IP地址段添加到云数据库ClickHouse的白名单中。如何添加，具体请参见[设置白名单](#)。

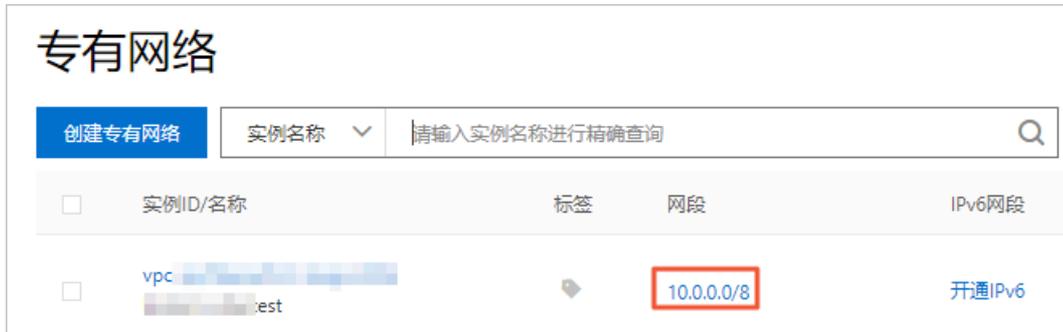
6.8. 获取集群的VPC网段IP

将数据从其他数据库导入云数据库ClickHouse时，需要将目标云数据库ClickHouse集群所在的VPC网段IP地址添加到源数据库的白名单中。本文介绍了如何获取云数据库ClickHouse集群所在的VPC网段IP地址。

操作步骤

1. 复制目标云数据库ClickHouse集群的VPC ID。
 - i. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
 - ii. 在页面左上角，选择集群所在地域。
 - iii. 在集群列表页面，选择默认实例列表或云原生版实例列表，单击目标集群ID。
 - iv. 在集群信息页面，查看并复制VPC ID。
2. 通过VPC控制台，获取云数据库ClickHouse集群所在VPC网段IP地址。
 - i. 使用阿里云账号登录[VPC控制台](#)。
 - ii. 在页面左上角，选择VPC所在地域。

- iii. 选择实例ID，在搜索框中粘贴上述步骤中复制的目标云数据库ClickHouse集群VPC ID，单击搜索。
- iv. 在列表中找到对应的网段IP地址。



后续步骤

将获取到的云数据库ClickHouse集群所在VPC网段IP地址添加到源数据库的白名单中。

6.9. 连接数据库

6.9.1. DMS连接ClickHouse

本文介绍如何通过数据管理服务DMS（Data Management Service）连接云数据库ClickHouse集群并对数据进行在线管理。

背景信息

DMS是阿里云提供的图形化数据管理工具，可用于管理关系型数据库和NoSQL数据库，支持数据管理、SQL操作、数据方案（数据导入/导出、数据库克隆等）、性能与优化、安全审计等功能。您可以使用DMS实现易用的数据库管理，在线进行云数据库ClickHouse库、表创建及在线数据查询，让数据更安全、管理更高效、数据价值更清晰。

注意事项

新建云数据库ClickHouse集群时，系统会自动将DMS服务器的IP地址添加至云数据库ClickHouse的白名单中，若自动添加失败需手动添加。DMS服务器不同地域的IP地址列表，具体请参见[DMS IP地址列表](#)。

操作步骤

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在集群列表页面，选择默认实例列表或云原生版实例列表，单击目标集群ID。
4. 如果您的集群是社区兼容版，请单击右上方导航栏的登录数据库。
如果您的集群是云原生版，请单击集群计算组操作列的登录。
5. 根据页面提示，配置如下参数。

参数	说明
数据库类型	默认为ClickHouse，云数据库ClickHouse集群的类型，无需修改。

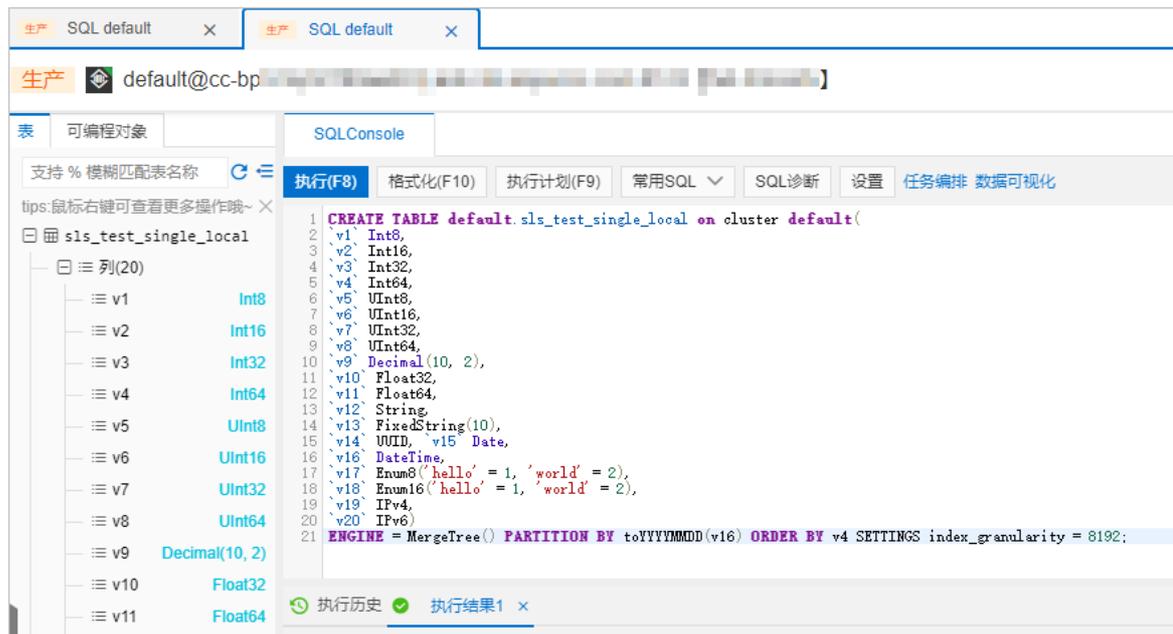
参数	说明
实例地区	默认为当前云数据库ClickHouse集群所在的地域，无需修改。  说明 若您需要登录其他地域下的云数据库ClickHouse集群，从下拉列表中选择目标集群所在的地域即可。
实例ID	默认为当前云数据库ClickHouse集群的集群ID，无需修改。  说明 若您需要登录其他云数据库ClickHouse集群，从下拉列表中选择目标集群ID即可。
数据库账号	数据库的登录账号。
数据库密码	数据库账号对应的密码。
记住密码	勾选 记住密码 ，下次登录此云数据库ClickHouse集群时，无需重复输入账号密码即可自动登录。

 **说明**

- 首次通过DMS登录云数据库ClickHouse集群时，管控模式默认为**自由操作**。登录成功后，您可以通过编辑实例功能来修改管控模式。更多信息，请参见[编辑实例](#)和[管控模式](#)。
- 配置完登录参数后，您可以单击左下角**测试连接**。如果测试连接失败，请按照报错提示检查录入的集群信息。

6. 单击**登录**。
7. 在**SQL Console**界面输入要执行的SQL语句。
8. 单击菜单栏**执行 (F8)** 按钮，并查看结果。

本文以建立本地表为例。



DMS系统支持执行各类SQL语句，同时支持以可视化的方式对目标数据库进行增删改查等操作。更多信息，请参见[SQL窗口介绍](#)。

6.9.2. ClickHouse命令行工具连接

本文介绍如何使用ClickHouse命令行工具连接云数据库ClickHouse集群。

操作步骤

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在集群列表页面，选择默认实例列表或云原生版实例列表，单击目标集群ID。
4. 如果您的集群是社区兼容版，请在集群信息页面，查看外网地址或VPC地址。
如果您的集群是云原生版，请在集群信息页面，单击集群计算组操作列的查看连接，查看外网地址或VPC地址。
5. 根据操作系统类型及云数据库ClickHouse版本下载对应的客户端。下载链接，请参见[clickhouse-client](#)。
6. 连接云数据库ClickHouse集群。

在clickhouse-client客户端的安装目录下执行如下命令。

```
./clickhouse-client --host=<host> --port=<port> --user=<user> --password=<password>;
```

参数说明如下。

参数	说明
host	外网地址或VPC地址。 如果clickhouse-client所在服务器与云数据库ClickHouse集群在同一VPC内，您可以使用VPC地址。否则，请使用外网地址。

参数	说明
port	TCP端口号。 如果您的集群是社区兼容版，请直接在 集群信息 页面查看。 如果您的集群是云原生版，请在 集群信息 页面，单击集群计算组操作列的 查看连接 查看。
user	您通过云数据库ClickHouse控制台创建的数据库账号。
password	数据库账号对应的密码。

6.9.3. 通过JDBC方式连接ClickHouse

本文通过代码示例说明如何通过JDBC方式连接ClickHouse进行应用开发。

操作步骤

1. 使用Eclipse或其他IDE工具创建Maven项目并引入ClickHouse驱动依赖包。

```
<dependency>
  <groupId>ru.yandex.clickhouse</groupId>
  <artifactId>clickhouse-jdbc</artifactId>
  <version>0.2.4</version>
</dependency>
```

2. 编写应用程序代码。

以下代码演示了如何使用JDBC驱动连接实例，使用过程需修改实例连接串和端口参数。如果应用和ClickHouse实例在同一个VPC环境，请使用实例VPC地址，如果不在相同的VPC环境则需使用实例外网地址，如何获取，请参见[申请外网地址](#)。

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.sql.Timestamp;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;
public class Main4 {
    private static final String DATE_FORMAT = "yyyy-MM-dd HH:mm:ss";
    private static final SimpleDateFormat SIMPLE_DATE_FORMAT = new SimpleDateFormat(DATE_
FORMAT);
    public static void main(String[] args) throws ClassNotFoundException, SQLException, I
nterruptedOperationException, ParseException {
        String url = "your url";
        String username = "your username";
        String password = "your password";
        Class.forName("ru.yandex.clickhouse.ClickHouseDriver");
        String connectionStr = "jdbc:clickhouse://" + url + ":8123";
        try (Connection connection = DriverManager.getConnection(connectionStr, username, p
assword);
```

```
Statement stmt = connection.createStatement() {
{
String createTableDDL = "create table test_table on cluster default " +
    "(id UInt32, " +
    "dt_str String, " +
    "dt_col DateTime) " +
    "engine=ReplicatedMergeTree('/clickhouse/tables/{database}/{table}/{shard}'
, '{replica}') " +
    "partition by toYYYYMM(dt_col)" +
    "order by (id)" +
    "primary key (id)" +
    "sample by (id)" +
    "settings index_granularity = 8192;";
stmt.execute(createTableDDL);
System.out.println("create local table done.");
}
{
String createTableDDL = "create table test_dist on cluster default " +
    "as default.test_table " +
    "engine=Distributed(default, default, test_table, rand());";
stmt.execute(createTableDDL);
System.out.println("create distributed table done.");
}
System.out.println("write 100000 rows...");
long startTime = System.currentTimeMillis();
// Write 10 batch
for (int batch = 0; batch < 10; batch++) {
    StringBuilder sb = new StringBuilder();
    // Build one batch
    sb.append("insert into test_dist values(" + (batch * 10000) + ", '2020-02-19 16
:00:00', '2020-02-19 16:00:00')");
    for (int row = 1; row < 10000; row++) {
        sb.append(", (" + (batch * 10000 + row) + ", '2020-02-19 16:00:00', '2020-02-
19 16:00:00')");
    }
    // Write one batch: 10000 rows
    stmt.execute(sb.toString());
}
long endTime = System.currentTimeMillis();
System.out.println("total time cost to write 10W rows: " + (endTime - startTime)
+ "ms");
Thread.sleep(2 * 1000);
System.out.println("Select count(id)...");
try (ResultSet rs = stmt.executeQuery("select count(id) from test_dist");) {
    while (rs.next()) {
        int count = rs.getInt(1);
        System.out.println("id count: " + count);
    }
}
try (ResultSet rs = stmt.executeQuery("select id, dt_str, dt_col from test_dist l
imit 10");) {
    while (rs.next()) {
        int id = rs.getInt(1);
        String dateStr = rs.getString(2);
```

```
Timestamp time = rs.getTimestamp(3);
String defaultDate = SIMPLE_DATE_FORMAT.format(new Date(time.getTime()));
System.out.println("id: " + id
    + ", date_str:" + dateStr
    + ", date_col:" + defaultDate);
}
}
}
}
}
```

6.9.4. 通过MySQL协议连接ClickHouse

云数据库ClickHouse支持通过MySQL客户端访问。本文介绍如何通过MySQL协议连接云数据库ClickHouse集群。

使用限制

仅20.8及以上版本的云数据库ClickHouse集群支持通过MySQL协议连接。

注意事项

使用MySQL协议连接云数据库ClickHouse集群时，会形成性能损失，并产生很多兼容性问题，建议您谨慎选择。

前提条件

- 已开通MySQL协议。如何开通，请参见[开通MySQL协议](#)。
- 已将连接源的IP地址添加到云数据库ClickHouse集群的白名单中。如何添加，请参见[设置白名单](#)。
- 如果连接源和云数据库ClickHouse集群不属于同一个VPC，请先为云数据库ClickHouse集群申请外网地址。

操作步骤

1. 通过外网地址或VPC地址连接云数据库ClickHouse集群，语法如下。

```
mysql -h <外网地址或VPC地址> -P <MySQL端口号> -u <数据库账号> -p<数据库账号的密码> --default-auth=mysql_native_password
```

示例如下。

```
mysql -h cc-bp1631724nkf****.clickhouse.ads.aliyuncs.com -P 9004 -u test -p123456Aa --default-auth=mysql_native_password
```

 **说明** MySQL端口号固定为9004。

2. 成功连接云数据库ClickHouse集群后，对数据库进行查询，查询语句如下。

```
show databases;
```

查询结果如下。

```
+-----+
| name          |
+-----+
| _temporary_and_external_tables |
| default      |
| system      |
+-----+
```

6.9.5. 通过HTTPS协议连接ClickHouse

您可以通过HTTPS协议访问云数据库ClickHouse。本文介绍如何通过HTTPS协议连接云数据库ClickHouse集群。

使用限制

- 云数据库ClickHouse的社区兼容版集群仅20.8及以上版本支持通过HTTPS协议连接。
- 云数据库ClickHouse的云原生版集群不支持通过HTTPS协议连接。

注意事项

- 通过HTTPS协议连接云数据库ClickHouse集群会增加网络连接响应时间。
- 通过HTTPS协议连接云数据库ClickHouse集群会显著增加CPU使用率，建议您在使用外网且有加密需求时，使用HTTPS协议访问云数据库ClickHouse。使用VPC网络相对安全，一般无需使用HTTPS协议访问云数据库ClickHouse。

前提条件

- 已开通HTTPS协议并下载SSL CA证书。如何开通和下载，请参见[开通HTTPS协议](#)。
- 已将连接源的IP地址添加到云数据库ClickHouse集群的白名单中。如何添加，请参见[设置白名单](#)。
- 如果连接源和云数据库ClickHouse集群不属于同一个VPC，请先为云数据库ClickHouse集群申请外网地址。

Java JDBC方式连接

语法如下。

```
public void run()
    throws InterruptedException {
    final ClickHouseProperties clickHouseProperties = new ClickHouseProperties();
    clickHouseProperties.setSslRootCertificate("<证书路径>");
    clickHouseProperties.setSsl(true);
    clickHouseProperties.setSslMode("<SSL Mode, 取值: strict、none>");
    clickHouseProperties.setUser("<数据库账号>");
    clickHouseProperties.setPassword("<数据库账号的密码>");
    clickHouseProperties.setSocketTimeout(<超时时间, 单位: 毫秒>);
    final BalancedClickhouseDataSource dataSource = new BalancedClickhouseDataSource("jdbc:
<外网地址、VPC地址或IP地址>:<HTTPS端口号>/<数据库名称>?ssl=true", clickHouseProperties);
    try {
        final ClickHouseConnection conn = dataSource.getConnection();
        conn.createStatement().executeQuery("select now()");
    } catch (Throwable e) {
        e.printStackTrace();
    }
}
```

 **说明** 如果通过IP地址连接，连接源和云数据库ClickHouse集群需属于同一个VPC且 `SSL Mode` 取值为 `none`。

示例如下。

```
public void run()
    throws InterruptedException {
    final ClickHouseProperties clickHouseProperties = new ClickHouseProperties();
    clickHouseProperties.setSslRootCertificate("/user/ck-root-ClickHouse-CA-Chain.pem");
    clickHouseProperties.setSsl(true);
    clickHouseProperties.setSslMode("strict");
    clickHouseProperties.setUser("test");
    clickHouseProperties.setPassword("123456Aa");
    clickHouseProperties.setSocketTimeout(2 * 3600 * 1000);
    final BalancedClickhouseDataSource dataSource = new BalancedClickhouseDataSource("jdbc:
cc-bp1631724nkf***.clickhouse.ads.aliyuncs.com:8443/default?ssl=true", clickHouseProperties);
    try {
        final ClickHouseConnection conn = dataSource.getConnection();
        conn.createStatement().executeQuery("select now()");
    } catch (Throwable e) {
        e.printStackTrace();
    }
}
```

curl命令行方式连接

语法如下。

```
curl --cacert <证书路径> https://<外网地址或VPC地址>:<HTTPS端口号>/ping
```

示例如下。

```
curl --cacert ./ck-root-ClickHouse-CA-Chain.pem https://cc-bp1631724nkf****.clickhouse.ads.aliyuncs.com:8443/ping
```

6.10. 账号与安全

6.10.1. 账号管理

云数据库ClickHouse支持对账号进行管理，本文介绍如何在控制台创建数据库账号、修改权限、修改密码和删除数据库账号。

创建数据库账号

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在集群列表页面，选择默认实例列表或云原生版实例列表，单击目标集群ID。
4. 在左侧导航栏单击账号管理。
5. 单击右上角的创建账号。
6. 在创建账号页面，按照页面提示配置如下参数。

社区兼容版集群，参数说明如下。

参数	描述
数据库账号	<p>数据库账号，设置规则如下。</p> <ul style="list-style-type: none"> 名称唯一。 由小写字母、数字或下划线组成。 以小写字母开头，以小写字母或数字结尾。 长度为2~64个字符。
授权访问范围	<p>授权数据库账号的访问范围。</p> <ul style="list-style-type: none"> 所有数据库和字典。 部分数据库和字典。 <p>您可以根据需求勾选数据库或字典后，单击  按钮或  按钮进行添加或移除授权。</p>
DML权限	<p>是否授予写权限。</p> <ul style="list-style-type: none"> 读写和设置权限：对授权的数据库和字典可进行读、写和设置的操作。 读取和设置权限：对授权的数据库和字典仅可进行读和设置的操作，不可写入数据。

参数	描述
DDL权限	是否授予DDL权限。 <ul style="list-style-type: none"> 允许DDL。 禁止DDL。
密码	数据库账号的密码，设置规则如下。 <ul style="list-style-type: none"> 由大写字母、小写字母、数字和特殊字符中的至少三种组成。 特殊字符为!@#%&*()_+ -= 长度为8~32个字符。
确认密码	再次输入相同的密码。
备注说明	数据库账号的描述，可选项，设置规则如下。 <ul style="list-style-type: none"> 不能以http://或https://开头。 长度为0~256个字符。

云原生版集群，参数说明如下。

参数	描述
数据库账号	数据库账号，设置规则如下。 <ul style="list-style-type: none"> 名称唯一。 由小写字母、数字或下划线组成。 以小写字母开头，以小写字母或数字结尾。 长度为2~64个字符。
密码	数据库账号的密码，设置规则如下。 <ul style="list-style-type: none"> 由大写字母、小写字母、数字和特殊字符中的至少三种组成。 特殊字符为!@#%&*()_+ -= 长度为8~32个字符。
确认密码	再次输入相同的密码。
备注说明	数据库账号的描述，可选项，设置规则如下。 <ul style="list-style-type: none"> 不能以http://或https://开头。 以中文、英文字母开头。 长度为2~256个字符。
权限类型	数据库账号的权限类型。

7. 单击**确定**，创建数据库账号。每个集群支持创建500个数据库账号。

修改权限

 **注意** 仅账号状态为可用情况下可进行修改权限操作。

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在**集群列表**页面，选择**默认实例列表**或**云原生版实例列表**，单击目标集群ID。
4. 在左侧导航栏单击**账号管理**。
5. 单击目标数据库账号操作列的**修改权限**。
6. 如果您的集群是社区兼容版，请根据需求修改数据库账号的授权访问范围、DML权限和DDL权限。
如果您的集群是云原生版，请根据需求修改权限类型。
7. 单击**确定**。

修改密码

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在**集群列表**页面，选择**默认实例列表**或**云原生版实例列表**，单击目标集群ID。
4. 在左侧导航栏单击**账号管理**。
5. 单击目标数据库账号操作列的**修改密码**。
6. 在**修改密码**页面中，输入新密码并确认新密码。
7. 单击**确定**。

删除数据库账号

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择集群所在地域。
3. 在**集群列表**页面，选择**默认实例列表**或**云原生版实例列表**，单击目标集群ID。
4. 在左侧导航栏单击**账号管理**。
5. 单击目标数据库账号操作列的**删除**。
6. 在弹出的**删除账号**窗口，单击**确定**。

6.10.2. 云盘加密

云数据库ClickHouse免费提供云盘加密功能，基于块存储对整个数据盘进行加密，即使数据备份泄露也无法解密，保护您的数据安全。

注意事项

- 云数据库ClickHouse的云原生版集群不支持云盘加密。
- 仅在创建ClickHouse社区兼容版集群时可以开启云盘加密，创建集群后无法开启。
- 云盘加密功能开启后无法关闭。
- 开启云盘加密后，实例生成的快照以及通过这些快照创建的云盘版实例将自动延续加密属性。
- 云盘加密不会影响业务，应用程序也无需修改。
- 使用云盘加密功能不会造成明显的性能损失。

功能说明

在创建了一块加密云盘并将其挂载到ECS实例后，系统将对以下数据进行加密：

- 云盘中的静态数据。
- 云盘和实例间传输的数据（不支持加密系统盘数据）。
- 从加密云盘创建的所有快照（即加密快照）。

计费

ClickHouse侧，云盘加密为免费功能，您在磁盘上的任何读写操作都不会产生额外费用。

密钥服务系统（KMS）涉及密钥管理费用和API调用费用，请参见[KMS计费说明](#)。

开启方式

新建社区兼容版集群时进行如下设置：

1. 存储类型选择ESSD云盘或高效云盘。
2. 加密类型选择云盘加密。
3. 选择加密密钥。如还未创建密钥，请根据界面提示，先开启密钥管理服务，再创建密钥。

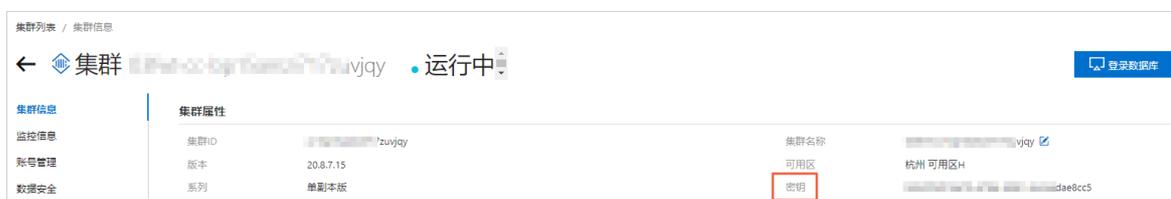
说明

- Clickhouse的云盘加密功能仅支持由用户手工创建的服务密钥，用户在创建普通密钥时需要将轮转周期设置为不开启。创建密钥详细操作请参见[创建密钥](#)。
- 用户授权开通KMS，会留下相关审计记录。请参见[使用操作审计查询密钥管理服务的操作事件](#)。

4. 单击立即购买，即可创建加密云盘。

查看密钥

1. 登录[ClickHouse控制台](#)。
2. 在集群列表页面，选择默认实例列表，单击目标集群ID。
3. 在集群信息页面的集群属性中查看密钥。



6.11. 备份与恢复

6.11.1. 数据备份

云数据库ClickHouse提供数据备份能力，您可以手动打开备份服务开关对数据进行备份设置。

前提条件

- 实例版本为20.3、20.8和21.8。

- 实例创建时间为2021年12月01日之前。

注意事项

- 一个数据库实例只允许设置一个备份计划。
- 备份内容包含当前实例所有库、表、视图、数据字典和元数据结构。
- 备份过程中会进行数据文件读取复制，影响实例的读写效率，请尽量在业务低峰期或运维时间段启动备份任务。
- 已启动的备份任务不能中断。
- 数据备份功能开通后，暂不支持关闭。
- 已开通冷热分层存储功能的实例不支持再开通数据备份功能。

计费说明

- 计费规则：云数据库ClickHouse提供了免费备份存储空间，免费备份存储空间=100%×实例存储空间。超出免费备份存储空间后，超出部分依据备份存储单价按量按小时进行收费。
- 计费公式：每小时备份费用=（备份使用量-免费备份存储空间）×备份存储单价。
- 示例：
 - 备份使用量不超过免费备份存储空间：
若免费备份存储空间为100 GB，备份使用量为50 GB，则本次备份数据免费。
 - 备份使用量大于免费备份存储空间：
若备份存储单价为0.0002元/GB/小时，免费备份存储空间为50 GB，备份使用量为100 GB，则本次备份数据每小时费用为（100-50）GB×0.0002元/GB/小时=0.01元/小时。

 **说明** 本示例中的价格仅用于举例，实际价格以控制台为准。关于备份空间价格，具体请参见[计费概述](#)。

开启数据备份

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择实例所在的地域。
3. 在集群列表页面，选择默认实例列表，单击目标实例ID。
4. 在左侧导航栏单击备份恢复。
5. 单击备份设置，打开备份服务开关。

 **注意** 首次开启备份，将立即执行一次备份操作，对实例写入查询有一定的影响。

6. 在弹出的备份设置页面，设置如下参数。

参数	说明
备份周期	备份任务以周为单位进行周期性执行，每天最多一次备份。您可以设置一周中任意若干天作为备份日期。
备份时间	默认备份时间为备份日期当天凌晨02:00~03:00，您可以设置为任意时段。

参数	说明
备份保留时间	默认备份保留时间为7天，您可以设置为7~730天。

7. 单击**确定**，完成备份设置。

备份任务成功执行后，会生成一份备份数据。您可以基于备份数据，将数据恢复至一个新实例。更多信息，请参见[数据恢复](#)。

6.11.2. 数据恢复

如果云数据库ClickHouse实例已经进行了数据备份，您可以通过数据恢复的方式将数据恢复至云数据库ClickHouse的一个新实例。

前提条件

满足以下条件的实例支持数据恢复：

- 实例版本为20.3、20.8和21.8。
- 实例创建时间为2021年12月01日之前。

计费说明

数据恢复至新实例，会收取购买新实例的费用。

 **说明** 新实例的计费方式与源集群相同，且不可变更。

功能说明

云数据库ClickHouse数据恢复功能为实例级别的数据恢复，不支持库、表级别的数据恢复。数据恢复时，会将源集群备份集的数据全量恢复至新实例（也称为克隆实例）。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择实例所在的地域。
3. 在[集群列表](#)页面，选择**默认实例列表**，单击目标实例ID。
4. 在左侧导航栏单击**备份恢复**。
5. 在[数据备份](#)页面，单击待恢复数据操作列的**数据恢复**。
6. 在[克隆实例](#)页面，查看源集群的配置信息，并对克隆实例进行如下配置。

参数	说明
克隆源集群	源集群的实例ID。
克隆源类型	固定为 备份集 。
克隆源备份集	备份集的备份开始时间。
地域和可用区	克隆实例所在的地域。默认与源集群相同，且不可变更。

参数	说明
可用区	克隆实例所在的可用区。
网络类型	目前只支持专有网络，也称为VPC（Virtual Private Cloud）。VPC是一种隔离的网络环境，安全性较高。
专有网络（VPC）	克隆实例所在的专有网络。
专有网络交换机	克隆实例的专有网络交换机。
内核版本	克隆实例的内核版本。默认与源集群相同，且不可变更。
副本配置	<p>克隆实例的副本数量。默认与源集群相同，且不可变更。</p> <ul style="list-style-type: none"> 双副本版：每个节点包含两个副本，某个副本服务不可用的时候，同一分片的另一个副本还可以继续服务。 单副本版：每个节点只有1个副本，该副本服务不可用时，会导致整个集群不可用，需要等待此副本完全恢复服务状态，集群才能继续提供稳定服务。
规格	克隆实例的节点规格。
节点数量	克隆实例的节点数量。默认与源集群相同，且不可变更。
存储类型	<p>克隆实例的存储类型。</p> <ul style="list-style-type: none"> ESSD云盘：基于新一代分布式块存储架构的超高性能云盘产品，存取效率较高。性能表现上，ESSD PL3云盘>ESSD PL2云盘>ESSD PL1云盘。 高效云盘：具备高性价比、中等随机读写性能、高可靠性的云盘产品。
加密类型	<p>克隆实例的加密类型。</p> <ul style="list-style-type: none"> 不加密：不进行云盘加密。 云盘加密：基于块存储对整个数据盘进行加密。如果选择云盘加密，还需要选择加密密钥。
存储空间	克隆实例的存储空间。默认与源集群相同，且不可变更。

7. 勾选服务协议。

 说明 克隆实例付费类型与源集群相同，且不可变更。

8. 单击立即购买。

购买成功后，您可以在[集群列表](#)页面查看克隆实例。

 说明 创建克隆实例一般需要10~15分钟，请耐心等待。

6.12. 监控与报警

6.12.1. 集群监控

云数据库ClickHouse提供了丰富的性能监控数据，方便您查看和掌握集群的运行状态。本文介绍如何在控制台查看集群监控信息。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在集群列表页面，选择默认实例列表或云原生版本实例列表，单击目标集群ID。
4. 单击左侧导航栏中的[监控告警](#)。
5. 在[监控告警](#)页面，选择[集群监控](#)，默认展示最新1小时的集群监控信息。

为提升用户体验，云数据库ClickHouse对控制台进行了优化，2021年12月01日之后购买的集群提供了更加全面的性能监控数据。

 **说明** 为方便理解，本文将优化之前的控制台称为旧控制台，优化之后的控制台称为新控制台，此分类仅适用于本功能。

2021年12月01日之后购买的集群，监控信息具体请参见[新控制台监控项](#)。2021年12月01日之前购买的集群，监控信息具体请参见[旧控制台监控项](#)。

新控制台监控项

社区兼容版

监控项	说明
磁盘吞吐	磁盘读写的吞吐量（带宽）。
磁盘IOPS	磁盘每秒进行读写操作的次数。
磁盘使用量	磁盘空间的使用量。单位：MByte。
磁盘使用率	磁盘空间的使用量与最多可使用的磁盘空间容量的比值。
inode使用率	inode的使用个数占最多可使用的总个数的比值。
Data Part个数	数据分片的总数。
内存使用量	集群各节点的内存使用量。单位：MByte。
Inactive Data Part个数	不活跃的数据分片个数。
CPU使用率	集群各节点的平均CPU使用率。
内存使用率	集群各节点的平均内存使用率。
每秒写入大小	集群各节点每秒写入的数据量。单位：MByte。
网络吞吐	网络带宽。

监控项	说明
QPS	每秒处理的查询数目。
每秒写入行数	集群各节点每秒写入的行数。
TCP连接数	集群的TCP连接个数。
TPS	每秒处理的事务数目。
运行Query个数	正在运行的查询语句个数。
运行Init Query个数	正在运行的非转发查询语句个数。
运行Mutation个数	正在运行的数据订正任务个数。
HTTP连接数	集群的HTTP连接个数。
分布式DDL个数	分布式DDL语句的个数。
失败Query个数	查询失败的语句个数。
MaterializeMySQL个数	使用MaterializeMySQL引擎创建的同步数据库个数。
失败Insert Query个数	插入失败的语句个数。
Kafka外表个数	已创建的Kafka外表个数。
失败Select Query个数	查询失败的语句个数。
冷存使用量	冷数据的存储量。
运行Merge个数	正在运行的合并任务个数。
MaterializeMySQL同步失败个数	使用MaterializeMySQL引擎创建的同步数据库同步失败个数。
延迟Insert个数	延迟的插入语句个数。
Kafka外表消费发生错误个数	同步失败的Kafka消费表个数。
分布式表临时文件个数	分布式表临时文件的个数。

云原生版

监控项	说明
磁盘吞吐	磁盘读写的吞吐量（带宽）。
磁盘IOPS	磁盘每秒进行读写操作的次数。
磁盘使用量	磁盘空间的使用量。单位：MByte。
磁盘使用率	磁盘空间的使用量与最多可使用的磁盘空间容量的比值。

监控项	说明
inode使用率	inode的使用个数占最多可使用的总个数的比值。
Data Part个数	数据分片的总数。
内存使用量	计算组的内存使用量。单位：MByte。
内存使用率	计算组的平均内存使用率。
CPU使用率	计算组的平均CPU使用率。
网络吞吐	网络带宽。
每秒写入大小	计算组每秒写入的数据量。单位：MByte。
每秒写入行数	计算组每秒写入的行数。
QPS	每秒处理的查询数目。
TPS	每秒处理的事务数目。
连接数	连接集群的客户端个数。
失败Query个数	查询失败的语句个数。
运行Query个数	正在运行的查询语句个数。
运行Merge个数	正在运行的合并任务个数。
运行Mutation个数	正在运行的数据订正任务个数。
延迟Insert个数	延迟的插入语句个数。
分布式DDL个数	分布式DDL语句的个数。
分布式表临时文件个数	分布式表临时文件的个数。
冷存使用量	冷数据的存储量。

旧控制台监控项

监控项	说明
CPU使用率	集群各节点的CPU使用率。
内存使用率	集群各节点的内存使用率。
内存使用量	集群各节点的内存使用量。单位：MByte。
磁盘使用比率	使用中的磁盘空间与规格最大可使用磁盘空间的比值。
磁盘使用量	使用中的磁盘空间大小。单位：MByte。

监控项	说明
磁盘IOPS	磁盘每秒进行读写操作的次数。
磁盘IOPS大小	磁盘每秒读写数据的大小。
数据库连接使用比率	数据库使用连接数与规格最大可使用连接数的比值。
数据库使用连接数	数据库使用中的连接数目。
TPS	每秒处理的事务数目。
每秒写入行数	各节点每秒写入的行数。
每秒写入大小	各节点每秒写入的大小。单位：MByte。
QPS	每秒处理的查询数目。
ZK的平均等待时长	反应当前ZooKeeper的响应性能。单位：毫秒。
平均IO等待时长	反应当前IO的响应性能。单位：毫秒。
平均CPU等待时长	反应当前CPU的响应性能。单位：毫秒。

6.12.2. 报警规则设置

云数据库ClickHouse提供集群状态监控及报警功能，通过对重要的监控指标设置报警规则，让您在第一时间得知指标数据发生异常，帮您迅速定位处理故障。

背景信息

监控报警是通过[云监控](#)产品实现的。通过阿里云云监控，您可以设置监控项，在触发监控项报警规则时，通知报警联系组中的所有联系人。您也可以维护报警监控项对应的报警联系组，以便发生报警时，能及时通知到相关联系人。

为提升用户体验，云数据库ClickHouse对控制台进行了优化，云原生版集群和2021年12月01日以后新购的社区兼容版集群，无需跳转至云监控控制台设置报警规则。本文分别介绍优化前后报警规则的设置方法。

 **说明** 为方便理解，本文将优化之前的控制台称为旧控制台，优化之后的控制台称为新控制台，此分类仅适用于本功能。

云原生版集群和2021年12月01日以后新购的社区兼容版集群，请参考[新控制台操作步骤](#)。2021年12月01日之前购买的社区兼容版集群，请参考[旧控制台操作步骤](#)。

新控制台操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在[集群列表](#)页面，选择默认实例列表或云原生版实例列表，单击目标集群ID。
4. 单击左侧导航栏的[监控信息](#)。
5. 选择[集群告警](#)，单击[创建报警](#)。

6. 在创建报警面板，执行以下操作。

- i. (可选) 从告警模板下拉列表，选择模板。
- ii. 在规则名称文本框，输入规则名称，例如：网络接收压力报警。
- iii. 在告警表达式文本框，输入告警表达式。例如：

```
(sum(rate(kube_state_metrics_list_total{job="kube-state-metrics",result="error"}[5m])) / sum(rate(kube_state_metrics_list_total{job="kube-state-metrics"}[5m]))) > 0.01
```

。

 **注意** PromQL语句中包含的 `$` 符号会导致报错，您需要删除包含\$符号的语句中 `=` 左右两边的参数及 `=`。例如：将 `sum (rate (container_network_receive_bytes_total{instance=~"^\$HostIp.*"}[1m]))` 修改为 `sum (rate (container_network_receive_bytes_total [1m]))`。

- iv. 在持续时间文本框，输入持续时间N，当连续N分钟满足告警条件的时候才触发告警。例如：1分钟，当告警条件连续1分钟都满足时才会发送告警。

 **说明** 持续N分钟满足告警条件是指在连续N分钟内，您上面设置的PromQL语句条件都能满足。Prometheus默认数据采集周期为15s，如果没有连续 $N \times 60 / 15 = 4N$ 个数据点满足告警条件（PromQL语句），就不会发送告警。如Prometheus告警规则默认持续时间为1分钟，既只有连续4个数据点都满足告警条件（PromQL语句）才会触发告警。如果您想在任何一个数据点满足告警条件（PromQL语句）就发送告警，请修改持续时间为0分钟。

- v. 在告警消息文本框，输入告警消息。
- vi. (可选) 在高级配置的标签区域，单击创建标签可以设置报警标签，设置的标签可用作分派规则的选项。

- vii. (可选) 在高级配置的注释区域, 单击创建注释, 设置键为 `message`, 设置值为 `{{变量名}}` 告警信息。设置完成后的格式为: `message:{{变量名}}告警信息`, 例如: `message:{{${labels.pod_name}}重启`。

您可以自定义变量名, 也可以选择已有的标签作为变量名。已有的标签包括:

- 报警规则表达式指标中携带的标签。
- 通过报警规则创建的标签。
- ARMS系统自带的默认标签, 默认标签说明如下。

标签	说明
<code>alertname</code>	告警名称, 格式为: 告警名称_集群名称。
<code>_aliyun_arms_alert_level</code>	告警等级。
<code>_aliyun_arms_alert_type</code>	告警类型。
<code>_aliyun_arms_alert_rule_id</code>	告警规则对应的ID。
<code>_aliyun_arms_region_id</code>	地域ID。
<code>_aliyun_arms_userid</code>	用户ID。
<code>_aliyun_arms_involvedObject_type</code>	关联对象子类型, 如 ManagedKubernetes, ServerlessKubernetes。
<code>_aliyun_arms_involvedObject_kind</code>	关联对象分类, 如app, cluster。
<code>_aliyun_arms_involvedObject_id</code>	关联对象ID。
<code>_aliyun_arms_involvedObject_name</code>	关联对象名称。

- viii. 从通知策略下拉列表, 选择通知策略。

如何创建通知策略, 请参见[通知策略](#)。

- ix. 单击确定。

报警配置页面显示创建的报警。

报警名称	报警分类	阈值	状态	通知策略	操作
网络负载压力报警	Kubernetes 工作负载	5m	已启用	网络负载	编辑 删除 更多

旧控制台操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角, 选择目标集群所在的地域。
3. 在集群列表页面, 选择默认实例列表, 单击目标集群ID。
4. 单击左侧导航栏的监控信息。
5. 单击右上方导航栏的告警监控。
6. 在云监控控制台中, 单击左侧导航栏中的报警服务 > 报警规则。
7. 在阈值报警页签下, 单击创建报警规则。
8. 在创建报警规则页面, 配置如下参数。

i. 设置关联资源，参数说明如下。

参数	说明
产品	下拉选择云数据库ClickHouse。
资源范围	<ul style="list-style-type: none">全部资源：所选产品下任何集群满足报警规则描述时，都会发送报警通知。集群：选择指定的集群，则选中的集群满足报警规则描述时，才会发送报警通知。
地域	资源范围为集群时，需配置该参数。 选择集群所属地域。
集群	资源范围为集群时，需配置该参数。 选择集群ID，可选择多个集群。

ii. 设置报警规则，如何设置，具体请参见[创建报警规则](#)。

iii. 设置报警通知方式。

 **说明** 联系人通知组需要预先创建好，如何创建，具体请参见[创建报警联系人或报警联系组](#)。

9. 完成上述参数配置后，单击**确认**，报警规则将自动生效。

6.13. 查询管理

本文介绍了如何使用云数据库ClickHouse控制台进行慢SQL监控和运行中查询管理。协助及时发现慢SQL，进行SQL优化，或者进行临时的Kill请求操作，释放资源。

背景信息

原始QueryLog表使用过程繁琐，需要进行客户端安装，编写SQL，结果排序等，无法进行直接的趋势展现，整体使用成本高，易用性和效率低。云数据库ClickHouse将QueryLog进行白屏化的呈现，提供关键维度的可视化检索，慢SQL的统计分析。通过简单点选就能轻松实现对于日志数据的分析汇总，帮助进行问题定位和处理。

慢SQL监控

慢SQL监控提供对选定时间区间内的慢SQL进行趋势统计、趋势监控和慢SQL详情。

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在[集群列表](#)页面，选择**默认实例列表**或**云原生版实例列表**，单击目标集群ID。
4. 单击左侧导航栏中的**查询管理**。
5. 单击**慢查询**页签。

如果您的集群是云原生版，请先选择计算组。

6. 自定义慢查询时间。默认查询时长超过3000 ms的为慢SQL。

说明 慢SQL执行时长最小值为500 ms。

7. 您可以通过以下两种方式查看慢SQL趋势或明细。

- 单击最近15分钟、最近30分钟或最近1小时，查看最近15分钟、最近半小时或最近1小时的慢日志情况。
- 设置查询起止日期，然后单击查看，可以查看最多7天的慢日志情况。

说明

- 起始时间从不能早于当前时刻往前7*24小时，结束时间不能超过当前时刻。
- 时间选择精度1分钟。

8. 查看慢SQL趋势。

- 横坐标：时间轴。
- 左纵坐标：慢SQL数量，单位：个。
- 右纵坐标：最大执行时间，单位：毫秒。

9. 查看慢SQL明细。

10. 导出慢SQL数据。

发起用户	访问源地址	初始查询ID	SQL语句	执行开始时间	运行时间(ms)	读取行数	读取数据大小	返回结果大小	内存使用量
没有数据									

每页显示 30 | 50 | 100

正在运行的SQL管理

展示运行中的SQL详情，支持将当前筛选条件下的运行中状态的SQL进行实时展现，同时提供终止选定SQL的处理，达到临时释放资源的目的。

- 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
- 在页面左上角，选择目标集群所在的地域。
- 在集群列表页面，选择默认实例列表或云原生版实例列表，单击目标集群ID。
- 单击左侧导航栏中的查询管理。
- 单击正在运行的查询页签。

如果您的集群是云原生版，请先选择计算组。

- 您可以单击下拉菜单选项，通过用户、SQL关键字、初始查询ID的维度过滤查询结果。

用户	请求ip	请求ID	开始时间	运行时长(ms)
没有数据				

每页显示 30 | 50 | 100

说明 不设置过滤查询条件时，默认展示状态为运行中的全部SQL。

- 当集群出现宕机、或资源使用达到告警水位时，您可以根据需要勾选部分慢SQL后点击终止选中查询或终止所有查询，终止当前集群中已勾选或所有的查询请求。

 **说明** 单击终止选中查询或终止所有查询后，弹出确认窗口，如果在确认窗口中单击取消，则后台不进行任何kill操作。

6.14. RDS MySQL分析实例

RDS MySQL分析实例本质是一个云数据库ClickHouse实例，本文介绍如何创建和管理RDS MySQL分析实例。

背景信息

RDS MySQL分析实例通过在云数据库ClickHouse中自动创建MaterializedMySQL引擎库，将RDS MySQL数据同步到云数据库ClickHouse中，利用其超高的分析性能解决多维聚合实时分析问题。

 **说明** 关于MaterializeMySQL引擎，请参见[MaterializeMySQL引擎](#)。

使用限制

- 云数据库ClickHouse的云原生版集群不支持RDS MySQL分析实例。
- 目标实例只支持20.8及其以上版本的云数据库ClickHouse社区兼容版集群。
- 用户具有RDS MySQL的Binlog访问权限以及RDS MySQL数据库的RELOAD、REPLICATION SLAVE、REPLICATION CLIENT和SELECT权限。

 **说明** 您可以通过 `GRANT RELOAD, REPLICATION SLAVE, REPLICATION CLIENT, SELECT ON *.* TO 'your-user-name';` 命令获取RDS MySQL数据库的相关权限。

- RDS MySQL实例版本需满足以下条件才可以创建RDS MySQL分析实例：
 - RDS MySQL 8.0三节点企业版
 - RDS MySQL 8.0高可用版（SSD云盘、ESSD云盘或本地SSD盘）
 - RDS MySQL 5.7三节点企业版
 - RDS MySQL 5.7高可用版（SSD云盘、ESSD云盘或本地SSD盘）
 - RDS MySQL 5.6
- RDS MySQL源实例和云数据库ClickHouse目标实例必须在同一地域、同一VPC网络。
- 同一集群的同一数据库不能重复配置同步任务。
- 没有主键的表无法同步。
- 云数据库ClickHouse的 `DateTime` 类型数据的时间范围为 `[1970-01-01 00:00:00,2106-02-07 06:28:15]`，如果RDS MySQL中的时间超出当前范围，会导致同步到云数据库ClickHouse的时间值不正确。

操作步骤

1. 创建同步任务。
 - i. 登录[云数据库ClickHouse控制台](#)。

 **说明** 您也可以直接登录[云数据库RDS控制台](#)，参考步骤v及以后的操作步骤。

- ii. 在[集群列表](#)页面，选择默认实例列表，单击目标集群ID。
- iii. 在左侧导航栏单击MySQL实时同步。

iv. 如果是首次创建同步任务，单击提示语句中的RDS for MySQL，跳转至云数据库RDS控制台。



v. 在实例列表页面，单击目标实例ID。

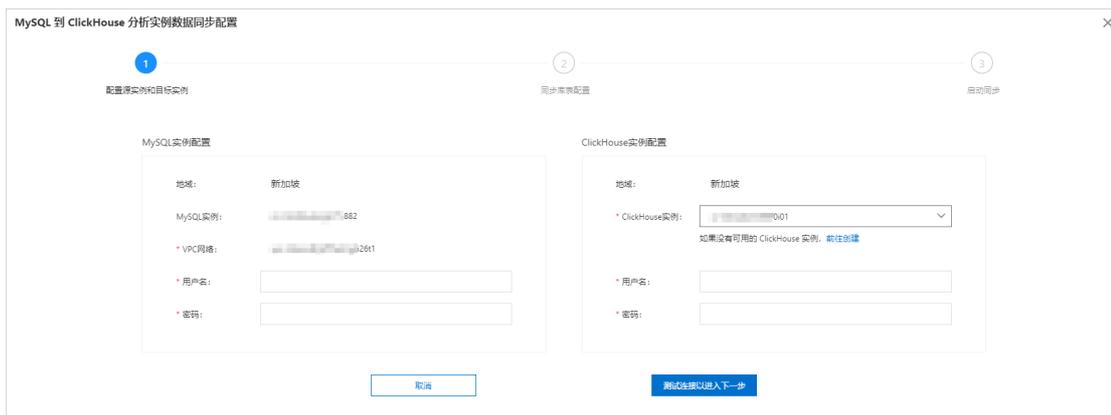
vi. 在基本信息页面的实例分布区域，单击分析实例后的云ClickHouse分析实例。



2. 如果是首次创建分析实例，需要进行授权。具体步骤如下。

说明 若您之前已完成授权操作，单击云ClickHouse分析实例直接跳转至MySQL到ClickHouse分析实例数据同步配置页面。

- i. 在弹出的创建服务关联角色窗口，单击AliyunServiceRoleForClickHouse。
 - ii. 单击确认。
3. 配置数据同步信息。
- i. 在MySQL到ClickHouse分析实例数据同步配置页面，选择ClickHouse实例，并分别填写MySQL实例和ClickHouse实例对应的用户名和密码。

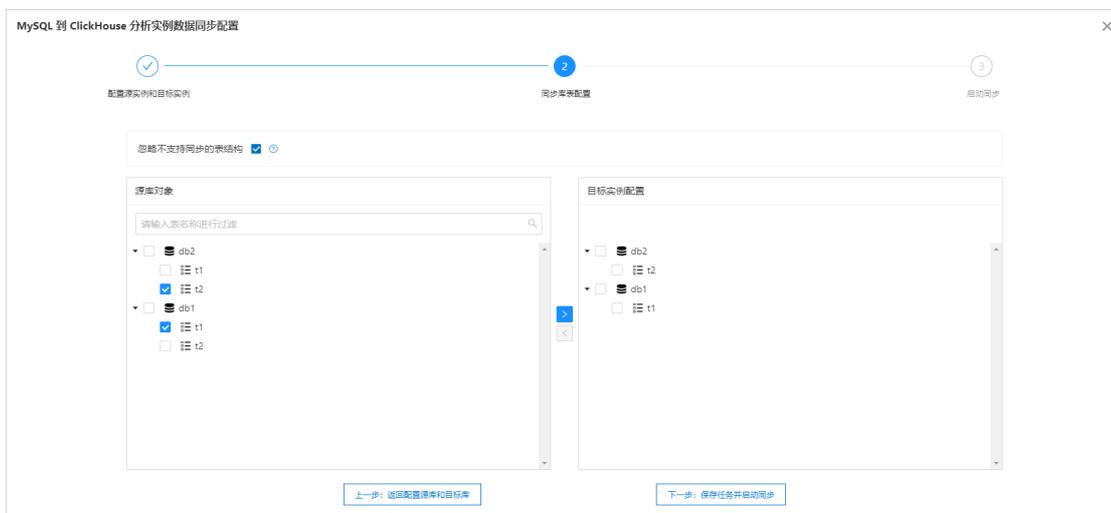


- ii. 单击测试连接以进行下一步，进行实例连通性校验。
 - 测试连接成功，进行下一步。
 - 测试连接失败，进行异常提示。请根据提示信息进行排查。
- iii. 在源库对象框中选中待同步的表。

说明

- 默认会选中忽略不支持同步的表结构，建议保持选中状态。目前不支持同步无主键的表。
- 不能只选中同步库，只有选中要同步的表才能配置成功。

iv. 单击  图标将其同步至目标实例配置框。



v. 单击下一步：保存任务并启动同步。

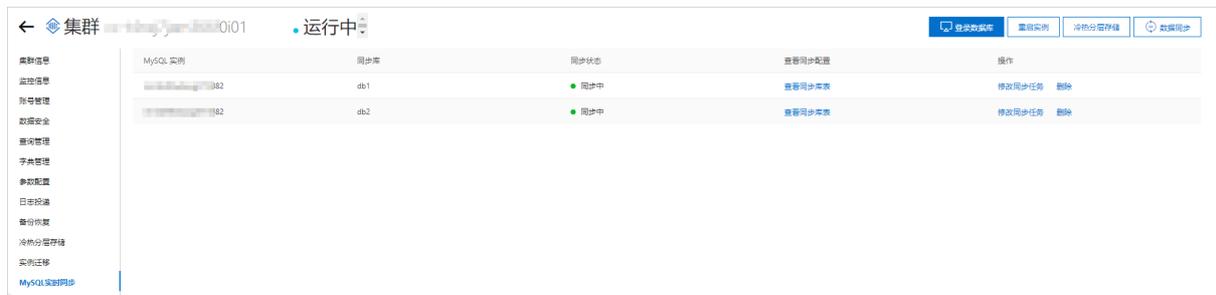
- 配置成功，进行下一步。
- 配置失败，进行异常提示。请按照提示信息进行排查。

vi. (可选) 配置成功后，数据同步任务启动，单击点击查看同步任务。

 说明 如果同步任务时出现异常提示：task inner error，请及时提交工单。

管理RDS MySQL分析实例

您可以在MySQL实时同步页面查看同步任务的相关信息并进行管理操作。



- 查看同步库表

在查看同步配置列单击查看同步库表，查看同步库表信息。



- 修改同步任务
 - i. 在操作列单击修改同步任务。
 - ii. 在MySQL到ClickHouse分析实例数据同步配置页面重新配置数据同步信息。具体请参见[配置数据同步信息](#)。
- 删除同步任务
 - i. 在操作列单击删除。
 - ii. 在弹出的提示窗口单击确定。

6.15. 操作审计日志

云数据库ClickHouse与阿里云操作审计（ActionTrail）集成，支持在操作审计（ActionTrail）中查看用户行为日志，同时支持通过操作审计（ActionTrail）将日志投递到日志服务 LogStore 或指定的OSS Bucket中，满足实时审计、问题回溯分析等需要。本文介绍如何查看云数据库ClickHouse的操作审计日志。

背景信息

操作审计（ActionTrail）监控并记录了阿里云账号访问云数据库ClickHouse服务时的下列事件：

- 售卖事件，具体事件列表请参见[售卖事件说明](#)。
- API事件，具体事件列表请参见[API事件说明](#)。

使用限制

- 只有单账号跟踪的事件可以通过操作审计控制台查询，且每秒最多可以查询两次。多账号跟踪的事件不能通过操作审计控制台查询，只能在对应的SLS Logstore或OSS存储空间中查询。具体操作，请参见[查询多账号跟踪的事件](#)。
- 操作审计事件查询功能只能查询当前地域最近90天的事件。
 - 如果需要查询超过90天的事件，您必须创建单账号跟踪并将事件投递到相应的存储服务。否则，将无法追溯90天以前的事件。具体操作，请参见[创建单账号跟踪](#)。
 - 如果需要同时查询多个地域超过90天的事件，或者使用多个搜索条件过滤查询事件，您可以使用事件高级查询功能。具体操作，请参见[自定义查询事件](#)。
- 云上操作后10分钟才可以通过操作审计控制台查询相关事件。

注意事项

- 操作审计支持免开通，任何已完成实名认证且未欠费的有效阿里云账号访问[ActionTrail控制台](#)。
- 如果您使用操作审计时，通过创建跟踪将事件投递到日志服务SLS或对象存储OSS，需要按照SLS或OSS的定价单独付费。

更多信息，请参见[计费说明](#)。

查看操作审计日志

1. 登录[操作审计控制台](#)。
2. 在左侧导航栏，单击[事件查询](#)。
3. 在顶部导航栏选择您想查询事件的地域。
4. 在[事件查询](#)页面找到待查询的事件，鼠标悬停至事件名称，[查询事件详情](#)。

 **说明** 您可以设置读写类型、用户名（RAM用户）、事件名称、资源类型、资源名称、服务名称、AccessKey ID等单个搜索条件来过滤查询事件。

5. （可选）如果需要查询事件代码记录，您可以单击事件前面的加号，然后单击[事件详情](#)。
ClickHouse事件详情代码示例如下：

```
{
  "ApiVersion": "2019-11-11",
  "RequestId": "76BEA6CF-****-****-****-12393F559EFF",
  "EventType": "ApiCall",
  "UserIdentity": {
    "Type": "ram-user",
    "InvokedBy": "",
    "AccountId": "",
    "UserName": "",
    "PrincipalId": "20*****95",
    "AccessKeyId": "TMP.******4K99m",
    "Arn": ""
  },
  "AcsRegion": "cn-shenzhen",
  "EventName": "CreateAccountAndAuthority",
  "IsBlack": false,
  "RequestParameters": {
    "AcsHost": "clickhouse-share.aliyuncs.com",
    "RequestId": "76BEA6CF-****-****-****-12393F559EFF",
    "DBClusterId": "cc-2z*****7q",
    "HostId": "clickhouse-share.aliyuncs.com",
    "AllowDatabases": "sh****g",
    "AccountPassword": "*****",
    "DdlAuthority": true,
    "DmlAuthority": "all",
    "AcsProduct": "clickhouse",
    "TotalDatabases": "de***lt,sh***g",
    "TotalDictionaries": "",
    "AllowDictionaries": "",
    "AcceptLanguage": "zh-CN",
    " charset": "UTF-8",
    "AccountName": "root"
  },
  "EventSource": "clickhouse-share.aliyuncs.com",
  "ServiceName": "ClickHouse",
  "EventTime": "2021-06-08T08:28:57.497+0000",
  "ReferencedResources": {},
  "UserAgent": "clickhouse.console.aliyun.com",
  "EventId": "76BEA6CF-****-****-****-12393F559EFF",
  "ResponseElements": {
    "RequestId": "76BEA6CF-****-****-****-12393F559EFF"
  },
  "ErrorCode": "",
  "ErrorMessage": "",
  "EventVersion": "",
  "SourceIpAddress": "11*.*.*.*7"
}
```

更多操作

操作审计日志投递至SLS/OSS的步骤说明，请参见[通过SLS或OSS控制台查询事件](#)。

 **注意** 使用审计日志投递会产生费用，更多信息请参见[计费说明](#)。

售卖事件说明

云数据库ClickHouse操作审计日志会记录以下售卖事件：

事件类型	事件名称	事件含义
AliyunServiceEvent	Create	登录产品售卖页面进行的购买操作
AliyunServiceEvent	Renew	登录产品售卖页面进行的续费操作
AliyunServiceEvent	Release	登录产品售卖页面进行的释放操作

API事件说明

云数据库ClickHouse操作审计日志会记录以下API事件：

事件名称	事件含义
AllocateClusterPublicConnection	创建实例的公网连接地址
CheckScaleOutBalanced	检查是否符合升降配条件
CheckServiceLinkedRole	检查是否符合授权链要求
CreateAccount	为实例创建高权限账号
CreateAccountAndAuthority	创建账号和权限
CreateBackupPolicy	创建备份策略
CreateDBInstance	创建集群实例
CreateOSSStorage	创建分层存储
CreatePortsForClickHouse	创建端口
CreateServiceLinkedRole	创建授权链
DeleteAccount	删除数据库高权限账号
DeleteDBCluster	释放按量付费的ClickHouse集群
DeleteLorneTask	删除SLS投递任务
DescribeAccountAuthority	查询账号权限
DescribeAccounts	查询指定集群的数据库登录账号
DescribeAllDataSource	枚举实例下所有Schema列表、Table列表和Columns列表

DescribeAvailableResource	查询是否有可用资源
DescribeBackupPolicy	查询备份策略
DescribeBackups	查询备份集
DescribeDBClusters	查询集群列表或被RAM授权的集群列表
DescribeDBClusterAccessWhiteList	查询实例IP白名单
DescribeDBClusterAttribute	查询指定实例的详细属性
DescribeDBClusterConfig	查询集群配置
DescribeDBClusterStatusSet	查询集群状态
DescribeDBClusterNetInfoItems	查询集群的网络信息
DescribeDBClusterPerformance	查询实例性能数据
DescribeDBConfig	查询字典配置
DescribeLogHubAttribute	查询SLS投递任务的属性
DescribeLoghubDetail	查询SLS投递任务的明细
DescribeLorneLog	查询SLS投递任务日志
DescribeLorneTasks	查询SLS投递任务列表
DescribeLorneTasksMCount	查询SLS投递指标
DescribeLorneTasksMetrics	查询SLS投递指标明细
DescribeOSSStorage	查询分层存储信息
DescribeRegions	查询可选的ClickHouse地域和可用区
DescribeProcessList	查询正在运行的SQL
DescribeSchemas	查询数据库结构列表
DescribeSlowLogRecords	查询慢SQL
DescribeSlowLogTrend	查询慢SQL趋势
DescribeTables	查询表结构列表
KillProcess	停止正在运行的SQL
ModifyAccountAuthority	修改账号权限
ModifyAccountDescription	修改ClickHouse数据库账号的备注信息

ModifyBackupPolicy	修改备份策略
ModifyDBCluster	集群升降配
ModifyDBClusterAccessWhiteList	修改实例的白名单
ModifyDBClusterConfig	修改集群配置
ModifyDBClusterDescription	修改实例的备注名
ModifyDBClusterMaintainTime	修改实例可维护时间段
ModifyDBConfig	修改字典配置
OperateLogHub	配置SLS投递任务
OperateLogTaskStatus	修改SLS投递任务状态
ReleaseClusterPublicConnection	删除集群的公网地址
ResetAccountPassword	重置高权限账号的密码

6.16. 冷热数据分层存储

为了降低海量数据存储成本，云数据库ClickHouse提供了冷热数据分层存储的功能。

背景知识

- 热数据：访问频次较高的数据，存储在热数据盘（即创建实例时所选的ESSD云盘或高效云盘）中，满足高性能访问的需求。
- 冷数据：访问频次较低的数据，存储在较低价的冷数据盘中，满足高性价比的存储需求。
- 冷热数据分层存储提供了两种存储策略，具体如下：
 - 默认存储策略：将新写入的数据存储在热数据盘中，提供高效查询。当热数据存储量达到业务使用阈值时，自动将最早写入热数据盘的数据移动到冷数据盘存储，从而释放热数据盘存储空间。
 - TTL分层存储策略：在默认存储策略基础上，添加TTL语句，实现将间隔时间之前的所有数据自动转移到冷数据盘中。

注意事项

- 云数据库ClickHouse的云原生版集群不支持冷热数据分层存储功能。
- 云数据库ClickHouse的社区兼容版集群仅20.8及以上版本支持冷热数据分层存储功能。对于不支持的低版本，可以先将数据迁移至20.8及以上版本的实例，再进行冷热数据分层存储。具体请参见[云数据库ClickHouse实例间的数据迁移](#)。
- 冷热数据分层存储功能开通时，实例会进行重启。
- 如果要使用冷热数据分层存储功能，必须在建表时添加 `SETTINGS storage_policy = 'hot_to_cold'`；语句指定存储策略。
- 存储策略一旦指定，后期无法更改。
- 冷热数据分层存储功能开通后，暂不支持关闭。
- 已开通数据备份功能的实例不支持再开通冷热分层存储功能。

计费说明

- 计费规则如下。
 - 热数据存储：购买集群后已经拥有了热数据存储空间，与购买集群的存储空间相同。
 - 冷数据存储：采用按量付费模式。无需提前购买固定的存储空间，先使用，后付费，并按照实际使用的存储空间每小时计费一次。
- 计费公式：每小时费用=冷数据存储单价×冷数据存储使用量。
- 示例：若冷数据存储单价为0.0002元/GB/小时，19:00-20:00使用冷数据存储2000 GB，则总费用为：0.0002元/GB/小时×2000 GB×1小时=0.4元。20:00-21:00释放了500 GB冷数据存储空间，实际使用了1500 GB存储空间，那么20:00-21:00的总费用为：0.0002元/GB/小时×1500 GB×1小时=0.3元。

 **说明** 本示例中的价格仅用于举例，实际价格以控制台为准。关于冷数据存储价格，具体请参见[计费概述](#)。

操作步骤

1. 登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择实例所在的地域。
3. 在集群列表页面，单击目标集群ID。
4. 在左侧导航栏单击冷热分层存储。
5. 单击点击开通冷热分层存储。
6. 在弹出的冷热数据分层开启提示窗口，单击确定。

确定开通后，实例会进行重启。待实例状态从配置变更中变为运行中，即完成开通。

开通冷热分层存储功能后，默认存储策略的相关参数如下表所示。

参数	说明
move_factor	<p>当热数据盘中的可用存储空间所占比例小于该参数时，自动将当前热数据盘中占用空间最大的数据文件移动到冷数据盘存储。</p> <p>取值范围为 <code>[0,1]</code>。取值为0时，代表关闭基于磁盘存储容量的冷热数据分层存储，您可以设置TTL分层存储策略。默认值为0.1，代表可用存储空间所占比例小于10%时，将自动进行数据移动。</p> <p> 说明 如何查看磁盘空间，请参见查看磁盘空间。</p>
prefer_not_to_merge	<p>冷数据盘中的数据是否进行合并。取值范围如下。</p> <ul style="list-style-type: none">• true: 不合并，默认值。• false: 合并。

设置TTL分层存储策略

在默认存储策略的基础上，添加TTL语句，实现将间隔时间之前的所有数据自动转移到冷数据盘中。

您可以参考如下语法添加TTL语句。

```
TTL <time_column> + INTERVAL <number> TO DISK 'cold_disk'
```

参数说明如下。

参数	说明
time_column	日期或日期时间类型的列。
number	间隔时间。 常用单位有：Day、Week、Month、Year。

 **说明** 关于TTL语句的更多信息，请参见[TTL](#)。

示例

以date列为依据，将90天之前的所有数据移动到冷数据盘。建表语句如下。

```
CREATE TABLE ttl_test_tbl
(
  `f1` String,
  `f2` String,
  `f3` Int64,
  `f4` Float64,
  `date` Date
)
ENGINE = MergeTree()
PARTITION BY date
ORDER BY f1
TTL date + INTERVAL 90 DAY TO DISK 'cold_disk'
SETTINGS storage_policy = 'hot_to_cold';
```

更改TTL分层存储策略

您可以更改TTL分层存储策略的日期或日期时间类型的列以及间隔时间。

说明

- 默认情况下，更改TTL分层存储策略后存量数据和新增数据全部按照新策略进行存储。
- 如果您不想更改存量数据的TTL分层存储策略，可以在更改TTL分层存储策略之前，执行 `set materialize_ttl_after_modify=0;` 语句。执行成功后，再更改TTL分层存储策略。存量数据按照原有策略进行存储，不发生改变。只有新增数据按照新策略进行存储。
- 更改TTL分层存储策略后，已经进入冷数据盘中的数据不会自动移动到热数据盘。如果需要移动，请参见[移动冷热数据盘的数据](#)。

您可以参考如下语法更改TTL分层存储策略。

```
ALTER TABLE <table_name> ON CLUSTER default MODIFY TTL <time_column> + INTERVAL <number> TO DISK 'cold_disk';
```

参数说明如下。

参数	说明
table_name	表名。
time_column	更改后的日期或日期时间类型的列。
number	更改后的间隔时间。 常用单位有：Day、Week、Month、Year。

移动冷热数据盘的数据

- 移动热数据盘数据到冷数据盘。

```
ALTER TABLE <table_name> ON CLUSTER default MOVE PARTITION <partition> TO DISK 'cold_disk';
```

- 移动冷数据盘数据到热数据盘。

```
ALTER TABLE <table_name> ON CLUSTER default MOVE PARTITION <partition> TO DISK 'default';
```

 说明 `partition` 为分区名称，您可以在system.parts系统表中查看。

查看冷热数据盘上的数据

- 查看热数据盘上的数据。

```
select * from system.parts where database = '<db_name>' and table = '<tbl_name>' and disk_name = 'default' and active = 1;
```

- 查看冷数据盘上的数据。

```
select * from system.parts where database = '<db_name>' and table = '<tbl_name>' and disk_name = 'cold_disk' and active = 1;
```

查看磁盘空间

开通冷热数据分层存储功能后，您可以执行如下语句查看磁盘空间。

```
select * from system.disks;
```

返回结果如下。

```
┌name┐┌path┐┌free_space┐┌total_space┐┌used_space┐┌keep_free_
space┐
├──┴──┤├──┴──┤├──┴──┤├──┴──┤├──┴──┤├──┴──┤
| cold_disk | /clickhouse/data/data/disks/cold_disk/ | 18446744073709551615 | 18446744073709
551615 | 115312080698 | 0 |
├──┴──┤├──┴──┤├──┴──┤├──┴──┤├──┴──┤├──┴──┤
| default | /clickhouse/data/data/ | 201663209472 | 207083249664
| 5420040192 | 0 |
├──┴──┤├──┴──┤├──┴──┤├──┴──┤├──┴──┤├──┴──┤
```

参数说明如下。

参数	说明
name	磁盘名称。取值范围如下。 <ul style="list-style-type: none">• cold_disk: 冷数据盘。• default: 热数据盘。
path	磁盘上的数据存储路径。
free_space	当前磁盘剩余可用空间。冷数据盘可用空间不受限制。
total_space	当前磁盘总空间。冷数据盘可用空间不受限制。
used_space	当前磁盘已经被使用的空间。
keep_free_space	预留的空闲空间。

6.17. 数据字典

本文介绍了新建、修改和使用ClickHouse外部扩展字典的操作步骤。

背景信息

ClickHouse数据字典分为外部扩展字典和内置字典，其中内置字典为ClickHouse预定义字典内容，外部扩展字典提供多种灵活的自定义的数据源定义，使用相对更加广泛，本文所讨论的数据字典为ClickHouse外部扩展字典。在数仓开发和数据分析过程中，当业务中有常量数据或者业务数据可以划分为业务事实表和业务维度表时，您可以通过使用数据字典来替代维度表实现，避免使用join语句，从而提升查询效率。

注意事项

云数据库ClickHouse云原生版集群的字典管理通过SQL支持。

新建字典配置

1. 使用阿里云账号登录[云数据库ClickHouse控制台](#)。
2. 在页面左上角，选择目标集群所在的地域。
3. 在集群列表页面，选择默认实例列表，单击目标集群ID。
4. 在左侧导航栏，单击字典管理，进入字典列表页面。
5. 单击右上角新增字典配置。
6. 在弹出的新增字典配置窗口编辑模板填写字典内容。

参数	描述
<name>	自定义字典名称，必须全局唯一。
<source>	配置字典的数据源，表示字典中的数据来源。目前ClickHouse支持的数据源为： <ul style="list-style-type: none">◦ MySQL◦ ClickHouse

参数	描述
<lifetime>	字典中数据的更新频率，单位：秒。
<layout>	内存中的数据格式类型，目前扩展字典共拥有7种类型。 <ul style="list-style-type: none"> ○ 单数值key <ul style="list-style-type: none"> ▪ flat ▪ hashed ▪ range_hashed ▪ cache ○ 复合key <ul style="list-style-type: none"> ▪ complex_key_hashed ▪ complex_key_cache ▪ ip_trie
<structure>	字典的数据结构。 <div style="border: 1px solid #ccc; background-color: #e0f2f1; padding: 5px; margin-top: 10px;"> ? 说明 单次只允许添加一个节点。 </div>

7. 填写完毕后单击**确定**，数据字典创建完成。

创建成功后，在数据字典对应的操作列，可进行查看、删除和修改。

字典表DDL

云数据库ClickHouse集群20.8之后的版本（您可以登录ClickHouse控制台-集群信息-查看版本号）增加了数据字典的DDL，您可以直接使用DDL语句新建扩展字典表。

```
CREATE DICTIONARY [IF NOT EXISTS] [db.]dictionary_name [ON CLUSTER cluster]
(
    key1 type1 [DEFAULT|EXPRESSION expr1] [HIERARCHICAL|INJECTIVE|IS_OBJECT_ID],
    key2 type2 [DEFAULT|EXPRESSION expr2] [HIERARCHICAL|INJECTIVE|IS_OBJECT_ID],
    attr1 type2 [DEFAULT|EXPRESSION expr3],
    attr2 type2 [DEFAULT|EXPRESSION expr4]
)
PRIMARY KEY key1, key2
SOURCE(SOURCE_NAME([param1 value1 ... paramN valueN]))
LAYOUT(LAYOUT_NAME([param_name param_value]))
LIFETIME([MIN val1] MAX val2)
```

数据字典的使用

字典表查询

```
SELECT
  name,
  type,
  key,
  attribute.names,
  attribute.types,
  bytes_allocated,
  element_count,
  source
FROM system.dictionaries
```

查询字典数据

使用dictGet语句结合函数获取数据字典中的数据。

```
dictGet('dict_name', 'attr_name', id_expr)
```

或

```
dictGetOrDefault('dict_name', 'attr_name', id_expr, default_value_expr)
```

7.SDK参考

7.1. SDK使用说明

云数据库ClickHouse支持三种语言的SDK，包括Java、Python和Go。

 **说明** 更多SDK信息，请参见[阿里云开放平台](#)。

下载地址

Alibaba Cloud SDK	ClickHouse	说明文档
Alibaba Cloud SDK for Java	Alibaba Cloud ClickHouse SDK for Java	<ul style="list-style-type: none">Java版SDK使用说明快速开始 (Java)
Alibaba Cloud SDK for Python	Alibaba Cloud ClickHouse SDK for Python	<ul style="list-style-type: none">Python版SDK使用说明快速开始 (Python)
Alibaba Cloud SDK for Go	Alibaba Cloud ClickHouse SDK for Go	无

Java版SDK使用说明

如果您使用Maven管理Java项目，只需在项目的 `pom.xml` 文件中添加Maven依赖即可。

```
<!-- 阿里云基础依赖包 -->
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-core</artifactId>
  <version>4.5.3</version>
</dependency>
<!-- clickhouse 依赖包 -->
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-clickhouse</artifactId>
  <version>1.1.4</version>
</dependency>
```

示例

```
import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.IAcsClient;
import com.aliyuncs.clickhouse.model.v20191111.DescribeDBClustersRequest;
import com.aliyuncs.clickhouse.model.v20191111.DescribeDBClustersResponse;
import com.aliyuncs.profile.DefaultProfile;
public class DescribeDBClustersMain {
    public static void main(String[] args) throws Exception{
        String accessKeyId = "";
        String accessKeySecret = "";
        String region = "";
        DefaultProfile profile = DefaultProfile.getProfile(
            "<your-region-id>", // 地域ID
            "<your-access-key-id>", // RAM账号的AccessKey ID
            "<your-access-key-secret>"); // RAM账号的AccessKey Secret
        IAcsClient client = new DefaultAcsClient(profile);
        DescribeDBClustersRequest request = new DescribeDBClustersRequest();
        DescribeDBClustersResponse response = client.getAcsResponse(request);
        System.out.println(response.getRequestId());
    }
}
```

Python版SDK使用说明

如果您使用Python版SDK，请先安装阿里云基础依赖和云数据库ClickHouse OpenAPI依赖。

- 如果您使用的是Python 2环境，请使用如下命令安装：

```
pip install aliyun-python-sdk-core
pip install aliyun-python-sdk-clickhouse
```

- 如果您使用的是Python 3环境，请使用如下命令安装：

```
pip3 install aliyun-python-sdk-core
pip3 install aliyun-python-sdk-clickhouse
```

 **注意** 不同环境依赖安装的命令是不一样的。如果代码执行的时候报某个类导入错误，请检查安装依赖的命令是否正确。

示例

```
#!/usr/bin/python
from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.acs_exception.exceptions import ClientException
from aliyunsdkcore.acs_exception.exceptions import ServerException
from aliyunsdkclickhouse.request.v20191111.DescribeRegionsRequest import DescribeRegionsRequest

# 创建AcsClient实例
client = AcsClient(
    "<your-access-key-id>", // RAM账号的AccessKey ID
    "<your-access-key-secret>"); // RAM账号的AccessKey Secret
    "<your-region-id>", // 地域ID
);
# 创建request，并设置参数。如果需要其他接口，请使用接口相应的Request
request = DescribeRegionsRequest()
response = client.do_action_with_exception(request)
print(response)
```

8. 常见问题

8.1. 常见问题

本文汇总了云数据库ClickHouse的常见问题及解决方案。

● 选型与购买

- 云数据库ClickHouse和官方版本对比多了哪些功能和特性？
- 购买实例时，推荐选择哪一个版本？
- 单双副本实例各有什么特点？
- 购买链路资源时显示“当前区域资源不足”，应该如何处理？

● 扩容与缩容

- 水平扩缩容耗时受什么影响？
- 扩缩容期间对实例有什么影响？
- 水平扩缩容升级有什么建议？

● 连接

- 每个端口的含义是什么？
- 每个Driver对应的端口是什么？
- Go、Python语言对应推荐什么Driver？
- 如何处理客户端工具连接集群时报错：connect timed out？
- 为什么MySQL、HDFS、Kafka等外表无法连通？
- 为什么程序无法连接ClickHouse？
- 如何处理ClickHouse超时问题？

● 迁移与同步

- 为什么OSS外表导入ORC、PARQUET等格式的数据，出现内存报错或OOM挂掉？
- 如何处理导入数据报错：too many parts？
- 为什么DataX导入速度慢？
- 为什么Hive导入后其数据行数跟ClickHouse对不上？
- 为什么Kafka导入后其数据行数跟ClickHouse对不上？
- 如何使用Spark、Flink导入数据？
- 如何从现有ClickHouse导入数据到云数据库ClickHouse？
- 使用MaterializeMySQL引擎同步MySQL数据时，为什么出现如下报错：The slave is connecting using CHANGE MASTER TO MASTER_AUTO_POSITION = 1, but the master has purged binary logs containing GTIDs that the slave requires？
- 使用MaterializeMySQL引擎同步MySQL数据时，为什么出现表停止同步？为什么系统表system.materialize_mysql中sync_failed_tables字段不为空？

● 数据写入与查询

- 如何处理insert into select XXX内存超限报错？
- 什么查询耗费的CPU、MEM高？
- 如何处理查询时内存超出限制？

- 如何处理查询报并发超限?
- 在数据停止写入时，同一个查询语句每次查询的结果不一致，应该如何处理?
- 为什么有时看不到已经创建好的表并且查询结果一直抖动时多时少?
- 如何处理往表里写入时间戳数据后查询出来的结果与实际数据不同?
- 如何处理建表后查询表不存在?
- 为什么Kafka外表建表后数据不增加?
- 为什么客户端看到的时间结果和时区显示的不一样?
- 为什么数据写入后不可见?
- 为什么optimize任务很慢?
- 为什么optimize后数据仍未主键合并?
- 为什么optimize后数据TTL仍未生效?
- 为什么optimize后更新删除操作没有生效?
- 如何进行DDL增加列、删除列、修改列操作?
- 为什么DDL执行慢，经常卡住?
- 如何处理分布式DDL报错：longer than distributed_ddl_task_timeout (=xxx) seconds?
- 如何处理语法报错：set global on cluster default?
- 有什么BI工具推荐?
- 有什么数据查询IDE工具推荐?
- **数据存储**
 - 如何查看每张表所占的磁盘空间?
 - 如何查看冷数据大小?
 - 如何查询哪些数据在冷存上?
 - 如何移动分区数据到冷存?
- **监控、升级、系统参数**
 - 为什么监控中存在数据中断情况?
 - 20.8后的版本是否支持平滑升级，不需要迁移数据?
 - 常用系统表有哪些?
 - 如何修改系统级别的参数？是否要重启，有什么影响?
 - 如何修改用户级别的参数?
 - 如何修改Quota?

云数据库ClickHouse和官方版本对比多了哪些功能和特性？

云数据库ClickHouse主要对社区版本进行稳定性Bug修复，同时提供特有的内核特性：

- **资源队列**进行用户角色级别的资源使用优先级配置。
- **二级索引**支持日志明细类点查询场景。

购买云数据库ClickHouse实例时，推荐选择哪一个版本？

云数据库ClickHouse根据开源社区公开的LTS内核稳定版提供服务，通常在版本推出3个月相对稳定后启动云服务售卖。当前建议购买20.8及以上版本。20.8版本提供**冷热数据分层存储**和**增强MaterializedMySQL**。

单双副本实例各有什么特点？

- 单副本实例每一个Shard节点无副本节点，无高可用服务保障。数据安全性基于云盘多副本存储，性价比高。
- 双副本实例每一个Shard节点对应一个副本服务节点，在主节点故障不能提供服务时副本节点可提供容灾服务支持。

购买链路资源时显示“当前区域资源不足”，应该如何处理？

解决方案：您可以选择同地域的其他区域购买或提交工单。VPC网络支持相同区域不同可用区之间打通，同区域网络延迟无感知。

水平扩缩容耗时受什么影响？

水平扩缩容过程涉及数据搬迁，实例里面数据越多搬得越多，耗时时间越长。

扩缩容期间对实例有什么影响？

为保证扩缩容中数据搬迁后的数据一致性，扩缩容期间实例处于可读不可写状态。

水平扩缩容升级有什么建议？

水平扩缩容耗时较长，当集群性能不满足时，请优先选择垂直升配。如何进行垂直升配，请参见[社区兼容版变配和扩缩容](#)。

每个端口的含义是什么？

社区兼容版

端口号	含义
3306	TCP端口。
8123	HTTP端口。
9004	MySQL端口。
8443	HTTPS端口。

云原生版

端口号	含义
9000	TCP端口。
8123	HTTP端口。
9004	MySQL端口。

每个Driver对应的端口是什么？

Driver名称	端口号
Java	8123

Driver名称	端口号
Python	3306
Go	

Go、Python语言对应推荐什么Driver?

详情请参见[第三方开发库](#)。

如何处理客户端工具连接集群时报错：connect timed out?

您可以采取如下解决方案。

- 检查网络是否畅通。通过 `ping` 命令检查网络是否通畅，通过 `telnet` 命令探测数据库3306和8123端口是否开放。
- 检查是否配置了ClickHouse白名单，配置方法请参见[设置白名单](#)。
- 检查客户端机器IP是否正确。通常公司办公网内的机器IP经常变动，用户看到的不是正确的IP地址。通过访问专业IP探查服务确定真实IP，示例请参见[whatsmyip](#)。

为什么MySQL、HDFS、Kafka等外表无法连通?

目前20.3和20.8版本在创建相关外表时程序内会自动进行验证，如果创建表成功，那说明网络是通的。如果无法创建成功，常见原因如下。

- 目标端和ClickHouse不在同一个VPC内，网络无法连通。
- MySQL端存在白名单相关设置，需要在MySQL端添加ClickHouse的白名单。

对于Kafka外表，表创建成功，但查询没有结果。常见原因是Kafka中数据通过表结构给出的字段和格式解析失败，报错信息会给出解析失败的具体位置。

为什么程序无法连接ClickHouse?

常见原因如下。

- VPC网络、公网网络环境不对。同一VPC内可用内网连接，不在同一VPC内需开设公网后连接。
- 白名单未配置。
- ECS安全组未放开。
- 公司设置了网络防火墙。

如何处理ClickHouse超时问题?

详情请参见[如何处理ClickHouse超时问题](#)。

为什么OSS外表导入ORC、PARQUET等格式的数据，出现内存报错或OOM挂掉?

常见原因：内存使用率比较高。

您可以采取如下解决方案。

- 把OSS上的文件拆分为一个一个的小文件，然后再进行导入。
- 进行内存的升配。如何升配请参见[社区兼容版变配和扩缩容](#)。

如何处理导入数据报错：too many parts?

ClickHouse每次写入都会生成一个data part，如果每次写入一条或者少量的数据，那会造成ClickHouse内部有大量的data part（会给merge和查询造成很大的负担）。为了防止出现大量的data part，ClickHouse内部做了很多限制，这就是too many parts报错的内在原因。出现该错误，第一要做的就是增加写入的批量大小。如果确实没有办法调整批量大小，可以适当的修改参数：`max_partitions_per_insert_block`。

为什么DataX导入速度慢？

常见原因及解决方案如下。

- 常见原因1：参数设置不合理。ClickHouse适合使用大batch、少数几个并发进行写入。多数情况下batch可以高达几万甚至几十万（取决于您的单行RowSize大小，一般按照每行100Byte进行评估，您需要根据实际数据特征进行估算）。

解决方案：并发数建议不超过10个。您可以调整不同参数进行尝试。

- 常见原因2：DataWorks独享资源组的ECS规格太小。比如独享资源的CPU、Memory太小，导致并发数、网络出口带宽受限；或者是batch设置太大而Memory太小，引起DataWorks进程Java GC等。

解决方案：您可以通过DataWorks的输出日志对ECS规格大小进行确认。

- 常见原因3：从数据源中读取慢。

解决方案：您可以在DataWorks输出日志中搜索totalWaitReaderTime、totalWaitWriterTime，如果发现totalWaitReaderTime明显大于totalWaitWriterTime，则表明主要耗时在读取端，而不是写入端。

- 常见原因4：使用了公网Endpoint。公网Endpoint的带宽非常有限，无法承载高性能的数据导入导出。

解决方案：您需要替换为VPC网络的Endpoint。

- 常见原因5：有脏数据。在没有脏数据的情况下，数据以batch方式写入。但是遇到了脏数据，正在写入的batch就会失败，并回退到逐行写入，生成大量的data part，大幅度降低了写入速度。

您可以参考如下两种方式判断是否有脏数据。

- 查看报错信息，如果返回信息包含 `Cannot parse`，则存在脏数据。

代码如下。

```
select written_rows, written_bytes, query_duration_ms, event_time, exception
from system.query_log
where event_time between '2021-11-22 22:00:00' and '2021-11-22 23:00:00' and lowerUTF8(
query) like '%insert into <table_name>%' and type != 'QueryStart' and exception_code !=
0
order by event_time desc limit 30;
```

- 查看batch行数，如果batch行数变为1，则存在脏数据。

代码如下。

```
select written_rows, written_bytes, query_duration_ms, event_time
from system.query_log
where event_time between '2021-11-22 22:00:00' and '2021-11-22 23:00:00' and lowerUTF8(
query) like '%insert into <table_name>%' and type != 'QueryStart'
order by event_time desc limit 30;
```

解决方案：您需要在数据源删除或修改脏数据。

为什么Hive导入后其数据行数跟ClickHouse对不上？

您可以通过以下手段进行排查。

1. 首先通过系统表query_log来查看导入的过程中是否有报错，如果有报错，那很有可能出现数据丢失的情况。
2. 确定使用的表引擎是否可以去重，比如使用ReplacingMergeTree，那很可能出现ClickHouse中的Count小于Hive中的情况。
3. 重新确认Hive中数据行数的正确性，很有可能出现源头的行数确定错误的情况。

为什么Kafka导入后其数据行数跟ClickHouse对不上？

您可以通过以下手段进行排查。

- 1.
2. 确定使用的表引擎是否可以去重，比如使用ReplacingMergeTree，那很可能出现ClickHouse中的Count小于Kafka中的情况。
3. 查看Kafka外表的配置是否有kafka_skip_broken_messages参数的配置，如果有该参数，那可能会跳过解析失败的Kafka消息，导致ClickHouse总的行数是小于Kafka中的。

如何使用Spark、Flink导入数据？

- 如何使用Spark导入数据请参见[从Spark导入](#)。
- 如何使用Flink导入数据请参见[从Flink SQL导入](#)。

如何从现有ClickHouse导入数据到云数据库ClickHouse？

您可以采取如下方案。

- 通过ClickHouse Client以导出文件的形式进行数据迁移，详情请参见[从自建ClickHouse迁移上云](#)。
- 通过Remote函数进行数据的迁移。

```
insert into <目的表> select * from remote('<连接串>', '<库>', '<表>', '<username>', '<passwd>');
```

使用MaterializeMySQL引擎同步MySQL数据时，为什么出现如下报错： The slave is connecting using CHANGE MASTER TO MASTER_AUTO_POSITION = 1, but the master has purged binary logs containing GTIDs that the slave requires ?

常见原因：MaterializeMySQL引擎停止同步的时间太久，导致MySQL Binlog日志过期被清理掉。

解决方案：删除报错的数据库，重新在云数据库ClickHouse中创建同步的数据库。

使用MaterializeMySQL引擎同步MySQL数据时，为什么出现表停止同步？为什么系统表system.materialize_mysql中 sync_failed_tables 字段不为空？

常见原因：同步过程中使用了云数据库ClickHouse不支持的MySQL DDL语句。

解决方案：重新同步MySQL数据，具体步骤如下。

1. 删除停止同步的表。

```
drop table <table_name> on cluster default;
```

 **说明** `table_name` 为停止同步的表名。如果停止同步的表有分布式表，那么本地表和分布式表都需要删除。

2. 重启同步进程。

```
alter database <database_name> on cluster default MODIFY SETTING skip_unsupported_tables = 1;
```

说明 <database_name> 为云数据库ClickHouse中同步的数据库。

如何处理insert into select XXX内存超限报错?

常见原因及解决方案如下。

- 常见原因1: 内存使用率比较高。
解决方案: 调整参数max_insert_threads, 减少可能的内存使用量。
- 常见原因2: 当前是通过 insert into select 把数据从一个ClickHouse集群导入到另外一个集群。
解决方案: 通过导入文件的方式来迁移数据, 更多信息请参见[从自建ClickHouse迁移上云](#)。

什么查询耗费的CPU、MEM高?

您可以在system.query_log系统表里自助查看CPU、MEM高期间的查询日志, 里面有每个查询的CPU消耗和内存消耗统计。更多信息请参见[system.query_log](#)。

如何处理查询时内存超出限制?

ClickHouse服务端对所有查询线程都配有memory tracker, 同一个查询下的所有线程tracker会汇报给一个memory tracker for query, 再上层还是memory tracker for total。您可以根据情况采取如下解决方案。

- 遇到 Memory limit (for query) 超限报错说明是查询内存占用过多(实例总内存的70%)导致失败, 这种情况下您需要垂直升配提高实例内存规模。
- 遇到 Memory limit (for total) 超限报错说明是实例总内存使用超限(实例总内存的90%), 这种情况下您可以尝试降低查询并发, 如果仍然不行则可能是后台异步任务占用了比较大的内存(常常是写入后主键合并任务), 您需要垂直升配提高实例内存规模。

如何处理查询报并发超限?

默认Server查询最大并发数为100, 您可以在控制台上进行修改。修改运行参数值具体操作步骤如下。

1. 登录[云数据库ClickHouse控制台](#)。
2. 在集群列表页面, 选择默认实例列表或云原生版本实例列表, 单击目标集群ID。
3. 在集群信息页面单击导航栏左侧的参数配置。
4. 修改max_concurrent_queries参数的运行参数值, 输入修改值并单击确定。

说明: 当前页面仅支持服务参数修改, 参考 [config.xml参数修改](#)。其他用户级参数说明和配置请参考 [user.xml参数修改](#) 只显示已修改的参数

参数名	参考值	运行参数值	是否重启	可修改参数值	参数描述
keep_alive_timeout	300	300 <input type="text"/>	是	>0	<input type="button" value="i"/>
max_connections	4096	4096 <input type="text"/>	是	>0	<input type="button" value="i"/>
max_partition_size_to_drop	0	0 <input type="text"/>	是	>=0	<input type="button" value="i"/>
max_concurrent_queries	100	100 <input type="text"/>	是	>0	<input type="button" value="i"/>
uncompressed_cache_size	1717986918	<input type="text"/>	是	>0	<input type="button" value="i"/>
timezone	Asia/Shanghai	<input type="text"/>	是	Legal time zone such as: Africa/Abidjan Asia/Shanghai Europe/Moscow	<input type="button" value="i"/>

输入范围: >0

5. 单击提交参数。
6. 单击确定。

在数据停止写入时，同一个查询语句每次查询的结果不一致，应该如何处理？

问题详细描述：通过 `select count(*)` 查询数据时只有整体数据的大概一半，或者数据一直在跳变。

您可以采取如下解决方案。

- 检查是否是多节点集群。多节点集群需要创建分布式表，往分布式表里写入数据并查询，每次查询结果一致。否则每次查询到不同分片的数据，结果不一致。如何创建分布式表请参见[创建分布式表](#)。
- 检查是否是多副本集群。多副本集群需要建Replicated系列表引擎的表，才能实现副本间数据同步。否则每次查到不同副本，结果不一致。如何创建Replicated系列表引擎的表请参见[表引擎](#)。

为什么有时看不到已经创建好的表并且查询结果一直抖动时多时少？

常见原因及解决方案如下。

- 常见原因1：建表流程存在问题。ClickHouse的分布式集群搭建并没有原生的分布式DDL语义。如果您在自建ClickHouse集群时使用 `create table` 创建表，查询虽然返回了成功，但实际这个表只在当前连接的Server上创建了。下次连接重置换一个Server，您就看不到这个表了。

解决方案：

- i. 建表时，请使用 `create table <table_name> on cluster default` 语句，`on cluster default` 声明会把这条语句广播给default集群的所有节点进行执行。示例代码如下。

```
Create table test on cluster default (a UInt64) Engine = MergeTree() order by tuple()  
;
```

- ii. 在test表上再创建一个分布式表引擎，建表语句如下。

```
Create table test_dis on cluster default as test Engine = Distributed(default, default,  
t, test, cityHash64(a));
```

- 常见原因2：ReplicatedMergeTree存储表配置有问题。ReplicatedMergeTree表引擎是对应MergeTree表引擎的主备同步增强版，在单副本实例上限定只能创建MergeTree表引擎，在双副本实例上只能创建ReplicatedMergeTree表引擎。

解决方案：在双副本实例上建表时，请使用 `ReplicatedMergeTree('/clickhouse/tables/{database}/{table}/{shard}', '{replica}')` 或 `ReplicatedMergeTree()` 配置ReplicatedMergeTree表引擎。其中，`ReplicatedMergeTree('/clickhouse/tables/{database}/{table}/{shard}', '{replica}')` 为固定配置，无需修改。

如何处理往表里写入时间戳数据后查询出来的结果与实际数据不同？

用 `SELECT timezone()` 语句，查看时区是否为当地时区，如果不是修改timezone配置项的值为当地时区。如何修改请参见[修改配置项运行参数值](#)。

如何处理建表后查询表不存在？

常见原因：DDL语句只在一个节点上执行。

解决方案：检查DDL语句是否有 `on cluster` 关键字。更多信息，请参见[建表语法](#)。

为什么Kafka外表建表后数据不增加？

您可以先对Kafka外表进行 `select * from` 的查询，如果查询报错，那可以根据报错信息确定原因（一般是数据解析失败）。如果查询正常返回结果，那需要进一步查看目的表（Kafka外表的具体存储表）和Kafka源表（Kafka外表）的字段是否匹配。如果数据写入失败，那说明字段是匹配不上的。示例语句如下。

```
insert into <目的表> as select * from <kafka外表>;
```

为什么客户端看到的时间结果和时区显示的不一样？

客户端设置了 `use_client_time_zone`，并设定在了错误时区上。

为什么数据写入后不可见？

一般原因是分布式表和本地表的表结构不一致造成的。您可以通过查询系统表 `system.distribution_queue` 来查看写入分布式表的时候是否发生错误。

为什么optimize任务很慢？

optimize任务非常占用CPU和磁盘吞吐，查询和optimize任务都会相互影响，在机器节点负载压力较大的时候就会表现出optimize很慢问题，目前没有特殊优化方法。

为什么optimize后数据仍未主键合并？

首先为了让数据有正确的主键合并逻辑，需要保证以下两个前提条件。

- 存储表里的partition by定义字段必须是包含在 `order by` 里的，不同分区的数据不会主键合并。
- 分布式表里定义的Hash算法字段必须是包含在 `order by` 里的，不同节点的数据不会主键合并。

optimize常用命令及相关说明如下。

命令	说明
<code>optimize table test;</code>	尝试选取MergeTree的data parts进行合并，有可能没有执行任务就返回。执行了也并不保证全表的记录都完成了主键合并，一般不会使用。
<code>optimize table test partition tuple();</code>	指定某个分区，选取分区中所有的data parts进行合并，有可能没有执行任务就返回。任务执行后代表某个分区下的数据都合并到了同一个data part，单分区下已经完成主键合并。但是在任务执行期间写入的数据不会参与合并，若是分区下只有一个data part也不会重复执行任务。 说明 对于没有分区键的表，其默认分区就是 <code>partition tuple()</code> 。
<code>optimize table test final;</code>	对全表所有分区强制进行合并，即使分区下只有一个data part也会进行重新合并，可以用于强制移除TTL过期的记录。任务执行代价最高，但也有可能没有执行合并任务就返回。

对于上面三种命令，您可以设置参数 `optimize_throw_if_noop` 通过异常报错感知是否执行任务。

为什么optimize后数据TTL仍未生效？

常见原因及解决方案如下。

- 常见原因1：数据的TTL淘汰是在主键合并阶段执行的，如果data part迟迟没有进行主键合并，那过期的数据就无法淘汰。

解决方案：

- 您可以通过手动 `optimize final` 或者 `optimize` 指定分区 的方式触发合并任务。
- 您可以在建表时设置`merge_with_ttl_timeout`、`ttl_only_drop_parts`等参数，提高含有过期数据data parts的合并频率。

- 常见原因2：表的TTL经过修改或者添加，存量的data part里缺少TTL信息或者不正确，这样也可能导致过期数据淘汰不掉。

解决方案：

- 您可以通过 `alter table materialize ttl` 命令重新生成TTL信息。
- 您可以通过 `optimize` 分区 更新TTL信息。

为什么optimize后更新删除操作没有生效？

云数据库ClickHouse中的更新删除都是异步执行的，目前没有机制可以干预其进度。您可以通过system.mutations系统表查看进度。

如何进行DDL增加列、删除列、修改列操作？

本地表的修改直接执行即可。如果要对分布式表进行修改，需分如下情况进行。

- 如果没有数据写入，您可以先修改本地表，然后修改分布式表。
- 如果数据正在写入，您需要区分不同的类型进行操作。

类型	操作步骤
增加Nullable的列	i. 修改本地表。 ii. 修改分布式表。
修改列的数据类型（类型可以相互转换）	
删除Nullable列	i. 修改分布式表。 ii. 修改本地表。
增加非Nullable的列	i. 停止数据的写入。 ii. 执行SYSTEM FLUSH DISTRIBUTED分布式表。 iii. 修改本地表。 iv. 修改分布式表。 v. 重新进行数据的写入。
删除非Nullable的列	
修改列的名称	

为什么DDL执行慢，经常卡住？

常见原因：DDL全局的执行是串行执行，复杂查询会导致死锁。

您可以采取如下解决方案。

- 等待运行结束。
- 在控制台尝试终止查询。
- 在云数据库ClickHouse控制台的参数配置页面，对任一参数的参数值进行编辑不修改原来的值，单击提交参数。

 说明 如何进行参数值修改请参见[修改参数运行值](#)。

- 如果上述方法仍然无法解决，请提交工单。

如何处理分布式DDL报错：longer than distributed_ddl_task_timeout (=xxx) seconds?

您可以通过使用 `set global on cluster default distributed_ddl_task_timeout=xxx` 命令修改默认超时时间，xxx为自定义超时时间，单位为秒。全局参数修改请参见[集群参数修改](#)。

如何处理语法报错：set global on cluster default?

常见原因及解决方案如下。

- 常见原因1：ClickHouse客户端会进行语法解析，而 `set global on cluster default` 是服务端增加的语法。在客户端尚未更新到与服务端对齐的版本时，该语法会被客户端拦截。

解决方案：

- 使用JDBC Driver等不会在客户端解析语法的工具，比如DataGrip、DBEaver。
- 编写JDBC程序来执行该语句。
- 常见原因2： `set global on cluster default key = value;` 中value是字符串，但是漏写了引号。

解决方案：在字符串类型的value两侧加上引号。

有什么BI工具推荐?

Quick BI。

有什么数据查询IDE工具推荐?

DataGrip、DBEaver。

如何查看每张表所占的磁盘空间?

您可以通过如下代码查看每张表所占的磁盘空间。

```
SELECT table, formatReadableSize(sum(bytes)) as size, min(min_date) as min_date, max(max_date) as max_date FROM system.parts WHERE active GROUP BY table;
```

如何查看冷数据大小?

示例代码如下。

```
select * from system.disks;
```

如何查询哪些数据在冷存上？

示例代码如下。

```
select * from system.parts where disk_name = 'cold_disk';
```

如何移动分区数据到冷存？

示例代码如下。

```
ALTER TABLE table_name MOVE PARTITION partition_expr TO DISK 'cold_disk';
```

为什么监控中存在数据中断情况？

常见原因如下。

- 查询触发OOM。
- 修改配置触发重启。
- 升降配后的实例重启。

20.8后的版本是否支持平滑升级，不需要迁移数据？

支持。

常用系统表有哪些？

常用系统表及作用如下。

名称	作用
system.processes	查询正在执行的SQL。
system.query_log	查询历史执行过的SQL。
system.merges	查询集群上的merge信息。
system.mutations	查询集群上的mutation信息。

如何修改系统级别的参数？是否要重启，有什么影响？

系统级别的参数对应config.xml内的部分配置项，具体修改步骤如下。

1. 登录[云数据库ClickHouse控制台](#)。
2. 在[集群列表](#)页面，选择默认实例列表或云原生版本实例列表，单击目标集群ID。
3. 在集群信息页面单击导航栏左侧的[参数配置](#)。
4. 修改max_concurrent_queries参数的运行参数值，输入修改值并单击确定。

说明：当前页面仅支持服务参数修改，参考 [config.xml参数修改](#)。其他用户级参数说明和配置请参考 [user.xml参数修改](#) 只显示已修改的参数

参数名	参考值	运行参数值	是否重启	可修改参数值	参数描述
keep_alive_timeout	300	300 <input type="checkbox"/>	是	>0	<input type="button" value="i"/>
max_connections	4096	4096 <input type="checkbox"/>	是	>0	<input type="button" value="i"/>
max_partition_size_to_drop	0	0 <input type="checkbox"/>	是	>=0	<input type="button" value="i"/>
max_concurrent_queries	100	100 <input type="checkbox"/>	是	>0	<input type="button" value="i"/>
uncompressed_cache_size	1717986918	<input type="text"/>	是	>0	<input type="button" value="i"/>
timezone	Asia/Shanghai	<input type="text"/>	是	Legal time zone such as: Africa/Abidjan Asia/Shanghai Europe/Moscow	<input type="button" value="i"/>

输入范围: >0

5. 单击提交参数。

6. 单击确定。

单击确定后，自动重启clickhouse-server，重启会造成约1min闪断。

如何修改用户级别的参数？

用户级别的参数对应users.xml内的部分配置项，你需要执行如下示例语句。

```
set global on cluster default ${key}=${value};
```

无特殊说明的参数执行成功后即可生效。

如何修改Quota？

您可以在执行语句的settings里增加，示例代码如下。

```
settings max_memory_usage = XXX;
```

8.2. 如何处理ClickHouse超时问题

本文介绍如何设置云数据库ClickHouse超时相关的参数以及遇到超时问题时如何查询后台状态。

背景信息

云数据库ClickHouse内核中有很多超时相关的参数设置，并且提供了多种协议进行交互，本文主要介绍如何设置HTTP协议和TCP协议的相关参数。

HTTP协议

HTTP协议是云数据库ClickHouse在生产环境中最常用的交互方式，包括官方提供的jdbc driver、阿里云DMS、DataGrip，后台使用的都是HTTP协议。HTTP协议常用的端口号为8123。

如何处理distributed_ddl_task_timeout超时问题

分布式DDL查询（带有 on cluster clause）的执行等待时间，系统默认是180s。您可以在DMS上执行以下命令来设置全局参数，设置后需要重启集群。

```
set global on cluster default distributed_ddl_task_timeout = 1800;
```

由于分布式DDL是基于zookeeper构建任务队列异步执行，执行等待超时并不代表查询失败，只表示之前发送还在排队等待执行，用户不需要重复发送任务。

如何处理max_execution_time超时问题

一般查询的执行超时时间，DMS平台上默认设置是7200s，jdbc driver、DataGrip上默认是30s。超时限制触发之后查询会自动取消。用户可以进行查询级别更改，例如 `select * from system.numbers settings max_execution_time = 3600`，也可以在DMS上执行以下命令来设置全局参数。

```
set global on cluster default max_execution_time = 3600;
```

如何处理socket_timeout超时问题

HTTP协议在监听socket返回结果时的等待时间，DMS平台上默认设置是7200s，jdbc driver、DataGrip上默认是30s。该参数不是Clickhouse系统内的参数，它属于jdbc在HTTP协议上的参数，但它是会影响前面的max_execution_time参数设置效果，因为它决定了客户端在等待结果返回上的时间限制。所以一般用户在调整max_execution_time参数的时候也需要配套调整socket_timeout参数，略微高于max_execution_time即可。用户设置参数时需要在jdbc链接串上添加socket_timeout这个property，单位是毫秒，例如：`'jdbc:clickhouse://127.0.0.1:8123/default?socket_timeout=3600000'`。

如何处理使用SLB链接时的异常断链

阿里云上的SLB链接在长时间内没有数据报文发送时会自动断开链接，客户端收到“read timeout”错误，查询无法追踪。对于这种情况您可以在DMS上执行以下命令来设置全局参数，设置后需要重启集群。

```
set global on cluster default send_progress_in_http_headers = 1;  
set global on cluster default http_headers_progress_interval_ms = 60000;
```

对于20.8最新版本您需要使用如下命令，设置失败时可以先申请小版本升版。

```
set global on cluster default http_server_enable_tcp_keep_alive = 1;  
set global on cluster default tcp_keep_alive_timeout = 60;
```

开启send_progress_in_http_headers后，ClickHouse服务端会不断发送包含查询进度的HTTP-Header报文给客户端，这样就会一直有数据报文流通，避免链接断开。

因为各种原因客户端失联后，HTTP协议发送的查询仍然会继续执行。用户可以在系统表追踪到查询是否成功执行。

- 查询集群中所有实例当前正在running的查询：

```
select * from remote(default, system, processes) where query like 'XXX'
```

- 查询当天的历史查询结果，包括是否成功、已经失败的错误信息：

```
select * from remote(default, system, query_log) where event_date = toDate(now()) and  
query like 'XXX'
```

使用ClickHouse服务端IP直接链接时的Client异常hang住

阿里云上的ECS在跨安全组链接时，有可能陷入静默链接错误。具体原因是jdbc客户端所在ECS机器的安全组白名单并没有开放给ClickHouse服务端机器。当客户端的请求经过超长时间才得到查询结果时，返回的报文可能因为路由表不通无法发送到客户端。此时客户端就陷入了异常hang住状态。

该问题的处理办法和SLB链接异常断链问题一样，开启`send_progress_in_http_headers`可以解决大部分问题。在极少数情况下，开启`send_progress_in_http_headers`仍不能解决问题的，您可以尝试配置jdbc客户端所在ECS机器的安全组白名单，把ClickHouse服务端地址加入到白名单中。

TCP协议

TCP协议最常使用的场景是ClickHouse自带的命令行工具进行交互分析时，社区兼容版集群常见端口号为3306，云原生版集群常见端口号为9000。因为TCP协议里有链接定时探活报文，所以它不会出现socket层面的超时问题。您只需关注`distributed_ddl_task_timeout`和`max_execution_time`参数的超时，设置方法和HTTP协议一致。

9. 性能测试指南

本文以On Time和Star Schema数据集为例，介绍如何将测试数据集导入云数据库ClickHouse并完成性能测试。

准备工作

- 已创建云数据库ClickHouse集群。如何创建，请参见[新建集群](#)。
- 已创建数据库账号。如何创建，请参见[创建账号](#)。
- 已准备一台Linux系统的机器，并将Linux机器的IP地址添加到云数据库ClickHouse集群的白名单中。如何添加，请参见[设置白名单](#)。
- 已安装与云数据库ClickHouse集群版本对应的clickhouse-client工具。下载链接，请参见[clickhouse-client](#)。

测试说明

本测试中的所有clickhouse-client命令均在clickhouse-client客户端的安装目录下执行。

本测试中使用的On Time和Star Schema数据集均引用自ClickHouse开源网站。更多信息，请参见[示例数据集](#)。

On Time数据集测试

1. 连接集群

在Linux环境中使用clickhouse-client工具连接云数据库ClickHouse集群。

```
./clickhouse-client --host=<数据库连接地址> --port=<TCP端口号> --user=<数据库账号> --password=<数据库账号的密码>
```

2. 建表

请根据云数据库ClickHouse集群的规格下载正确的建表语句脚本并执行。

- [单副本版集群建表语句](#)
- [双副本版集群建表语句](#)

3. 下载On Time数据

🔍 说明

On Time数据集覆盖的时间区间非常大，下载数据时间较长，同时数据下载过程中会遇到一部分数据已丢失的错误。

```
echo https://transtats.bts.gov/PREZIP/On_Time_Reporting_Carrier_On_Time_Performance_1987_pr  
esent_{1987..2021}_{1..12}.zip | xargs -P10 wget --no-check-certificate --continue
```

4. 导入数据至云数据库ClickHouse

请根据如下Shell命令，正确配置ck_url、ck_user、ck_pass和ck_port环境变量，再进行数据导入。

```
export ck_url=<数据库连接地址>
export ck_user=<数据库账号>
export ck_pass=<数据库账号的密码>
export ck_port=<TCP端口号>
for i in *.zip; do echo $i; unzip -cq $i '*.csv' | sed 's/\.00//g' | ./clickhouse-client -h
<数据库连接地址> -u <数据库账号> --password <数据库账号的密码> --port <TCP端口号> --query="INSERT
INTO ontime FORMAT CSVWithNames"; done
```

5. 测试集

🔍 说明

测试集引用自ClickHouse开源网站。更多信息，请参见[On Time数据集](#)。

Q0

```
SELECT avg(c1)
FROM
(
  SELECT Year, Month, count(*) AS c1
  FROM ontime
  GROUP BY Year, Month
);
```

Q1

```
SELECT DayOfWeek, count(*) AS c
FROM ontime
WHERE Year>=2000 AND Year<=2008
GROUP BY DayOfWeek
ORDER BY c DESC;
```

Q2

```
SELECT DayOfWeek, count(*) AS c
FROM ontime
WHERE DepDelay>10 AND Year>=2000 AND Year<=2008
GROUP BY DayOfWeek
ORDER BY c DESC;
```

Q3

```
SELECT Origin, count(*) AS c
FROM ontime
WHERE DepDelay>10 AND Year>=2000 AND Year<=2008
GROUP BY Origin
ORDER BY c DESC
LIMIT 10;
```

Q4

```
SELECT Carrier, count(*)
FROM ontime
WHERE DepDelay>10 AND Year=2007
GROUP BY Carrier
ORDER BY count(*) DESC;
```

Q5

```
set joined_subquery_requires_alias=0;
set any_join_distinct_right_table_keys=1;
SELECT Carrier, c, c2, c*100/c2 as c3
FROM
(
  SELECT
    Carrier,
    count(*) AS c
  FROM ontime
  WHERE DepDelay>10
    AND Year=2007
  GROUP BY Carrier
)
ANY INNER JOIN
(
  SELECT
    Carrier,
    count(*) AS c2
  FROM ontime
  WHERE Year=2007
  GROUP BY Carrier
) USING Carrier
ORDER BY c3 DESC;
```

Q6

```
set joined_subquery_requires_alias=0;
SELECT Carrier, c, c2, c*100/c2 as c3
FROM
(
  SELECT
    Carrier,
    count(*) AS c
  FROM ontime
  WHERE DepDelay>10
    AND Year>=2000 AND Year<=2008
  GROUP BY Carrier
)
ANY INNER JOIN
(
  SELECT
    Carrier,
    count(*) AS c2
  FROM ontime
  WHERE Year>=2000 AND Year<=2008
  GROUP BY Carrier
) USING Carrier
ORDER BY c3 DESC;
```

Q7

```
set joined_subquery_requires_alias=0;
SELECT Year, c1/c2
FROM
(
  select
    Year,
    count(*)*100 as c1
  from ontime
  WHERE DepDelay>10
  GROUP BY Year
)
ANY INNER JOIN
(
  select
    Year,
    count(*) as c2
  from ontime
  GROUP BY Year
) USING (Year)
ORDER BY Year;
```

Q8

```
SELECT DestCityName, uniqExact(OriginCityName) AS u
FROM ontime
WHERE Year>=2000 and Year<=2010
GROUP BY DestCityName
ORDER BY u DESC
LIMIT 10;
```

Q9

```
SELECT Year, count(*) AS c1
FROM ontime
GROUP BY Year;
```

Q10

```
SELECT
  min(Year), max(Year), Carrier, count(*) AS cnt,
  sum(ArrDelayMinutes>30) AS flights_delayed,
  round(sum(ArrDelayMinutes>30)/count(*),2) AS rate
FROM ontime
WHERE
  DayOfWeek NOT IN (6,7) AND OriginState NOT IN ('AK', 'HI', 'PR', 'VI')
  AND DestState NOT IN ('AK', 'HI', 'PR', 'VI')
  AND FlightDate < '2010-01-01'
GROUP by Carrier
HAVING cnt>100000 and max(Year)>1990
ORDER by rate DESC
LIMIT 1000;
```

Star Schema数据集测试

1. 连接集群

在Linux环境中使用clickhouse-client工具连接云数据库ClickHouse集群。

```
./clickhouse-client --host=<数据库连接地址> --port=<TCP端口号> --user=<数据库账号> --password=<数据库账号的密码>
```

2. 建表

请根据云数据库ClickHouse集群的规格下载正确的建表语句脚本并执行。

- [单副本版集群建表语句](#)
- [双副本版集群建表语句](#)

3. 生成数据文件

请根据如下Shell命令，首先克隆数据生成项目ssb-dbgen，然后通过make编译，用命令生成数据文件customer.tbl、lineorder.tbl、part.tbl和supplier.tbl。

说明

示例中 `-s 100` 参数会产生大概6亿行的数据（67 GB），您可以根据情况适当调整测试数据量。

```
git clone https://github.com/vadimtk/ssb-dbgen.git
cd ssb-dbgen
make
./dbgen -s 100 -T c
./dbgen -s 100 -T l
./dbgen -s 100 -T p
./dbgen -s 100 -T s
```

4. 导入数据至云数据库ClickHouse

请根据如下Shell命令，正确配置ck_url、ck_user、ck_pass和ck_port环境变量，再进行数据导入。

```
export ck_url=<数据库连接地址>
export ck_user=<数据库账号>
export ck_pass=<数据库账号的密码>
export ck_port=<TCP端口号>
./clickhouse-client -h <数据库连接地址> -u <数据库账号> --password <数据库账号的密码> --port <TCP端口号> --query "INSERT INTO customer FORMAT CSV" < ssb-dbgen/customer.tbl
./clickhouse-client -h <数据库连接地址> -u <数据库账号> --password <数据库账号的密码> --port <TCP端口号> --query "INSERT INTO part FORMAT CSV" < ssb-dbgen/part.tbl
./clickhouse-client -h <数据库连接地址> -u <数据库账号> --password <数据库账号的密码> --port <TCP端口号> --query "INSERT INTO supplier FORMAT CSV" < ssb-dbgen/supplier.tbl
./clickhouse-client -h <数据库连接地址> -u <数据库账号> --password <数据库账号的密码> --port <TCP端口号> --query "INSERT INTO lineorder FORMAT CSV" < ssb-dbgen/lineorder.tbl
```

5. 测试集

说明

测试集引用自ClickHouse开源网站。更多信息，请参见[Star Schema数据集](#)。

Q0

```
INSERT INTO lineorder_flat
SELECT
  l.LO_ORDERKEY AS LO_ORDERKEY,
  l.LO_LINENUMBER AS LO_LINENUMBER,
  l.LO_CUSTKEY AS LO_CUSTKEY,
  l.LO_PARTKEY AS LO_PARTKEY,
  l.LO_SUPPKEY AS LO_SUPPKEY,
  l.LO_ORDERDATE AS LO_ORDERDATE,
  l.LO_ORDERPRIORITY AS LO_ORDERPRIORITY,
  l.LO_SHIPPRIORITY AS LO_SHIPPRIORITY,
  l.LO_QUANTITY AS LO_QUANTITY,
  l.LO_EXTENDEDPRICE AS LO_EXTENDEDPRICE,
  l.LO_ORDTOTALPRICE AS LO_ORDTOTALPRICE,
  l.LO_DISCOUNT AS LO_DISCOUNT,
  l.LO_REVENUE AS LO_REVENUE,
  l.LO_SUPPLYCOST AS LO_SUPPLYCOST,
  l.LO_TAX AS LO_TAX,
  l.LO_COMMITDATE AS LO_COMMITDATE,
  l.LO_SHIPMODE AS LO_SHIPMODE,
  c.C_NAME AS C_NAME,
  c.C_ADDRESS AS C_ADDRESS,
  c.C_CITY AS C_CITY,
  c.C_NATION AS C_NATION,
  c.C_REGION AS C_REGION,
  c.C_PHONE AS C_PHONE,
  c.C_MKTSEGMENT AS C_MKTSEGMENT,
  s.S_NAME AS S_NAME,
  s.S_ADDRESS AS S_ADDRESS,
  s.S_CITY AS S_CITY,
  s.S_NATION AS S_NATION,
  s.S_REGION AS S_REGION,
  s.S_PHONE AS S_PHONE,
  p.P_NAME AS P_NAME,
  p.P_MFGR AS P_MFGR,
  p.P_CATEGORY AS P_CATEGORY,
  p.P_BRAND AS P_BRAND,
  p.P_COLOR AS P_COLOR,
  p.P_TYPE AS P_TYPE,
  p.P_SIZE AS P_SIZE,
  p.P_CONTAINER AS P_CONTAINER
FROM lineorder AS l
INNER JOIN customer AS c ON c.C_CUSTKEY = l.LO_CUSTKEY
INNER JOIN supplier AS s ON s.S_SUPPKEY = l.LO_SUPPKEY
INNER JOIN part AS p ON p.P_PARTKEY = l.LO_PARTKEY;
```

Q1

```
SELECT sum(LO_EXTENDEDPRICE * LO_DISCOUNT) AS revenue
FROM lineorder_flat
WHERE toYear(LO_ORDERDATE) = 1993 AND LO_DISCOUNT BETWEEN 1 AND 3 AND LO_QUANTITY < 25;
```

Q2

```
SELECT sum(LO_EXTENDEDPRICE * LO_DISCOUNT) AS revenue
FROM lineorder_flat
WHERE toYYYYMM(LO_ORDERDATE) = 199401 AND LO_DISCOUNT BETWEEN 4 AND 6 AND LO_QUANTITY BETWEEN 26 AND 35;
```

Q3

```
SELECT sum(LO_EXTENDEDPRICE * LO_DISCOUNT) AS revenue
FROM lineorder_flat
WHERE toISOWeek(LO_ORDERDATE) = 6 AND toYear(LO_ORDERDATE) = 1994
AND LO_DISCOUNT BETWEEN 5 AND 7 AND LO_QUANTITY BETWEEN 26 AND 35;
```

Q4

```
SELECT
    sum(LO_REVENUE),
    toYear(LO_ORDERDATE) AS year,
    P_BRAND
FROM lineorder_flat
WHERE P_CATEGORY = 'MFGR#12' AND S_REGION = 'AMERICA'
GROUP BY
    year,
    P_BRAND
ORDER BY
    year,
    P_BRAND;
```

Q5

```
SELECT
    sum(LO_REVENUE),
    toYear(LO_ORDERDATE) AS year,
    P_BRAND
FROM lineorder_flat
WHERE P_BRAND >= 'MFGR#2221' AND P_BRAND <= 'MFGR#2228' AND S_REGION = 'ASIA'
GROUP BY
    year,
    P_BRAND
ORDER BY
    year,
    P_BRAND;
```

Q6

```
SELECT
    sum(LO_REVENUE),
    toYear(LO_ORDERDATE) AS year,
    P_BRAND
FROM lineorder_flat
WHERE P_BRAND = 'MFGR#2239' AND S_REGION = 'EUROPE'
GROUP BY
    year,
    P_BRAND
ORDER BY
    year,
    P_BRAND;
```

Q7

```
SELECT
    C_NATION,
    S_NATION,
    toYear(LO_ORDERDATE) AS year,
    sum(LO_REVENUE) AS revenue
FROM lineorder_flat
WHERE C_REGION = 'ASIA' AND S_REGION = 'ASIA' AND year >= 1992 AND year <= 1997
GROUP BY
    C_NATION,
    S_NATION,
    year
ORDER BY
    year ASC,
    revenue DESC;
```

Q8

```
SELECT
    C_CITY,
    S_CITY,
    toYear(LO_ORDERDATE) AS year,
    sum(LO_REVENUE) AS revenue
FROM lineorder_flat
WHERE C_NATION = 'UNITED STATES' AND S_NATION = 'UNITED STATES' AND year >= 1992 AND year <
= 1997
GROUP BY
    C_CITY,
    S_CITY,
    year
ORDER BY
    year ASC,
    revenue DESC;
```

Q9

```
SELECT
  C_CITY,
  S_CITY,
  toYear(LO_ORDERDATE) AS year,
  sum(LO_REVENUE) AS revenue
FROM lineorder_flat
WHERE (C_CITY = 'UNITED KI1' OR C_CITY = 'UNITED KI5') AND (S_CITY = 'UNITED KI1' OR S_CITY
= 'UNITED KI5') AND year >= 1992 AND year <= 1997
GROUP BY
  C_CITY,
  S_CITY,
  year
ORDER BY
  year ASC,
  revenue DESC;
```

Q10

```
SELECT
  C_CITY,
  S_CITY,
  toYear(LO_ORDERDATE) AS year,
  sum(LO_REVENUE) AS revenue
FROM lineorder_flat
WHERE (C_CITY = 'UNITED KI1' OR C_CITY = 'UNITED KI5') AND (S_CITY = 'UNITED KI1' OR S_CITY
= 'UNITED KI5') AND toYYYYMM(LO_ORDERDATE) = 199712
GROUP BY
  C_CITY,
  S_CITY,
  year
ORDER BY
  year ASC,
  revenue DESC;
```

Q11

```
SELECT
  toYear(LO_ORDERDATE) AS year,
  C_NATION,
  sum(LO_REVENUE - LO_SUPPLYCOST) AS profit
FROM lineorder_flat
WHERE C_REGION = 'AMERICA' AND S_REGION = 'AMERICA' AND (P_MFGR = 'MFGR#1' OR P_MFGR = 'MFG
R#2')
GROUP BY
  year,
  C_NATION
ORDER BY
  year ASC,
  C_NATION ASC;
```

Q12

```
SELECT
    toYear(LO_ORDERDATE) AS year,
    S_NATION,
    P_CATEGORY,
    sum(LO_REVENUE - LO_SUPPLYCOST) AS profit
FROM lineorder_flat
WHERE C_REGION = 'AMERICA' AND S_REGION = 'AMERICA' AND (year = 1997 OR year = 1998) AND (P
_MFGR = 'MFGR#1' OR P_MFGR = 'MFGR#2')
GROUP BY
    year,
    S_NATION,
    P_CATEGORY
ORDER BY
    year ASC,
    S_NATION ASC,
    P_CATEGORY ASC;
```

Q13

```
SELECT
    toYear(LO_ORDERDATE) AS year,
    S_CITY,
    P_BRAND,
    sum(LO_REVENUE - LO_SUPPLYCOST) AS profit
FROM lineorder_flat
WHERE S_NATION = 'UNITED STATES' AND (year = 1997 OR year = 1998) AND P_CATEGORY = 'MFGR#14
'
GROUP BY
    year,
    S_CITY,
    P_BRAND
ORDER BY
    year ASC,
    S_CITY ASC,
    P_BRAND ASC;
```

10. 管理规范

10.1. 安全白皮书

法律声明

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

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

安全隔离

云数据库ClickHouse中所有计算是在受限的沙箱(多层次的应用沙箱)中运行的,从KVM级到Kernel级。系统沙箱配合鉴权管理机制,用来保证数据的安全,以避免出现内部人员恶意或粗心造成服务器故障。

网络隔离

云数据库ClickHouse只支持VPC网络,集群中所有的节点都运行在同一VPC网络内部,默认只提供VPC内网访问;用户可以手动开通外网访问链接;只有配置在访问IP白名单中的客户端才能访问云数据库ClickHouse。

鉴权认证

身份认证:云数据库ClickHouse管理员创建数据库账户和密码,通过数据库账号密码登录验证识别身份。

权限控制:云数据库ClickHouse提供集群级别的访问权限控制。

数据安全

云数据库ClickHouse集群数据存储阿里云ESSD云盘、SSD云盘或高效云盘中，数据采用3备份方式存储，保证数据可靠性和一致性。

日志审计

云数据库ClickHouse提供用户行为日志审计功能，包括运行记录、安全信息等内容。

最佳实践

1.VPC网络

为了保障安全，建议使用VPC网络访问集群。购买一台ECS服务器作为客户端，连接ClickHouse数据库，作为写入和查询的客户端服务器。

2.访问客户端白名单

无论通过VPC网络还是外网网络访问ClickHouse数据库，都需要在控制台配置客户端IP白名单，否则会被拒绝访问。如果配置了白名单，依然连接不上，注意检查客户端IP是否正确。

为了降低数据泄露风险，建议白名单开放范围越小越好，只对特定生产网段开放，不允许设置为0.0.0.0/0。

3.访问日志导出

支持通过命令行工具，将访问日志导出，命令如下：
`SELECT * FROM system.query_log INTO OUTFILE access.log`

11. 相关协议

11.1. 云数据库服务协议

本产品最新版服务协议，请在阿里云[服务协议](#)汇总页查找获取。

11.2. 服务等级协议

本产品最新版服务协议，请在阿里云[服务协议](#)汇总页查找获取。