

数据传输服务

数据同步

产品文档



腾讯云

【版权声明】

©2013-2024 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

文档目录

数据同步

数据同步支持的数据库

云数据库跨账号实例间同步

同步至 MySQL 系列

同步至 MySQL

MySQL/MariaDB/Percona 同步至 MySQL

MySQL 数据同步至 MySQL

TDSQL-C MySQL 同步至 MySQL

同步至 MariaDB

MySQL/MariaDB/Percona/TDSQL-C MySQL 同步至 MariaDB

TDSQL MySQL 同步至 MariaDB

同步至 TDSQL-C MySQL

TDSQL-C MySQL 同步至 TDSQL-C MySQL

MariaDB 同步至 TDSQL-C MySQL

TDSQL MySQL 同步至 TDSQL-C MySQL

MySQL 同步至 TDSQL-C MySQL

同步至 TDSQL MySQL

TDSQL MySQL 同步至 TDSQL MySQL

支持能力

使用说明

同步操作指导

MySQL/MariaDB/Percona 同步至 TDSQL MySQL

MySQL/MariaDB/Percona 同步至 TDSQL TDSStore

同步至 TDSQL TDSStore

MySQL/MariaDB/Percona/TDSQL-C MySQL 同步至 TDSQL TDSStore

MySQL 系列同步高级操作

同步高级对象

创建数据一致性校验

设置 SQL 过滤策略

库表重命名

同步 Online DDL 临时表

列过滤

MySQL 系列前置校验不通过处理

连接 DB 检查

周边检查

版本检查
源实例权限检查
部分实例参数检查
目标实例内容冲突检查
目标实例空间检查
Binlog 参数检查
外键依赖检查
视图检查
高级对象检查
警告项检查
TDSQL Proxy 检查
TDSQL 内核检查
同库表对象 DDL 环形链路同步检测
同目标同库表对象 DDL 同步冲突检测

同步至 PostgreSQL

PostgreSQL 同步至 PostgreSQL

PostgreSQL 前置校验不通过处理

连接 DB 检查
对象依赖检查
版本检查
源实例或目标实例权限检查
增量迁移预检查
结构兼容性检查
结构冲突检查
账号冲突检查
参数配置冲突检查
插件兼容性检查
主键依赖检查
实例关键参数检查
多任务冲突检测

同步至 Kafka

数据同步至 Kafka 与数据订阅的区别

数据投递到 Kafka 策略设置

MySQL/MariaDB/Percona/TDSQL-C MySQL 数据库同步至 Kafka

支持能力
数据同步操作指导
数据消费操作指导
Demo 说明

Avro Demo 说明

JSON Demo 说明

Canal Demo 说明（Canal ProtoBuf/Canal JSON）

TDSQL MySQL 同步至 Kafka

支持能力

使用说明

数据同步操作指导

数据消费操作指导

任务管理

任务状态说明

查看任务

配置任务

校验任务

启动任务

重试任务（MySQL 系列）

限速

结束任务

修改同步配置

暂停任务

恢复任务

重置任务

删除任务

变更配置

按量计费转包年包月

销毁/退货实例

续费任务

解隔离任务

下线实例

创建类似任务

查看日志

释放任务

数据同步

数据同步支持的数据库

最近更新时间：2024-04-01 15:40:29

场景概述

数据同步功能指两个数据源之间的数据实时同步，数据同步属于持续性任务，任务创建后会一直同步数据（几乎实时同步），保持源库和目标库的数据一致性。

数据同步和**数据迁移**的基本功能类似，但**数据同步**可支持在源和目标端同时进行数据写入，增加了数据冲突处理、任务配置动态调整、构建复杂拓扑结构等功能，更方便用户对长期任务的管理。

腾讯云 DTS 支持源端/目标端为自建数据库、云数据库、第三方云厂商数据库的同步，典型场景示例如下。

场景	说明
云上云下同步	将本地 IDC 自建数据库、CVM 自建数据库等，同步到腾讯云数据库实例，并且支持建立反向同步，实现云上云下双向同步。
多云厂商间同步	将第三方云厂商数据库同步到腾讯云数据库实例，实现多云同步。
腾讯云数据库实例间同步	跨地域、跨境数据库同步，以及不同腾讯云账号下的数据库实例同步等。
IDC 数据库之间同步	本地 IDC 自建数据库之间的相互同步，实现数据打通和汇聚。

功能说明

同步数据类型

支持结构同步、全量同步和增量同步。

结构同步：同步任务配置中，**初始化类型**选择“结构初始化”。

全量同步：同步任务配置中，**初始化类型**选择“结构初始化+全量初始化”。

增量同步：同步任务配置中，**初始化类型**都不勾选，然后在**同步操作类型**中选择 DML 和 DDL。

接入方式

DTS 支持的接入方式有“公网/云主机自建/专线接入/VPN 接入/云数据库/云联网/私有网络 VPC”，数据库的部署形态不同，可以选择的接入方式也不同。每种接入方式需要具备相应网络条件，请参考 [网络准备概述](#)。

IDC 自建数据库/其他云厂商数据库：接入方式可选择“公网/专线接入/VPN 接入/云联网”。

CVM 上的自建数据库：接入方式选择“云主机自建”。

腾讯云数据库实例：接入方式选择“云数据库”。

拓扑结构

数据同步支持复杂拓扑结构，如一对多同步、多对一同步、双向同步、环形同步等，如需构建复杂拓扑，请参考对应配置指导，[构建双向同步数据结构](#)，[构建多对一同步数据结构](#)，[构建多活数据中心](#)。

同步至 MySQL

说明：

当前如果用户需要使用如下链路同步，需 [提交工单](#) 进行申请。

MySQL/MariaDB/Percona 数据同步至 MariaDB。

MariaDB/Percona 数据同步至 MySQL。

TDSQL MySQL（作为源库或者作为目标库）的同步。

MySQL 数据同步至 TDSQL PostgreSQL。

MySQL > MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 MySQL 5.5、5.6、5.7、8.0 CVM 上自建 MySQL 5.5、5.6、5.7、8.0	自建数据库 IDC 自建/CVM 自建 MySQL 5.5、5.6、5.7、8.0 第三方云厂商 阿里云 RDS 5.6、5.7、8.0 阿里云 PolarDB 5.6、5.7、8.0 AWS RDS MySQL 5.6、5.7、8.0 AWS Aurora MySQL 5.6、5.7 云数据库 MySQL 5.5、5.6、5.7、8.0	支持	MySQL/MariaDB/Percona 同步至 MySQL
第三方云厂商 阿里云 RDS 5.6、5.7、8.0 阿里云 PolarDB 5.6、5.7、8.0 AWS RDS MySQL 5.6、5.7、8.0 AWS Aurora MySQL 5.6、5.7	自建数据库 IDC 自建/CVM 自建 MySQL 5.5、5.6、5.7、8.0 第三方云厂商 阿里云 RDS 5.6、5.7、8.0 阿里云 PolarDB 5.6、5.7、8.0 AWS RDS MySQL 5.6、5.7、8.0 AWS Aurora MySQL 5.6、5.7 云数据库 MySQL 5.5、5.6、5.7、8.0		
云数据库 MySQL 5.5、5.6、	自建数据库		

5.7、8.0 相同腾讯云主账号下数据库之间同步 不同腾讯云主账号下数据库之间同步	IDC 自建/CVM 自建 MySQL 5.5、5.6、5.7、8.0 第三方云厂商 阿里云 RDS 5.6、5.7、8.0 阿里云 PolarDB 5.6、5.7、8.0 AWS RDS MySQL 5.6、5.7、8.0 AWS Aurora MySQL 5.6、5.7 云数据库 MySQL 5.5、5.6、5.7、8.0		
---	--	--	--

MariaDB > MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 上自建 MariaDB 5.5、10.0-10.6	自建数据库 IDC 自建/CVM 自建 MySQL 5.5、5.6、5.7、8.0 云数据库 MySQL 5.5、5.6、5.7、8.0	支持	MySQL/MariaDB/Percona 同步至 MySQL
云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4） 相同腾讯云主账号下数据库之间同步 不同腾讯云主账号下数据库之间同步	自建数据库 IDC 自建/CVM 自建 MySQL 5.5、5.6、5.7、8.0 云数据库 MySQL 5.5、5.6、5.7、8.0		

Percona > MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 Percona 5.5、5.6、5.7、8.0 CVM 上自建 Percona 5.5、5.6、5.7、8.0	自建数据库 IDC 自建/CVM 自建 MySQL 5.5、5.6、5.7、8.0 云数据库 MySQL 5.5、5.6、5.7、8.0	支持	MySQL/MariaDB/Percona 同步至 MySQL

TDSQL-C MySQL > MySQL

--	--	--	--

源数据库及版本	目标数据库及版本	双向同步	参考文档
云数据库 TDSQL-C MySQL 5.7、8.0 相同主账号下数据库之间同步 不同主账号下数据库之间同步	自建数据库 IDC 自建/CVM 自建 MySQL 5.5、5.6、5.7、8.0 云数据库 MySQL 5.5、5.6、5.7、8.0	支持	TDSQL-C MySQL 同步至 MySQL

TDSQL MySQL > MySQL

源数据库	目标数据库	双向同步	参考文档
云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1） 相同主账号下数据库之间同步 不同主账号下数据库之间同步	自建数据库 IDC 自建/CVM 自建 MySQL 5.5、5.6、5.7、8.0 云数据库 MySQL	不支持	TDSQL MySQL 同步至 MySQL

同步至 MariaDB

MySQL > MariaDB

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 MySQL 5.5、5.6、5.7、8.0 CVM 自建 MySQL 5.5、5.6、5.7、8.0	自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 上自建 MariaDB 5.5、10.0-10.6 云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4）	支持	MySQL/MariaDB/Percona/TDSQL-C MySQL 同步至 MariaDB
云数据库 MySQL 5.5、5.6、5.7、8.0 相同主账号下数据库之间同步 不同主账号下数据库之间同步	自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 上自建 MariaDB 5.5、10.0-10.6 云数据库 MariaDB（数据库版本 MySQL 8.0、Percona		

	5.7、MariaDB 10.1、MariaDB 10.4)		
--	--------------------------------	--	--

MariaDB > MariaDB

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 自建 MariaDB 5.5、10.0-10.6	自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 上自建 MariaDB 5.5、10.0-10.6 云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4）	支持	MySQL/MariaDB/Percona/TDSQL-C MySQL 同步至 MariaDB
云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4） 相同主账号下数据库之间同步 不同主账号下数据库之间同步	自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 上自建 MariaDB 5.5、10.0-10.6 云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4）		

Percona > MariaDB

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 Percona 5.5、5.6、5.7、8.0 CVM 上自建 Percona 5.5、5.6、5.7、8.0	自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 上自建 MariaDB 5.5、10.0-10.6 云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4）	支持	MySQL/MariaDB/Percona/TDSQL-C MySQL 同步至 MariaDB

TDSQL-C MySQL > MariaDB

--	--	--	--

源数据库及版本	目标数据库及版本	双向同步	参考文档
云数据库 TDSQL-C MySQL 5.7、8.0 相同主账号下数据库之间同步 不同主账号下数据库之间同步	自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 上自建 MariaDB 5.5、10.0-10.6 云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4）	支持	MySQL/MariaDB/Percona/TDSQL-C MySQL 同步至 MariaDB

TDSQL MySQL > MariaDB

源数据库及版本	目标数据库及版本	双向同步	参考文档
云数据库 TDSQL MySQL （数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1） 相同主账号下数据库之间同步 不同主账号下数据库之间同步	自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 上自建 MariaDB 5.5、10.0-10.6 云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4）	不支持	MySQL/MariaDB/Percona/TDSQL-C MySQL 同步至 MariaDB

同步至 TDSQL-C MySQL

MySQL > TDSQL-C MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 MySQL 5.5、5.6、5.7、8.0 CVM 自建 MySQL 5.5、5.6、5.7、8.0	云数据库 TDSQL-C MySQL 5.7、8.0	支持	MySQL/MariaDB/Percona 同步至 TDSQL-C MySQL
第三方云厂商 阿里云 RDS 5.6、5.7、8.0 阿里云 PolarDB 5.6、5.7、8.0	云数据库 TDSQL-C MySQL 5.7、8.0		

AWS RDS MySQL 5.6、5.7、8.0 AWS Aurora MySQL 5.6、5.7			
云数据库 MySQL 5.5、5.6、5.7、8.0 相同主账号下数据库之间同步 不同主账号下数据库之间同步	云数据库 TDSQL-C MySQL 5.7、8.0		

MariaDB > TDSQL-C MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 自建 MariaDB 5.5、10.0-10.6	云数据库 TDSQL-C MySQL 5.7、8.0	支持	MySQL/MariaDB/Percona 同步至 TDSQL-C MySQL
云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4） 相同主账号下数据库之间同步 不同主账号下数据库之间同步	云数据库 TDSQL-C MySQL 5.7、8.0		

Percona > TDSQL-C MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 Percona 5.5、5.6、5.7、8.0 CVM 上自建 Percona 5.5、5.6、5.7、8.0	云数据库 TDSQL-C MySQL 5.7、8.0	支持	MySQL/MariaDB/Percona 同步至 TDSQL-C MySQL

TDSQL-C MySQL > TDSQL-C MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
云数据库 TDSQL-C MySQL 5.7、8.0	云数据库 TDSQL-C MySQL 5.7、8.0	支持	TDSQL-C MySQL

相同主账号下数据库之间同步			同步至 TDSQL-C MySQL
不同主账号下数据库之间同步			

TDSQL MySQL > TDSQL-C MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1） 相同主账号下数据库之间同步 不同主账号下数据库之间同步	云数据库 TDSQL-C MySQL 5.7、8.0	不支持	TDSQL-C MySQL 同步至 TDSQL-C MySQL

同步至 TDSQL MySQL

MySQL > TDSQL MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 MySQL 5.6、5.7、8.0 CVM 自建 MySQL 5.6、5.7、8.0	云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1）	不支持	MySQL/MariaDB/Percona 同步至 TDSQL MySQL
云数据库 MySQL 5.6、5.7、8.0 相同主账号下数据库之间同步 不同主账号下数据库之间同步	云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1）		

MariaDB > TDSQL MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 自建 MariaDB 5.5、10.0-10.6	云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1）	不支持	MySQL/MariaDB/Percona 同步至 TDSQL MySQL

云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4） 相同主账号下数据库之间同步 不同主账号下数据库之间同步	云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1）		
---	--	--	--

Percona > TDSQL MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 Percona 5.5、5.6、5.7、8.0 CVM 上自建 Percona 5.5、5.6、5.7、8.0	云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1）	不支持	MySQL/MariaDB/Percona 同步至 TDSQL MySQL

TDSQL-C MySQL > TDSQL MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
云数据库 TDSQL-C MySQL 5.7、8.0 相同主账号下数据库之间同步 不同主账号下数据库之间同步	云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1）	不支持	MySQL/MariaDB/Percona/TDSQL-C MySQL 同步至 TDSQL MySQL

TDSQL MySQL > TDSQL MySQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1） 相同主账号下数据库之间同步 不同主账号下数据库之间同步	自建 TDSQL MySQL 云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1）	不支持	TDSQL MySQL 同步至 TDSQL MySQL
自建 TDSQL MySQL	云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1）		

MySQL > TDSQL TDSStore

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 MySQL 5.5、5.6、5.7、8.0 CVM 自建 MySQL 5.5、5.6、5.7、8.0	云数据库 TDSQL TDSStore	不支持	MySQL/MariaDB/Percona 同步至 TDSQL TDSStore
云数据库 MySQL 5.5、5.6、5.7、8.0 相同主账号下数据库之间同步 不同主账号下数据库之间同步	云数据库 TDSQL TDSStore		

MariaDB > TDSQL TDSStore

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6 CVM 自建 MariaDB 5.5、10.0-10.6	云数据库 TDSQL TDSStore	不支持	MySQL/MariaDB/Percona 同步至 TDSQL TDSStore
云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4） 相同主账号下数据库之间同步 不同主账号下数据库之间同步	云数据库 TDSQL TDSStore		

Percona > TDSQL TDSStore

源数据库及版本	目标数据库及版本	双向同步	参考文档
自建数据库 IDC 自建 Percona 5.5、5.6、5.7、8.0 CVM 上自建 Percona 5.5、5.6、5.7、8.0	云数据库 TDSQL TDSStore	不支持	MySQL/MariaDB/Percona 同步至 TDSQL TDSStore

同步至 TDSQL PostgreSQL

MySQL > TDSQL PostgreSQL

源数据库及版本	目标数据库及版本	双向同步	参考文档
云数据库 MySQL 5.5、5.6、5.7、8.0 相同主账号下数据库之间同步 不同主账号下数据库之间同步	云数据库 TDSQL PostgreSQL	不支持	云数据库 MySQL 同步至 TDSQL PostgreSQL

同步至 Kafka

MySQL > Kafka

源数据库及版本	目标数据库及版本	参考文档
自建数据库 IDC 自建 MySQL 5.5、5.6、5.7、8.0 CVM 自建 MySQL 5.5、5.6、5.7、8.0	消息队列 CKafka 1.1.1 版本及以后	MySQL/MariaDB/Percona/TDSQL-C MySQL 数据库同步至 Kafka
第三方云厂商 阿里云 RDS 5.6、5.7、8.0 阿里云 PolarDB 5.6、5.7、8.0 AWS RDS MySQL 5.6、5.7、8.0 AWS Aurora MySQL 5.6、5.7	消息队列 CKafka 1.1.1 版本及以后	
云数据库 MySQL 5.5、5.6、5.7、8.0 相同主账号下数据库之间同步 不同主账号下数据库之间同步	消息队列 CKafka 1.1.1 版本及以后	

MariaDB > Kafka

源数据库及版本	目标数据库及版本	参考文档
自建数据库 IDC 自建 MariaDB 5.5、10.0-10.6	消息队列 CKafka 1.1.1 版本及以后	MySQL/MariaDB/Percona/TDSQL-C MySQL 数据库同步至 Kafka

CVM 自建 MariaDB 5.5、10.0-10.6		
云数据库 MariaDB（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1、MariaDB 10.4） 相同主账号下数据库之间同步 不同主账号下数据库之间同步	消息队列 CKafka 1.1.1 版本及以后	

Percona > Kafka

源数据库及版本	目标数据库及版本	参考文档
自建数据库 IDC 自建 Percona 5.5、5.6、5.7、8.0 CVM 上自建 Percona 5.5、5.6、5.7、8.0	消息队列 CKafka 1.1.1 版本及以后	MySQL/MariaDB/Percona/TDSQL-C MySQL 数据库同步至 Kafka

TDSQL-C MySQL > Kafka

源数据库及版本	目标数据库及版本	参考文档
云数据库 TDSQL-C MySQL 5.7、8.0 相同主账号下数据库之间同步 不同主账号下数据库之间同步	消息队列 CKafka 1.1.1 版本及以后	MySQL/MariaDB/Percona/TDSQL-C MySQL 数据库同步至 Kafka

TDSQL MySQL > Kafka

源数据库及版本	目标数据库及版本	参考文档
云数据库 TDSQL MySQL（数据库版本 MySQL 8.0、Percona 5.7、MariaDB 10.1） 相同主账号下数据库之间同步 不同主账号下数据库之间同步	消息队列 CKafka 1.1.1 版本及以后	TDSQL MySQL 同步至 Kafka

云数据库跨账号实例间同步

最近更新时间：2023-11-21 20:40:55

操作场景

云数据库跨账号实例间数据同步，指源库和目标库都属于腾讯云数据库实例，但所属不同的主账号名下，这种数据库实例之间的数据同步。本章节介绍通过 DTS 数据同步功能实现从其他账号腾讯云数据库实例同步数据至本账号下云数据库实例。

支持范围

支持腾讯云数据库实例 MySQL/TDSQL MySQL/TDSQL-C MySQL/MariaDB/PostgreSQL 间的跨账号同步，具体请参见 [数据同步支持的数据库](#) 中的表信息 **跨账号同步** 列。

前提条件

已创建目标数据库实例。

注意事项

本操作中涉及多处账号信息配置，如下列出了主要配置逻辑，以方便用户理解和正确配置。

数据同步方向：源数据库（其他账号的数据库实例）> 目标数据库（本账号的数据库实例）。

执行同步任务的账号可以是目标数据库的主账号，也可以是目标数据库的子账号。

使用主账号执行同步任务。操作任务前，需要请求源数据所属主账号持有人进行角色授权（给目标数据库的主账号），使目标数据库主账号可以访问源数据库。

使用子账号执行同步任务。操作任务前，需要先请求源数据所属主账号持有人进行角色授权（给目标数据库的主账号），使目标数据库主账号可以访问源数据库。然后再请求目标数据库所属主账号持有人进行策略授权（对目标数据库的子账号），使目标数据库子账号可以访问源数据库。

授权账号

使用主账号执行同步任务，请操作步骤1 - 6，使用子账号执行同步任务，请操作步骤1 - 11。

1. 使用源数据库所属的腾讯云主账号登录 [访问管理控制台](#)（如果子账号有CAM和角色相关的权限，也可以使用子账号登录）。
2. 左侧导航单击**角色**，进入角色管理页面，然后单击**新建角色**。
3. 在选择角色载体页面，选择**腾讯云账户**方式。

Select role entity



Tencent Cloud Product Service

Authorize Tencent Cloud service to use your cloud resources via roles



Tencent Cloud Account

Authorize your root account or other root accounts to use your cloud resources via roles



IdPs

Authorize external user identity (such as enterprise user directory) to use your cloud resources

4. 在**输入角色载体信息**页面，配置相关信息，单击**下一步**。

1 Enter Role Entity Info

2 Configure Role Policy

3 Set Role Tag

Tencent Cloud account *☐ Current root account ☒ Other root account

Account ID *

10137

The root account ID of Target Databases

Console access☐ Allow the current role to access console

External ID☐ Enable Verification (You're advised to enable this feature when a third-party platform uses the role)

i

The external ID is a string of characters that you define for this role. To use this role, a third-party platform must verify the role's identity. This improves the security of role assuming by preventing unauthorized use of the role. You're advised to enable external ID verification if you will allow a third-party platform to assume the role.

Next

云账号类型：选择**其他主账号**。

账号 ID：填入目标数据库所属的腾讯云主账号 ID，主账号 ID 可在 [账号信息](#) 中查看。目标数据库实例属于子账号名下时，此处也填写主账号 ID。

外部 ID：可依据情况，选择性开启。

说明

如果使用了外部 ID，请用户自行记录和保存该 ID，后续配置需要填写。

5. 在**配置角色策略**页面，选择 DTS 策略和源数据库服务对应策略，单击**下一步**。

DTS 服务策略，选择 QcloudDTSReadOnlyAccess。

源数据库服务对应策略。

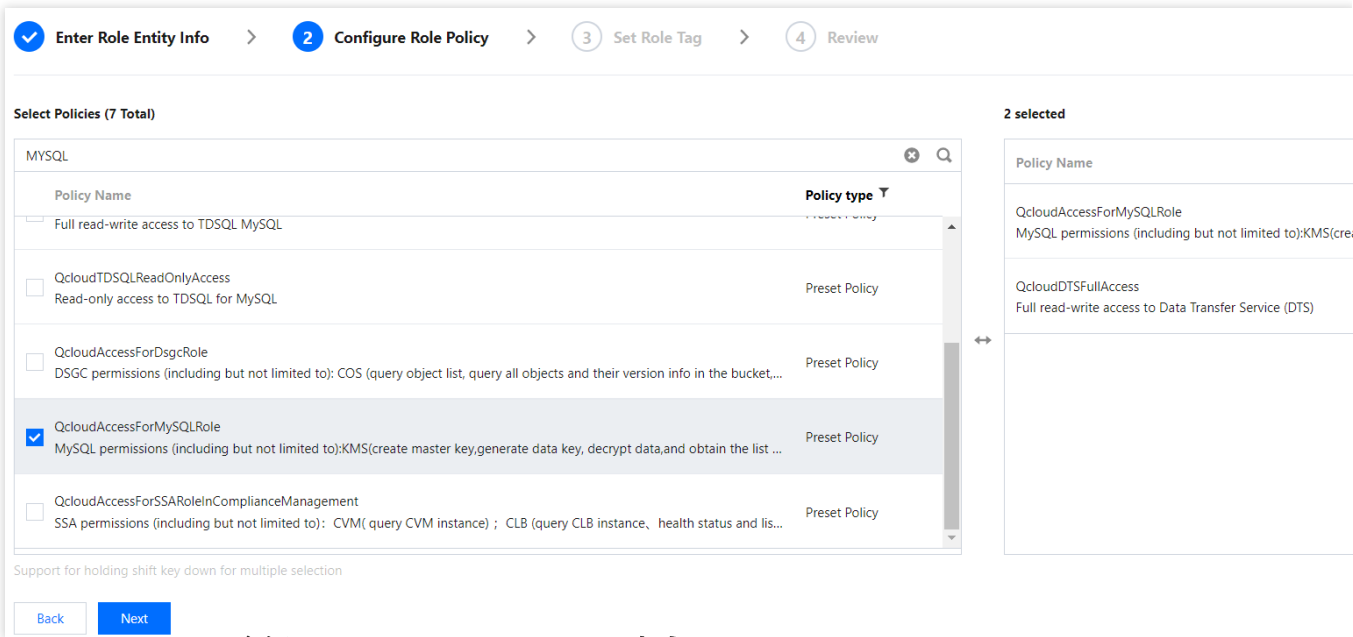
源数据库为云数据库 MySQL，选择 QcloudCDBReadOnlyAccess（云数据库 CDB 相关资源只读访问权限），QcloudCDBInnerReadOnlyAccess（CDB 的读取列表权限）。

源数据库为云数据库 MariaDB，选择 QcloudMariaDBReadOnlyAccess（云数据库 MariaDB 只读访问权限）。

源数据库为云数据库 TDSQL MySQL，选择 QcloudTDSQLReadOnlyAccess（TDSQL MySQL 版（TDSQL for MySQL）只读访问权限）。

源数据库为云数据库 TDSQL-C MySQL，选择 QcloudCynosDBReadOnlyAccess（云原生数据库 TDSQL-C（CynosDB）只读访问权限）。

源数据库为云数据库 PostgreSQL，选择 QcloudPostgreSQLReadOnlyAccess（云数据库 PostgreSQL 只读访问权限）。



6. 配置角色标签，然后在**审阅**页面，设置角色名称，单击**完成**后该角色创建完成。

说明

配置后记录该名称，后续创建同步任务时需要输入。

✓

Enter Role Entity Info

>

✓

Configure Role Policy

>

✓

Set Role Tag

>

4

Role Name *

DTS-test

Description

Role Entity

Account-

Access Type

Programming access

Tag

No tag

Policy Name	Description
QcloudDTSFullAccess	Full read-write access to Data Transfer Service (DTS)
QcloudAccessForMySQLRole	MySQL permissions (including but not limited to):KMS(create master key,generate data key, decrypt

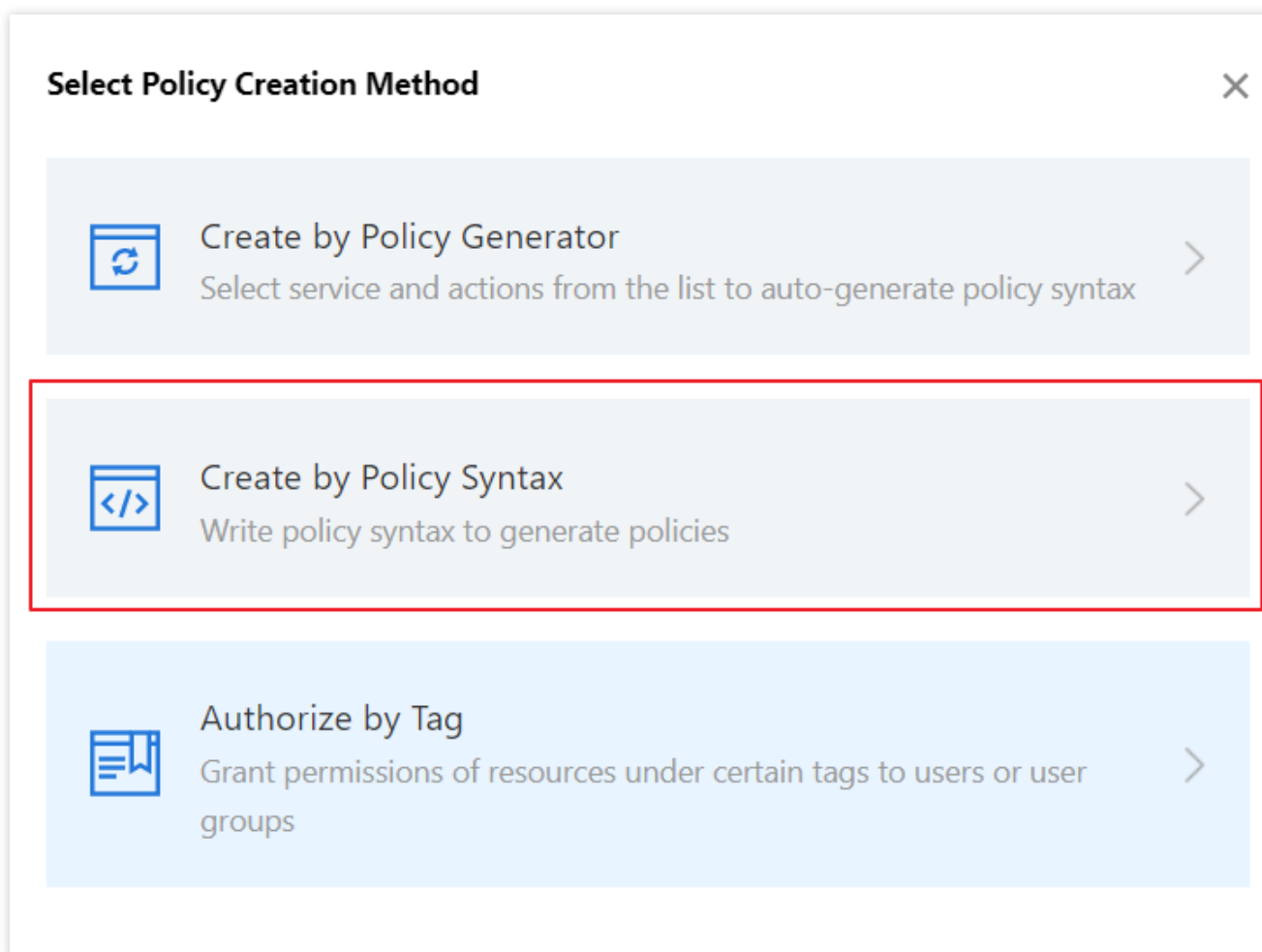
Back

Complete

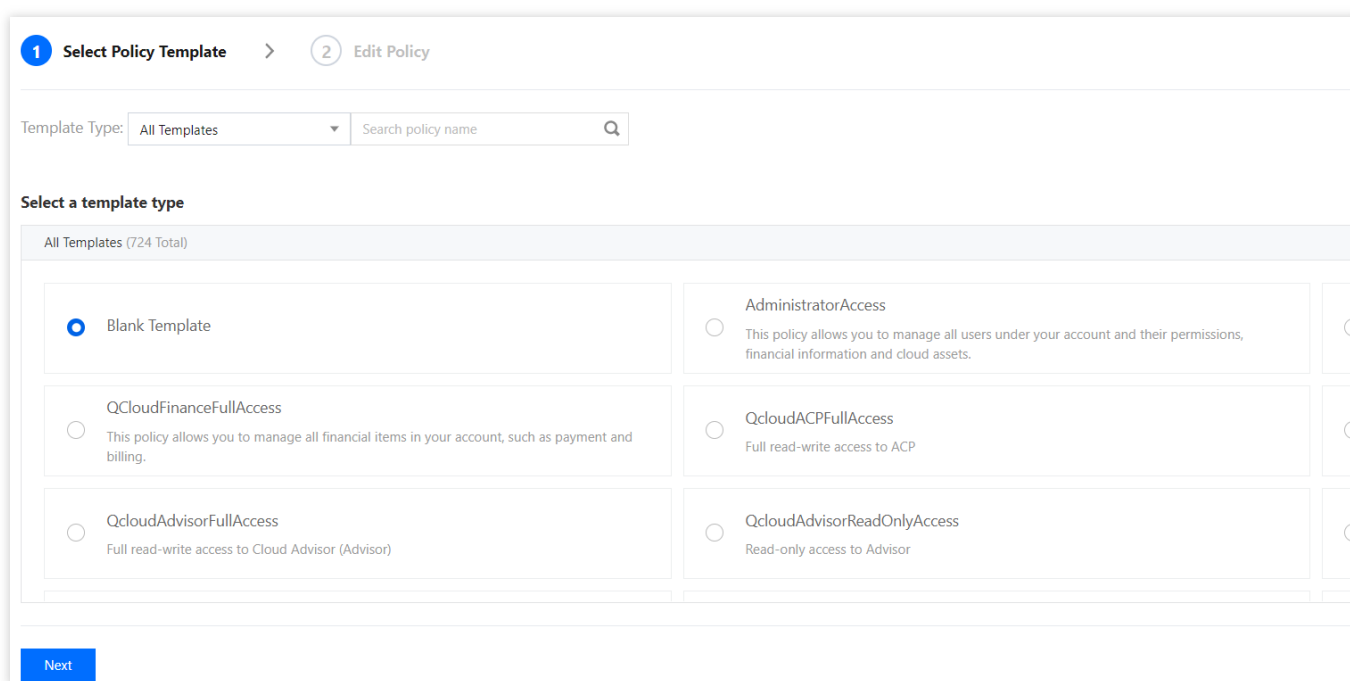
说明

如果执行同步任务的账号为主账号，授权步骤到此结束；如果为子账号，还需要继续进行如下步骤7-11，请求当前主账号持有人，对子账号进行策略授权。

7.（可选）使用目标数据库所属的腾讯云主账号登录 [访问管理控制台](#)，在左侧导航单击**策略**，然后在右侧单击**新建自定义策略**，并选择**按策略语法创建**。



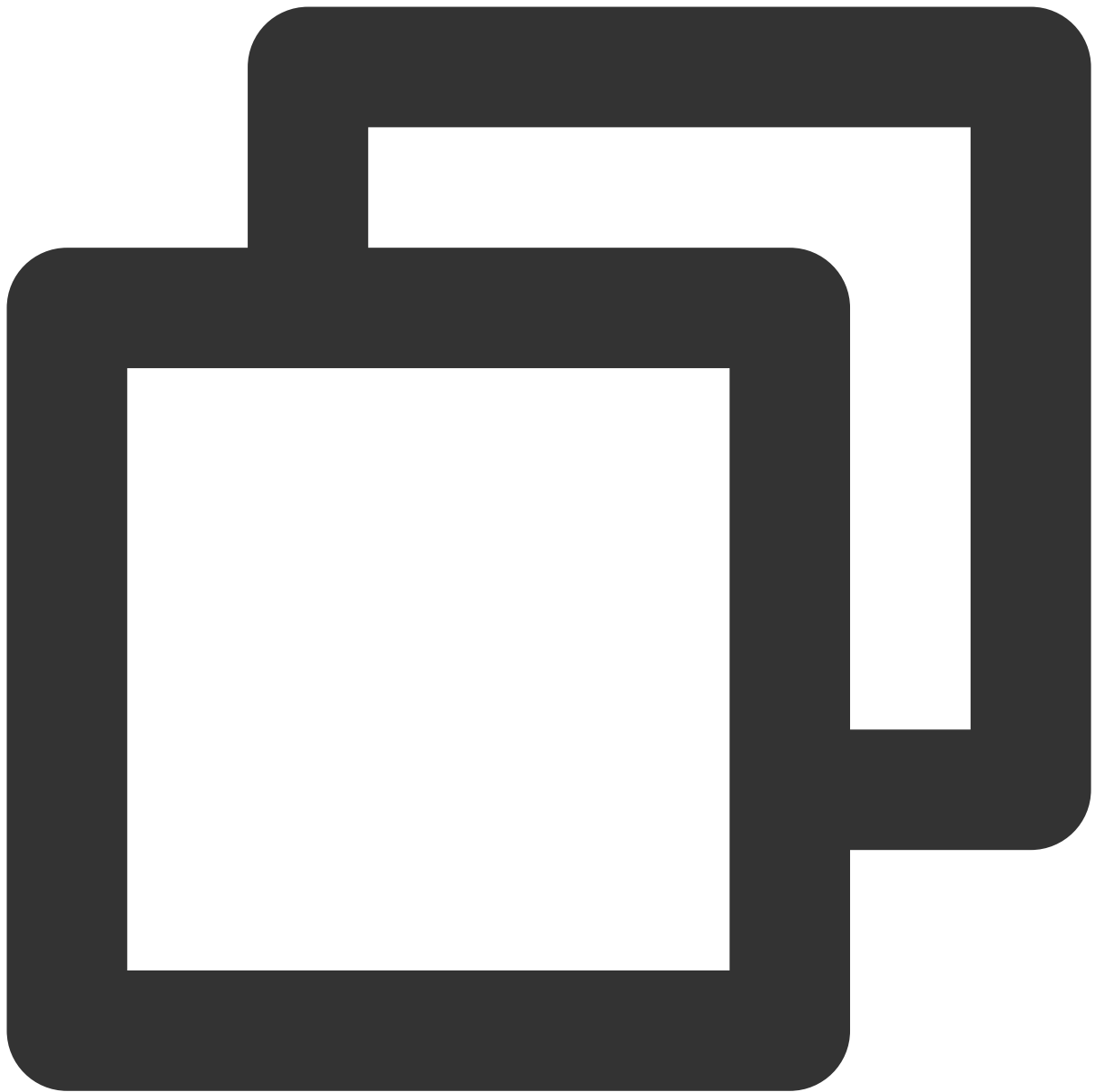
8. (可选) 选择空白模板，然后单击下一步。



9. （可选）创建一个策略，策略的名称以及描述可以根据自己的需求填写，策略内容复制示例代码后，将红框中的内容替换成对应的信息。

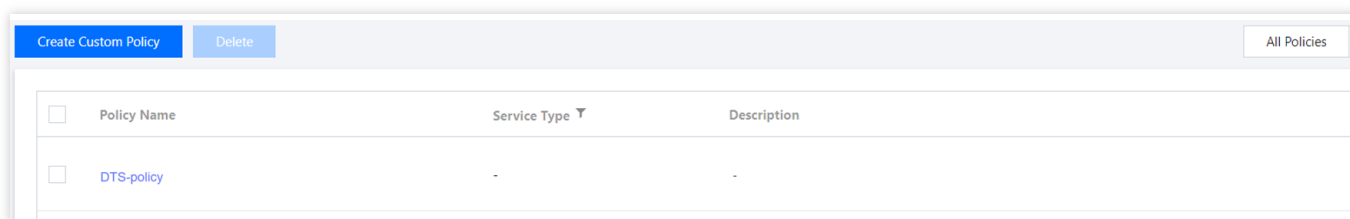
Previous Complete

第25 共300页

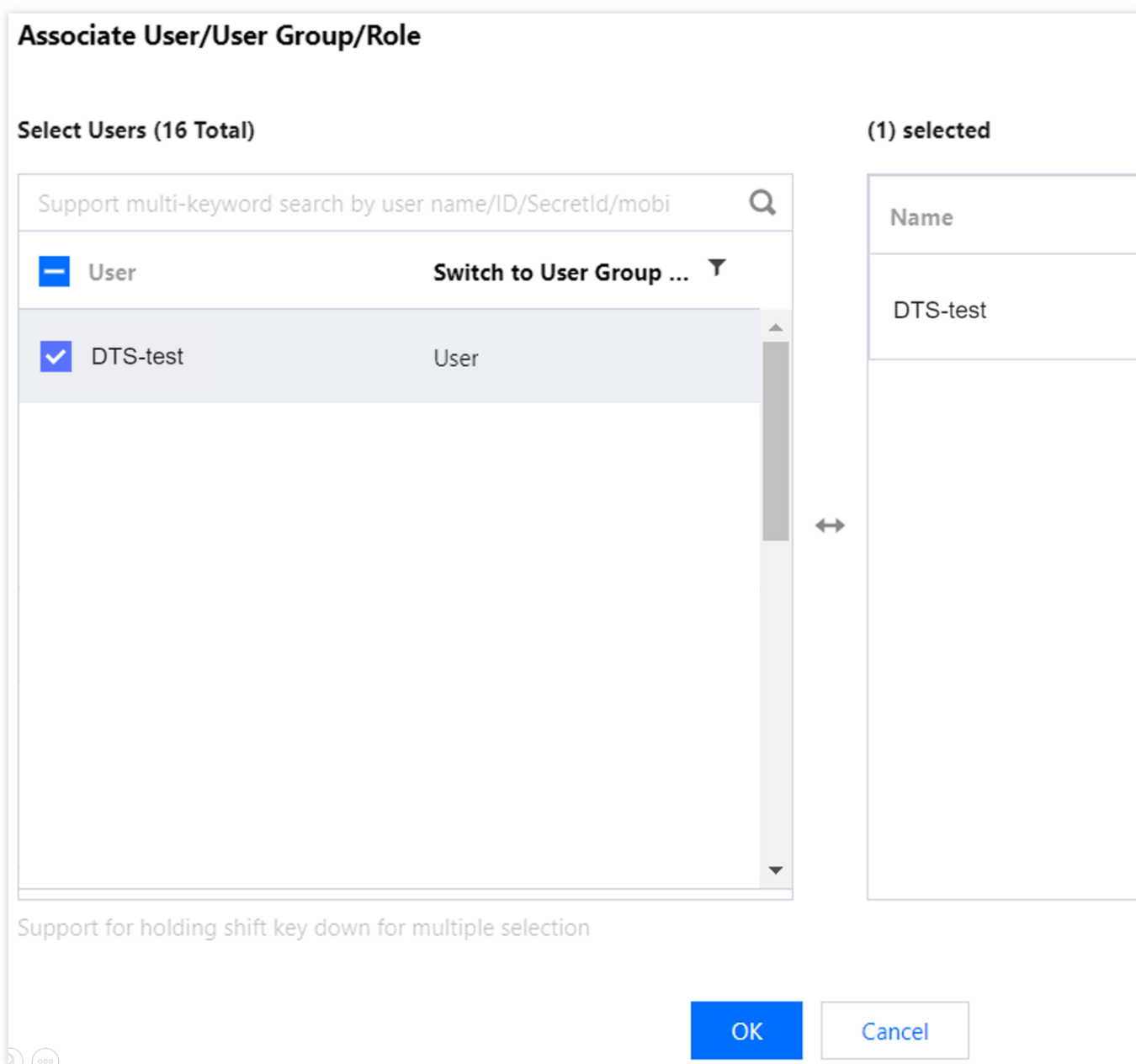


```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "allow",
      "action": ["name/sts:AssumeRole"],
      "resource": ["qcs::cam::uin/10*****8:roleName/DTS-role"]
    }
  ]
}
```

10.（可选）单击**完成**后返回到策略列表页，在列表页中单击**关联用户/组**。



11.（可选）选择目标数据库实例所属子账号（即执行同步任务的子账号），单击**确定**，如下图所示。



创建同步任务

1. 使用目标数据库实例所属腾讯云账号，登录 [DTS 控制台](#)。
2. 选择**数据同步** > **新建同步任务**，购买一个新的同步任务。
3. 购买完成后，返回数据同步列表，单击**操作**列的**配置**，进入配置同步任务页面。
4. 在设置源和目标数据页面，配置源库和目标库信息。下面以 MySQL 到 MySQL 的同步为例进行介绍。

1

Set source and target databases

>

2

Set migration options and select migration o

Task Configuration

Task Name *

stacy-test

Running Mode *

Immediate execution

Scheduled execution

Source Database Settings

Source Database Type *

MySQL

Service Provider *

Others

Access Type *

Public Network

Self-Build on CVM

Direct Connect

VPN Access

Please add the DTS IP addresses to the security group allowlist in advance so that the connect

Cross-/Intra-Account *

Intra-account

Cross-account

Peer Account ID *

10 8

The root account ID
Source Database

Peer Account Role Name *

DTS-role

Role External ID

Please enter the external ID of the role

Region

South China(Guangzhou)

Database Instance *

cdbro- (1 1)

Account *

root

Password *

Please enter password

Test Connectivity

跨账号关键参数配置如下：

接入类型：选择**云数据库**，表示源数据库属于腾讯云数据库实例。

是否跨账号：选择**跨账号**。

跨腾讯云账号 ID：填入源数据库所属的主账号 ID。

跨账号授权角色名称。即前文 [授权账号](#) 步骤6中创建的**角色名称**。您可以通过 [角色](#) 以及 [跨账号角色](#) 了解更多关于角色的概念。

角色外部 ID：如果前文 [授权账号](#) 步骤4中设置了外部 ID，则这里则必须按照设置的内容正确填写，否则会发生权限报错。没有设置则不需要填写。

说明

完成上述配置后，选择对应的**所属地域**，即可获取到跨账号下的实例列表，如果获取实例出现报错，则可能为配置错误，或者未授权，请参考 [常见问题](#) 进行处理。

5. 在设置同步选项和同步对象页面，对数据初始化选项、数据同步选项、同步对象选项进行设置，在设置完成后单击**保存并下一步**。

6. 在校验任务页面，完成校验并全部校验项通过后，单击**启动任务**。

如果校验任务不通过，可以参考 [校验不通过处理方法](#) 修复问题后重新发起校验任务。

7. 返回数据同步任务列表，任务开始进入**运行中**状态。

说明

选择**操作列**的**更多 > 结束**可关闭同步任务，请您确保数据同步完成后再关闭任务。

常见问题

1. 跨账号拉取实例列表报错：role not exist[InternalError.GetRoleError]

请确认**跨腾讯云账号 ID**（应该为源数据库的主账号 ID）和**跨账号授权角色名称**（应该为 [授权账号](#) 步骤6中创建的**角色名称**）配置是否正确。如果还无法拉取，可能没有授权源数据库的服务权限（参考 [授权账号](#) 中的步骤5）。

2. 获取云数据库实例列表失败：InternalError:InternalInnerCommonError

角色中没有授权源数据库所属的腾讯云服务策略，请参考 [授权账号](#) 中的步骤5进行授权。

3. 跨账号拉取实例列表报错：you are not authorized to perform operation (sts:AssumeRole) , resource (qcs::cam::uin/1xx5:roleName/xxxx) has no permission

出错原因：您当前创建同步任务使用的账号是子账号，并且当前子账号没有 sts:AssumeRole 权限。

解决方法：

使用主账号来创建同步任务。

请求目标数据库所属的主账号持有人，参考 [授权账号](#) 对子账号授权，策略语法中的 resource，填写报错框中蓝色字段部分。

4. DTS 任务配置未填写角色外部 ID，或者填写错误，导致权限报错或者拉取数据库实例失败。

出错原因：[授权账号](#) 时设置了外部 ID，但在 DTS 任务配置中未填写外部 ID，或者填写错误，导致权限报错或者拉取数据库实例失败。

解决方案：授权账号时如果设置了外部 ID，则 DTS 任务配置时需要正确填写，未设置则不需要填。

同步至 MySQL 系列

同步至 MySQL

MySQL/MariaDB/Percona 同步至 MySQL

最近更新时间：2024-02-02 16:16:46

本文为您介绍使用数据传输服务 DTS 从 MySQL、MariaDB、Percona 数据库同步数据至腾讯云数据库 MySQL 的过程。

源数据库支持的部署类型如下：

自建 MySQL、第三方云厂商 MySQL、腾讯云数据库 MySQL。

自建 MariaDB、腾讯云数据库 MariaDB。

自建 Percona。

因为 MySQL、MariaDB、Percona 同步至腾讯云数据库 MySQL，三种场景的同步要求和操作步骤基本一致，本章节仅以 MySQL 到 MySQL 的数据同步为例进行介绍，其他场景请参考相关内容。

注意事项

DTS 在执行全量数据同步时，会占用一定源端实例资源，可能会导致源实例负载上升，增加数据库自身压力。如果您数据库配置过低，建议您在业务低峰期进行。

为了避免数据重复，请确保需要同步的表具有主键或者非空唯一键。

默认采用无锁方式，同步过程中对源库不加全局锁（FTWRL），仅对无主键的表加表锁，其他不加锁。

数据同步时，DTS 会使用执行同步任务的账号在源库中写入系统库 `__tencentdb__`，用于记录同步任务过程中的数据对比信息。

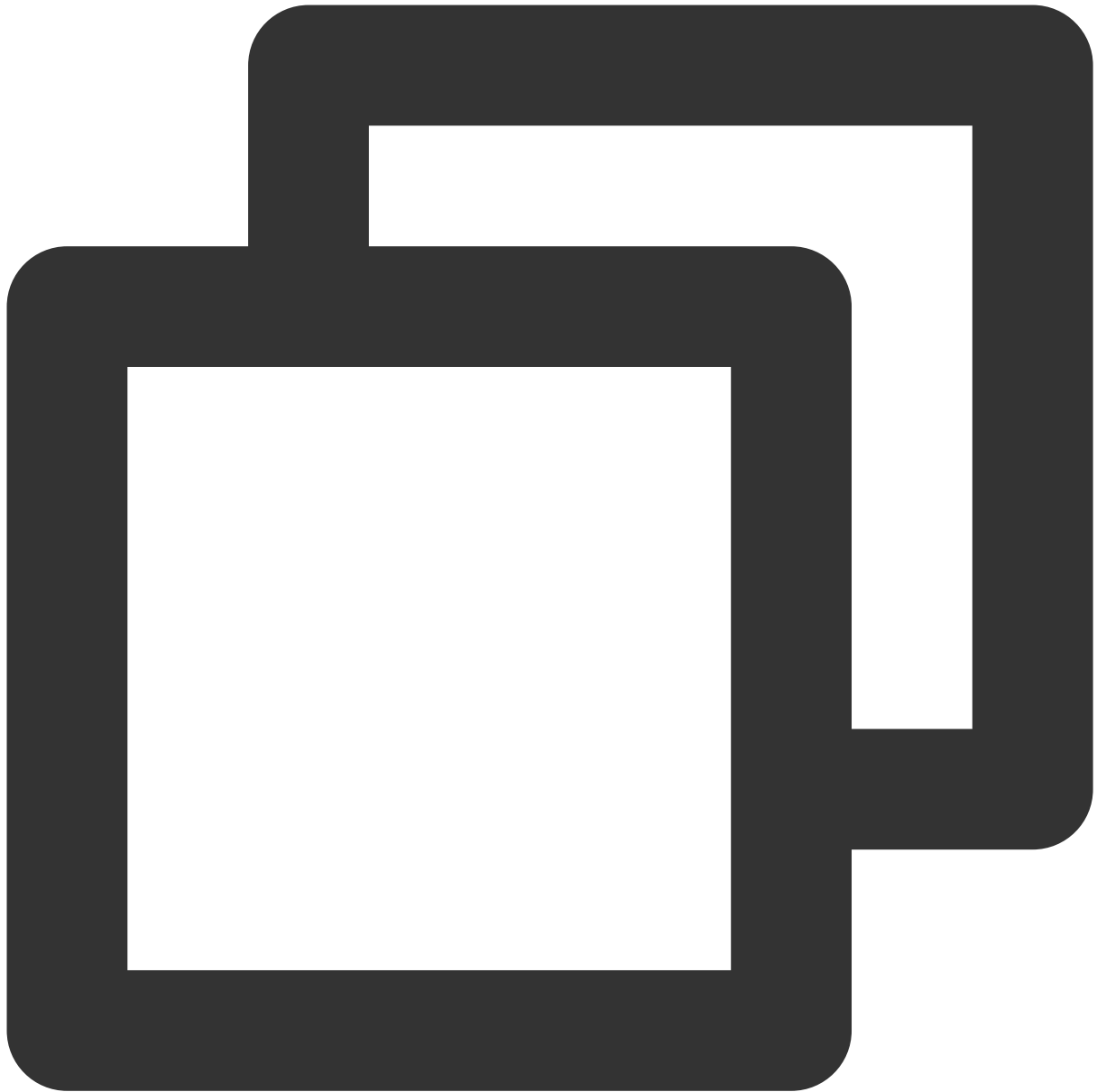
为保证后续数据对比问题可定位，同步任务结束后不会删除源库中的 `__tencentdb__`。

`__tencentdb__` 系统库占用空间非常小，约为源库存储空间的千分之一到万分之一（例如源库为50G，则 `__tencentdb__` 系统库约为 5K-50K），并且采用单线程，等待连接机制，所以对源库的性能几乎无影响，也不会抢占资源。

前提条件

源数据库和目标数据库符合同步功能和版本要求，请参考 [数据同步支持的数据库](#) 进行核对。

需要具备源数据库的权限如下：



```
GRANT RELOAD,LOCK TABLES,REPLICATION CLIENT,REPLICATION SLAVE,SHOW VIEW,PROCESS,SEL  
GRANT ALL PRIVILEGES ON `__tencentdb__`.* TO '帐号'@'%';  
FLUSH PRIVILEGES;
```

需要具备目标数据库的权限：ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE USER, CREATE VIEW, DELETE, DROP, EVENT, EXECUTE, INDEX, INSERT, LOCK TABLES, PROCESS, REFERENCES, RELOAD, SELECT, SHOW DATABASES, SHOW VIEW, TRIGGER, UPDATE。

应用限制

只支持同步基础表、视图、存储过程和函数。

在同步视图、存储过程和函数时，DTS 会检查源库中 `DEFINER` 对应的 `user1`（`[DEFINER = user1]`）和同步账号 `user2` 是否一致，如果不一致，同步后 DTS 会修改 `user1` 在目标库中的 `SQL SECURITY` 属性，由 `DEFINER` 转换为 `INVOKER`（`[INVOKER = user1]`），同时设置目标库中 `DEFINER` 为同步账号 `user2`（`[DEFINER = 同步账号 user2]`）。如果源库中视图定义过于复杂，可能会导致任务失败。

源端如果是非 GTID 实例，DTS 不支持源端 HA 切换，一旦源端 MySQL 发生切换可能会导致 DTS 增量同步中断。只支持同步 InnoDB、MyISAM、TokuDB 三种数据库引擎，如果存在这三种以外的数据引擎表则默认跳过不进行同步。

相互关联的数据对象需要一起同步，否则会导致同步失败。常见的关联关系：视图引用表、视图引用视图、主外键关联表等。

增量同步过程中，若源库存在分布式事务或者产生了类型为 `STATEMENT` 格式的 Binlog 语句，则会导致同步失败。

源数据库为阿里云 MySQL，则阿里云 MySQL 5.6 版本待同步表不能存在无主键表，MySQL 5.7 及以后版本不限制。源数据库为 AWS MySQL，则 AWS MySQL 待同步表不能存在无主键表。

源数据库 Binlog 的 GTID 如果存在空洞，可能会影响同步任务的性能并导致任务失败。

不支持同时包含 DML 和 DDL 语句在一个事务的场景，遇到该情况任务会报错。

不支持 Geometry 相关的数据类型，遇到该类型数据任务报错。

不支持 `ALTER VIEW` 语句，遇到该语句任务跳过不同步。

操作限制

同步过程中请勿进行如下操作，否则会导致同步任务失败。

请勿修改、删除源数据库和目标数据库中用户信息（包括用户名、密码和权限）和端口号。

请勿在源库上执行分布式事务。

请勿在源库写入 Binlog 格式为 `STATEMENT` 的数据。

请勿在源库上执行清除 Binlog 的操作。

在同步增量阶段，请勿删除系统库表 `__tencentdb__`。

支持同步的 SQL 操作

操作类型	SQL 操作语句
DML	INSERT、UPDATE、DELETE

DDL	CREATE DATABASE、DROP DATABASE、ALTER DATABASE、CREATE TABLE、ALTER TABLE、DROP TABLE、TRUNCATE TABLE、RENAME TABLE、CREATE VIEW、DROP VIEW、CREATE INDEX、DROP INDEX 说明： 不支持同步涉及分区（Partition）的 DDL。
-----	---

环境要求

类型	环境要求
源数据库要求	源库和目标库网络能够连通。 实例参数要求： 源库 <code>server_id</code> 参数需要手动设置，且值不能设置为0。 源库表的 <code>row_format</code> 不能设置为 <code>FIXED</code>。 源库和目标库 <code>lower_case_table_names</code> 变量必须设置一致。 源库变量 <code>connect_timeout</code> 设置数值必须大于10。 Binlog 参数要求： 源端 <code>log_bin</code> 变量必须设置为 <code>ON</code>。 源端 <code>binlog_format</code> 变量必须设置为 <code>ROW</code>。 源端 <code>binlog_row_image</code> 变量必须设置为 <code>FULL</code>。 MySQL 5.6 及以上版本 <code>gtid_mode</code> 变量不为 <code>ON</code> 时会报警告，建议打开 <code>gtid_mode</code>。 不允许设置 <code>do_db</code>, <code>ignore_db</code>。 源实例为从库时，<code>log_slave_updates</code> 变量必须设置为 <code>ON</code>。 建议源库 Binlog 日志至少保留3天及以上，否则可能会因任务暂停/中断时间大于 Binlog 日志保留时间，造成任务无法续传，进而导致任务失败。 外键依赖： 外键依赖只能设置为 <code>NO ACTION</code>，<code>RESTRICT</code> 两种类型。 部分库表同步时，有外键依赖的表必须齐全。
目标数据库要求	目标库的版本必须大于等于源库的版本。 目标库需要有足够的存储空间，如果初始类型选择“全量数据初始化”，则目标库的空间大小须是源库待同步库表空间的1.2倍以上。 目标库不能有和源库同名的表、视图等同步对象。 目标库 <code>max_allowed_packet</code> 参数设置数值至少为4M。
其他要求	环境变量 <code>innodb_stats_on_metadata</code> 必须设置为 <code>OFF</code> 。

操作步骤

1. 登录 [数据同步购买页](#)，选择相应配置，单击**立即购买**。

参数	描述
----	----

计费模式	支持包年包月和按量计费。
源实例类型	选择 MySQL，购买后不可修改。
源实例地域	选择源实例所在地域，购买后不可修改。
目标实例类型	选择 MySQL，购买后不可修改。
目标实例地域	选择目的实例所在地域，购买后不可修改。
规格	请根据业务诉求选择规格，规格越高，性能越好。

2. 购买完成后，返回 [数据同步列表](#)，可看到刚创建的数据同步任务，刚创建的同步任务需要进行配置后才可以使
用。
3. 在数据同步列表，单击**操作**列的**配置**，进入配置同步任务页面。
4. 在配置同步任务页面，配置源端实例、帐号密码，配置目标端实例、帐号和密码，测试连通性后，单击**下一步**。

1 Set source and target databases >

2 Set sync options and objects >

Task Configuration

Task Name *

sync-

Running Mode *

Immediate execution

Scheduled execution

Source Instance Settings

Source Instance Type *

MySQL

Source Instance Region

South China(Guangzhou)

Service Provider *

Others

AWS

Alibaba Cloud

Access Type *

Public Network

Self-Build on CVM

Direct Connect

VPN Access

Cross-/Intra-Account *

Intra-account

Cross-account

Instance ID *

cdb-

Account *

root

Password *

.....

Test Connectivity

Test passed

Target Instance Settings

Target Instance Type *

MySQL

Target Instance Region

South China(Guangzhou)

Access Type *

Public Network

Self-Build on CVM

Direct Connect

VPN Access

Instance ID *

cdb-

Account *

root

Password *

.....

Test Connectivity

Test passed

因源数据库部署形态和接入类型的交叉场景较多，各场景同步操作步骤类似，如下仅提供典型场景的配置示例，其他场景请用户参考配置。

示例一：本地自建数据库通过专线/VPN方式同步至腾讯云数据库

设置项	参数	描述
任务设置	任务名称	DTS 会自动生成一个任务名称，用户可以根据实际情况进行设置。
	运行模式	支持立即执行和定时执行两种模式。
源实例设置	源实例类型	购买时所选择的源实例类型，不可修改。
	源实例地域	购买时选择的源实例所在地域，不可修改。
	服务提供商	自建数据库（包括云服务器上的自建）或者腾讯云数据库，请选择“普通”；第三方云厂商数据库，请选择对应的服务商。 本场景选择“普通”。
	接入类型	请根据您的场景选择，本场景选择“专线接入”或“VPN接入”，该场景需要配置 VPN 和 IDC 之间的互通，其他接入类型的准备工作请参考 准备工作概述 。

		公网：源数据库可以通过公网 IP 访问。 云主机自建：源数据库部署在 腾讯云服务器 CVM 上。 专线接入：源数据库可以通过 专线接入 方式与腾讯云私有网络打通。 VPN接入：源数据库可以通过 VPN 连接 方式与腾讯云私有网络打通。 云数据库：源数据库属于腾讯云数据库实例。 云联网：源数据库可以通过 云联网 与腾讯云私有网络打通。 私有网络 VPC：源数据和目标数据库都部署在腾讯云上，且有 私有网络。 如果需要使用私有网络 VPC接入类型，请 提交工单 申请。
	私有网络专线网关/VPN 网关	专线接入时只支持私有网络专线网关，请确认网关关联网络类型。 VPN 网关，请选择通过 VPN 网关接入的 VPN 网关实例。
	私有网络	选择私有网络专线网关和 VPN 网关关联的私有网络和子网。
	主机地址	源实例 MySQL 访问 IP 地址或域名。
	端口	源实例 MySQL 访问端口。
	帐号	源实例帐号，帐号权限需要满足要求。
	密码	源实例帐号的密码。
目标实例设置	目标实例类型	购买时选择的目标实例类型，不可修改。
	目标实例地域	购买时选择的目标实例地域，不可修改。
	接入类型	根据您的场景选择，本场景选择“云数据库”。
	实例 ID	选择目标实例 ID。
	帐号	目标实例帐号，帐号权限需要满足要求。
	密码	目标实例帐号的密码。

示例二：腾讯云数据库同步至腾讯云数据库

设置项	参数	描述
任务设置	任务名称	DTS 会自动生成一个任务名称，用户可以根据实际情况进行设置。
	运行模式	支持立即执行和定时执行两种模式。
源实例设置	源实例类型	购买时所选择的源实例类型，不可修改。
	源实例地域	购买时选择的源实例所在地域，不可修改。
	服务提供商	自建数据库（包括云服务器上的自建）或者腾讯云数据库，请选择“普通”；第三方云厂商数据库，请选择对应的服务商。

		本场景选择“普通”。
	接入类型	请根据您的场景选择，本场景选择“云数据库”，不同接入类型的准备工作请参考准备工作概述。 公网：源数据库可以通过公网 IP 访问。 云主机自建：源数据库部署在 腾讯云服务器 CVM 上。 专线接入：源数据库可以通过 专线接入 方式与腾讯云私有网络打通。 VPN接入：源数据库可以通过 VPN 连接 方式与腾讯云私有网络打通。 云数据库：源数据库属于腾讯云数据库实例。 云联网：源数据库可以通过 云联网 与腾讯云私有网络打通。 私有网络 VPC：源数据和目标数据库都部署在腾讯云上，且有 私有网络。如果需要使用私用网络 VPC接入类型，请 提交工单 申请。
	实例 ID	源实例 ID。可在 实例列表 查看源实例信息。
	帐号	源实例帐号，帐号权限需要满足要求。
	密码	源实例帐号的密码。
目标实例设置	目标实例类型	购买时选择的目标实例类型，不可修改。
	目标实例地域	购买时选择的目标实例地域，不可修改。
	接入类型	根据您的场景选择，本场景选择“云数据库”。
	实例 ID	选择目标实例 ID。
	帐号	目标实例帐号，帐号权限需要满足要求。
	密码	目标实例帐号的密码。

示例三：阿里云 RDS 通过公网方式同步至腾讯云数据库

设置项	参数	描述
任务设置	任务名称	DTS 会自动生成一个任务名称，用户可以根据实际情况进行设置。
	运行模式	支持立即执行和定时执行两种模式。
源实例设置	源实例类型	购买时所选择的源实例类型，不可修改。
	源实例地域	购买时选择的源实例所在地域，不可修改。
	服务提供商	自建数据库（包括云服务器上的自建）或者腾讯云数据库，请选择“普通”；第三方云厂商数据库，请选择对应的服务商。 本场景选择“阿里云”。

	接入类型	对于第三方云厂商数据库，一般可以选择公网方式，也可以选择 VPN 接入，专线或者云联网的方式，需要根据实际的网络情况选择。 本场景选择“公网”，不同接入类型的准备工作请参考准备工作概述。 公网：源数据库可以通过公网 IP 访问。 云主机自建：源数据库部署在 腾讯云服务器 CVM 上。 专线接入：源数据库可以通过 专线接入 方式与腾讯云私有网络打通。 VPN接入：源数据库可以通过 VPN 连接 方式与腾讯云私有网络打通。 云数据库：源数据库属于腾讯云数据库实例。 云联网：源数据库可以通过 云联网 与腾讯云私有网络打通。 私有网络 VPC：源数据和目标数据库都部署在腾讯云上，且有 私有网络。 如果需要使用私有网络 VPC接入类型，请 提交工单 申请。
	主机地址	源实例访问 IP 地址或域名。
	端口	源实例端口。
	帐号	源实例帐号，帐号权限需要满足要求。
	密码	源实例帐号的密码。
目标实例设置	目标实例类型	购买时选择的目标实例类型，不可修改。
	目标实例地域	购买时选择的目标实例地域，不可修改。
	接入类型	根据您的场景选择，本场景选择“云数据库”。
	实例 ID	选择目标实例 ID。
	帐号	目标实例帐号，帐号权限需要满足要求。
	密码	目标实例帐号的密码。

5. 在设置同步选项和同步对象页面，将对数据初始化选项、数据同步选项、同步对象选项进行设置，在设置完成后单击**保存并下一步**。

说明：

当**初始化类型**仅选择**全量数据初始化**，系统默认用户在目标库已经创建了表结构，不会进行表结构同步，也不会校验源库和目标库是否有同名表，所以当用户同时在**已存在同名表**项选择**前置校验并报错**，则校验并报错功能不生效。

仅选择**全量数据初始化**的场景，用户需要提前在目标库创建好表结构。

如果用户在同步过程中确定会对某张表使用 **rename** 操作（例如将 **table A** rename 为 **table B**），则**同步对象**需要选择 **table A** 所在的整个库（或者整个实例），不能仅选择 **table A**，否则系统会报错。

1 Set source and target databases > 2 Set sync options and objects > 3 Verify task

Data Initialization Option

Initialization Type

Structure Initialization

Full data initialization

If Target Already Exists *

Precheck and report error

Ignore and execute

Data Sync Option

Primary Key Conflict Resolution *

Report

Ignore

Overwrite

SQL Type

DML

Insert

Update

Delete

DDL

DDL

Custom DDL *

Database

Create

Drop

Alter

Table

Create

Drop

Alter

Truncate

Rename

View

Create

Drop

Index

Create

Drop

Sync Object Option

Advanced Migration Object ⓘ

☒ Procedure

☒ Function

Advanced objects can only be copied once, which means you cannot copy new objects once the task is started.

ⓘ Up to 200 results can be displayed. If the objects you need are not shown in the result list, you can search them by object name.

Source Database Object

Search database name, supporting fuzzy match

ⓘ 1 database in total, with 1 displayed More

▶ ☒ db-dst

Refresh Select all Clear

Selected Object ⓘ

Globally search for original object names, v

☐ db-dst (Entire database selecte

Unfold all Fold all Select all Clear Revert to

设置项	参数	描述

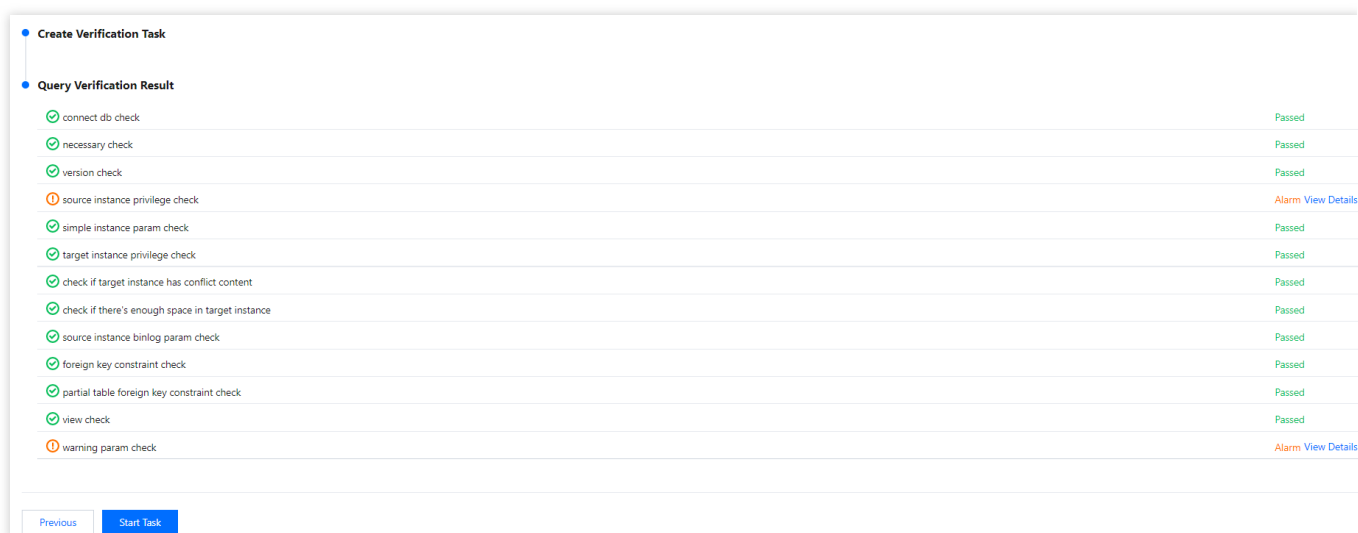
数据初始化选项	初始化类型	结构初始化：同步任务执行时会先将源实例中表结构初始化到目标实例中。 全量数据初始化：同步任务执行时会先将源实例中数据初始化到目标实例中。默认两者都勾选，可根据实际情况取消。
	已存在同名表	前置校验并报错：存在同名表则报错，流程不再继续。 忽略并继续执行：全量数据和增量数据直接追加目标实例的表中。
数据同步选项	冲突处理机制	冲突报错：在同步时发现表主键冲突，报错并暂停数据同步任务。 冲突忽略：在同步时发现表主键冲突，保留目标库主键记录。 冲突覆盖：在同步时发现表主键冲突，用源库主键记录覆盖目标库主键记录。
	同步操作类型	支持操作：Insert、Update、Delete、DDL。勾选“DDL自定义”，可以根据需要选择不同的DDL同步策略。详情请参考 设置 SQL 过滤策略。
同步对象选项	源实例库表对象	选择待同步的对象，支持基础库表、视图、存储过程和函数。高级对象的同步是一次性动作，仅支持同步在任务启动前源库中已有的高级对象，在任务启动后，新增的高级对象不会同步到目标库中。更多详情，请参考 同步高级对象。
	已选对象	支持库表映射（库表重命名），将鼠标悬浮在库名、表名上即显示编辑按钮，单击后可在弹窗中填写新的名称。 选择高级对象进行同步时，建议不要进行库表重命名操作，否则可能会导致高级对象同步失败。 支持同步 Online DDL 临时表（使用 gh-ost、pt-online-schema-change 工具），单击表的编辑按钮，在弹窗中即可选择临时表名。更多详情请参考 同步 Online DDL 临时表。

6. 在校验任务页面，完成校验并全部校验项通过后，单击**启动任务**。

如果校验任务不通过，可以参考 [校验不通过处理方法](#) 修复问题后重新发起校验任务。

失败：表示校验项检查未通过，任务阻断，需要修复问题后重新执行校验任务。

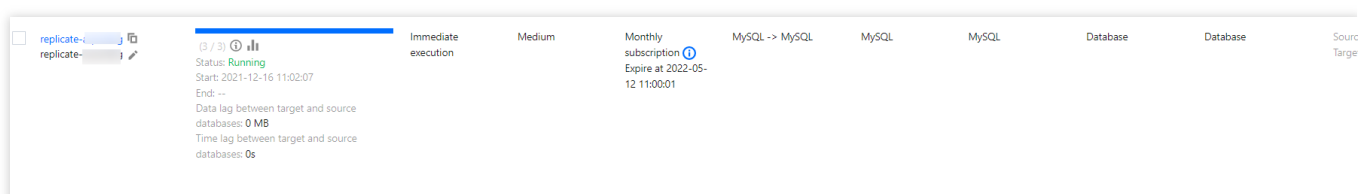
警告：表示检验项检查不完全符合要求，可以继续任务，但对业务有一定的影响，用户需要根据提示自行评估是忽略警告项还是修复问题再继续。



7. 返回数据同步任务列表，任务开始进入**运行中**状态。

说明：

选择**操作**列的**更多 > 结束**可关闭同步任务，请您确保数据同步完成后再关闭任务。



8. （可选）您可以单击任务名，进入任务详情页，查看任务初始化状态和监控数据。

MySQL 数据同步至 MySQL

最近更新时间：2024-02-02 16:17:17

TDSQL MySQL 同步至腾讯云数据库 MySQL 的要求和指导，与 [TDSQL MySQL 同步至 TDSQL MySQL](#) 的内容基本一致，请参考相关内容进行操作。

TDSQL-C MySQL 同步至 MySQL

最近更新时间：2024-02-02 16:17:41

TDSQL-C MySQL 同步至腾讯云数据库 MySQL 的要求和指导，与 TDSQL-C MySQL 同步至 TDSQL-C MySQL 的内容基本一致，请参考相关内容进行操作。

同步至 MariaDB

MySQL/MariaDB/Percona/TDSQL-C MySQL

同步至 MariaDB

最近更新时间：2024-07-08 17:33:33

本文为您介绍使用数据传输服务 DTS 从 MySQL、MariaDB、Percona 数据库同步数据至腾讯云数据库 MariaDB 的过程。

源数据库支持的部署类型如下：

自建 MySQL、腾讯云数据库 MySQL。

自建 MariaDB、腾讯云数据库 MariaDB。

自建 Percona。

因为 MySQL、MariaDB、Percona 同步至腾讯云数据库 MariaDB，三种场景的同步要求和操作步骤基本一致，本章节仅以 MariaDB 到 MariaDB 的数据同步为例进行介绍，其他场景请参考相关内容。

注意事项

DTS 在执行全量数据同步时，会占用一定源端实例资源，可能会导致源实例负载上升，增加数据库自身压力。如果您数据库配置过低，建议您在业务低峰期进行。

为了避免数据重复，请确保需要同步的表具有主键或者非空唯一键。

默认采用无锁方式，同步过程中对源库不加全局锁（FTWRL），仅对无主键的表加表锁，其他不加锁。

数据同步时，DTS 会使用执行同步任务的账号在源库中写入系统库 `__tencentdb__`，用于记录同步任务过程中的数据对比信息。

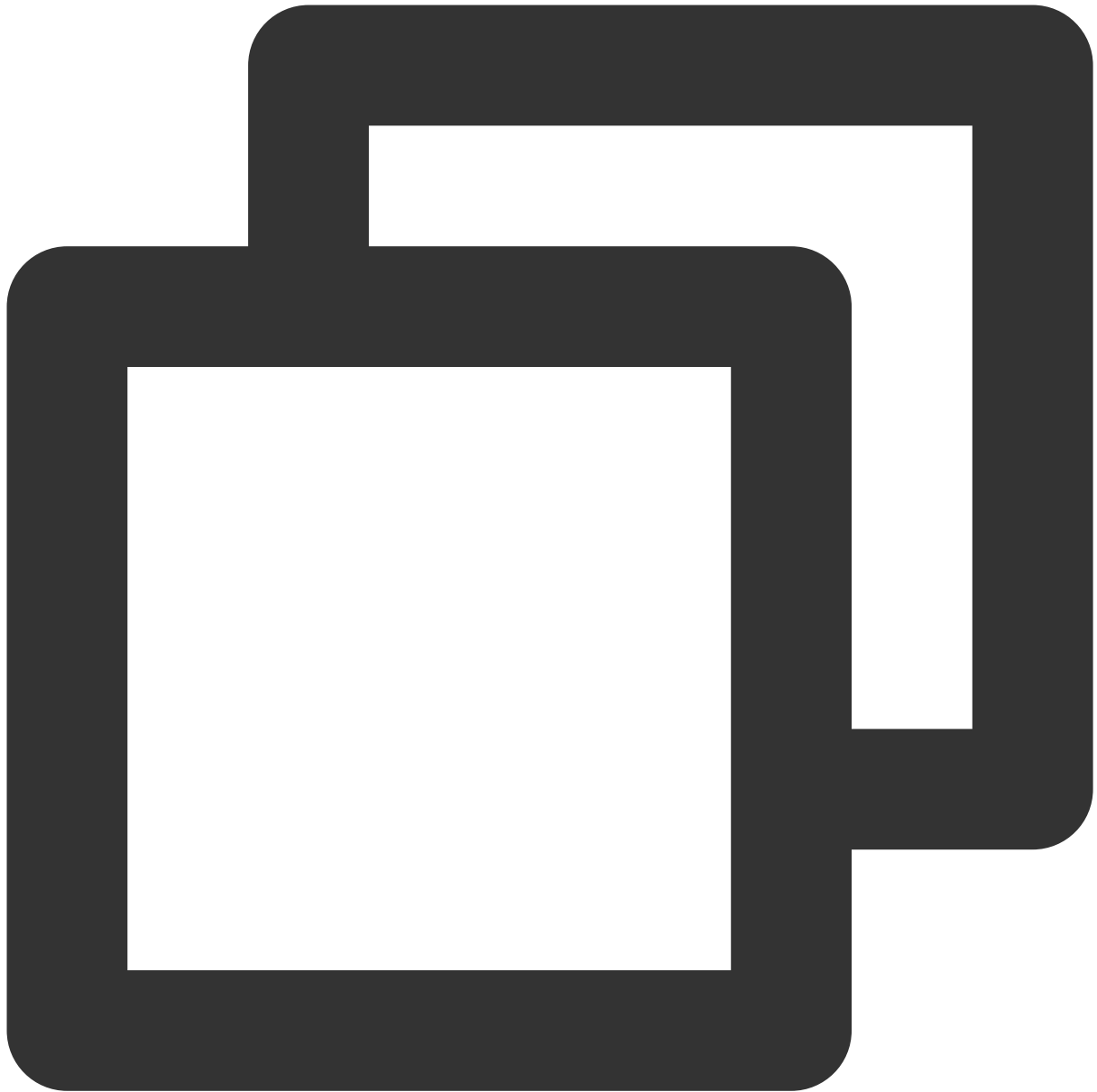
为保证后续数据对比问题可定位，同步任务结束后不会删除源库中的 `__tencentdb__`。

`__tencentdb__` 系统库占用空间非常小，约为源库存储空间的千分之一到万分之一（例如源库为 50GB，则 `__tencentdb__` 系统库约为 5MB - 50MB），并且采用单线程，等待连接机制，所以对源库的性能几乎无影响，也不会抢占资源。

前提条件

源数据库和目标数据库符合同步功能和版本要求，请参考 [数据同步支持的数据库](#) 进行核对。

需要具备源数据库的权限如下：



```
GRANT RELOAD,LOCK TABLES,REPLICATION CLIENT,REPLICATION SLAVE,SHOW VIEW,PROCESS,SEL  
GRANT ALL PRIVILEGES ON `__tencentdb__`.* TO '帐号'@'%';  
FLUSH PRIVILEGES;
```

需要具备目标数据库的权限：ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE USER, CREATE VIEW, DELETE, DROP, EVENT, EXECUTE, INDEX, INSERT, LOCK TABLES, PROCESS, REFERENCES, RELOAD, SELECT, SHOW DATABASES, SHOW VIEW, TRIGGER, UPDATE。

应用限制

只支持同步基础表、视图、存储过程和函数。

在同步视图、存储过程和函数时，DTS 会检查源库中 `DEFINER` 对应的 `user1`（`[DEFINER = user1]`）和同步账号 `user2` 是否一致，如果不一致，同步后 DTS 会修改 `user1` 在目标库中的 `SQL SECURITY` 属性，由 `DEFINER` 转换为 `INVOKER`（`[INVOKER = user1]`），同时设置目标库中 `DEFINER` 为同步账号 `user2`（`[DEFINER = 同步账号 user2]`）。如果源库中视图定义过于复杂，可能会导致任务失败。

源端如果是非 GTID 实例，DTS 不支持源端 HA 切换，一旦源端 MySQL 发生切换可能会导致 DTS 增量同步中断。只支持同步 InnoDB、MyISAM、TokuDB 三种数据库引擎，如果存在这三种以外的数据引擎表则默认跳过不进行同步。其中，源库如果存在压缩模式的 TokuDB 引擎数据，需要目标库同步支持压缩模式才可以同步，否则任务会报错。

相互关联的数据对象需要一起同步，否则会导致同步失败。常见的关联关系：视图引用表、视图引用视图、主外键关联表等。

增量同步过程中，若源库存在分布式事务或者产生了类型为 `STATEMENT` 格式的 Binlog 语句，则会导致同步失败。

源数据库为阿里云 MySQL，则阿里云 MySQL 5.6 版本待同步表不能存在无主键表，MySQL 5.7 及以后版本不限制。源数据库为 AWS MySQL，则 AWS MySQL 待同步表不能存在无主键表。

源数据库 Binlog 的 GTID 如果存在空洞，可能会影响同步任务的性能并导致任务失败。

不支持同时包含 DML 和 DDL 语句在一个事务的场景，遇到该情况任务会报错。

不支持 Geometry 相关的数据类型，遇到该类型数据任务报错。

不支持 `ALTER VIEW` 语句，遇到该语句任务跳过不同步。

操作限制

同步过程中请勿进行如下操作，否则会导致同步任务失败。

请勿修改、删除源数据库和目标数据库中用户信息（包括用户名、密码和权限）和端口号。

请勿在源库上执行分布式事务。

请勿在源库写入 Binlog 格式为 `STATEMENT` 的数据。

请勿在源库上执行清除 Binlog 的操作。

在同步增量阶段，请勿删除系统库表 `__tencentdb__`。

支持同步的 SQL 操作

操作类型	SQL 操作语句

DML	INSERT、UPDATE、DELETE
DDL	CREATE DATABASE、DROP DATABASE、ALTER DATABASE、CREATE TABLE、ALTER TABLE、DROP TABLE、TRUNCATE TABLE、RENAME TABLE、CREATE VIEW、DROP VIEW、CREATE INDEX、DROP INDEX

说明：

不支持同步涉及分区（Partition）的 DDL。

暂不支持 CREATE TABLE 表名 AS SELECT 语句。

环境要求

类型	环境要求
源数据库要求	源库和目标库网络能够连通。 实例参数要求： 源库 <code>server_id</code> 参数需要手动设置，且值不能设置为0。 源库表的 <code>row_format</code> 不能设置为 <code>FIXED</code>。 源库和目标库 <code>lower_case_table_names</code> 变量必须设置一致。 源库变量 <code>connect_timeout</code> 设置数值必须大于10。 Binlog 参数要求： 源端 <code>log_bin</code> 变量必须设置为 <code>ON</code>。 源端 <code>binlog_format</code> 变量必须设置为 <code>ROW</code>。 源端 <code>binlog_row_image</code> 变量必须设置为 <code>FULL</code>。 MySQL 5.6 及以上版本 <code>gtid_mode</code> 变量不为 <code>ON</code> 时会报警告，建议打开 <code>gtid_mode</code>。 不允许设置 <code>do_db</code>, <code>ignore_db</code>。 源实例为从库时，<code>log_slave_updates</code> 变量必须设置为 <code>ON</code>。 建议源库 Binlog 日志至少保留3天及以上，否则可能会因任务暂停/中断时间大于 Binlog 日志保留时间，造成任务无法续传，进而导致任务失败。 外键依赖： 外键依赖只能设置为 <code>NO ACTION</code>，<code>RESTRICT</code> 两种类型。 部分库表同步时，有外键依赖的表必须齐全。 环境变量 <code>innodb_stats_on_metadata</code> 必须设置为 <code>OFF</code>。
目标数据库要求	目标库的版本必须大于等于源库的版本。 目标库需要有足够的存储空间，如果初始类型选择“全量数据初始化”，则目标库的空间大小须是源库待同步库表空间的1.2倍以上。 目标库不能有和源库同名的表、视图等同步对象。 目标库 <code>max_allowed_packet</code> 参数设置数值至少为4M。

操作步骤

本场景操作步骤与 [MySQL 数据同步至 MySQL](#) 的一致，请参考 MySQL 同步场景的操作步骤。

TDSQL MySQL 同步至 MariaDB

最近更新时间：2024-02-02 16:21:17

TDSQL MySQL 同步至腾讯云数据库 MariaDB 的要求和指导，与 [TDSQL MySQL 同步至 TDSQL MySQL](#) 的内容基本一致，请参考相关内容进行操作。

同步至 TDSQL-C MySQL

TDSQL-C MySQL 同步至 TDSQL-C MySQL

最近更新时间：2024-02-02 16:27:16

本文介绍使用数据传输服务 DTS 从 TDSQL-C MySQL 数据库同步数据至 TDSQL-C MySQL 数据库的操作指导。

如下场景与 TDSQL-C MySQL 到 TDSQL-C MySQL 的数据同步要求一致，可参考本场景相关内容。

MySQL 到 TDSQL-C MySQL 的数据同步

TDSQL-C MySQL 到 MySQL 的数据同步

注意事项

DTS 在执行全量数据同步时，会占用一定源端实例资源，可能会导致源实例负载上升，增加数据库自身压力。如果您数据库配置过低，建议您在业务低峰期进行。

为了避免数据重复，请确保需要同步的表具有主键或者非空唯一键。

默认采用无锁方式，同步过程中对源库不加全局锁（FTWRL），仅对无主键的表加表锁，其他不加锁。

数据同步时，DTS 会使用执行同步任务的账号在源库中写入系统库 `__tencentdb__`，用于记录同步任务过程中的数据对比信息。

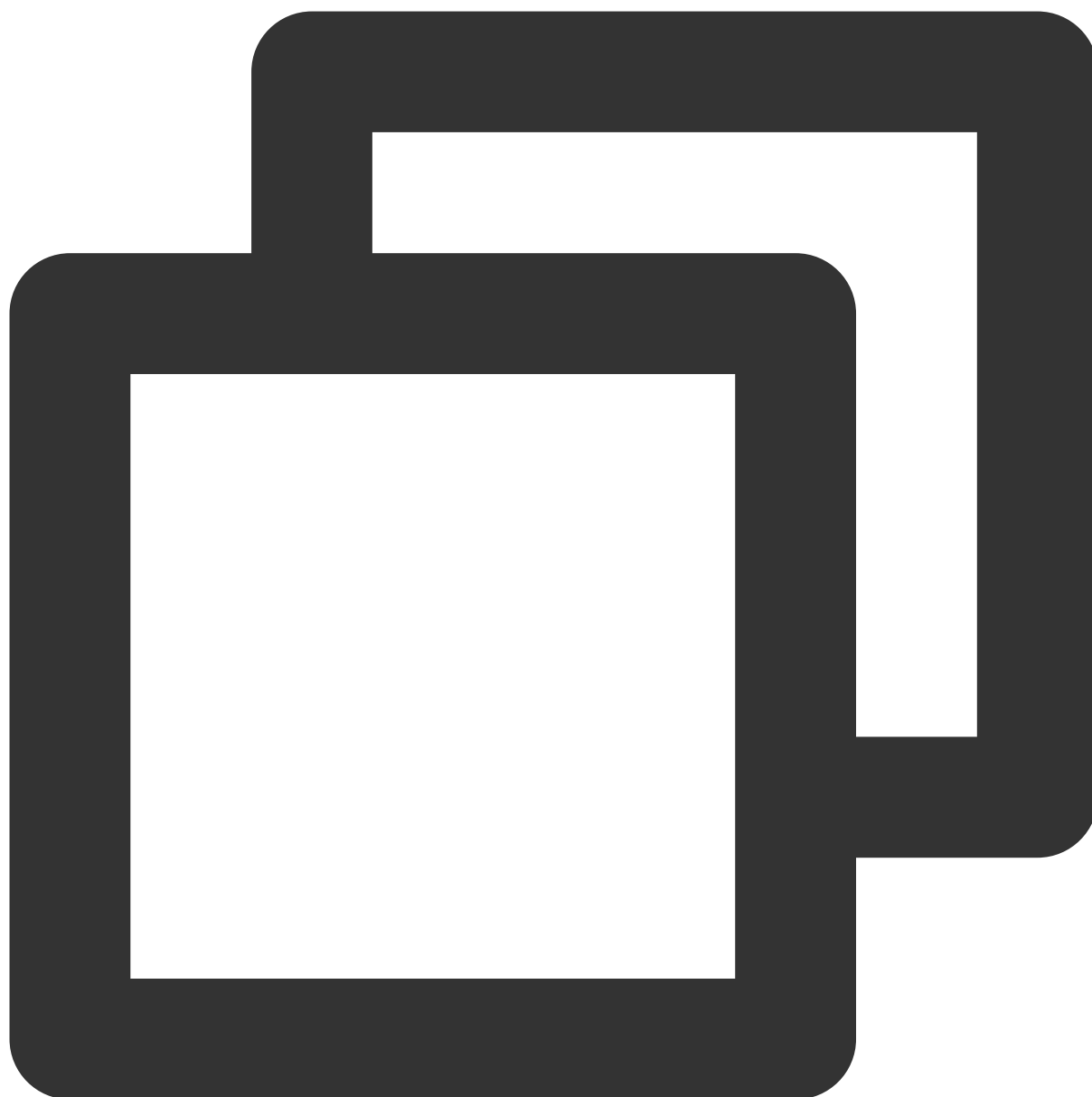
为保证后续数据对比问题可定位，同步任务结束后不会删除源库中的 `__tencentdb__`。

`__tencentdb__` 系统库占用空间非常小，约为源库存储空间的千分之一到万分之一（例如源库为50GB，则 `__tencentdb__` 系统库约为5MB - 50MB），并且采用单线程，等待连接机制，所以对源库的性能几乎无影响，也不会抢占资源。

前提条件

已 [创建 TDSQL-C MySQL](#)。

需要具备源数据库的权限如下：



```
GRANT RELOAD, LOCK TABLES, REPLICATION CLIENT, REPLICATION SLAVE, SHOW VIEW, PROCESS, SEL  
GRANT ALL PRIVILEGES ON `__tencentdb__`. * TO '迁移帐号'@'%';  
FLUSH PRIVILEGES;
```

需要具备目标数据库的权限：ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE USER, CREATE VIEW, DELETE, DROP, EVENT, EXECUTE, INDEX, INSERT, LOCK TABLES, PROCESS, REFERENCES, RELOAD, SELECT, SHOW DATABASES, SHOW VIEW, TRIGGER, UPDATE。

源数据库和目标数据库符合同步功能和版本要求，请参考 [数据同步支持的数据库](#) 进行核对。

应用限制

只支持同步基础表、视图、存储过程和函数。

在导出视图结构时，DTS 会检查源库中 `DEFINER` 对应的 `user1`（`[DEFINER = user1]`）和同步用户的 `user2` 是否一致，如果不一致，则会修改 `user1` 在目标库中的 `SQL SECURITY` 属性，由 `DEFINER` 转换为 `INVOKER`（`[INVOKER = user1]`），同时设置目标库中 `DEFINER` 为同步用户的 `user2`（`[DEFINER = user2]`）。如果源库中视图定义过于复杂，可能会导致任务失败。

源端如果是非 GTID 实例，DTS 不支持源端 HA 切换，一旦源端 TDSQL-C 发生切换可能会导致 DTS 增量同步中断。

只支持同步 InnoDB、MyISAM、TokuDB 三种数据库引擎，如果存在这三种以外的数据引擎表则默认跳过不进行同步。其中，源库如果存在压缩模式的 TokuDB 引擎数据，需要目标库同步支持压缩模式才可以同步，否则任务会报错。

增量同步过程中，若源库存在分布式事务或者产生了类型为 `STATEMENT` 格式的 Binlog 语句，则会导致同步失败。

源数据库为阿里云 MySQL，则阿里云 MySQL 5.6 版本待同步表不能存在无主键表，MySQL 5.7 及以后版本不限制。源数据库为 AWS MySQL，则 AWS MySQL 待同步表不能存在无主键表。

源数据库 Binlog 的 GTID 如果存在空洞，可能会影响同步任务的性能并导致任务失败。

不支持同时包含 DML 和 DDL 语句在一个事务的场景，遇到该情况任务会报错。

不支持 Geometry 相关的数据类型，遇到该类型数据任务报错。

不支持 `ALTER VIEW` 语句，遇到该语句任务跳过不同步。

操作限制

同步过程中请勿进行如下操作，否则会导致同步任务失败。

请勿修改、删除源数据库和目标数据库中用户信息（包括用户名、密码和权限）和端口号。

请勿在源库上执行分布式事务。

请勿在源库写入 Binlog 格式为 `STATEMENT` 的数据。

请勿在源库上执行清除 Binlog 的操作。

在同步增量阶段，请勿删除系统库表 `__tencentdb__`。

支持的 SQL 操作

操作类型	支持同步的 SQL 操作
DML	INSERT、UPDATE、DELETE

DDL	CREATE DATABASE、DROP DATABASE、ALTER DATABASE、CREATE TABLE、ALTER TABLE、DROP TABLE、TRUNCATE TABLE、RENAME TABLE、CREATE VIEW、DROP VIEW、CREATE INDEX、DROP INDEX
-----	--

说明：

不支持同步涉及分区（Partition）的 DDL。

暂不支持 CREATE TABLE 表名 AS SELECT 语句。

环境要求

类型	环境要求
源数据库要求	源库和目标库网络能够连通。 实例参数要求： 源库 <code>server_id</code> 参数需要手动设置，且值不能设置为0。 源库表的 <code>row_format</code> 不能设置为 <code>FIXED</code>。 源库和目标库 <code>lower_case_table_names</code> 变量必须设置一致。 源库变量 <code>connect_timeout</code> 设置数值必须大于10。 Binlog 参数要求： 源端 <code>log_bin</code> 变量必须设置为 <code>ON</code>。 源端 <code>binlog_format</code> 变量必须设置为 <code>ROW</code>。 源端 <code>binlog_row_image</code> 变量必须设置为 <code>FULL</code>。 TDSQL-C MySQL 5.6 及以上版本 <code>gtid_mode</code> 变量不为 <code>ON</code> 时会报警告，建议打开 <code>gtid_mode</code>。 不允许设置 <code>do_db</code>，<code>ignore_db</code>。 源实例为从库时，<code>log_slave_updates</code> 变量必须设置为 <code>ON</code>。 建议源库 Binlog 日志至少保留3天及以上，否则可能会因任务暂停/中断时间大于 Binlog 日志保留时间，造成任务无法续传，进而导致任务失败。 外键依赖： 外键依赖只能设置为 <code>NO ACTION</code>，<code>RESTRICT</code> 两种类型。 部分库表同步时，有外键依赖的表必须齐全。 环境变量 <code>innodb_stats_on_metadata</code> 必须设置为 <code>OFF</code>。
目标数据库要求	目标库的版本必须大于等于源库的版本。 目标库需要有足够的存储空间，如果初始类型选择“全量数据初始化”，则目标库的空间大小须是源库待同步库表空间的1.2倍以上。 目标库不能有和源库同名的表、视图等同步对象。 目标库 <code>max_allowed_packet</code> 参数设置数值至少为4M。

操作步骤

1. 登录 [数据同步购买页](#)，选择相应配置，单击**立即购买**。

参数	描述
计费模式	支持包年包月和按量计费。
源实例类型	选择 TDSQL-C MySQL，购买后不可修改。
源实例地域	选择源实例所在地域，购买后不可修改。
目标实例类型	选择 TDSQL-C MySQL，购买后不可修改。
目标实例地域	选择目的实例所在地域，购买后不可修改。
规格	请根据业务诉求选择规格，规格越高，性能越好。详情请参考 计费概述 。

2. 购买完成后，返回 [数据同步列表](#)，可看到刚创建的数据同步任务，刚创建的同步任务需要进行配置后才可以使

用。

3. 在数据同步列表，单击**操作**列的**配置**，进入配置同步任务页面。

4. 在配置同步任务页面，配置源端实例、帐号密码，配置目标端实例、帐号和密码，测试连通性后，单击**下一步**。

设置项	参数	描述
任务设置	任务名称	DTS 会自动生成一个任务名称，用户可以根据实际情况进行设置。
	运行模式	支持立即执行和定时执行两种模式。
	自动重试	设置后，迁移任务因网络异常等引起的任务临时中断，DTS 将在设置的时间范围内自动重试和恢复任务，不需要用户手动操作。 支持设置的时间范围为5分钟-720分钟。
源实例设置	源实例类型	购买时所选择的云数据库实例类型，不可修改。
	源实例地域	购买时选择的云数据库实例所在地域，不可修改。
	接入类型	根据实际情况选择，本场景选择“云数据库”。 公网：源数据库可以通过公网 IP 访问。 云主机自建：源数据库部署在 腾讯云服务器 CVM 上。 专线接入：源数据库可以通过 专线接入 方式与腾讯云私有网络打通。 VPN接入：源数据库可以通过 VPN 连接 方式与腾讯云私有网络打通。 云数据库：源数据库属于腾讯云数据库实例。 云联网：源数据库可以通过 云联网 与腾讯云私有网络打通。 私有网络 VPC：源数据和目标数据库都部署在腾讯云上，且有 私有网络 。 对于第三方云厂商数据库，一般可以选择公网方式，也可以选择 VPN 接入，专线或者云联网的方式，需要根据实际的网络情况选择。不同接入类型的准备工

		作请参考 准备工作概述 。
	实例 ID	源数据库实例 ID。
	帐号	源数据库帐号。
	密码	源数据库密码。
	连接方式	SSL 安全连接指 DTS 与数据库通过 SSL（Secure socket layer）安全连接，对传输链路进行加密。 选择 SSL 安全连接可能会增加数据库的连接响应时间，一般腾讯云内网链路相对较安全，无需开启 SSL 安全连接，采用公网/专线等传输方式，并且对数据安全要求较高的场景，需要开启 SSL 安全连接。选择 SSL 安全连接前，需要先在源数据库中开启 SSL 加密。
目标实例设置	目标实例类型	所选择的目标实例类型，不可修改。
	目标实例地域	选择的目标实例所在地域，不可修改。
	接入类型	选择目标数据库类型。
	实例 ID	目标数据库实例 ID。
	帐号	目标数据库帐号。
	密码	目标数据库密码。
	连接方式	SSL 安全连接指 DTS 与数据库通过 SSL（Secure socket layer）安全连接，对传输链路进行加密。 选择 SSL 安全连接可能会增加数据库的连接响应时间，一般腾讯云内网链路相对较安全，无需开启 SSL 安全连接，采用公网/专线等传输方式，并且对数据安全要求较高的场景，需要开启 SSL 安全连接。选择 SSL 安全连接前，需要先在源数据库中开启 SSL 加密。

5. 在设置同步选项和同步对象页面，将对数据初始化选项、数据同步选项、同步对象选项进行设置，在设置完成后单击**保存并下一步**。

说明

当**初始化类型**仅选择**全量数据初始化**，系统默认用户在目标库已经创建了表结构，不会进行表结构迁移，也不会校验源库和目标库是否有同名表，所以当用户同时在**已存在同名表**中选择**前置校验并报错**，则校验并报错功能不生效。

如果用户在同步过程中确定会对某张表使用 rename 操作（例如将 table A rename 为 table B），则**同步对象**需要选择 table A 所在的整个库（或者整个实例），不能仅选择 table A，否则系统会报错。

设置项	参数	描述
-----	----	----

数据初始化选项	初始化类型	结构初始化：同步任务执行时会先将源实例中表结构初始化到目标实例中。 全量数据初始化：同步任务执行时会先将源实例中数据初始化到目标实例中。仅选择全量数据初始化的场景，用户需要提前在目标库创建好表结构。 默认两者都勾选，可根据实际情况取消。
	已存在同名表	前置校验并报错：存在同名表则报错，流程不再继续。 忽略并继续执行：全量数据和增量数据直接追加目标实例的表中。
数据同步选项	冲突处理机制	冲突报错：在同步时发现表主键冲突，报错并暂停数据同步任务。 冲突忽略：在同步时发现表主键冲突，保留目标库主键记录。 冲突覆盖：在同步时发现表主键冲突，用源库主键记录覆盖目标库主键记录。
	同步操作类型	支持操作：Insert、Update、Delete、DDL。勾选“DDL自定义”，可以根据需要选择不同的 DDL 同步策略。详情请参考 设置 SQL 过滤策略 。
同步对象选项	源实例库表对象	选择待同步的对象，支持库级别和表及视图级别。
	已选对象	支持库表映射（库表重命名），将鼠标悬浮在库名、表名上即显示编辑按钮，单击后可在弹窗中填写新的名称。 选择高级对象进行同步时，建议不要进行库表重命名操作，否则可能会导致高级对象同步失败。
	是否同步 Online DDL 临时表	如果使用 gh-ost、pt-osc 工具对源库中的表执行 Online DDL 操作，DTS 支持将 Online DDL 变更产生的临时表迁移到目标库。 勾选 gh-ost，DTS 会将 gh-ost 工具产生的临时表名（`_表名_ghc`、`_表名_gho`、`_表名_del`）迁移到目标库。 勾选 pt-osc，DTS 会将 pt-osc 工具产生的临时表名（`_表名_new`、`_表名_old`）迁移到目标库。 更多详情请参考 同步 Online DDL 临时表。

6. 在校验任务页面，完成校验并全部校验项通过后，单击**启动任务**。

如果校验任务不通过，可以参考 [校验不通过处理方法](#) 修复问题后重新发起校验任务。

失败：表示校验项检查未通过，任务阻断，需要修复问题后重新执行校验任务。

警告：表示校验项检查不完全符合要求，可以继续任务，但对业务有一定的影响，用户需要根据提示自行评估是忽略警告项还是修复问题再继续。

7. 返回数据同步任务列表，任务开始进入**运行中**状态。

说明

选择**操作列**的**更多 > 结束**可关闭同步任务，请您确保数据同步完成后再关闭任务。

8. （可选）您可以单击任务名，进入任务详情页，查看任务初始化状态和监控数据。

MariaDB 同步至 TDSQL-C MySQL

最近更新时间：2024-02-02 16:28:39

MariaDB 同步至 TDSQL-C MySQL 的要求和指导，与 [TDSQL-C MySQL 同步至 TDSQL-C MySQL](#) 的内容基本一致，请参考相关内容进行操作。

TDSQL MySQL 同步至 TDSQL-C MySQL

最近更新时间：2023-09-05 15:10:00

TDSQL MySQL 同步至腾讯云数据库 TDSQL-C MySQL 的要求和指导，与 [TDSQL MySQL 同步至 TDSQL MySQL](#) 的内容基本一致，请参考相关内容进行操作。

MySQL 同步至 TDSQL-C MySQL

最近更新时间：2022-09-07 15:11:09

MySQL 同步至 TDSQL-C MySQL 的要求和指导，与 TDSQL-C MySQL 同步至 TDSQL-C MySQL 的内容基本一致，请参考相关内容进行操作。

同步至 TDSQL MySQL

TDSQL MySQL 同步至 TDSQL MySQL

支持能力

最近更新时间：2024-07-08 17:34:22

功能大类	功能子项或说明	支持能力
同步对象	-	普通对象表、索引、视图
同步类型	-	结构同步 全量同步（全量结束后自动接续增量） 增量同步
断点续传	-	全量导出部分支持（结构导出、无主键表导出阶段不支持续传）； 全量导入、增量支持。
任务通道关键操作	重试	支持
	暂停、恢复	支持
	修改同步配置（动态增减表对象）	不支持
	创建类似任务	支持
	限速	支持
同步方向	正向/反向	不涉及
	多对一（多实例合一/多表合一）	支持
	一对多	支持
	双向同步	不支持
结构同步	库表过滤	支持
	列过滤	不支持
	库表名映射	支持
	列名映射	不支持
	自定义分区（指定分区类型，分区	MySQL/MariaDB/Percona 同步到 TDSQL

	键)	MySQL，源端的分区表到目标端转化为单表。
	跳过同步表及订正	不支持
全量同步	Where 条件过滤	支持
增量同步	主键冲突策略	仅冲突覆盖
	DML 同步 (INSERT/UPDATE/DELETE)	支持
	DML 过滤	支持
	DDL 同步	CREATE DATABASE、DROP DATABASE、ALTER DATABASE。 CREATE TABLE、ALTER TABLE、DROP TABLE、TRUNCATE TABLE、RENAME TABLE。 CREATE VIEW、DROP VIEW。 CREATE INDEX、DROP INDEX。 不支持 CREATE TABLE 表名 AS SELECT 语句。
	DDL 过滤	支持
	Where 条件过滤	支持
	指定时间点同步	支持
	OnlineDDL	支持
数据一致性校验	整实例校验	支持
	行数校验	支持
	抽样校验	支持
	Where 条件过滤后支持一致性校验	不支持

使用说明

最近更新时间：2024-07-08 17:34:46

类别	说明
同步对象	1. 只支持同步基础表、视图，不支持同步函数、触发器、存储过程等对象。 2. 只支持同步 InnoDB 数据库引擎，如果存在其他数据引擎表则任务校验时会报错。 3. 相互关联的数据对象需要同时同步，否则会导致同步失败。 4. 源端 TDSQL MySQL 中对表的数量有限制，整个实例最多为5000个，超出后 DTS 任务会报错；同时，表的数量太多会导致源端的访问耗时变大，引起性能抖动和下降。 5. 增量同步阶段，源库的表名如含有“TDSQLagent”、“tdsql_sub”字符可能会被过滤或者引起同步异常，因为这些表名与 TDSQL 系统的临时表名相同，TDSQLagent 为扩容时的临时表，tdsql_sub 表为 hash-lish 和 hash-range 的子表，因此建议源端待同步的表名不要设置为这些类型。
同步功能	目前主键冲突处理策略只支持冲突覆盖，对于全量、增量阶段的主键数据冲突，都会进行冲突覆盖。
源库影响	1. DBbridge 在执行全量数据同步时，会占用一定源库资源，可能会导致源库负载上升，增加数据库自身压力。如果您数据库配置过低，建议您在业务低峰期进行。 2. 数据同步时，DBbridge 会使用执行同步任务的账号在源库中写入系统库`__tencentdb__`，用于记录事务标记 ID 等元信息，需要确保源库对`__tencentdb__`的读写权限。为保证后续数据对比问题可定位，同步任务结束后不会删除源库中的`__tencentdb__`。`__tencentdb__`系统库占用空间非常小，约为源库存储空间的千分之一到万分之一（例如源库为50GB，则`__tencentdb__`系统库约为5MB - 50MB），并且采用单线程，等待连接机制，所以对源库的性能几乎无影响，也不会抢占资源。 3. 默认采用无锁同步方式，全量数据导出阶段不会对源库加全局锁（FTWRL），仅对无主键的表加表锁。
操作限制	同步过程中请勿进行如下操作，否则会导致同步任务失败。 1. 在全量导出阶段，请勿在源库上执行库或表结构变更的 DDL 操作。 2. 同步任务过程中，请勿修改、删除源数据库和目标数据库中用户信息（包括用户名、密码和权限）和端口号。 3. 请勿在源库上执行清除 Binlog 的操作。
数据类型	1. 增量同步期间不支持修改主键，包括主键列、分区表分布键、对主键列的 COMMENT 注释，增删改列字段和长度。 2. 不支持 Geometry 相关的数据类型，遇到该类型数据任务报错。 3. 对于存在浮点类型的表，可能因为全量和增量同步的精度存在差异，导致同步结果的精度不一致。 4. 增量同步过程中，若源库产生了类型为 STATEMENT 格式的 Binlog 语句，则会导致同步失败。 5. 源端 TDSQL MySQL 为 MariaDB 10.1.x 内核时，使用 timestamp 类型不支持指定精度（例如 timestamp(3)），否则 DTS 任务会报错，需要去掉精度，然后重建任务。

事务	- 不支持同时包含 DML 和 DDL 语句在一个事务的场景，遇到该情况任务会报错。 - 源库为 TDSQL MySQL（内核 MariaDB 5.6）时，不支持 XA 事务，遇到 XA 事务任务会报错。
HA 切换和扩容	- 源库如果是非 GTID 数据库，DBbridge 不支持源端 HA 切换，一旦源端 TDSQL MySQL 发生切换可能会导致 DBbridge 增量同步中断。 - 源库如果是自建数据库，采用 SET 连接 TDSQL MySQL 的场景，当同步任务启动后，源端进行了增加或者删除 SET 节点，DBbridge 同步任务会报错，需要用户修改 DBbridge 中源数据库的 SET 配置信息（与源端实际的 SET 保持一致），然后重启任务，这样才能同步到新增或者删除 SET 的信息。
分区表同步	- 全量阶段支持同步一级/二级分区表，但分区语法需要符合 TDSQL MySQL 的规范，一级 Hash 分区表仅支持通过 shardkey 方式创建。 TDSQL MySQL 创建分区表的关键语法如下，详细语法请参考 TDSQL MySQL 创建表语法示例。 一级 Hash 分区：shardkey 一级 Range 分区：TDSQL_DISTRIBUTED BY RANGE 一级 List 分区：TDSQL_DISTRIBUTED BY LIST 一级 Hash 分区 + 二级 Range/List 分区：shardkey + PARTITION BY RANGE/LIST 一级 Range 分区 + 二级 Range/List 分区：TDSQL_DISTRIBUTED BY RANGE + PARTITION BY RANGE/LIST 一级 List 分区 + 二级 Range/List 分区：TDSQL_DISTRIBUTED BY LIST + PARTITION BY RANGE/LIST - 增量同步阶段，不支持进行密集的 create 二级分区表，再 drop 二级分区表，再 create 二级分区表的操作，否则可能会由于表类型冲突导致任务异常；先 drop 不存在的二级分区表，再 create 二级分区表，可能会导致死锁，任务无报错，需要手动解锁。 - TDSQL MySQL 同步到 MySQL/MariaDB/Percona 链路，如果源端待同步的库表中包含二级分区表，则同步到目标端后为单表。
一致性校验	- 数据一致性校验的范围，仅对比源数据库选择的库表对象、和同步到目标数据库的库表对象，如果用户在同步任务过程中向目标库进行数据写入，则这部分数据不包含在校验范围内，也不包括其他高级对象（如存储过程、函数、视图等）。如果同步任务配置中未选择“结构初始化”（表示不同步表结构），则进行一致性校验时也不校验表结构。 - 当前校验任务对 DDL 操作不感知，如果在同步过程中，用户对源库做了 DDL 操作，会出现校验结果不一致，需要用户重新发起校验任务才能得到准确的对比结果。 - 一致性校验任务中，DTS 单次查询源或者目标端数据的超时限制为10min，单次查询指每个分块校验会查询一次，行校验会查询一次等。如果单次查询时间超出10min（例如源端查询的表为大表时），则会导致校验任务失败。 - 同步任务配置中如果勾选了部分 DML，或者设置 Where 条件过滤，会造成源与目标库的数据不一致，所以不支持一致性校验，需要勾选全部的 DML、DDL 才可以进行一致性校验。 - 仅支持单向同步的数据校验；不支持多到一、一到多等复杂拓扑结构的数据校验。 - 同步任务进行如下配置，可能导致数据校验的结果不一致，请在创建校验任务时知晓。 数据初始化类型未勾选“全量数据初始化”，未勾选则可能存在源和目标的数据不一致，并最终导致数据校验的结果不一致。 - 数据一致性校验任务可能会增加源数据库实例的负载，因此请在业务低峰期进行操作。

8. 数据一致性校验的任务可以重复执行，但一个 DBbridge 任务在同一时刻只能发起一个一致性校验任务。
9. 完整对比、抽样对比需要校验的表必须具有主键或唯一键，否则将跳过不进行校验，行数校验不要求有主键或者唯一键。
10. 如果在数据一致性校验任务还未结束时，用户选择“结束”同步任务，则一致性校验任务会失败。
11. 一致性校验需要使用执行同步任务的账号，在源数据库中创建新表 `CRC_xxx_CMP_xxx`，用于记录同步任务过程中的数据对比信息。

同步操作指导

最近更新时间：2024-07-08 17:35:08

操作场景

本文为您介绍使用数据传输服务 DTS 从 TDSQL MySQL 数据库同步数据至 TDSQL MySQL 数据库的过程。支持的场景如下：

入腾讯云：自建 TDSQL MySQL 同步至腾讯云 TDSQL MySQL。

入腾讯云：腾讯云 TDSQL MySQL 同步至腾讯云 TDSQL MySQL。

出腾讯云：腾讯云 TDSQL MySQL 同步至自建 TDSQL MySQL。

注意事项

DTS 在执行全量数据同步时，会占用一定源端实例资源，可能会导致源实例负载上升，增加数据库自身压力。如果您数据库配置过低，建议您在业务低峰期进行。

为了避免数据重复，请确保需要同步的表具有主键或者非空唯一键。

数据同步时，DTS 会使用执行同步任务的账号在源库中写入系统库 `__tencentdb__`，用于记录同步任务过程中的数据对比信息。

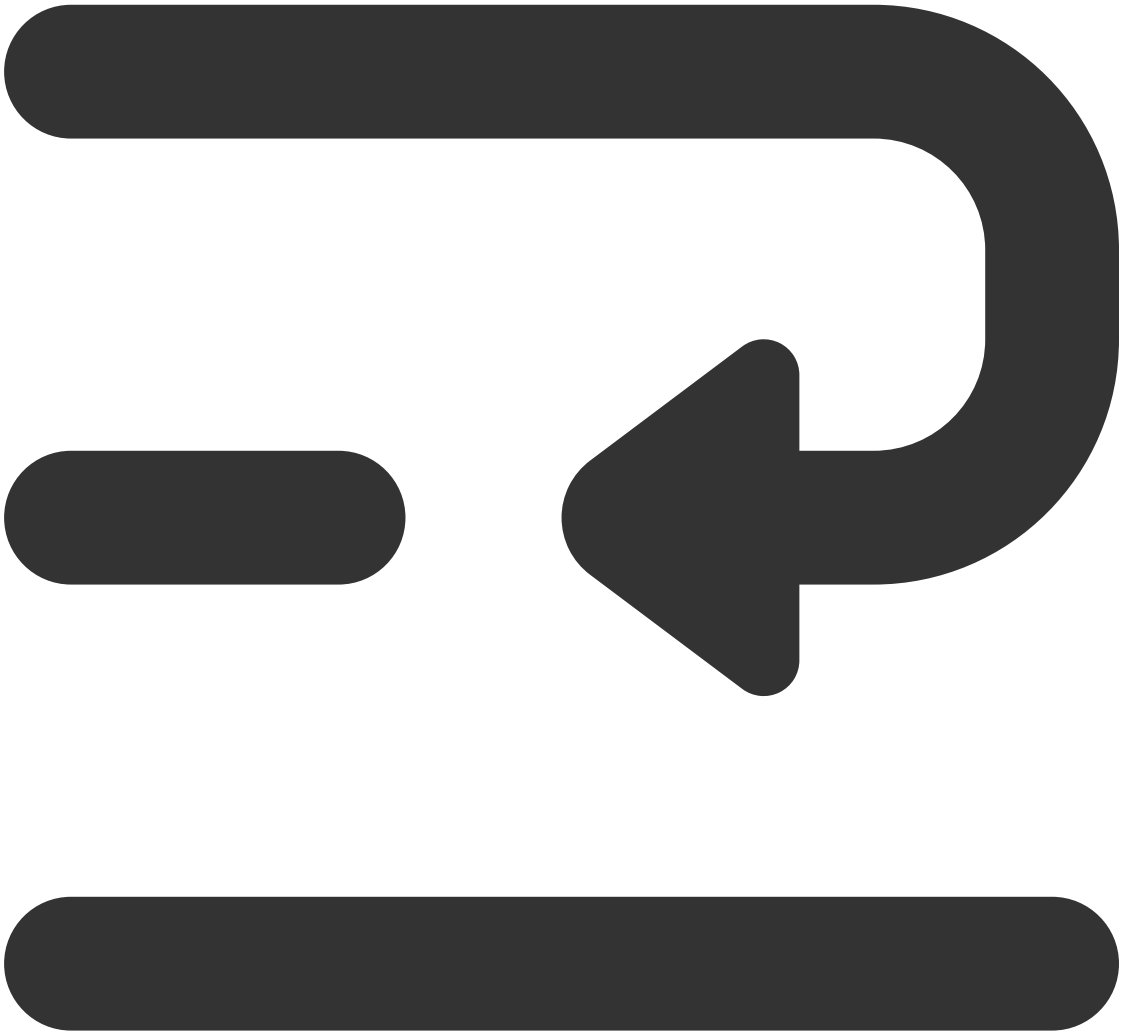
为保证后续数据对比问题可定位，同步任务结束后不会删除源库中的 `__tencentdb__`。

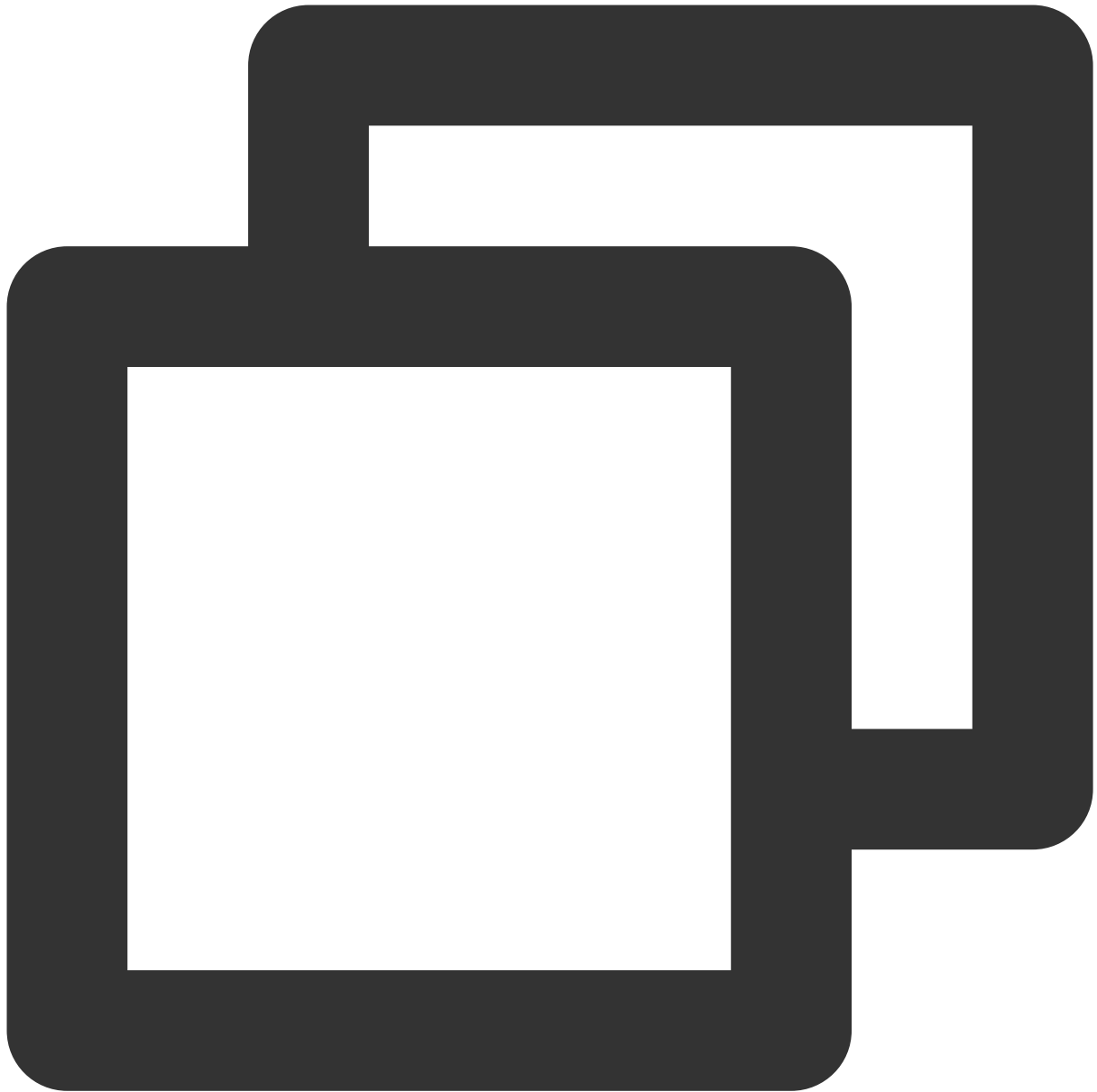
`__tencentdb__` 系统库占用空间非常小，约为源库存储空间的千分之一到万分之一（例如源库为50GB，则 `__tencentdb__` 系统库约为5MB - 50MB），并且采用单线程，等待连接机制，所以对源库的性能几乎无影响，也不会抢占资源。

前提条件

源数据库和目标数据库符合同步功能和版本要求，请参考 [数据同步支持的数据库](#) 进行核对。

需要具备源数据库的权限如下：





```
GRANT RELOAD,LOCK TABLES,REPLICATION CLIENT,REPLICATION SLAVE,SHOW VIEW,PROCESS,SEL  
GRANT ALL PRIVILEGES ON `__tencentdb__`.* TO '同步账号'@'%';  
FLUSH PRIVILEGES;
```

需要具备目标数据库的权限：ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE USER, CREATE VIEW, DELETE, DROP, EVENT, EXECUTE, INDEX, INSERT, LOCK TABLES, PROCESS, REFERENCES, RELOAD, SELECT, SHOW DATABASES, SHOW VIEW, TRIGGER, UPDATE。

环境要求

类型	环境要求
源数据库要求	源库和目标库网络能够连通。 实例参数要求： 源库 <code>server_id</code> 参数需要手动设置，且值不能设置为0。 源库表的 <code>row_format</code> 不能设置为 <code>FIXED</code>。 源库和目标库 <code>lower_case_table_names</code> 变量必须设置一致。 源库变量 <code>connect_timeout</code> 设置数值必须大于10。 Binlog 参数要求： 源端 <code>log_bin</code> 变量必须设置为 <code>ON</code>。 源端 <code>binlog_format</code> 变量必须设置为 <code>ROW</code>。 源端 <code>binlog_row_image</code> 变量必须设置为 <code>FULL</code>。 MySQL 5.6 及以上版本 <code>gtid_mode</code> 变量不为 <code>ON</code> 时会报警告，建议打开 <code>gtid_mode</code>。 不允许设置 <code>do_db</code>, <code>ignore_db</code>。 源实例为从库时，<code>log_slave_updates</code> 变量必须设置为 <code>ON</code>。 建议源库 Binlog 日志至少保留3天及以上，否则可能会因任务暂停/中断时间大于 Binlog 日志保留时间，造成任务无法续传，进而导致任务失败。 外键依赖： 外键依赖只能设置为 <code>NO ACTION</code>，<code>RESTRICT</code> 两种类型。 部分库表同步时，有外键依赖的表必须齐全。 环境变量 <code>innodb_stats_on_metadata</code> 必须设置为 <code>OFF</code>。
目标数据库要求	目标库为分布式数据库时，推荐提前手动创建分表，并规划 <code>shardkey</code>，否则 DTS 会按照源库的表样式来在目标库创建表，如果源库为单机实例，则目标库会创建为单表。 目标库的版本必须大于等于源库的版本。 目标库需要有足够的存储空间，如果初始类型选择“全量数据初始化”，则目标库的空间大小须是源库待同步库表空间的1.2倍以上。 目标库 <code>max_allowed_packet</code> 参数设置数值至少为4M。

操作步骤

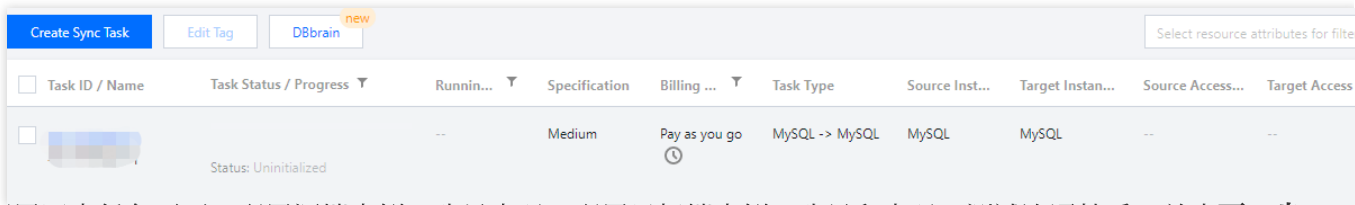
因为 TDSQL MySQL 迁移到 TDSQL MySQL 入云，和出云的操作步骤类似，如下以自建 TDSQL MySQL 迁移到腾讯云 TDSQL MySQL 为例进行介绍。

1. 登录 [数据同步购买页](#)，选择相应配置，单击**立即购买**。

参数	描述
计费模式	支持包年包月和按量计费。
源实例类型	选择 TDSQL MySQL，购买后不可修改。

	云数据库 TDSQL MySQL 分为三种内核版本（MySQL/MariaDB/Percona），这里无需区分内核版本，选择本身的数据库类型 TDSQL MySQL。
源实例地域	选择 DTS 迁移服务的源端地域，如果源数据库为腾讯云数据库，这里请选择源数据库所属地域，如果源库为自建数据库，这里请选择离源数据库最近的一个地域，以便 DTS 选择最优同步路径，降低同步时长。
目标实例类型	选择 TDSQL MySQL，购买后不可修改。
目标实例地域	选择目的实例所在地域，购买后不可修改。
规格	目前只支持标准版。

2. 购买完成后，返回 [数据同步列表](#)，可看到刚创建的数据同步任务，刚创建的同步任务需要进行配置后才可以使
- 用。
3. 在数据同步列表，单击操作列的**配置**，进入配置同步任务页面。



4. 在配置同步任务页面，配置源端实例、账号密码，配置目标端实例、账号和密码，测试连通性后，单击**下一步**。

设置项	参数	描述
任务设置	任务名称	DTS 会自动生成一个任务名称，用户可以根据实际情况进行设置。
	运行模式	支持立即执行和定时执行两种模式。
源实例设置	源实例类型	购买时所选择的源实例类型，不可修改。
	源实例地域	购买时选择的源实例所在地域，不可修改。
	接入类型	请根据您的场景选择，本场景选择“专线接入”或“VPN 接入”，该场景需要 配置 VPN 和 IDC 之间的互通 ，其他接入类型的准备工作请参考 准备工作概述 。 公网：源数据库可以通过公网 IP 访问。 云主机自建：源数据库部署在 腾讯云服务器 CVM 上。 专线接入：源数据库可以通过 专线接入 方式与腾讯云私有网络打通。 VPN 接入：源数据库可以通过 VPN 连接 方式与腾讯云私有网络打通。 云数据库：源数据库属于腾讯云数据库实例。 云联网：源数据库可以通过 云联网 与腾讯云私有网络打通。 私有网络 VPC：源数据和目标数据库都部署在腾讯云上，且有 私有网络 。如果需要使

	私有网络专线网关/VPN 网关	专线接入时只支持私有网络专线网关，请确认网关关联网络类型。 VPN 网关 ，请选择通过 VPN 网关 接入的 VPN 网关 实例。
	私有网络	选择私有网络专线网关和 VPN 网关 关联的私有网络和子网。
	主机 - Proxy	填入 Proxy 地址和端口。
	节点 - SET	填入 TDSQL MySQL 分片节点 IP，每分片需输入一个节点 IP，多个节点请换行输入。 源库如果是自建数据库，首次配置 SET 节点，连接保存后，无法再增加或者删除，如果用户需要增加或者删除 SET 节点，需要重新建立同步任务。
	账号	源实例账号，账号权限需要满足要求。
	密码	源实例账号的密码。
目标实例设置	目标实例类型	购买时选择的目标实例类型，不可修改。
	目标实例地域	购买时选择的目标实例地域，不可修改。
	接入类型	根据您的场景选择，本场景选择“云数据库”。如果源实例接入类型选择了 云数据库 ，目标实例接入类型这里可以选择 公网/云主机自建/专线接入/VPN 接入/云数据库/云联网 等方式。
	实例 ID	选择目标实例 ID。
	账号	目标实例账号，账号权限需要满足要求。
	密码	目标实例账号的密码。

5. 在设置同步选项和同步对象页面，将对数据初始化选项、数据同步选项、同步对象选项进行设置，在设置完成后单击**保存并下一步**。

说明：

如果用户在同步过程中确定会使用 **gh-ost**、**pt-osc** 等工具对某张表做 **Online DDL**，则**同步对象**需要选择这个表所在的整个库（或者整个实例），不能仅选择这个表，否则无法同步 **Online DDL** 变更产生的临时表数据到目标数据库。如果用户在同步过程中确定会对某张表使用 **rename** 操作（例如将 **table A** rename 为 **table B**），则**同步对象**需要选择 **table A** 所在的整个库（或者整个实例），不能仅选择 **table A**，否则 **rename** 操作后，**table B** 的数据不会同步到目标库。

库表映射：在已选对象中，鼠标放在右侧将出现编辑按钮，单击后可在弹窗中填写映射名。

设置项	参数	描述
-----	----	----

数据初始化选项	初始化类型	结构初始化：同步任务执行时会先将源实例中表结构初始化到目标实例中。 全量数据初始化：同步任务执行时会先将源实例中数据初始化到目标实例中。 仅选择全量数据初始化的场景，用户需要提前在目标库创建好表结构。 默认两者都勾上，可根据实际情况取消。仅选择“全量数据初始化”时，用户需要提前在目标库创建好表结构。
	已存在同名表	前置校验并报错：存在同名表则报错，流程不再继续。 忽略并继续执行：全量数据和增量数据直接追加目标实例的表中。
数据同步选项	冲突处理机制	冲突覆盖：在同步时发现表主键冲突，用源库主键记录覆盖目标库主键记录。
	同步操作类型	支持操作：Insert、Update、Delete、DDL。
同步对象选项	源实例库表对象	选择待同步的对象，支持库级别和表级别。
	已选对象	展示已选择的同步对象，支持库表映射。

6. 在校验任务页面，完成校验并全部校验项通过后，单击**启动任务**。

如果校验任务不通过，可以参考 [校验不通过处理方法](#) 修复问题后重新发起校验任务。

失败：表示校验项检查未通过，任务阻断，需要修复问题后重新执行校验任务。

警告：表示校验项检查不完全符合要求，可以继续任务，但对业务有一定的影响，用户需要根据提示自行评估是忽略警告项还是修复问题再继续。

7. 返回数据同步任务列表，任务开始进入**运行中**状态。

说明：

选择**操作**列的**更多 > 结束**可关闭同步任务，请您确保数据同步完成后再关闭任务。

8. （可选）您可以单击任务 ID，进入任务详情页，查看任务初始化状态和监控数据。

MySQL/MariaDB/Percona 同步至 TDSQL MySQL

最近更新时间：2024-02-02 16:35:26

MySQL、MariaDB、Percona、TDSQL-C MySQL 同步至腾讯云数据库 TDSQL MySQL 的要求和指导，与 [TDSQL MySQL 同步至 TDSQL MySQL](#) 的内容基本一致，请参考相关内容进行操作。

MySQL/MariaDB/Percona 同步至 TDSQL TDSore

最近更新时间：2023-11-16 17:22:31

DTS 支持同步到 TDSQL TDSore，源端支持的类型如下：

自建 MySQL、第三方云厂商 MySQL、腾讯云数据库 MySQL。

自建 MariaDB、腾讯云数据库 MariaDB。

自建 Percona。

MySQL/MariaDB/Percona 数据同步到 TDSQL TDSore 的操作指导，与 [MySQL/MariaDB/Percona 同步至 MySQL](#)

主要差异如下：

不支持外键依赖，如果选择的同步对象中包含外键依赖数据，会出现校验报错，需要去勾选相关对象。

其他链路能力相同，请参考 [MySQL/MariaDB/Percona 同步至 MySQL](#) 执行。

同步至 TDSQL TDStore

MySQL/MariaDB/Percona/TDSQL-C MySQL

同步至 TDSQL TDStore

最近更新时间：2024-08-13 17:25:59

MySQL/MariaDB/Percona/TDSQL-C MySQL 数据同步到 TDSQL TDStore 的操作指导，与 [MySQL/MariaDB/Percona 同步至 MySQL](#) 内容基本一致，请参考相关内容执行。

MySQL 系列同步高级操作

同步高级对象

最近更新时间：2023-03-03 15:00:31

操作场景

DTS 支持同步的高级对象有函数、存储过程。高级对象的同步是一次性动作，仅支持同步在任务启动前源库中已有的高级对象，在任务启动后，源库新增的高级对象不会同步到目标库中。

说明

当前支持高级对象同步的场景为 MySQL、TDSQL-C MySQL、MariaDB、Percona 之间的数据同步。

注意事项

同步高级对象时，建议不要进行库表重命名操作，否则可能会导致高级对象同步失败。

高级对象同步失败并不会影响整个同步任务，所以整个同步任务成功不能保证高级对象也同步成功，建议用户在同步任务完成后，自行在进度详情页查看高级对象是否同步成功。

在同步存储过程和函数时，DTS 会检查源库中 `DEFINER` 对应的 `user1`（`[DEFINER = user1]`）和同步账号 `user2` 是否一致，如果不一致，同步后 DTS 会修改 `user1` 在目标库中的 `SQL SECURITY` 属性，由 `DEFINER` 转换为 `INVOKER`（`[INVOKER = user1]`），同时设置目标库中 `DEFINER` 为同步账号 `user2`（`[DEFINER = 同步账号 user2]`）。

对于跨版本的同步，如果源库中高级对象设置的 `sql_mode`，目标库不支持，高级对象同步到目标库后会将 `sql_mode` 改为 `NO_AUTO_VALUE_ON_ZERO`。

如果勾选的高级对象定义中存在 `' '`（转义），会碰到 [MySQL 自身限制](#)，从而导致迁移任务失败，建议不要勾选这类高级对象。

操作步骤

1. 在 [数据同步任务](#) 的 **设置同步选项及同步对象** 页面，设置高级对象功能。默认会勾选高级对象，如果不需要同步高级对象，可以取消勾选。



2. 校验任务中会增加高级对象的检查项，具体请参考 [高级对象检查](#)。

创建数据一致性校验

最近更新时间：2023-03-02 11:40:31

操作场景

数据一致性校验，即 DTS 对数据同步的源库和目标库的表数据进行对比，并给出对比结果和不一致详情，方便用户快速对不一致数据进行处理。数据一致性校验任务是独立进行的，不会影响源数据库的正常业务，也不会影响 DTS 的任务。

说明

当前支持数据一致性校验的同步链路如下：

MySQL/MariaDB/Percona/TDSQL-C MySQL > MySQL

MySQL/MariaDB/Percona > MariaDB

MySQL/MariaDB/TDSQL-C MySQL > TDSQL-C MySQL

注意事项

数据一致性校验的范围，仅对比源数据库选择的库表对象、和同步到目标数据库的库表对象，如果用户在同步任务过程中向目标库进行数据写入，则这部分数据不包含在校验范围内，也不包括其他高级对象（如存储过程、函数）、视图等。

数据一致性校验任务可能会增加源数据库实例的负载，因此请在业务低峰期进行操作。

数据一致性校验的任务可以重复执行，但一个 DTS 实例在同一时刻只能发起一个数据一致性校验任务。

需要校验的表必须具有主键或唯一键，否则 DTS 将跳过，不进行校验。

如果在数据一致性校验任务还未结束时，用户选择**结束**同步任务，则数据一致性校验任务会失败。

因为一致性校验需要在源数据库中创建新库 `__tencentdb__`，并在该库下写入 CheckSum 表，所以源数据库为只读时将会跳过一致性校验。

约束限制

当前校验任务对 DDL 操作不感知，如果在同步过程中，用户对源库做了 DDL 操作，会出现校验结果不一致，需要用户重新发起校验任务才能得到准确的对比结果。

仅支持单向同步、双向同步的数据校验；不支持多到一、一到多、环形、星形同步等复杂拓扑结构的数据校验。

同步任务配置中如果勾选了部分 DML、DDL，或者进行 Where 条件过滤，会造成源与目标库的数据不一致，所以不支持一致性校验，需要勾选全部的 DML、DDL 才可以进行一致性校验。

同步任务进行如下配置，可能导致数据校验的结果不一致，请在创建校验任务时知晓。

数据初始化类型未勾选“全量数据初始化”，未勾选则可能存在源和目标的数据不一致，并最终导致数据校验的结果不一致。

主键冲突勾选了“冲突忽略”，勾选后发生冲突后可能会导致源和目标的数据不一致，并最终导致数据校验的结果不一致。

在同步一致性校验功能发布之前（2023年1月12日之前）已经存在的存量任务，因 DTS 版本过低，暂不支持直接创建校验任务，需要[提交工单](#)先升级版本后才能创建。

实现原理

DTS 对 MySQL 系数据库的一致性校验是基于 row 模式（binlog_format=row），row 模式可以实现 master 和 slave 的正确复制，保障数据的安全性。

1. 在源库创建校验库 `__tencentdb__.Checksums`，用于存储同步任务过程中的数据对比信息。
2. 选择待校验表的非空唯一键作为校验固定字段。
3. 计算源数据库的校验值 `crc1` 和行数 `count1`，并写入到源库的 `__tencentdb__.Checksums` 中。
`crc` 的计算方法类似于分块校验，根据校验固定字段，选取一个固定的范围（例如选取A表中，主键从1-1000的数据），将这些数据按行拼接起来计算出一个 `crc`，这样每个分块数据得到一个 `crc`，同时计算源库总的的数据得到 `crc1`。
4. DTS 解析 binlog 中的 row 模式数据，还原出在源库写入校验值的 SQL，然后将这个 SQL 在目标库上重放。目标库上使用与源库相同的变量计算校验值和行数，得到目标库的 `crc2` 和 `count2`。
5. 对比源和目标库的校验值和行数，显示对比结果。

创建数据一致性校验

1. 登录 [DTS 控制台](#)。
2. 在数据同步页面，选择需要校验的同步任务，选择 **操作 > 更多**，然后单击**创建数据一致性校验**。
3. 单击**创建数据一致性校验**。

说明

一致性校验需要在“目标与源库数据差距”小于100MB时，才可以创建。如果界面按钮呈灰色，则同步任务状态不满足条件，如任务配置中勾选了部分 DML 或DDL、设置了 Where 条件过滤、任务失败、源和目标的数据差距大于100MB、同步拓扑结构为复杂拓扑等。

4. 在弹出的对话框中，单击**确定**。
5. 配置数据一致性校验参数后，单击**创建并启动一致性校验**。

参数	说明
校验对象	全部同步对象：校验范围为同步任务勾选的全部对象。 自定义选择：在勾选的同步对象中，选择进行校验的对象。

对比类型	完整对比：对所选检验对象的完整数据进行校验。 抽样对比：对所选检验对象抽选一定的比例进行校验，抽样比例支持10%，20%，30%.....90%。 行数对比：对所选校验对象，仅对比数据行数。
线程数选择	设置范围为1~5，请根据实际情况选择，提高线程数可加快一致性校验速度，但也会对源和目标库造成负载。

查看数据一致性校验结果

- 在**数据同步**页面，选择需要查看的同步任务，选择 **操作 > 更多**，然后单击**创建数据一致性校验**。
- 在操作列单击**查看**，即可查看校验结果。

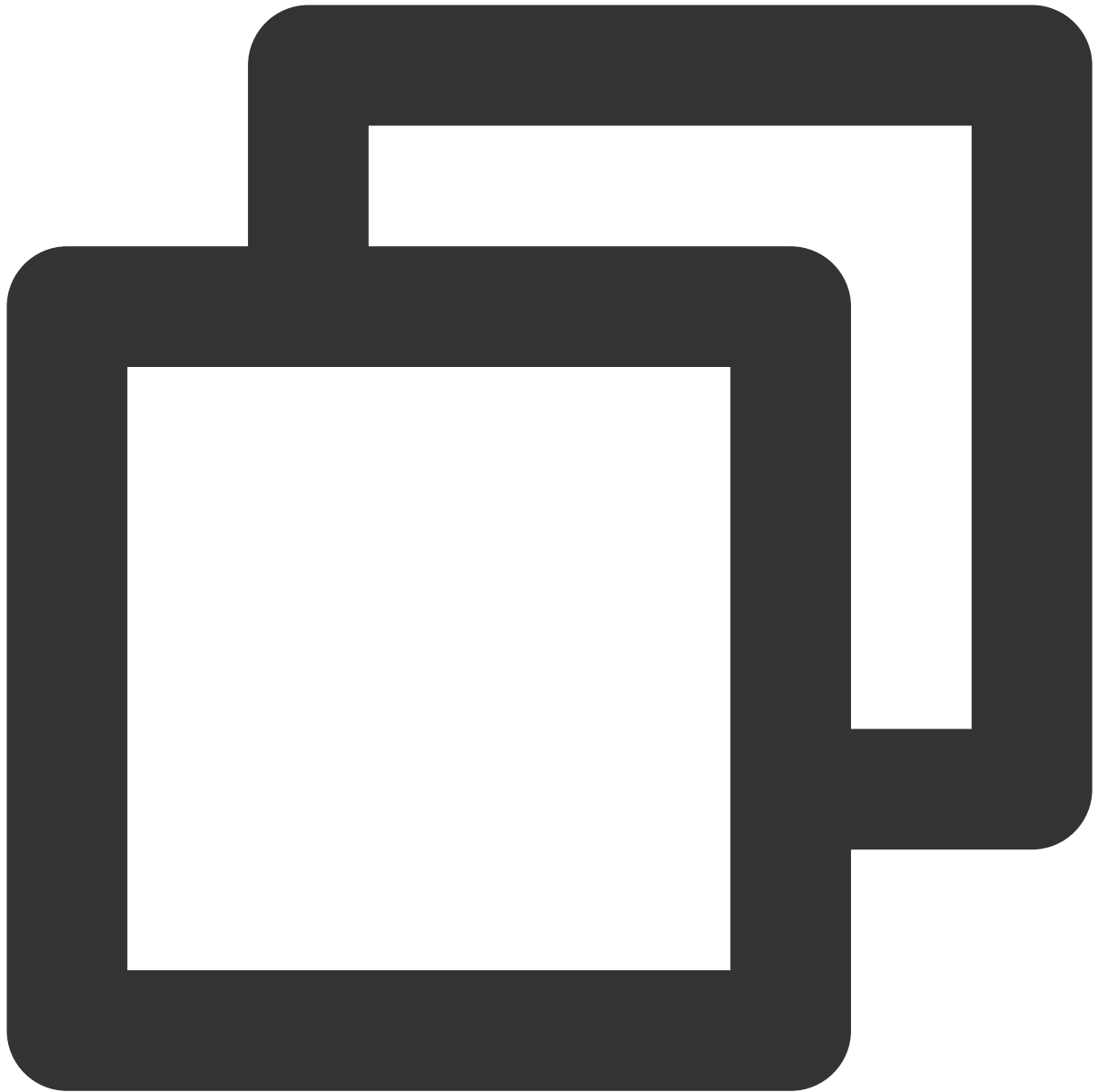
校验一致的结果示例：

可以查看预估表总数、已检查表数量、不一致表数量、不一致分块数。其中预估表总数，为预计校验的表总数的估算值，与最终实际校验表总数会有少许差异，因为准确提供预计校验表总数会影响整体校验性能。

未检查表原因为：无主键或者非空唯一键、空表、不支持的引擎类型、表不存在。

针对不一致性的结果，需要用户手动对比源数据库和目标数据库的对应内容。请按照界面提示的**数据库、数据表、索引名称、索引上边界、索引下边界**这些参数定位到具体位置进行对比。

参考操作如下：



1. 登录源数据库，查询提示的索引范围。
`select * from table_name where col_index >=1 and <=5;`
2. 登录目标数据库，查询提示的索引范围。
3. 对比目标和源的数据差距。

设置 SQL 过滤策略

最近更新时间：2022-09-02 18:32:32

操作场景

在配置同步任务时，可以选择 SQL 同步策略，设置后，只有满足设置条件的 SQL 数据才会同步到目标数据库，这样用户可以根据场景，灵活设置多个数据库之间的同步方案，或者进行数据表的拆分。

DTS 支持如下维度的 SQL 过滤：

DML 过滤：支持类型 Insert、Update、Delete。

DDL 过滤：支持选择具体的 DDL 操作，如 CREATE TABLE、CREATE VIEW、DROP INDEX 等。

Where 条件过滤：支持对单个表设置自定义过滤条件。

说明：

当前支持在如下数据同步链路中进行 DDL 过滤和 Where 条件过滤。

MySQL > MySQL/MariaDB/TDSQL-C MySQL

MariaDB > MySQL/MariaDB

Percona > MySQL/MariaDB

TDSQL-C MySQL > MySQL/TDSQL-C MySQL

约束限制

Where 条件过滤支持对单张表进行设置，需要对多张表进行过滤时，对每张表分别进行设置即可。

操作步骤

1. 在 [同步任务](#) 的 **设置同步选项和同步对象** 页面中，选择不同的 SQL 策略。

DML：支持类型 Insert、Update、Delete。

DDL：

勾选 **DDL**，不勾选 **DDL 自定义** 开关，则同步源数据库的所有 DDL 到目标数据库。

勾选 **DDL**，同时勾选 **DDL 自定义** 开关，可以自定义选择具体的 DDL 策略，只有勾选的策略会同步到目标库。

不勾选 **DDL**，则所有的 DDL 都不会同步到目标库。

1 Set source and target databases

2 Set sync options and objects

3 Verify task

Data Initialization Option

Initialization Type

Structure Initialization

Full data initialization

If Target Already Exists *

Precheck and report error

Ignore and execute

Data Sync Option

Primary Key Conflict Resolution *

Report

Ignore

Overwrite

SQL Type

DML

Insert

Update

Delete

DDL

DDL

Custom DDL *

Database

Create

Drop

Alter

Table

Create

Drop

Alter

Truncate

Rename

View

Create

Drop

Index

Create

Drop

Sync Object Option

Advanced Migration Object ⓘ

Procedure

Function

Advanced objects can only be copied once, which means you cannot copy new objects once the task is started.

ⓘ

 Up to 200 results can be displayed. If the objects you need are not shown in the result list, you can search them by object name.

Source Database Object

Search database name, supporting fuzzy match

ⓘ 1 database in total, with 1 displayed

More

▶ ☒ db-dst

Refresh

Select all

Clear

Selected Object ⓘ

Globally search for original object name

☐ db-dst (Entire database sele

Unfold all

Fold all

Select all

Clear

Rever

2. 在同步对象选项右侧的已选对象中，选择对应的表，单击右边的编辑按钮，设置 Where 条件过滤。

版权所有：腾讯云计算（北京）有限责任公司

第83 共300页

说明：

Where 条件过滤设置规则说明：

对于 **INSERT** 操作，需要插入的数据满足条件过滤规则才能生效，对于 **DELETE** 操作，需要删除的数据满足条件过滤规则才能生效，对于 **UPDATE** 操作，则需要更新前后的数据都满足条件过滤规则才能生效。

输入的规则必须是一个合法的 **BOOL** 表达式，并且表达式的规则相对于 **MySQL** 更为严格，一些在 **MySQL** 中支持但可能产生 **WARNING** 的语法（如字符串同数字比较，`c1 + c2 < "abc"`），此处不支持。逻辑运算、算术运算、比较运算规则和优先级同 **MySQL** 一致，支持通过括号改变运算优先级，操作数中有 **NULL** 时，运算规则也同 **MySQL** 一致。**DTS** 系统会对输入的条件过滤规则进行验证，如果不合法会给出提醒。

基本的运算规则如下：

支持引入列名作为变量。

支持逻辑运算（**NOT**，**AND**，**OR**，**XOR**，**&&**，**||**）。

支持数字类型（有符号/无符号的整型 **TINYINT**、**SMALLINT**、**MEDIUMINT**、**INT**、**BIGINT**，浮点类型 **FLOAT** 和 **DOUBLE**，精确类型 **DECIMAL**）及其算术运算（**+**、**-**、*****、**/**、**%**、**DIV**、**MOD**）和比较运算（**=**、**!=**、**>**、**<**、**>=**、**<=**、**<>**、**<=>**）。

支持字符串类型（**CHAR**，**VARCHAR**）及其比较运算（二进制比较）。

支持日期类型（**DATE**，**DATETIME**，**TIMESTAMP**）及其比较运算。

支持时间类型（**TIME**）及其比较运算。支持日期/时间类型变量同字符串比较，此时字符串被转换为日期/时间类型常量，按照日期/时间比较规则进行。

输入 **TIMESTAMP** 默认时区为 **UTC+0** 时区，比较时会转换为 **UTC+0** 时区进行。

示例：输入过滤规则为 `'c1 > "2016-10-01 09:00:00"`，其中 **c1** 为 **TIMESTAMP** 类型的列。则 `"2016-10-01 09:00:00"` 解析为 **UTC+0** 时区时间，相当于 `"2016-10-01 09:00:00 +00:00"`，比较时 **c1** 也会转换为 **UTC+0** 时区时间。

3. 单击**确定**。

4. 单击**保存并下一步**进行后续同步任务流程。

库表重命名

最近更新时间：2022-12-01 10:46:49

操作场景

库表重命名，也称库表映射。在数据同步场景中，用户可以通过库表重命名功能对同步到目标库中的库和表进行重新命名，以避免源库和目标库中的库表因为同名引起冲突。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，创建同步任务。
2. 在“设置同步源和目标数据库”步骤中，在选择同步对象右侧“已选对象”中，将鼠标悬浮在需要修改的对象上，即可显示编辑按钮。

1 Set source and target databases

2 Set sync options and objects

3 Verify task

Data Initialization Option

Initialization Type

Structure Initialization

Full data initialization

If Target Already Exists *

Precheck and report error

Ignore and execute

Data Sync Option

Primary Key Conflict Resolution *

Report

Ignore

Overwrite

SQL Type

DML

Insert

Update

Delete

DDL

DDL

Custom DDL *

Sync Object Option

Advanced Migration Object ⓘ

Procedure

Function

Advanced objects can only be copied once, which means you cannot copy new objects once the task is started.

ⓘ

Up to 200 results can be displayed. If the objects you need are not shown in the result list, you can search them by object

Source Database Object

Search database name, supporting fuzzy match

ⓘ 1 database in total, with 1 displayed

More

▶

✔

db-dst

Refresh

Select all

Clear

Selected Object ⓘ

Globally search for original obj

db-dst (Entire data

▶

◀

Unfold all

Fold all

Select all

Cle

3. 在弹出的对话框，修改对象的名称，单击**确定**。

库名映射：在弹出的对话框，设置该数据库在目标实例中的名称。

表名映射：在弹出的对话框，设置该数据表在目标实例中的名称。

版权所有：腾讯云计算（北京）有限责任公司

第86 共300页

同步 Online DDL 临时表

最近更新时间：2023-11-16 17:19:25

操作场景

使用 gh-ost、pt-online-schema-change（下文简称 pt-osc）等工具对源库中的表做 Online DDL，需要将 Online DDL 变更产生的临时表同步到目标库。

DTS 支持在选择同步对象时，提前关联对象表的临时表名，在后续源库产生临时表时一并进行同步。

使用 gh-ost 工具对表 `表名` 做 Online DDL，DTS 支持同步临时表 `_表名_ghc`、`_表名_gho`、`_表名_del` 到目标库。

使用 pt-osc 工具对表 `表名` 做 Online DDL，DTS 支持同步临时表 `_表名_new`、`_表名_old` 到目标库。

适用范围

同步 Online DDL 临时表支持的链路范围如下：

MySQL/MariaDB/Percona/TDSQL-C MySQL > MySQL

MySQL/MariaDB/Percona > MariaDB

MySQL/MariaDB/TDSQL-C MySQL > TDSQL-C MySQL

MySQL/MariaDB/Percona/TDSQL-C MySQL > Kafka

MySQL/MariaDB/Percona > TDSQL TdStore

约束限制

表映射（表重命名）功能与迁移 Online DDL 临时表功能冲突，只能选择一个功能使用。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，创建同步任务。
2. 在“设置同步选项和同步对象”步骤中，勾选**是否同步 Online DDL 临时表**，和 **Online DDL 工具**。
勾选 gh-ost，DTS 会将 gh-ost 工具产生的临时表名（`_表名_ghc`、`_表名_gho`、`_表名_del`）迁移到目标库。
勾选 pt-osc，DTS 会将 pt-osc 工具产生的临时表名（`_表名_new`、`_表名_old`）迁移到目标库。

说明

右方**已选对象**中，需要勾选指定的表对象，下面才会显示“是否同步 Online DDL 临时表”的按钮，勾选库名，或者勾选 Tables，无法显示“是否同步 Online DDL 临时表”的按钮。

如果源库中已存在与临时表名 `_表名_new`、`_表名_old` 相同的表名，则 pt-osc 会产生其他临时表，由于 pt-osc 产生的其他临时表名并非固定，DTS 无法同步其他临时表。这种情况**同步对象**不能仅选择这个表，需要选择这个表所在的整个库（或者整个实例），否则无法同步 Online DDL 变更产生的临时表数据到目标数据库。

Refresh Select all Clear

列过滤

最近更新时间：2024-07-08 17:35:52

操作场景

数据同步过程中，当用户勾选对象时，可以选择库对象、表对象的级别，也可以选择到列的级别，本文为您介绍列级别的数据同步。

适用范围

MySQL/MariaDB/Percona/TDSQL-C MySQL 相互之间的数据同步链路。

约束限制

1. 主键、唯一键的列不支持过滤。系统默认会将主键、唯一键的列同步到目标端。特殊情况下如果主键、唯一键被过滤，前置校验会报错。
2. 如果过滤的列是联合主键、外键、分区列，或者过滤列中包含索引，则源端不支持对该列进行 DML 操作，否则任务可能会报错。
3. 列过滤与视图的交互说明如下：

如果同步对象中勾选了视图，则视图相关列数据需要一并同步，不能过滤，否则任务会报错。

如果同步对象中勾选了视图，后续源库在创建视图时关联的是系统库表，则不能在同步任务中同时再使用列过滤，否则任务会报错。

4. Rename 操作与列过滤功能交互的特殊说明如下：

表 A 进行了列过滤，然后源端删除了表 A，又重新创建了一个结构完全相同的表 A，这种场景的同步结果，新创建的表 A 依然会进行列过滤。

表 A 进行了列过滤，然后源端删除了表 A，再将表 B Rename 为表 A，A 和 B 的表结构相同，这种场景，表 B 的同步结果，不会进行列过滤。

5. 配置列过滤后，不支持再修改同步配置；修改同步配置时，也不支持对列进行过滤。

6. 列过滤与 Where 条件过滤的交互说明如下：

如果一个表进行了列过滤，用户同时还对该表设置 Where 规则，需要用户自行保证输入 Where 规则中的列名，包含在待同步的表中，否则 Where 规则可以设置成功，但后续任务启动后会报错。

7. 如果对表 A 的 X 列进行了列过滤，用户在源端执行 CREATE TABLE table_B like table_A 语句，建议对表 B 的 X 列也进行过滤。在这种场景下 DTS 对表 B 的同步会按照表 A 列过滤后的表结构，如果未对表 B 的 X 列进行过滤，后续源端对表 B 的 X 列进行数据更新时，DTS 任务会报错。

8. 如果表对象中唯一键列包含函数，则同步对象勾选该表时，无法展开到列的维度，不能进行列过滤。
9. 仅选择同步数据，不同步表结构的场景，要求用户在 DTS 任务前，在目标端创建相同的表结构；如果任务还需要设置列过滤，则用户在目标端创建的表结构为列过滤后的表结构，否则同步到目标端可能会报错。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，创建同步任务。
2. 在**设置同步源和目标数据库**步骤，左侧**源库对象**中，可以选择表级别、列级别的数据，勾选后单击



，右侧即可显示待同步的对象。

MySQL 系列前置校验不通过处理

连接 DB 检查

最近更新时间：2024-07-08 17:51:43

检查详情

源数据库和目标数据库需要能正常连通，如果未连通，会报“连接源实例失败”。

问题原因

源数据库所在网络或服务器设置了安全组或防火墙。

源数据库对来源 IP 地址进行了限制。

网络端口未放通。

数据库帐号或密码不正确。

源数据库所在网络或服务器设置了安全组或者防火墙

检测方法

安全组功能与防火墙功能类似，安全组是针对云上数据库的网络安全设置。

请根据现场情况，进行以下检查步骤：

查看源数据库所在的服务器是否配置了防火墙策略。

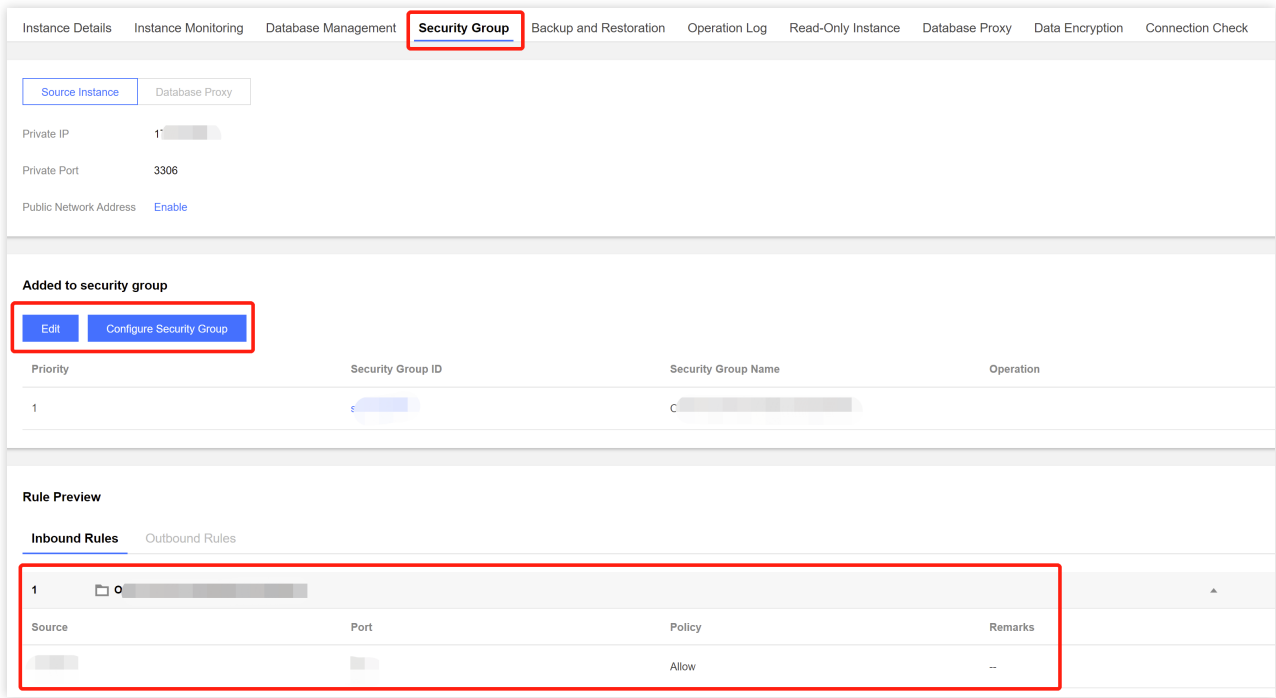
Windows 系统：打开控制面板找到 Windows 防火墙，查看是否配置了防火墙策略。

Linux 系统：请执行 `iptables -L` 命令，检查服务器是否配置了防火墙策略。

查看数据库配置的安全组是否限制了 DTS 的 IP 地址段。

1.1 登录 [对应的数据库](#)，在实例列表，单击实例 ID，进入实例管理页面。

1.2 在实例管理页面，选择**安全组**页，查看是否存在禁止 DTS 的 SNAT IP 地址段的策略。



修复方法

请根据现场情况，选择以下对应的步骤：
服务器开启了防火墙。

1.1 关闭服务器的防火墙，然后登录重新执行校验任务。

说明：

该方法 Windows 和 Linux 系统都适用。

1.2 将 DTS 的 IP 地址段“策略”配置为**允许**。
安全组的配置限制了DTS的SNAT IP地址段。

- 1.1 在安全组页面，单击对应的安全组 ID。
- 2. 修改 DTS 的 IP 地址段的策略，配置为**允许**。

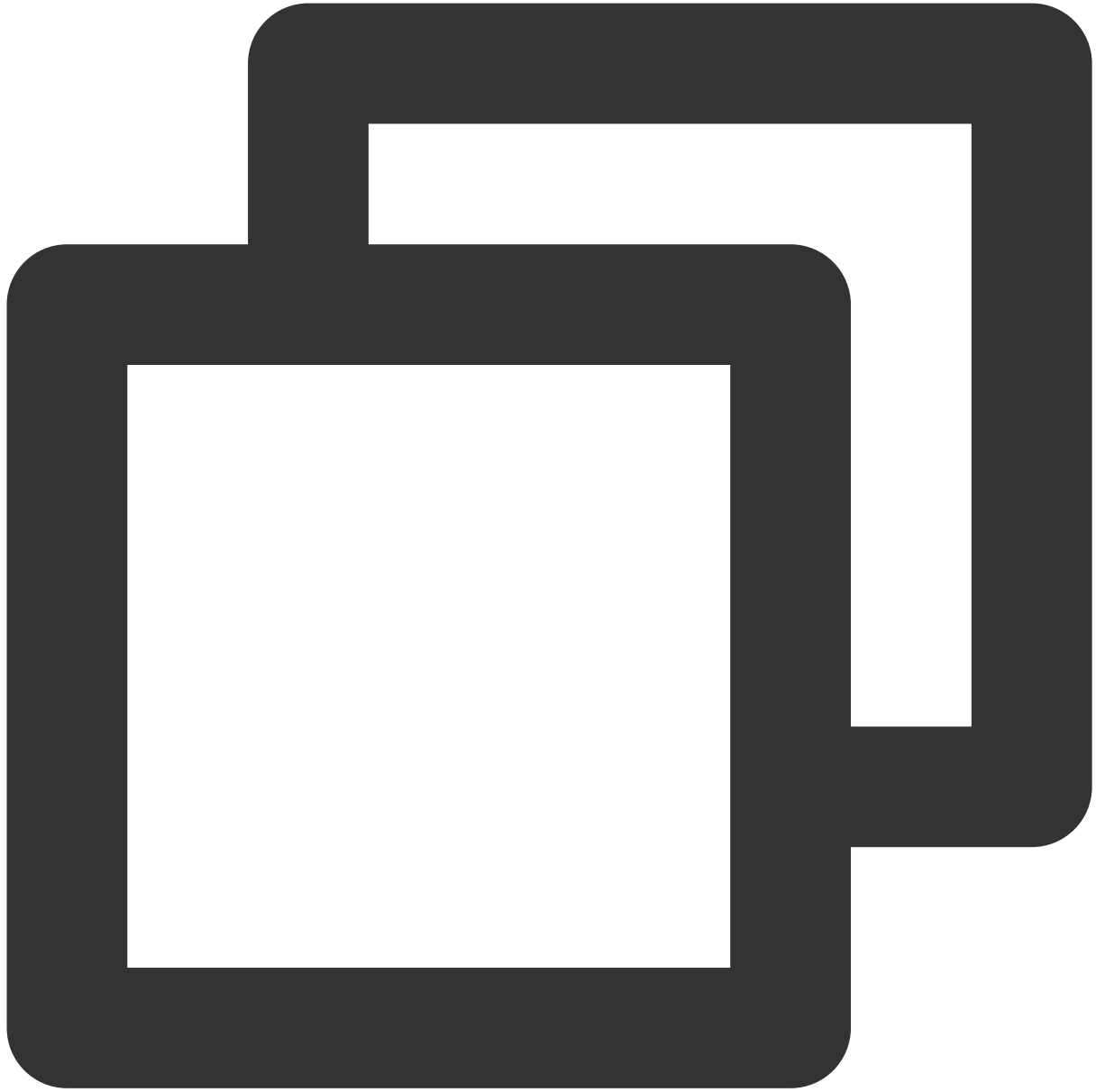
源数据库对来源 IP 地址进行了限制

检测方法

MySQL 检测方法

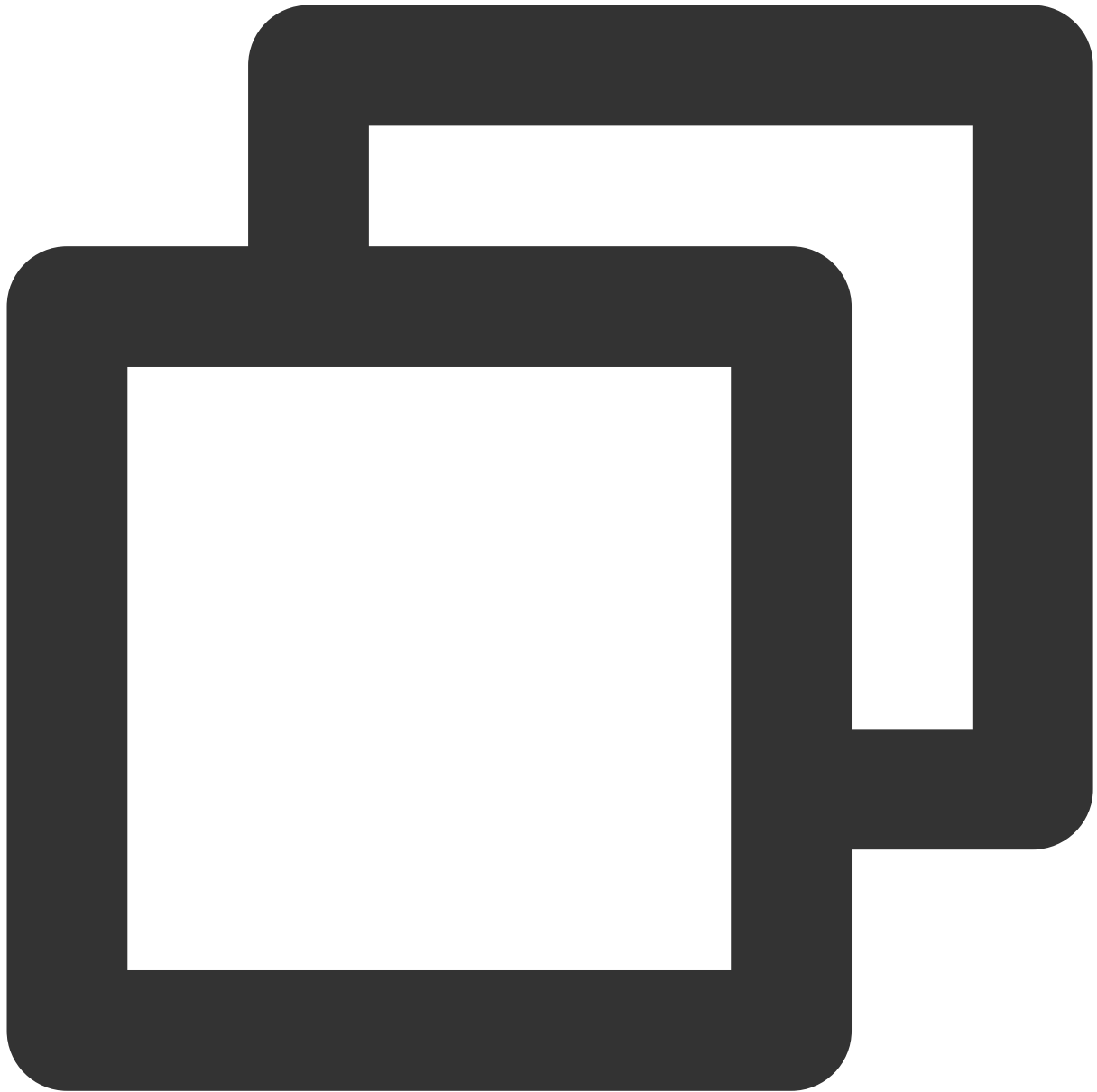
在源数据库部署的服务器上，使用数据迁移任务中填入的数据库帐号和数据库密码连接源数据库。如果连接正常，说明源数据库可能限制了来源 IP 地址。
如果是自建数据库，需要在数据库上确认 bind-address 的配置，如果不是0.0.0.0，则 IP 受限。
如果源数据库为 MySQL，您可以使用 MySQL 客户端连接源数据库，执行以下 SQL 语句进行检查，检查输出结果中的授权 IP 地址列表中是否包含 DTS 的 SNAT IP 地址。

在给用户进行数据库的授权时，授权的 IP 中需要包含 SNAT IP，否则会发生 IP 受限问题。示例如下：



```
root@10.0.0.0/8 //授权用户通过指定10.0.0.0/8访问，其他 IP 会受限（错误配置）
root@%         //授权用户访问所有的 IP，其中需要包含 SNAT IP（正确配置）
```

您可以通过如下方法验证。



```
select host,user,authentication_string,password_expired,account_locked from
mysql.user WHERE user='[\\$Username]';    //[\\$Username]为数据迁移任务中填写的数据库帐号
```

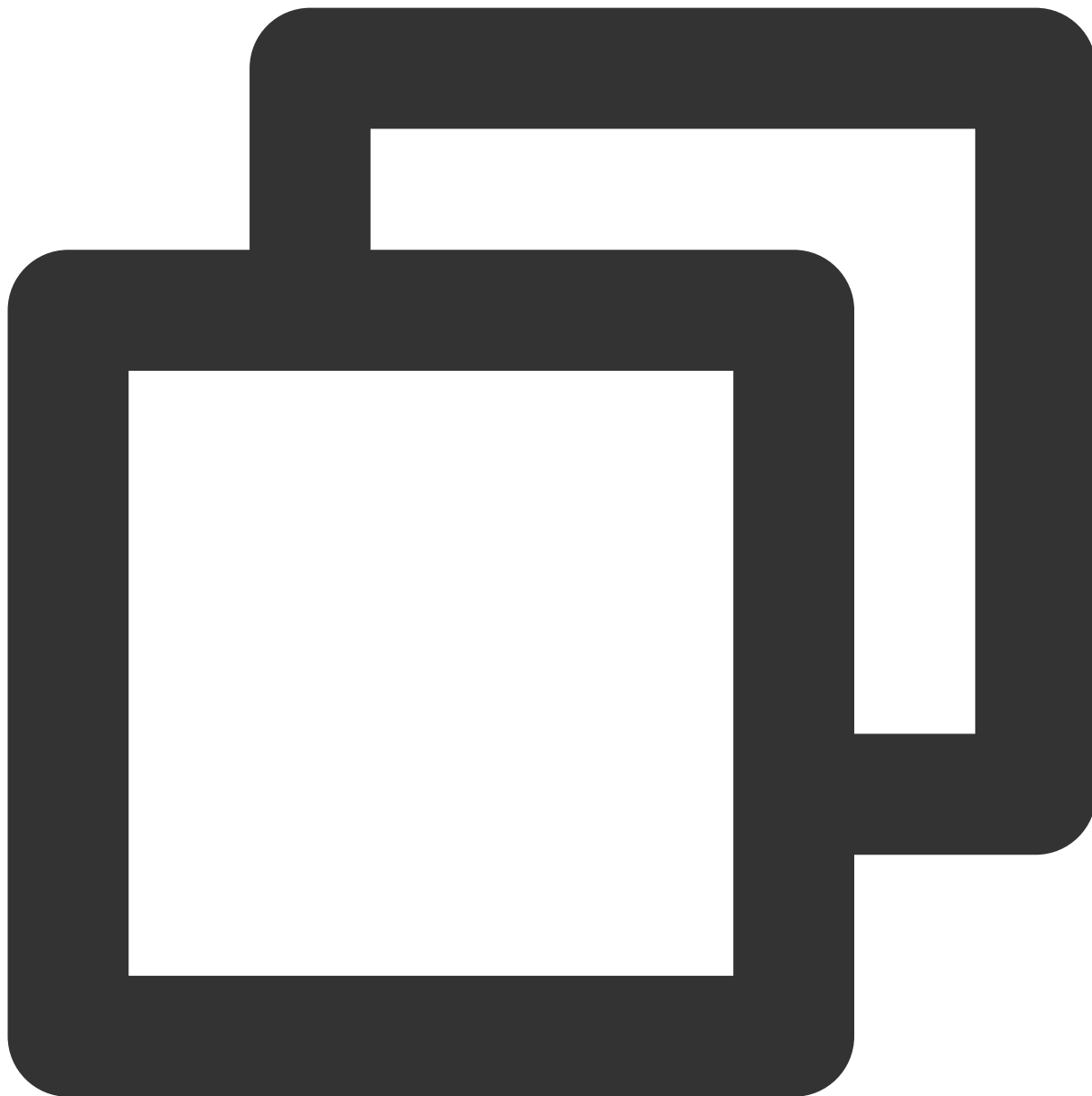
SQL Server 检测方法

检查源数据库中是否有 Endpoint 或 Trigger 限制了访问来源 IP 地址。

PostgreSQL 检测方法

如果源端数据库为其他云数据库时，请首先检查源端云数据库实例的安全访问策略是否有所限制。请根据不同云厂商的限制方法来进行检查。

如果源端数据库为自建的 PostgreSQL 数据库，请进入 `$PGDATA` 目录下的 `data` 目录，找到 `pg_hba.conf` 文件。查看此文件中是否存在 `deny` 策略，或者仅允许部分网络端的 IP 地址访问。



```
# cat pg_hba.conf
local    replication    all                                     trust
host     replication    all             127.x.x.1/32          trust
host     replication    all             ::1/128              trust
host     all             all             0.0.0.0/0            md5
host     all             all             172.x.x.0/20         md5
```

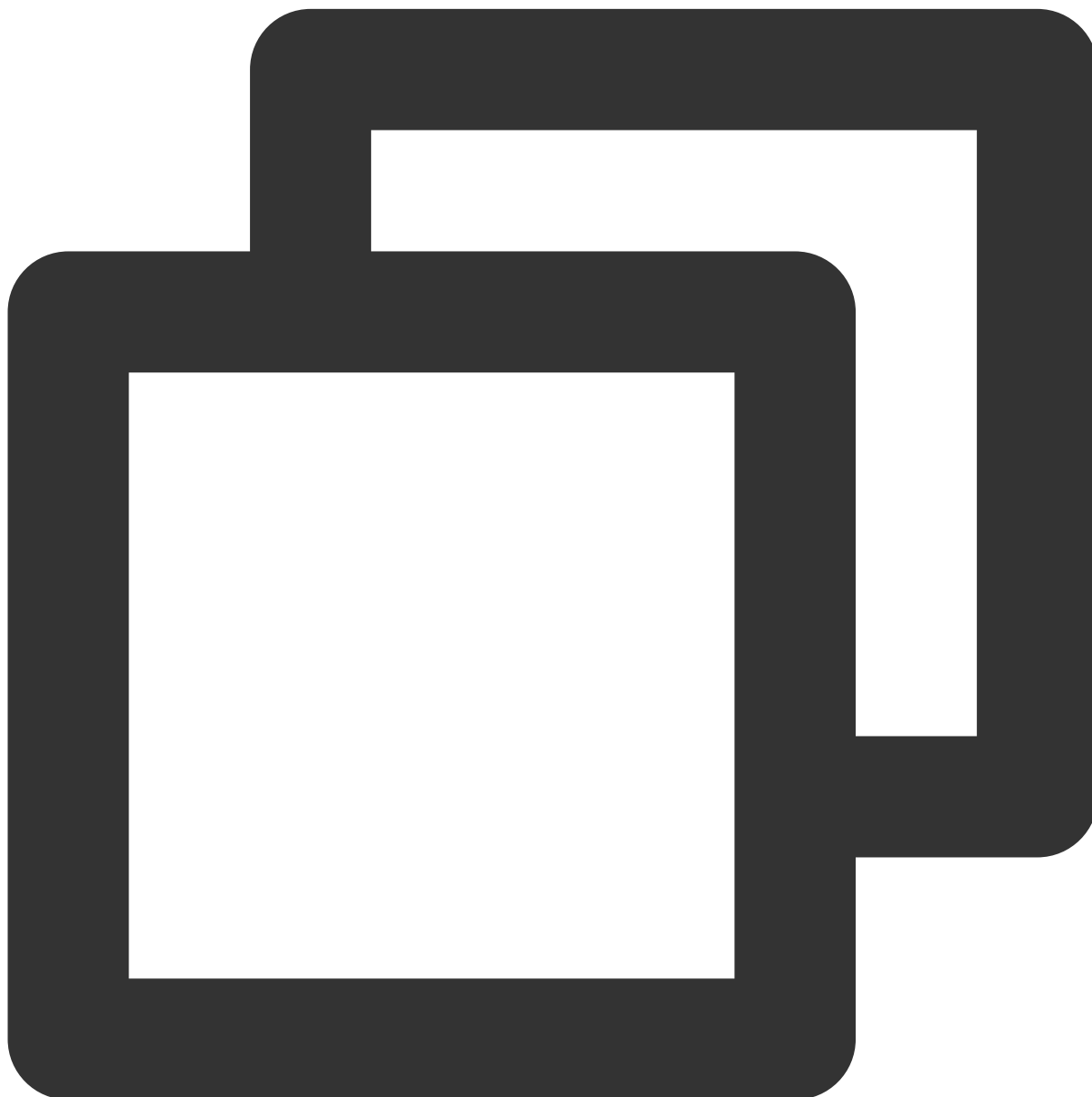
MongoDB 检测方法

如果是自建数据库，需要在数据库上确认 `bind-address` 的配置，如果不是 `0.0.0.0`，则 IP 受限。

修复方法

MySQL 修复方法

1. 源数据库为 MySQL，请在源数据库中执行以下 SQL 语句，重新给数据迁移使用的用户授权。



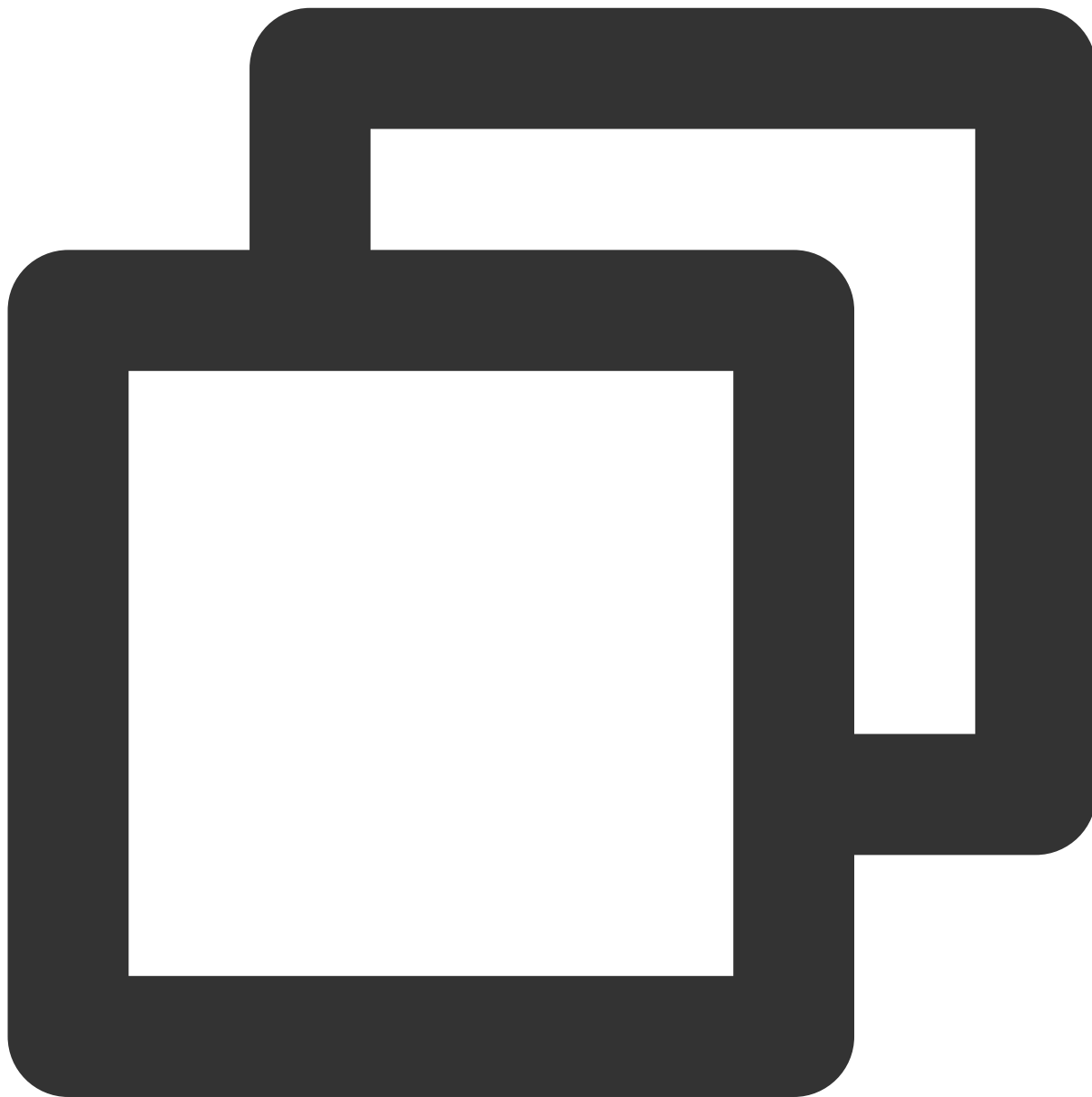
```
mysql> grant all privileges on . to '[\\$UserName]@'%'; // [\\$Username] 为数据迁移使用的用户名
mysql> flush privileges;
```

2. 对于自建数据库，如果 bind-address 配置异常，请参考如下指导修改。

2.1. 在 `/etc/my.cnf` 文件中增加如下内容。

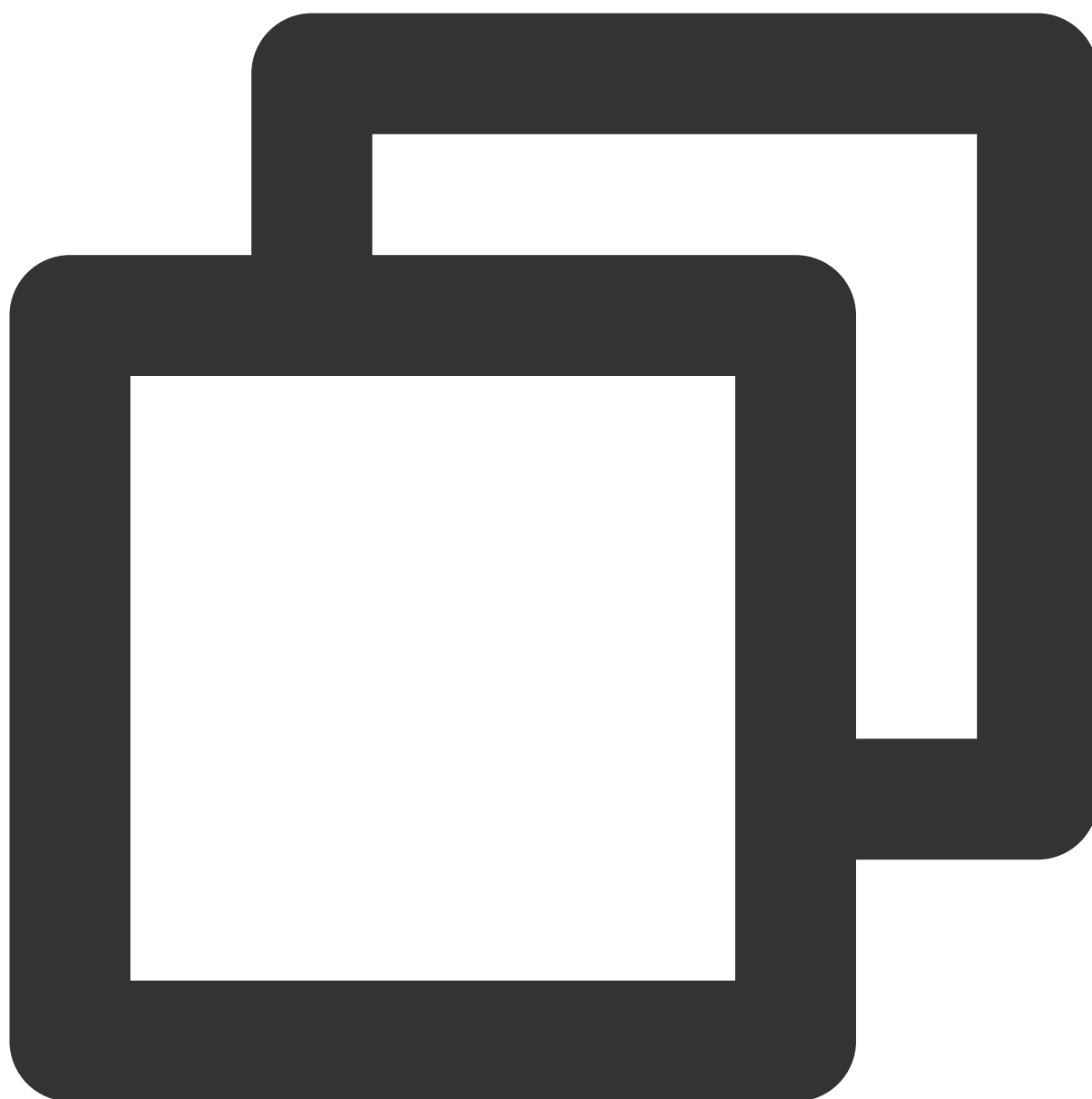
说明：

`my.cnf` 配置文件的默认路径为 `/etc/my.cnf`，现场以实际情况为准。



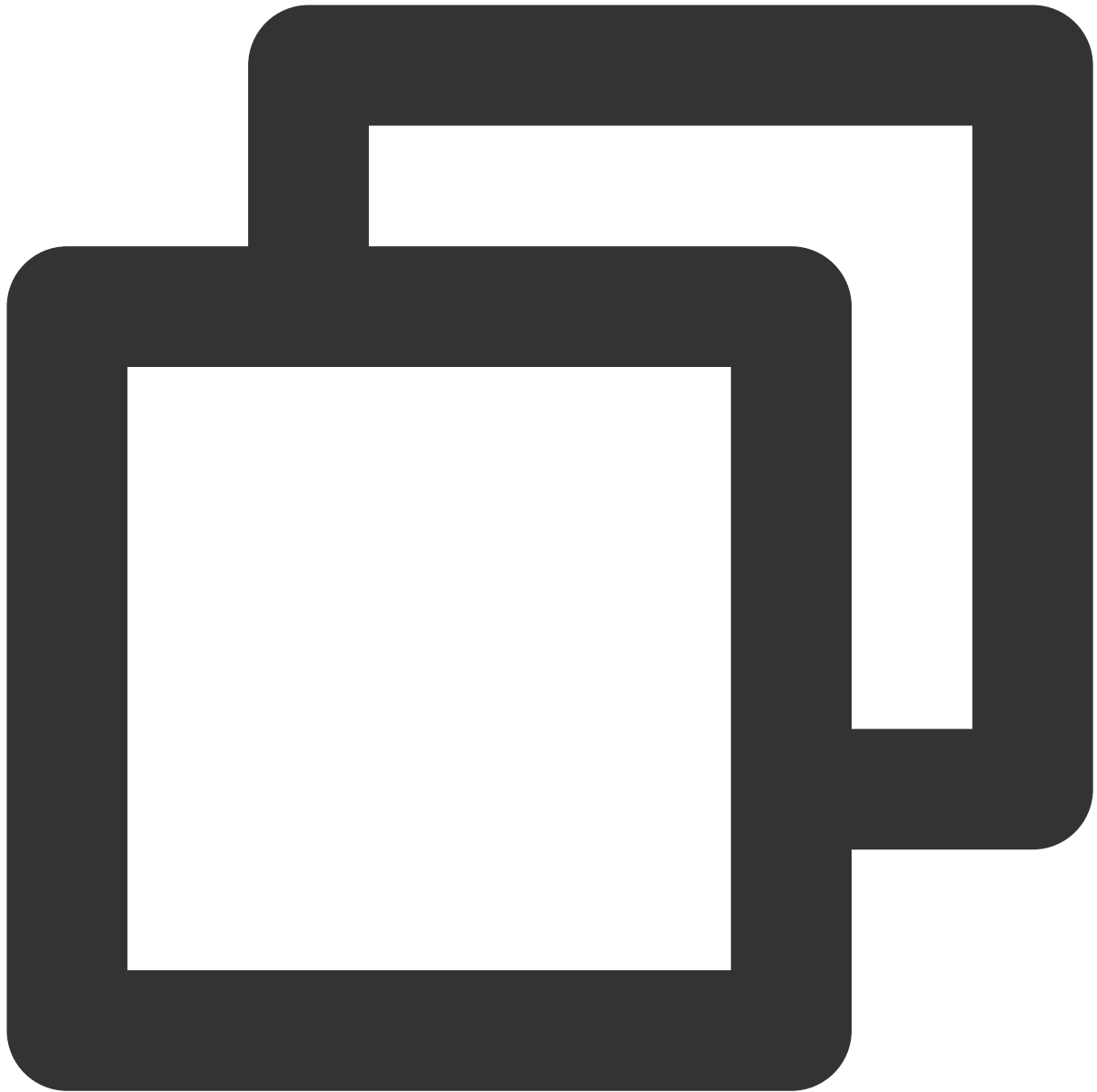
```
bind-address=0.0.0.0 #全部地址或者指定的 IP 地址
```

2.2. 重启数据库。



```
service mysqld restart
```

2.3. 验证配置是否生效。



```
netstat -tln
```

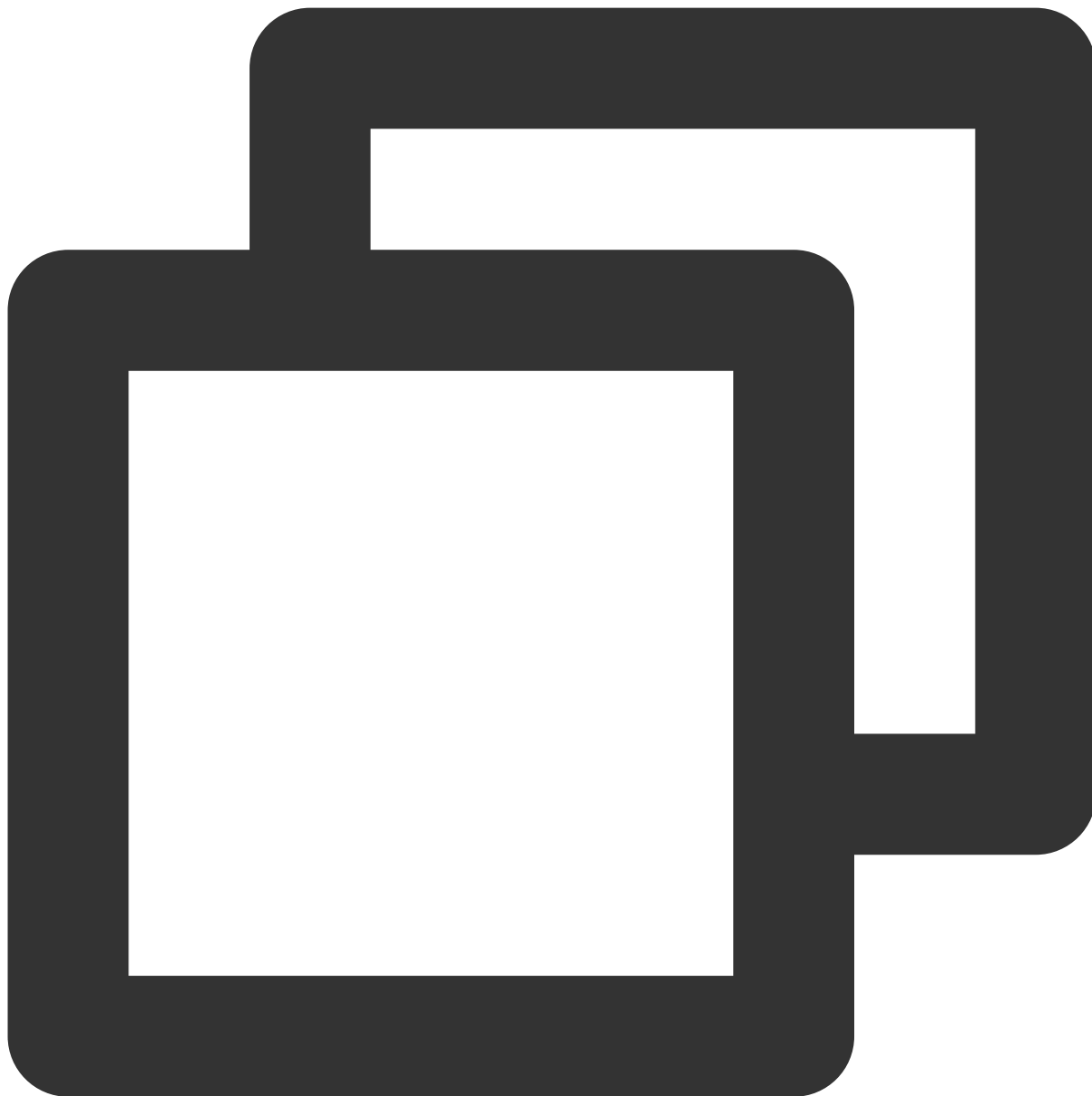
3. 重新执行校验任务。

SQL Server 修复方法

关闭防火墙或禁用 trigger。

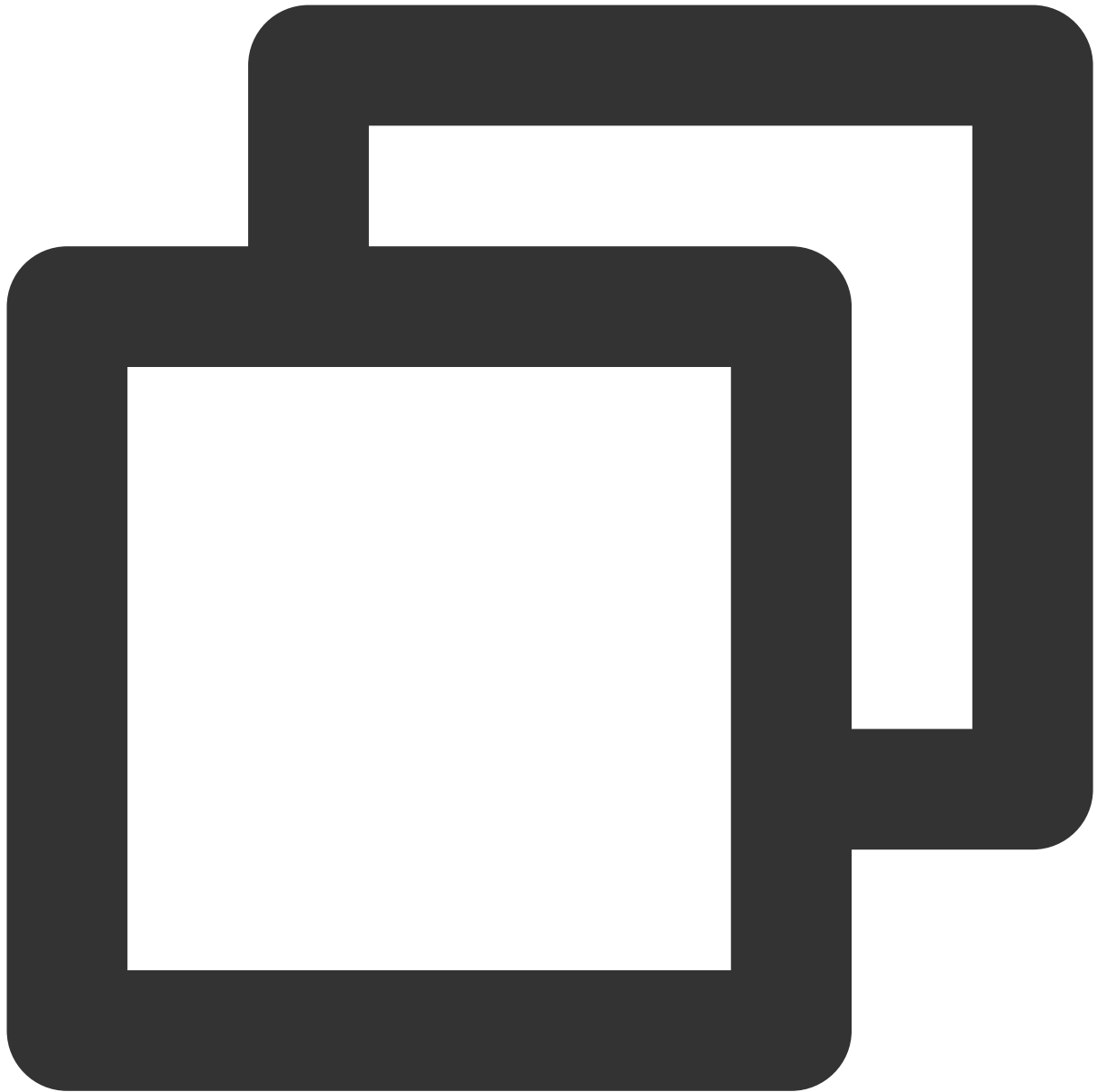
PostgreSQL 修复方法

1. 请在 `pg_hba.conf` 文件中加入允许 DTS 网络段的访问策略。或者在迁移过程中临时放开所有网段的访问策略。如在此文件中添加一行：



host	all	all	0.0.0.0/0	md5
------	-----	-----	-----------	-----

2. 修改完成后，可重启数据库实例，让配置生效：



```
pg_ctl -D $PGDATA restart
```

3. 重新执行校验任务。

MongoDB 修复方法

参考 [MySQL](#) 中的方法配置 bind-address。

网络端口未放通

检测方法

常见数据库默认端口如下，需要确认这些端口已放通。如果用户修改了默认端口，请按实际情况修改放通的端口。

MySQL：3306

SQL Server：1433

PostgreSQL：5432

MongoDB：27017

Redis：6379

修复方法

放通相应的数据库端口。

如果源数据库为 SQL Server，还需要同时放通文件共享服务端口445。

数据库帐号或密码不正确

检测方法

登录源数据库，验证帐号和密码是否正确。

修复方法

在 [DTS 控制台](#) 修改数据迁移任务，输入正确的数据库帐号和密码后重新执行校验任务。

周边检查

最近更新时间：2024-07-08 17:51:43

MySQL/TDSQL MySQL/TDSQL-C 检查详情

检查要求：源数据库环境变量参数 `innodb_stats_on_metadata` 需要设置为 `OFF`。

检查说明：

`innodb_stats_on_metadata` 参数开启时，每当查询 `information_schema` 元数据库里的表，Innodb 就会更新 `information_schema.statistics` 表，导致访问时间变长。关闭后可加快对于 `schema` 库表的访问。

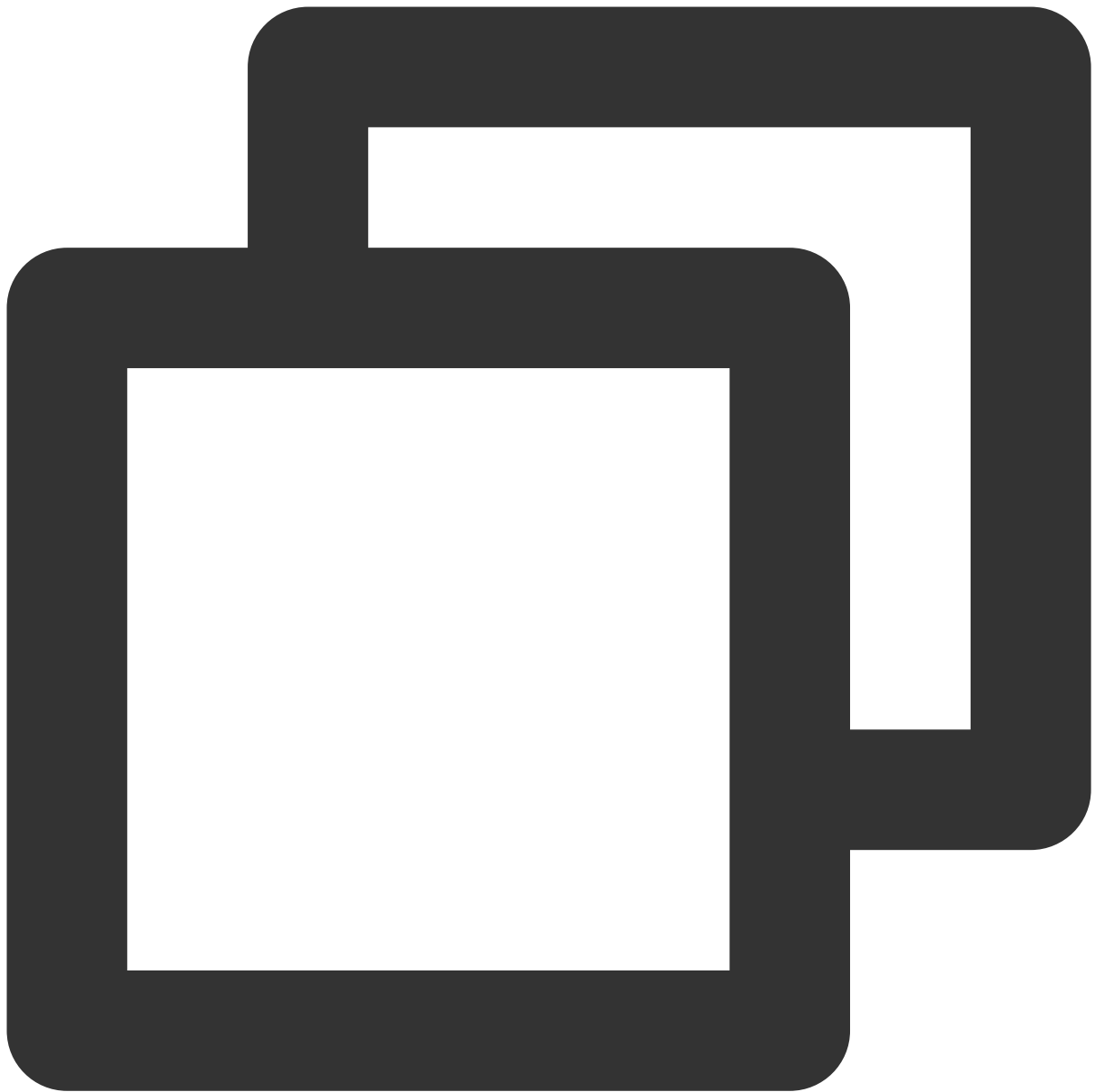
MySQL 5.6.6 之前版本 `innodb_stats_on_metadata` 参数预设值为 `ON`，需要修改为 `OFF`。MySQL 5.6.6 及其以后的版本预设值为 `OFF`，不存在问题。

TDSQL PostgreSQL 版检查详情

DN 节点的 `wal_level` 必须是 `logical`。

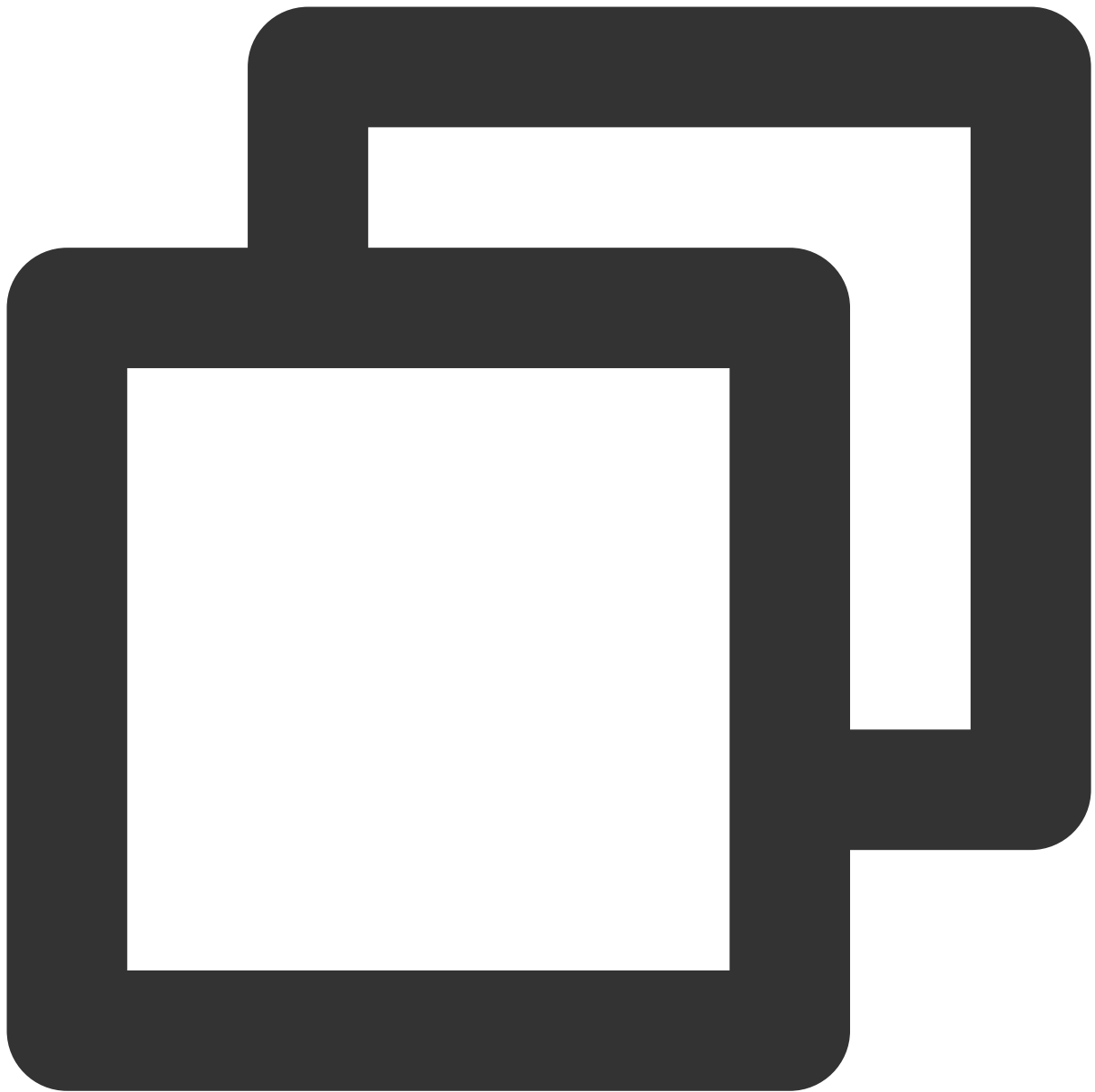
修复方法

1. 登录源数据库。
2. 修改 `innodb_stats_on_metadata` 为 `OFF`。



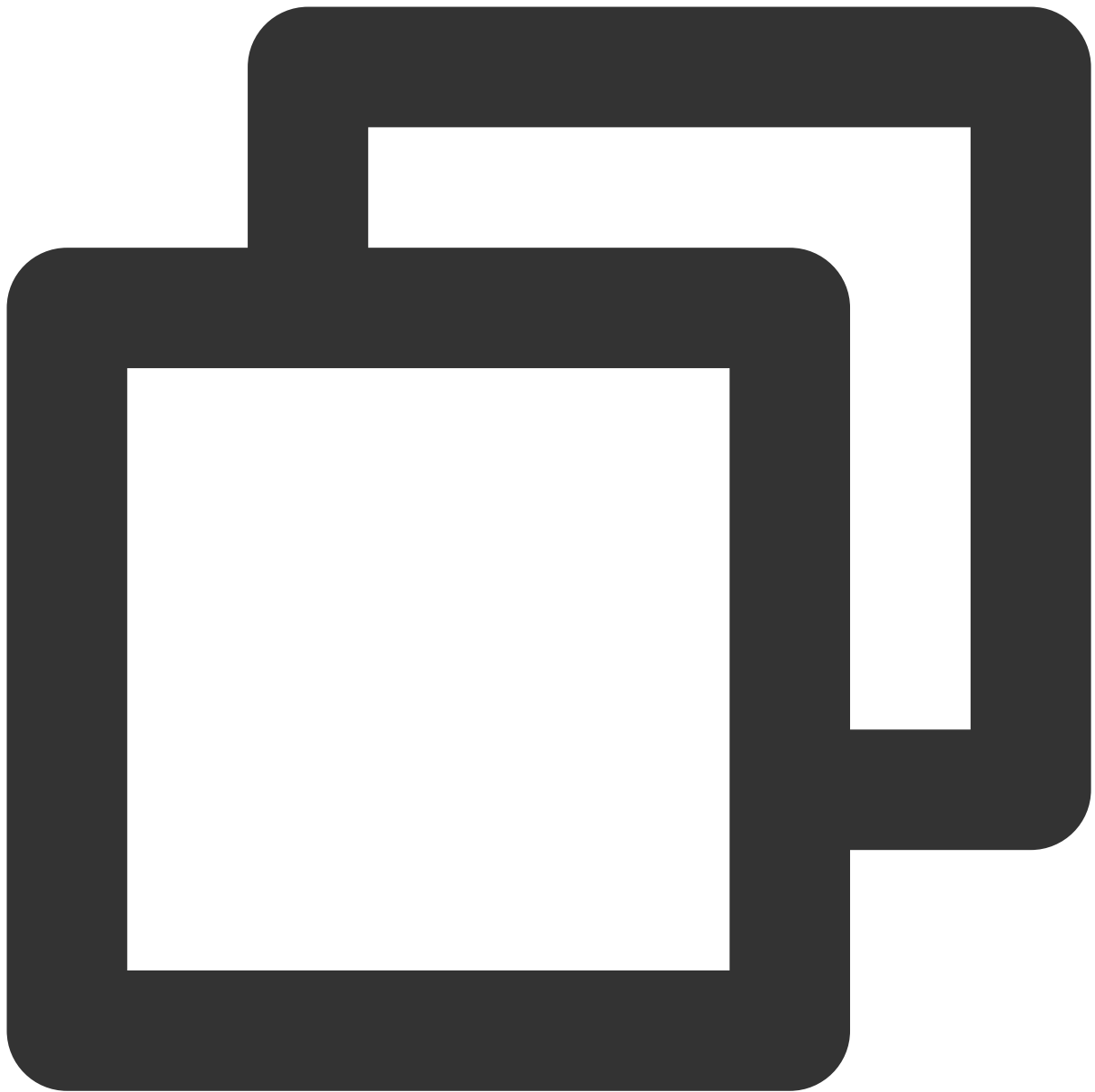
```
set global innodb_stats_on_metadata = OFF;
```

3. 查看配置是否生效。



```
show global variables like '%innodb_stats_on_metadata%';
```

系统显示结果类似如下：



```
mysql> show global variables like '%innodb_stats_on_metadata%';
+-----+-----+
| Variable_name          | Value |
+-----+-----+
| innodb_stats_on_metadata | OFF   |
+-----+-----+
1 row in set (0.00 sec)
```

4. 重新执行校验任务。

版本检查

最近更新时间：2024-07-08 17:51:43

检查详情

检查要求：目标数据库版本必须大于或等于源数据库版本，且所有迁移和同步的版本符合版本要求。

检查说明：此处版本以大版本号区分，如5.6.x支持迁移或同步到5.6.x、5.7.x及以后版本，最后一位属于小版本号，小版本号不限制，如5.6.5可以迁移或同步到5.6.4，但是可能会有兼容性问题。

修复方法

请按 [数据迁移支持的数据库](#) 和 [数据同步支持的数据库](#) 中的版本要求检查源库和目标库，如果源库或者目标库版本不支持，请升级目标实例版本或者使用更高版本的数据库实例。

源实例权限检查

最近更新时间：2024-07-08 17:51:43

检查详情

检查用户是否具备对数据库的操作权限，具体可参考以下对应权限要求：

[数据迁移权限要求](#)

[数据同步权限要求](#)

[数据订阅权限要求](#)

修复方法

用户若不具备操作权限，请按照检查要求中的对应权限要求对用户进行授权，然后重新执行校验任务。

部分实例参数检查

最近更新时间：2024-07-08 17:51:43

检查详情

源库表的 `row_format` 不能为 `FIXED`。

源库和目标库 `lower_case_table_names` 变量必须一致。

目标库 `max_allowed_packet` 参数设置至少为4M。

源库变量 `connect_timeout` 必须大于10。

在 MySQL/TDSQL MySQL/TDSQL-C 迁移到 MySQL 的场景中，如果源实例存在耗时较长的 SQL 在运行，则会触发警告，提示“源实例有耗时较长的 SQL 在运行，可能导致锁表，请稍后重试或对源实例中的 SQL 进行处理”。

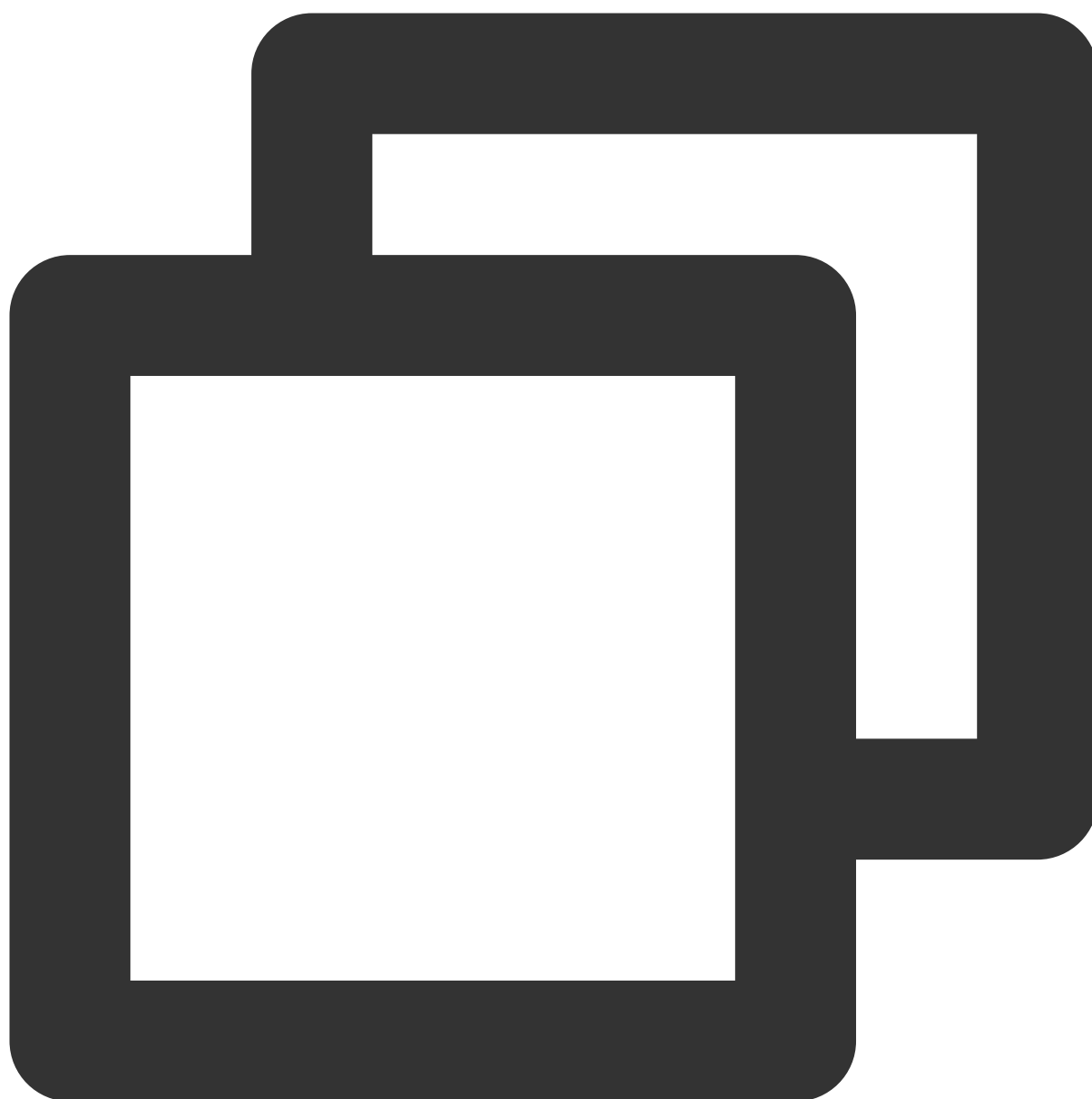
修复方法

修改源库 `row_format` 参数

数据库中表的 `row_format` 的取值为 `FIXED` 时，表格中每行的存储长度超过限制值时会溢出，发生报错。因此需要修改为其他模式，如 `DYNAMIC`，使每行的存储长度会随内容的长度而变化。

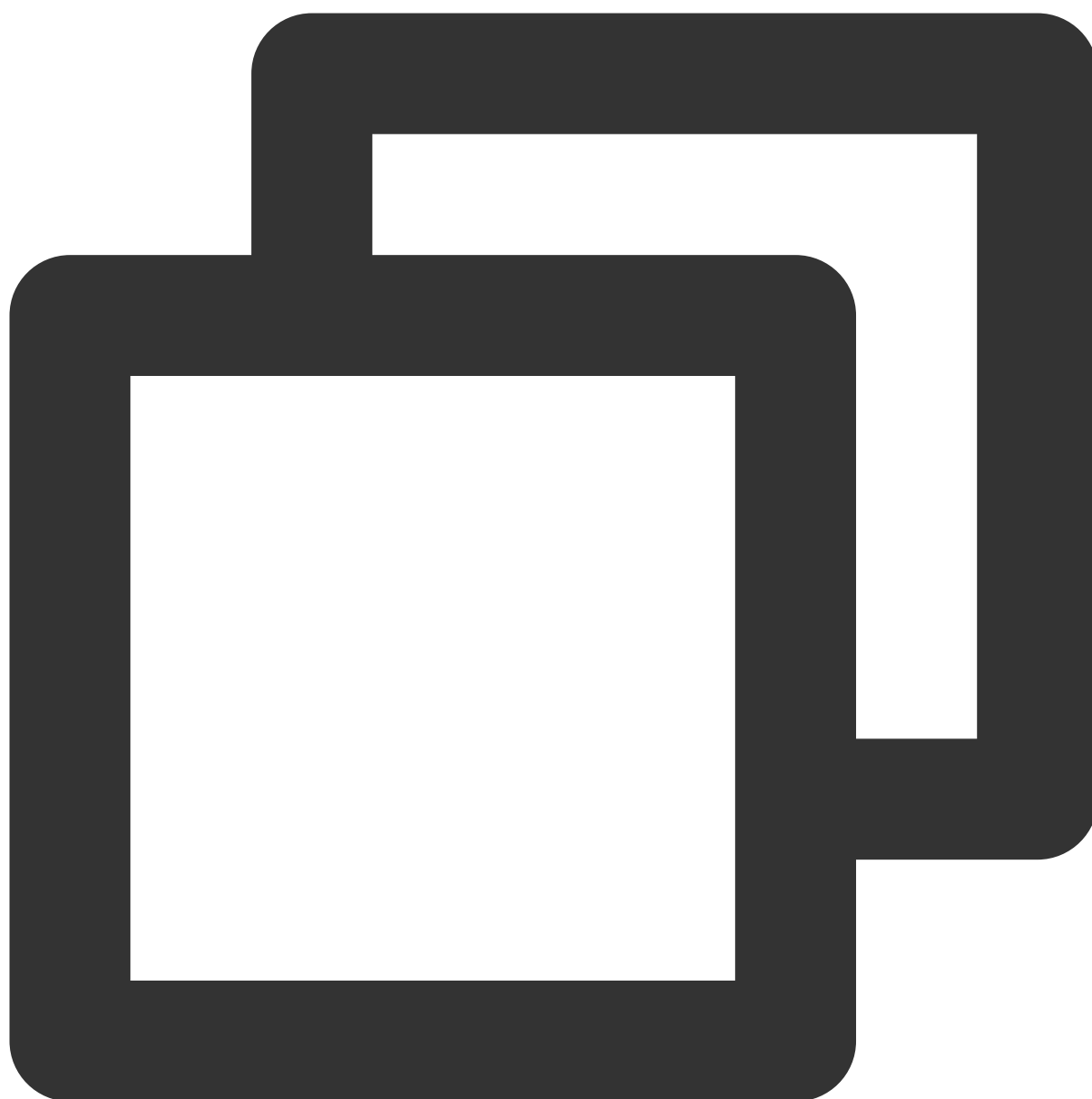
如发生此类报错，请参考如下指导进行修复：

1. 登录源数据库。
2. 修改 `row_format` 参数为 `DYNAMIC`。



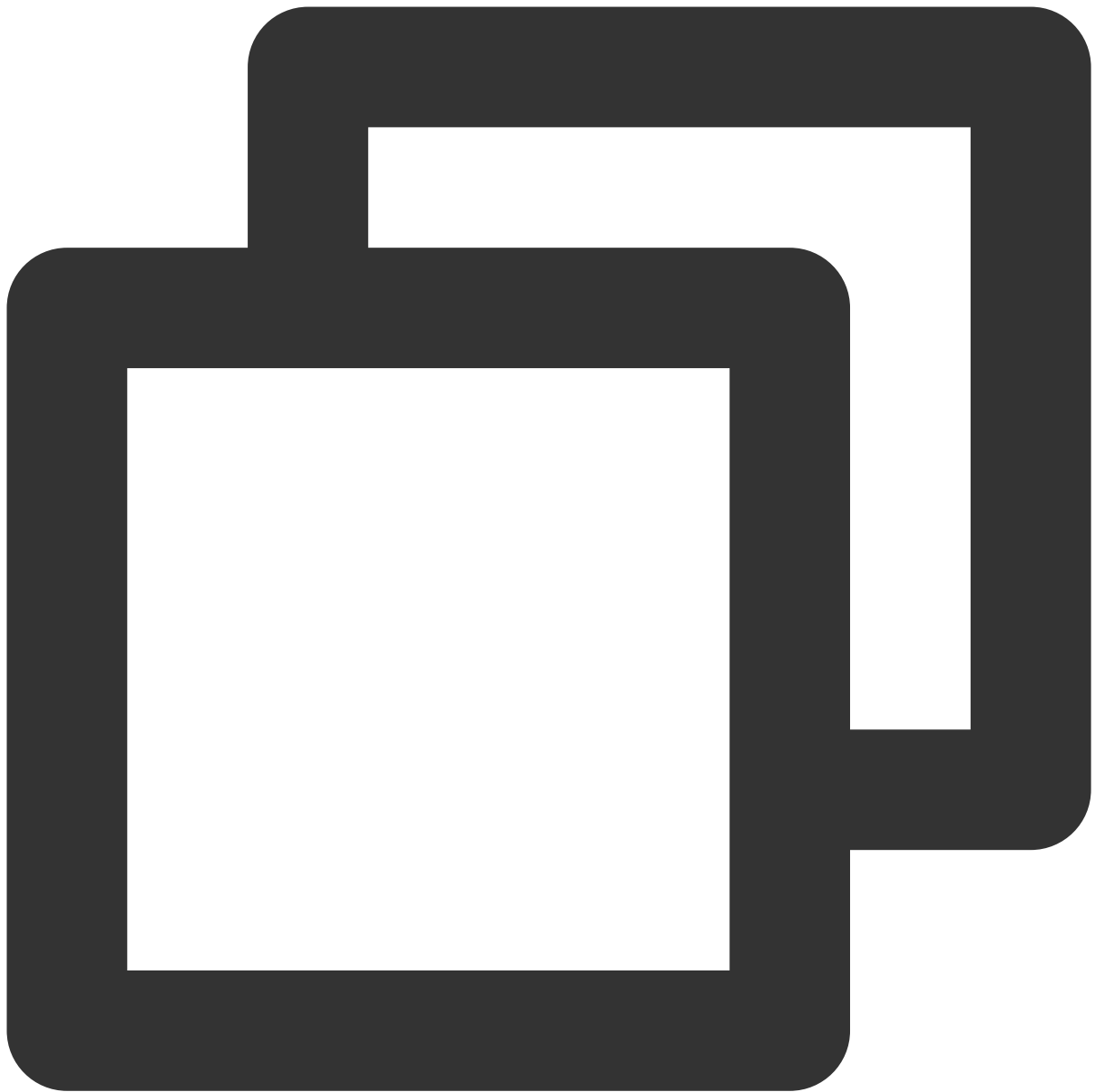
```
alter table table_name row_format = DYNAMIC;
```

3. 查看配置是否生效。



```
show table status like 'table_name'\\G;
```

系统显示结果类似如下：



```
mysql> show table status like 'table_name'\\G;
***** 1. row *****
      Name: table_name
      Engine: InnoDB
      Version: 10
      Row_format: Dynamic
      Rows: 5
      .....
1 row in set (0.00 sec)
```

4. 重新执行校验任务。

修改源库和目标库 `lower_case_table_names` 变量保持一致

`lower_case_table_names` 是 MySQL 设置大小写是否敏感的一个参数，不同的取值情况如下：

Windows 或 macOS 环境对大小写是不敏感的，但是 Linux 环境却是敏感的，为了保证不同系统的兼容性，需要将大小写敏感规则设置统一。

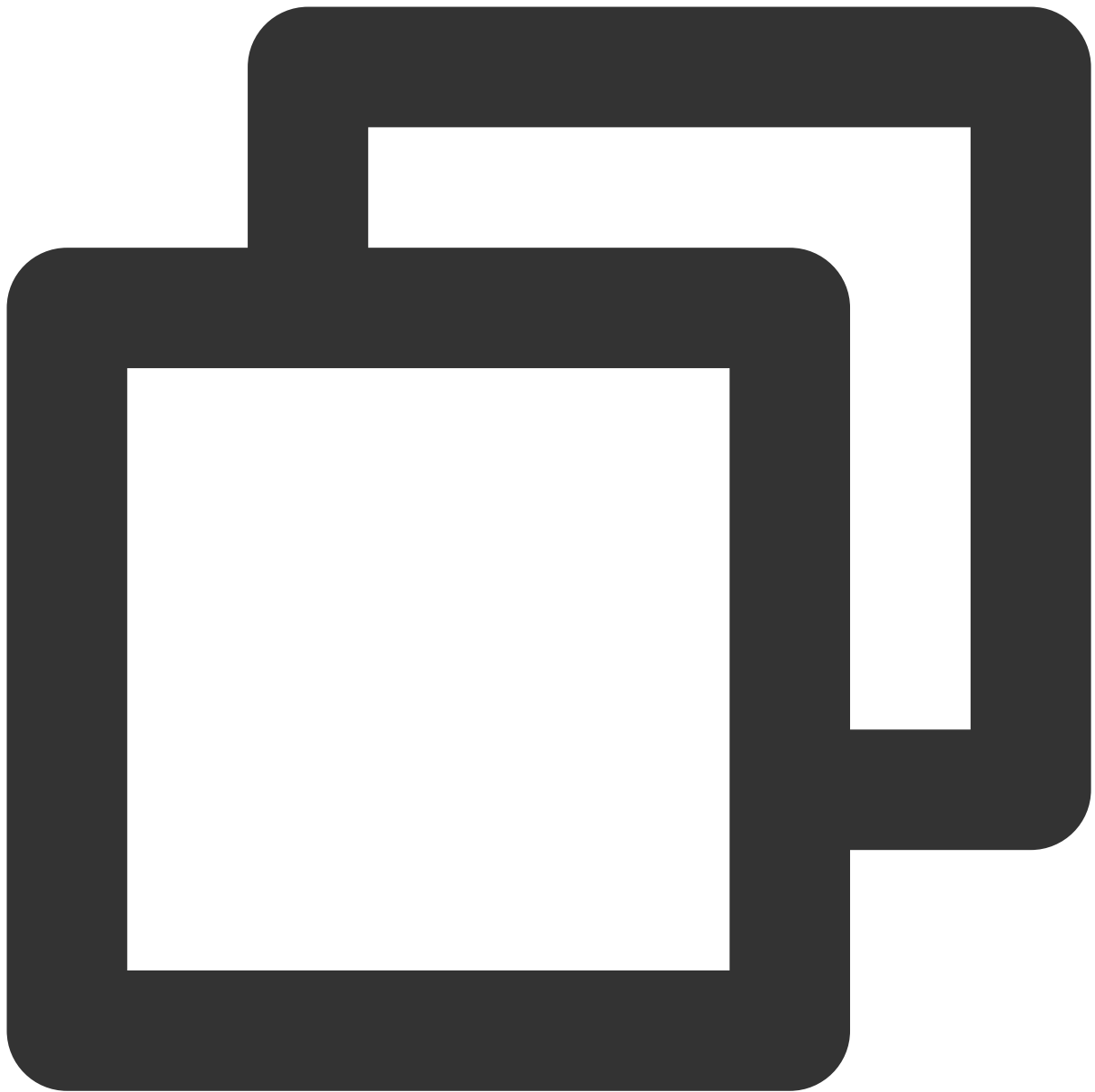
0：表名存储为给定的大小写，比较的时候区分大小写。

1：表名存储在磁盘是小写的，比较的时候不区分大小写。

2：表名存储为给定的大小写，比较的时候是小写的。

如发生此类报错，请参考如下指导将源库和目标库的参数改为统一。

1. 登录源数据库。
2. 查看源库和目标库中的 `lower_case_table_names` 取值。

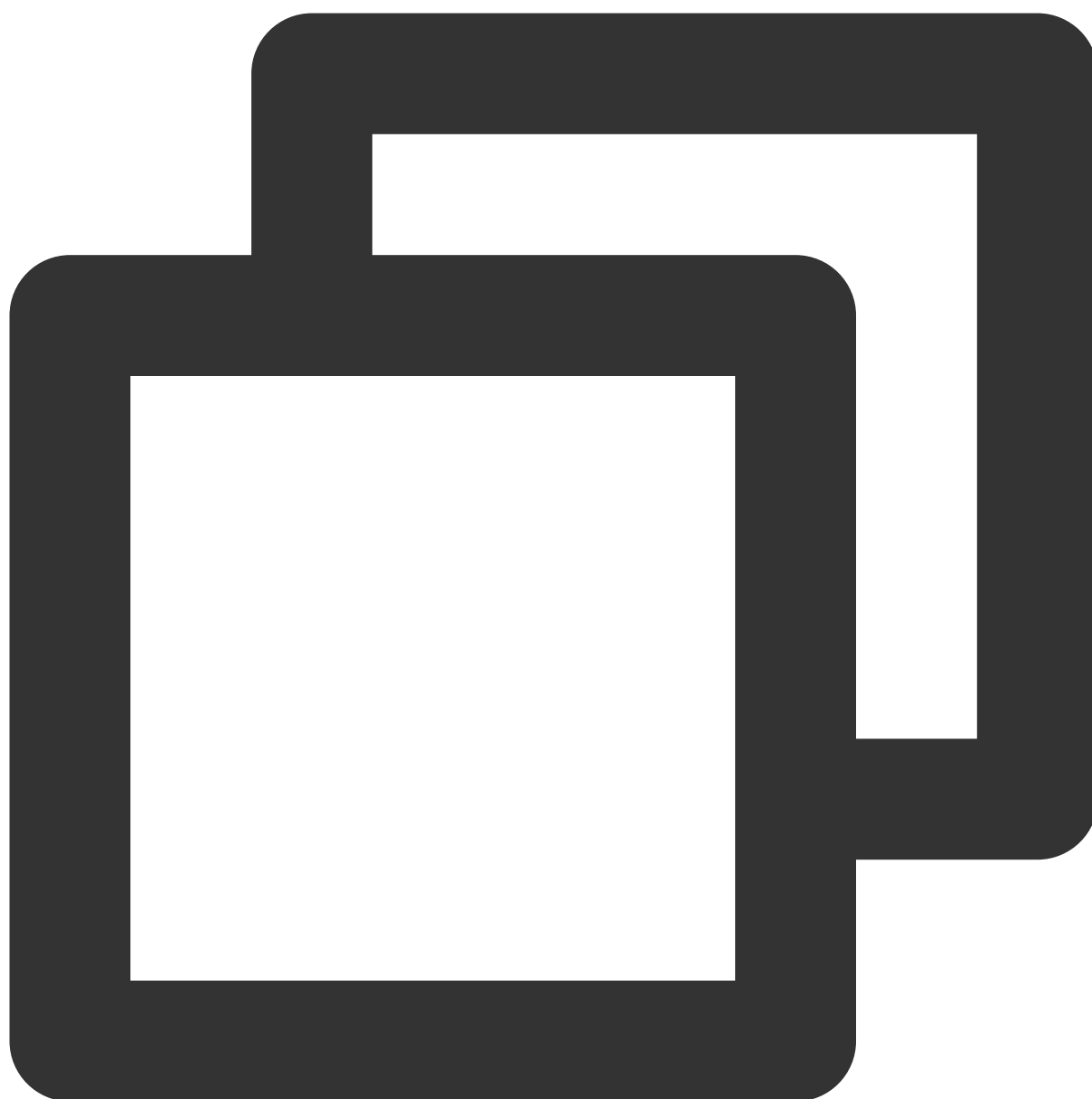


```
show variables like '%lower_case_table_names%';
```

3. 参考如下内容修改源数据库的配置文件 `my.cnf` 。

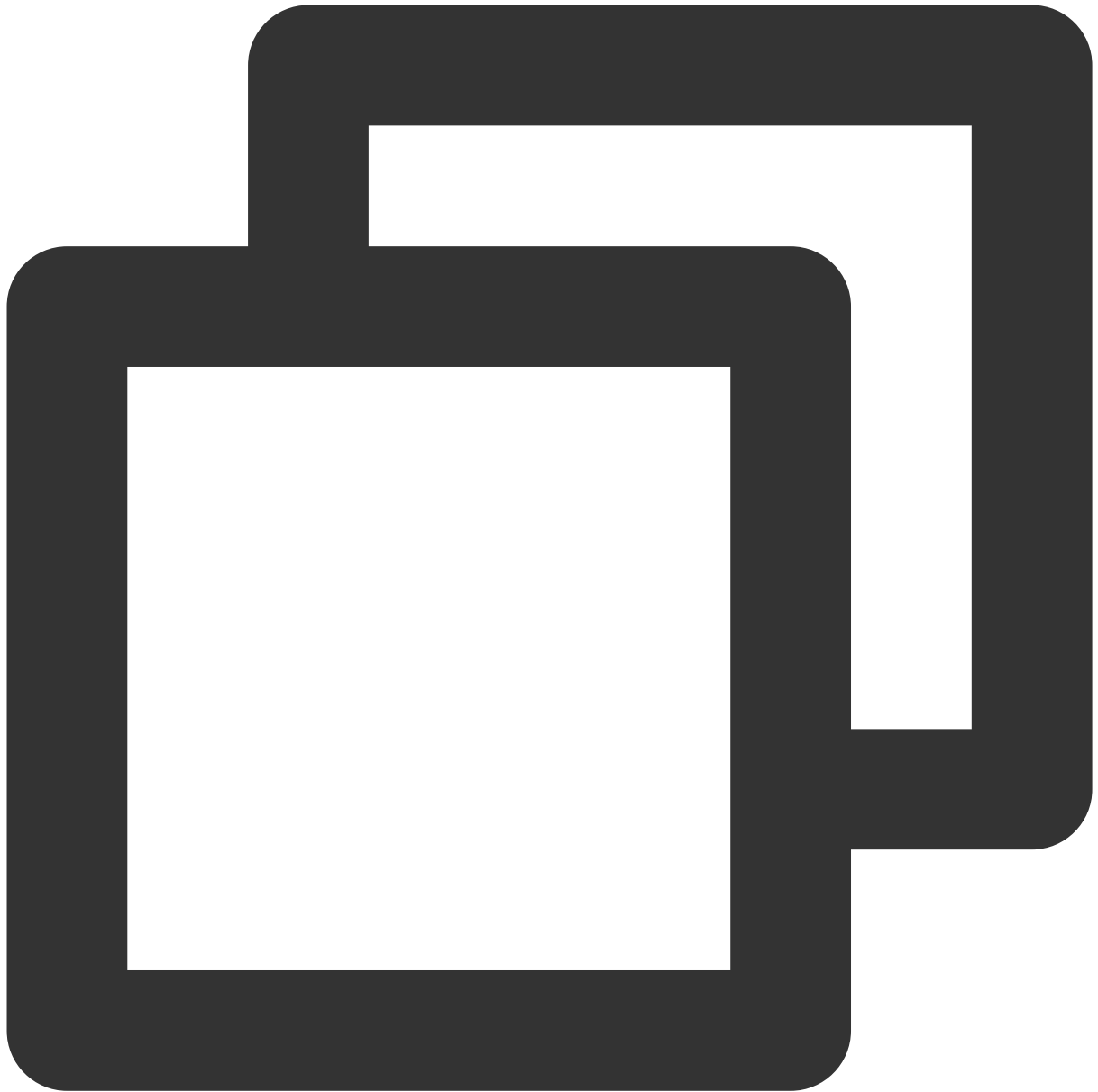
说明

`my.cnf` 配置文件的默认路径为 `/etc/my.cnf`，现场以实际情况为准。



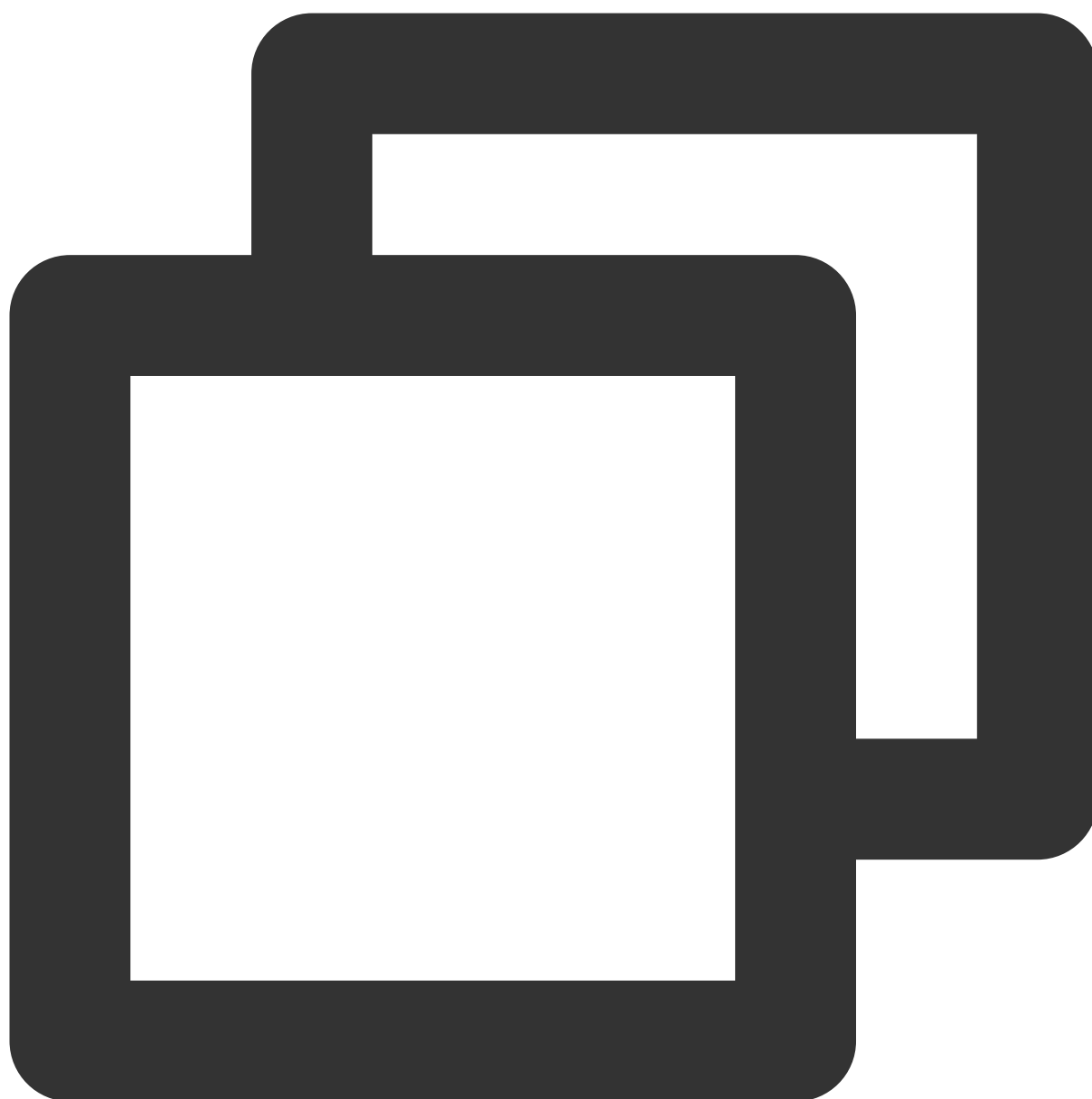
```
lower_case_table_names = 1
```

4. 参考如下命令重启数据库。



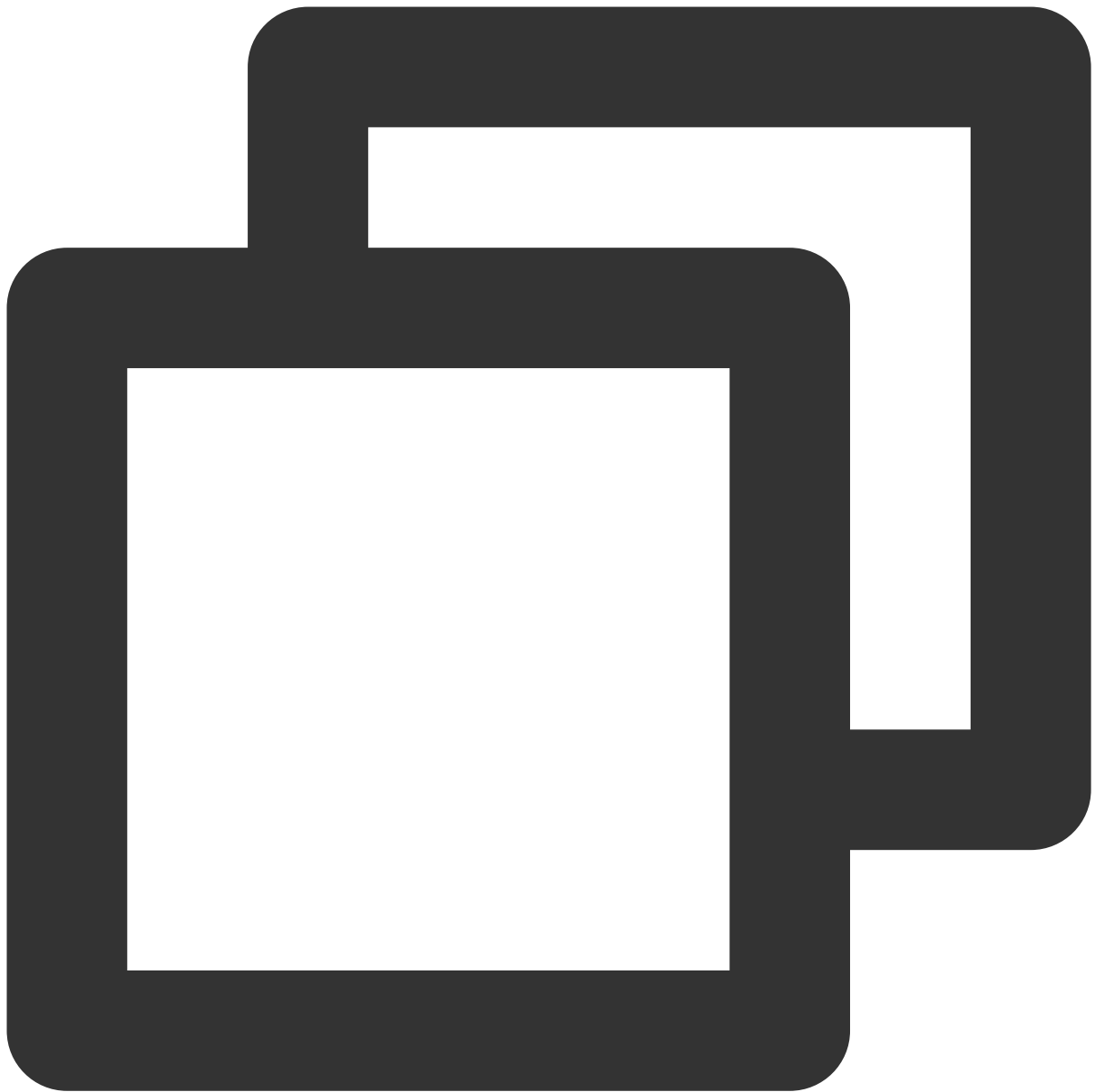
```
[$Mysql_Dir]/bin/mysqladmin -u root -p shutdown  
[$Mysql_Dir]/bin/safe_mysqld &
```

5. 查看配置是否生效。



```
show variables like '%lower_case_table_names%';
```

系统显示结果类似如下：



```
mysql> show variables like '%lower_case_table_names%';
+-----+-----+
| Variable_name          | Value |
+-----+-----+
| lower_case_table_names | 1      |
+-----+-----+
1 row in set (0.00 sec)
```

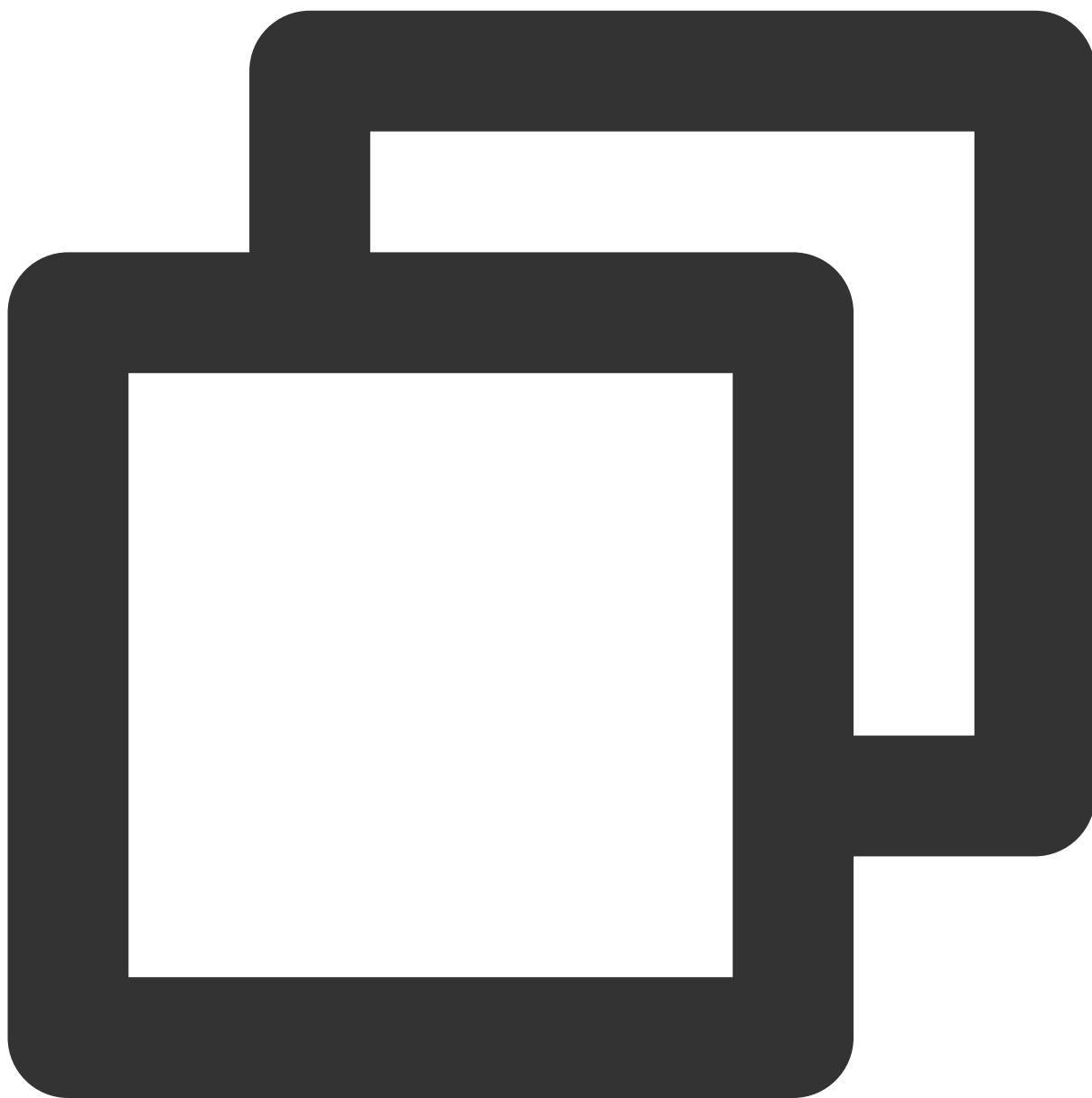
6. 重新执行校验任务。

修改目标库 `max_allowed_packet` 参数

`max_allowed_packet` 为最大允许的传输包。设置太大，会使用更多内存导致丢包，无法捕捉异常大事物包 SQL；设置太小，可能会导致程序报错，备份失败，也会导致频繁的收发网络包，影响系统性能。

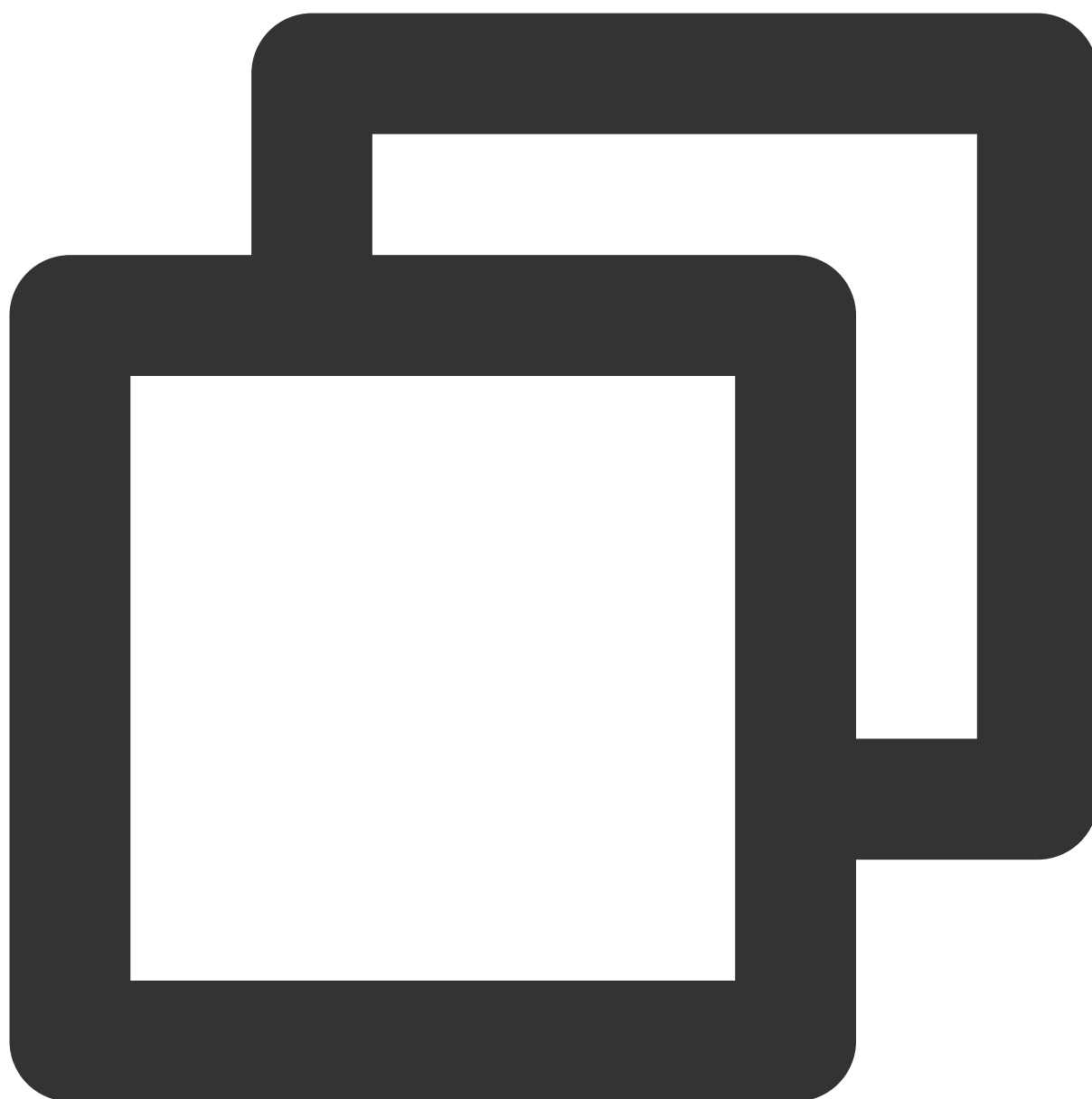
如发生此类报错，请参考如下指导进行修复：

1. 登录目标数据库。
2. 修改 `max_allowed_packet` 参数。



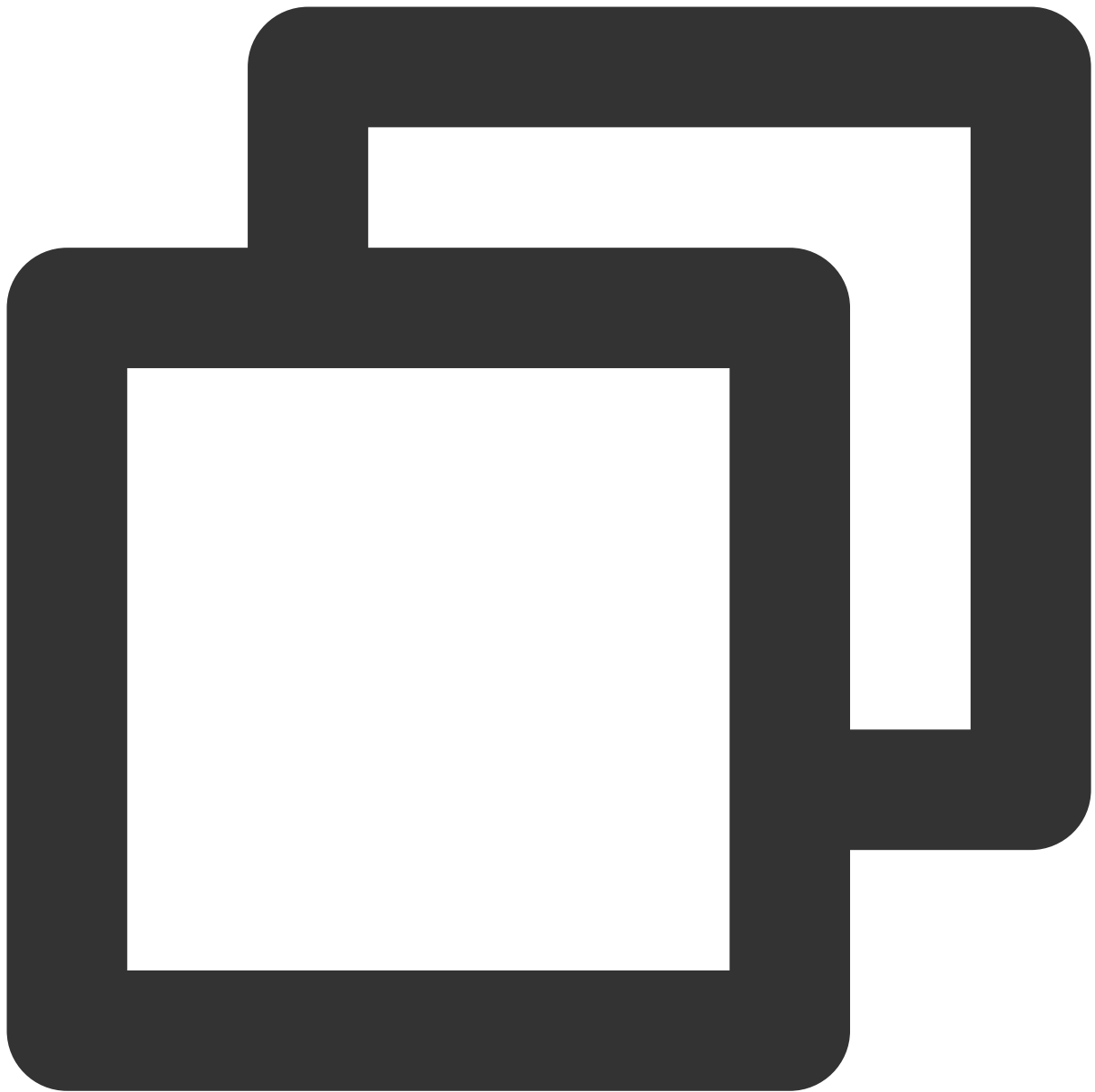
```
set global max_allowed_packet = 4*1024*1024;
```

3. 查看配置是否生效。



```
show global variables like '%max_allowed_packet%';
```

系统显示结果类似如下：



```
mysql> show global variables like '%max_allowed_packet%';
+-----+
| Variable_name      | Value      |
+-----+
| max_allowed_packet | 4194304    |
+-----+
1 row in set (0.00 sec)
```

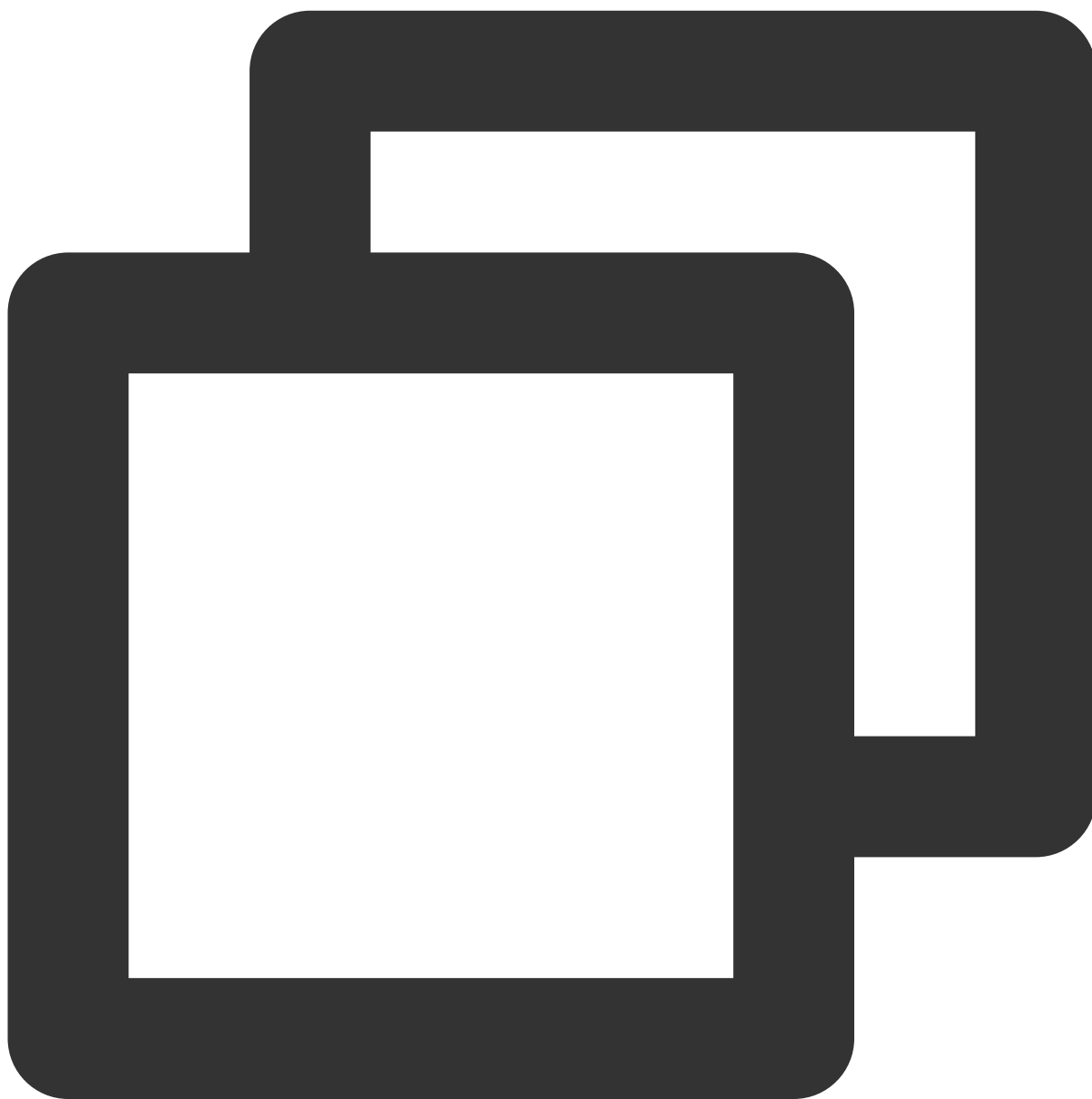
4. 重新执行校验任务。

修改源库变量 `connect_timeout`

`connect_timeout` 为数据库的连接时间，超过 `connect_timeout` 设置值的连接请求将会被拒绝。如果设置过小，会导致数据库连接频繁断开，影响处理效率，因此建议该参数取值大于10。

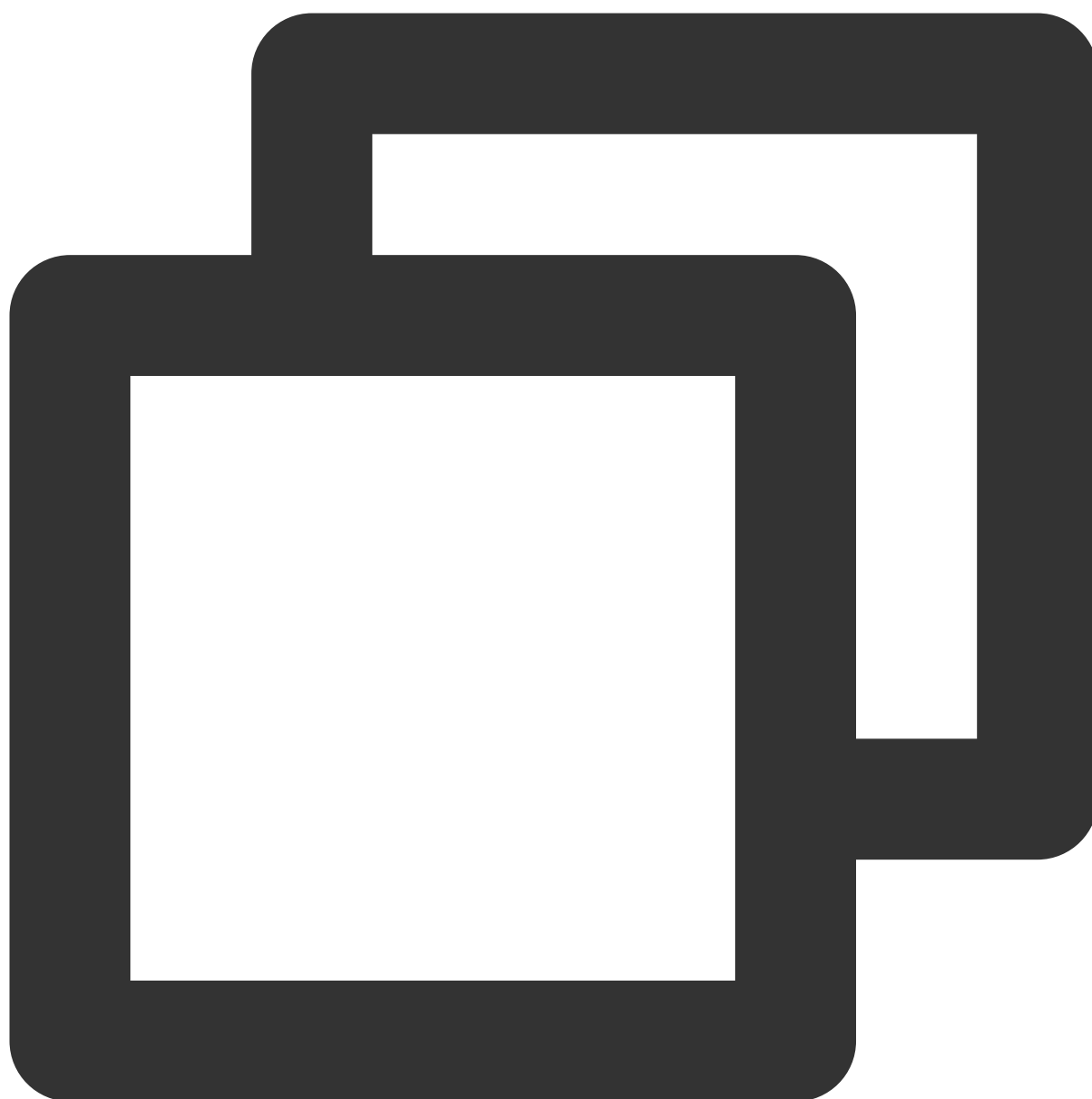
如发生此类报错，请参考如下指导进行修复：

1. 登录源数据库。
2. 修改 `connect_timeout` 参数。



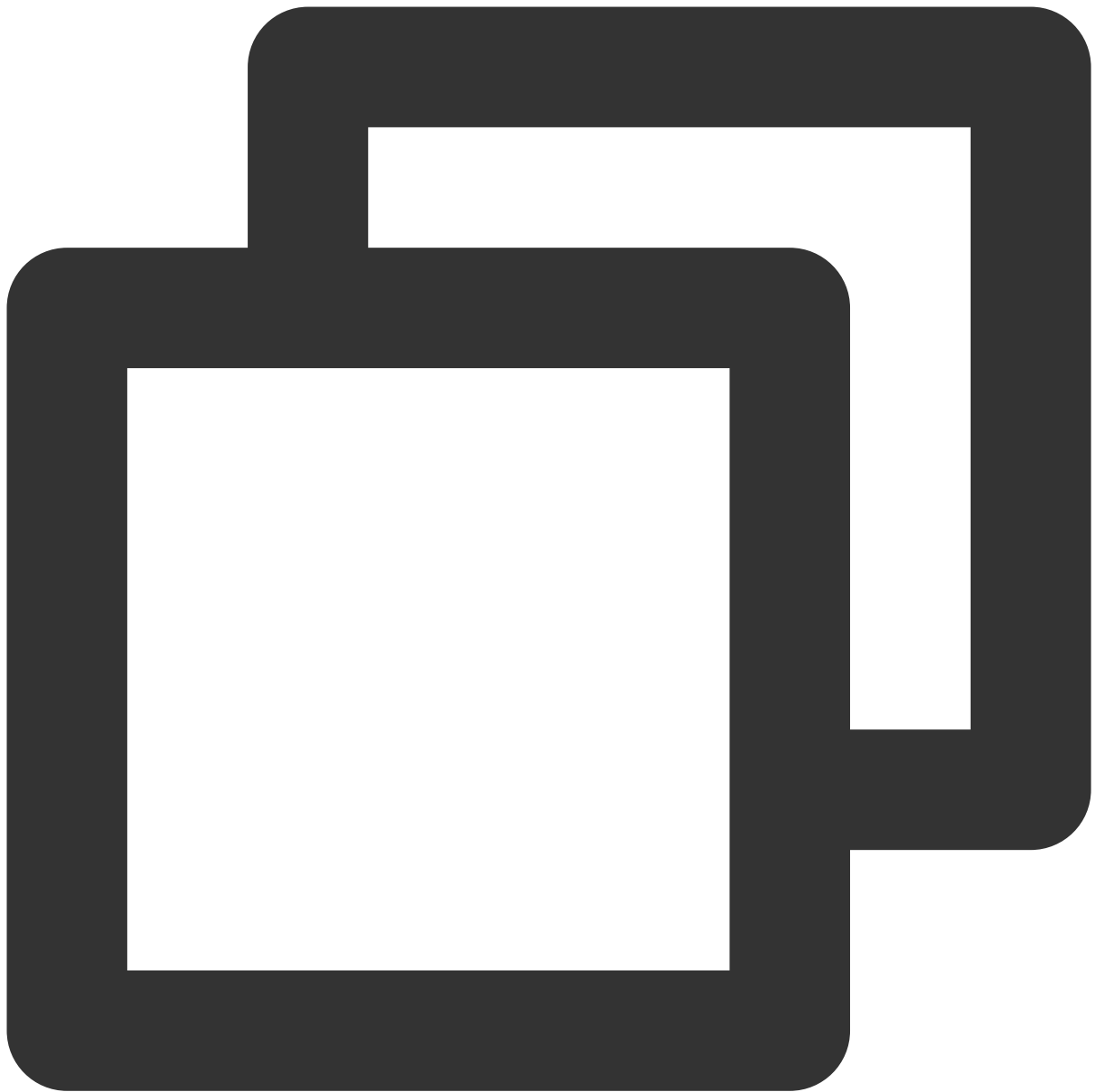
```
set global connect_timeout = 10;
```

3. 查看配置是否生效。



```
show global variables like '%connect_timeout%';
```

系统显示结果类似如下：



```
mysql> show global variables like '%connect_timeout%';
+-----+
| Variable_name      | Value  |
+-----+
| connect_timeout    | 10     |
+-----+
1 row in set (0.00 sec)
```

4. 重新执行校验任务。

目标实例内容冲突检查

最近更新时间：2024-07-08 17:51:43

检查详情

目标实例不能有和源库同名的对象。如果存在冲突报错，可任选以下一个方法进行修复。

方法一：[使用库表映射](#)。

方法二：[修改或删除目标数据库中的同名对象](#)。

方法三：[从迁移对象中移除同名对象](#)。

整个实例迁移时，目标实例必须为空。如果存在冲突报错，需要删除实例内容。

选择高级对象时，目标库不能有冲突的高级对象。如果存在冲突报错，需要删除冲突的对象。

修复方法

使用库表映射（仅适用于 MySQL/MariaDB/Percona/TDSQL-C MySQL/TDSQL MySQL）

使用 DTS 库表映射功能，将同名的待迁移对象映射为目标数据库中的其他名称。

1. 登录 [DTS 控制台](#)，选择对应的迁移任务，在**操作列**，选择**更多 > 修改**。
2. 在选择迁移对象右侧“已选对象”中，将鼠标悬浮在需要修改的对象上即可显示编辑按钮，然后重命名对象。
3. 重新执行校验任务。

修改目标数据库中的同名对象

登录目标数据库，重命名或删除目标数据库中和迁移对象同名的对象。

从迁移对象中移除同名对象

修改迁移任务配置，从迁移对象中移除同名对象，该对象不进行数据迁移。

1. 登录 [DTS 控制台](#)，选择对应的迁移任务，在**操作列**，选择**更多 > 修改**。
2. 在迁移对象中，移除同名的对象。
3. 重新执行校验任务。

删除目标库中的内容

登录目标数据库，删除目标数据库中的同名对象或者整库内容，然后重新执行校验任务。

目标实例空间检查

最近更新时间：2024-07-08 17:51:43

检查详情

目标库存储空间需要在源库待迁移库表空间的1.2倍以上。

全量数据迁移会并发执行 **INSERT** 操作，导致目标数据库的表产生碎片，因此全量迁移完成后目标数据库的表存储空间很可能会比源实例的表存储空间大。

修复方法

删除目标库中的部分数据，以便腾出足够的空间。

[升级目标库存储规格](#)，使用更大容量的实例进行迁移。

Binlog 参数检查

最近更新时间：2024-07-08 17:51:43

检查详情

源数据库 binlog 相关参数需要按照如下要求配置，如果校验不通过，请参考本文后续指导进行修复。

`log_bin` 变量必须设置为 `ON`。

`binlog_format` 变量必须设置为 `ROW`。

`binlog_row_image` 必须设置为 `FULL`。

如果源数据库为 MySQL 5.6 及以上版本，`gtid_mode` 只支持设置为 `ON` 和 `OFF`，建议将 `gtid_mode` 设置为 `ON`，设置为 `OFF` 会报警告，设置为 `ON_PERMISSIVE` 和 `OFF_PERMISSIVE` 会报错。

`server_id` 参数需要手动设置，且值不能设置为0。

不允许设置 `do_db`，`ignore_db`。

对于源实例为从库的情况，`log_slave_updates` 变量必须设置为 `ON`。

建议源库 Binlog 日志至少保留3天及以上，否则可能会因任务暂停/中断时间大于 Binlog 日志保留时间，造成任务无法续传，进而导致任务失败。

修复方法

开启 binlog

`log_bin` 是 binlog 的开关控制参数，需要将 binlog 打开，以便记录所有的数据库表结构和表数据变更日志。

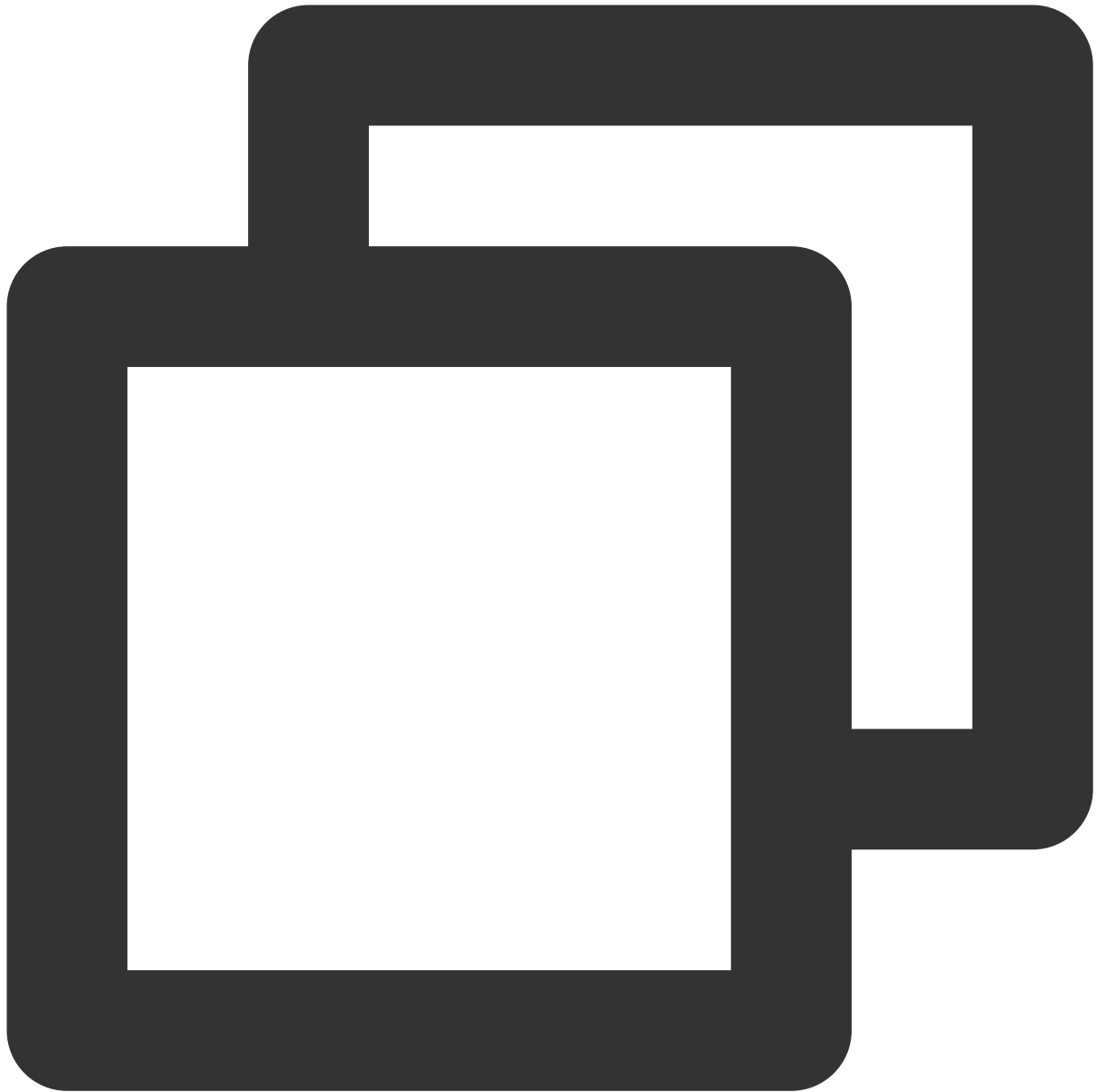
如发生类似报错，请参考如下指导进行修复：

1. 登录源数据库。
2. 参考如下内容修改源数据库的配置文件 `my.cnf`。

因 `log_bin` 参数修改后需要重启数据库后才能生效，如果 `binlog_format`、`server_id`、`binlog_row_image`、`expire_logs_days` 这几个参数也在校验阶段提示报错，请一并修改完成后再重启数据库，让所有参数都生效。

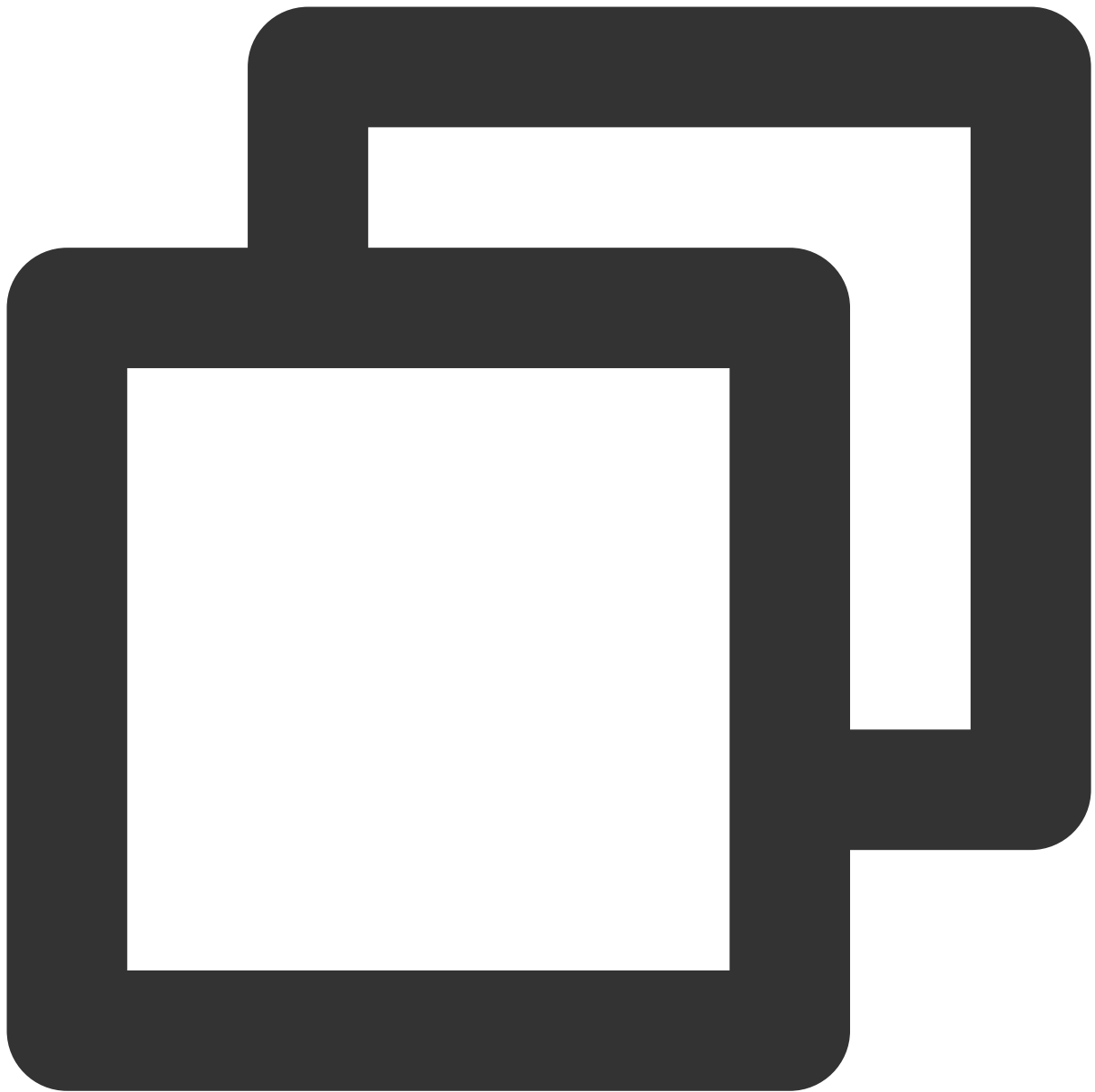
说明

`my.cnf` 配置文件的默认路径为 `/etc/my.cnf`，现场以实际情况为准。



```
log_bin = MYSQL_BIN
binlog_format = ROW
server_id = 2      //建议设为大于1的整数，此处仅为示例
binlog_row_image = FULL
expire_logs_days = 3    //修改 binlog 的保留时间，建议大于等于3天
```

3. 参考如下命令重启源数据库。

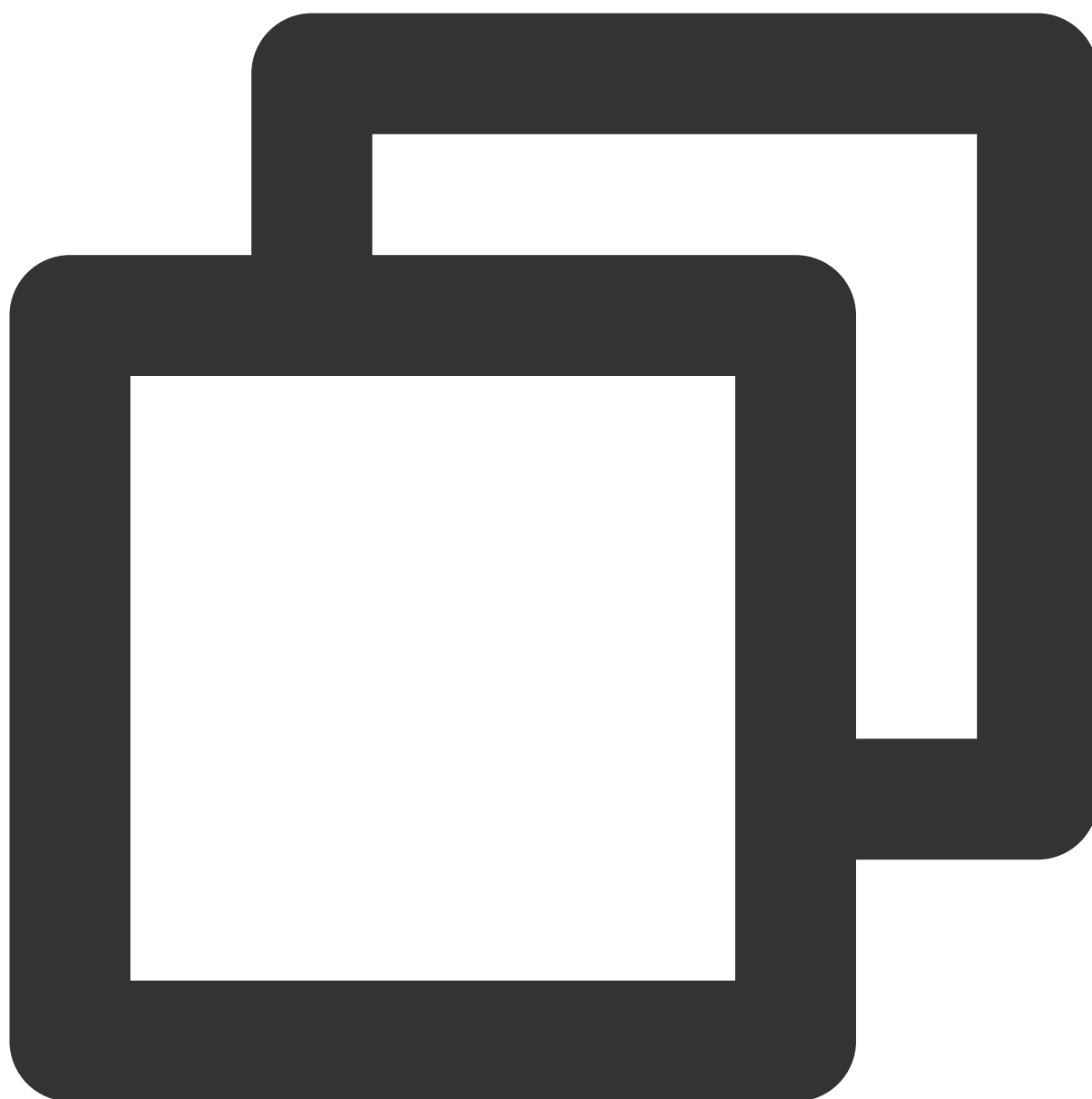


```
[$Mysql_Dir]/bin/mysqladmin -u root -p shutdown  
[$Mysql_Dir]/bin/safe_mysqld &
```

说明

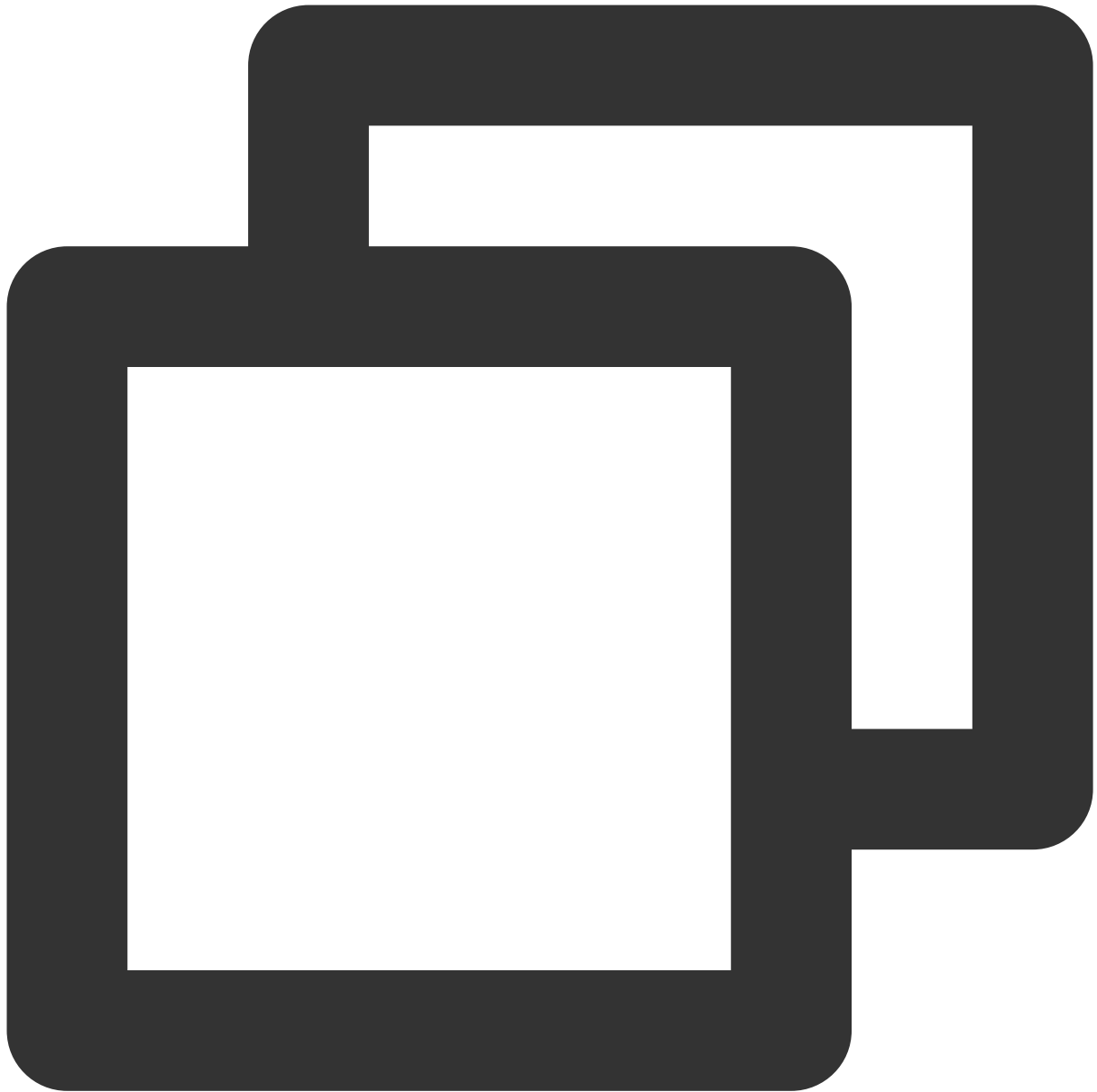
[\$Mysql_Dir] 指源数据库的安装路径，请替换为实际的源数据库安装目录。

4. 确认 binlog 功能是否已启用。



```
show variables like '%log_bin%';
```

系统显示结果类似如下：



```
mysql> show variables like '%log_bin%';
```

```
+-----+-----+
| Variable_name | Value |
+-----+-----+
| log_bin       | ON    |
+-----+-----+
| binlog_format | ROW   |
+-----+-----+
| binlog_row_image | FULL |
+-----+-----+
1 row in set (0.00 sec)
```


5. 重新执行校验任务。

修改 binlog_format 参数

`binlog_format` 为 binlog 的记录模式，有以下三种：

`STATEMENT`：每一条会修改数据的 SQL 都会记录到 master 的 binlog 中，slave 在复制的时候，会执行和原来 master 端相同的 SQL。该模式可以减少 binlog 日志量，但是对某些特定的函数进行复制时，slave 端不能正确复制。

`ROW`：binlog 日志中会记录成每一行数据修改的形式，然后在 slave 端再对相同的数据进行修改。该模式会保证 master 和 slave 的正确复制，但是 binlog 日志量会增加。

`MIXED`：前两种模式的结合，MySQL 会根据执行的每一条具体的 SQL 语句来区分对待记录的日志形式，在

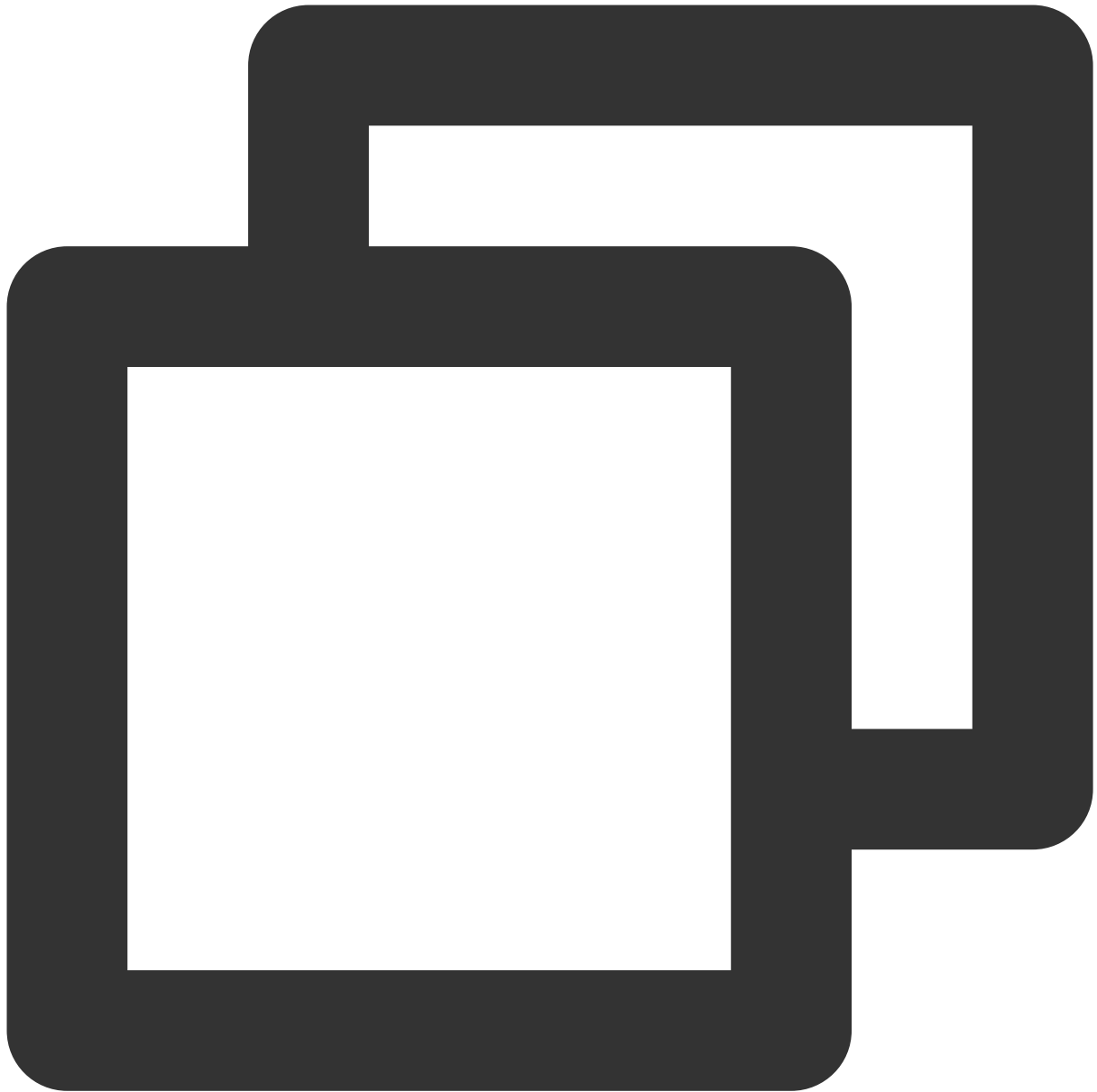
`STATEMENT` 和 `ROW` 之间选择一种。

综上，为了保证 master 和 slave 的正确复制，`binlog_format` 参数需要设置为 `ROW`。如发生类似报错，请参考如下指导进行修复。

说明

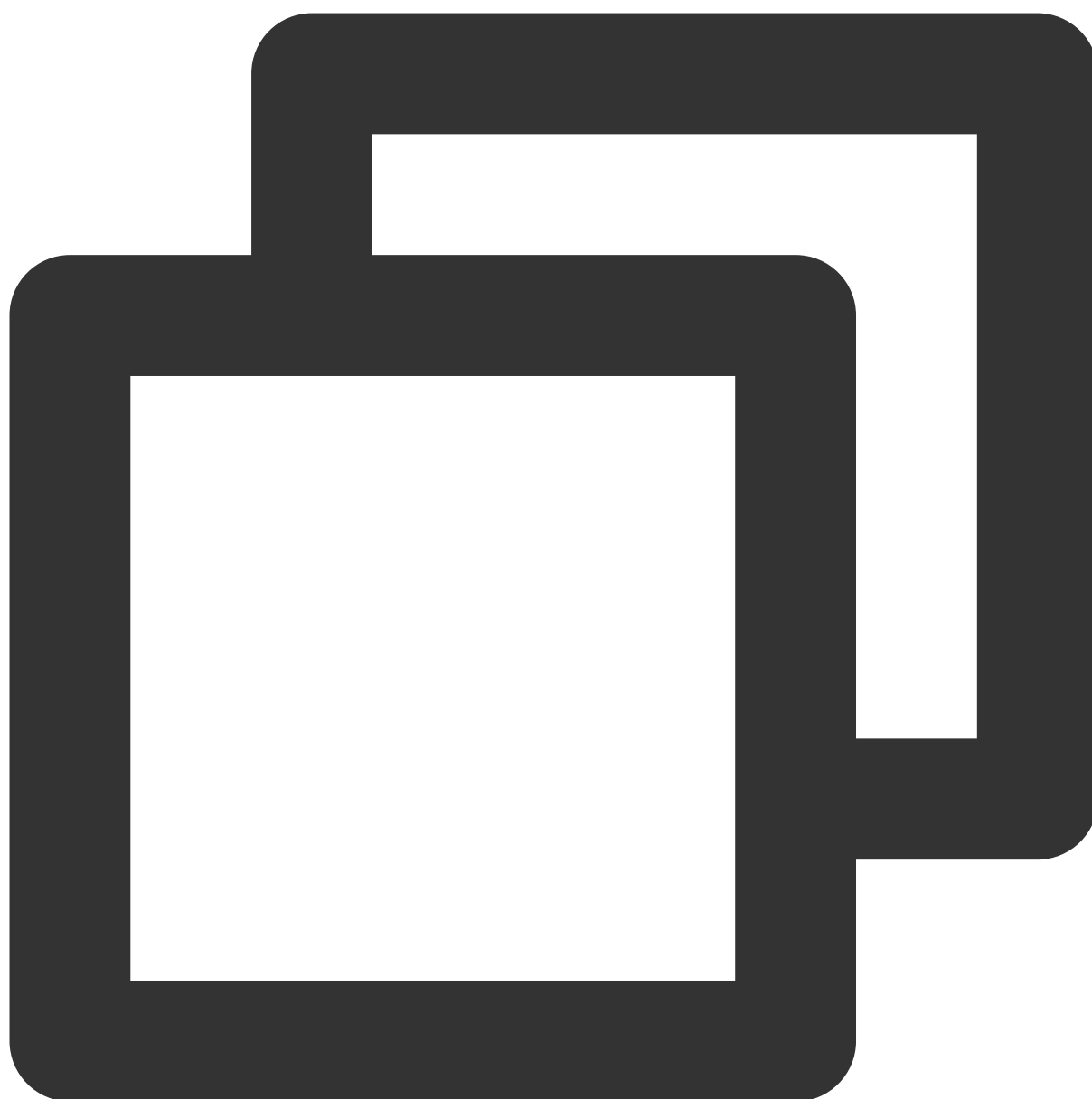
该参数修改需要重置数据库上的所有连接才能生效，当源库为从库时，还需重启主从同步 SQL 线程，避免当前业务连接继续使用修改前的模式写入。

1. 登录源数据库。
2. 参考如下命令修改 `binlog_format`。



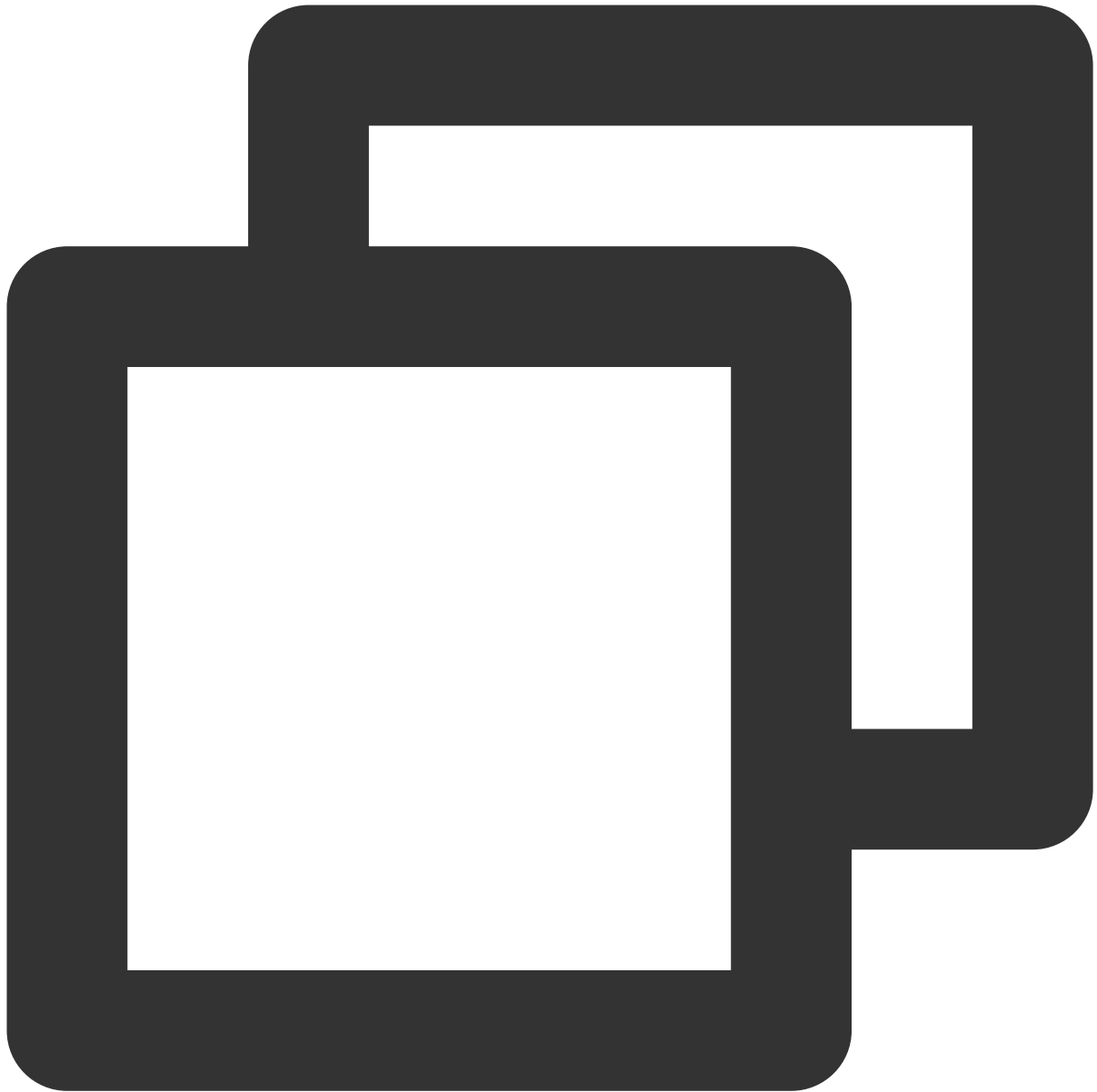
```
set global binlog_format = ROW;
```

3. 重启线程使配置生效，然后通过如下命令查看参数修改是否生效。



```
show variables like '%binlog_format%';
```

系统显示结果类似如下：



```
mysql> show variables like '%binlog_format%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| binlog_format | ROW   |
+-----+-----+
1 row in set (0.00 sec)
```

4. 重新执行校验任务。

修改 binlog_row_image 参数

`binlog_row_image` 参数决定了 `binlog` 是如何记录前镜像（记录修改前的内容）和后镜像（记录修改后的内容）的，这会直接影响到数据闪回、主从复制等功能。

`binlog_row_image` 参数只在 `binlog_format` 配置为 `ROW` 模式下生效。具体取值影响如下：

`FULL` ：在 `ROW` 模式下，`binlog` 会记录前镜像和后镜像的所有列的数据信息。

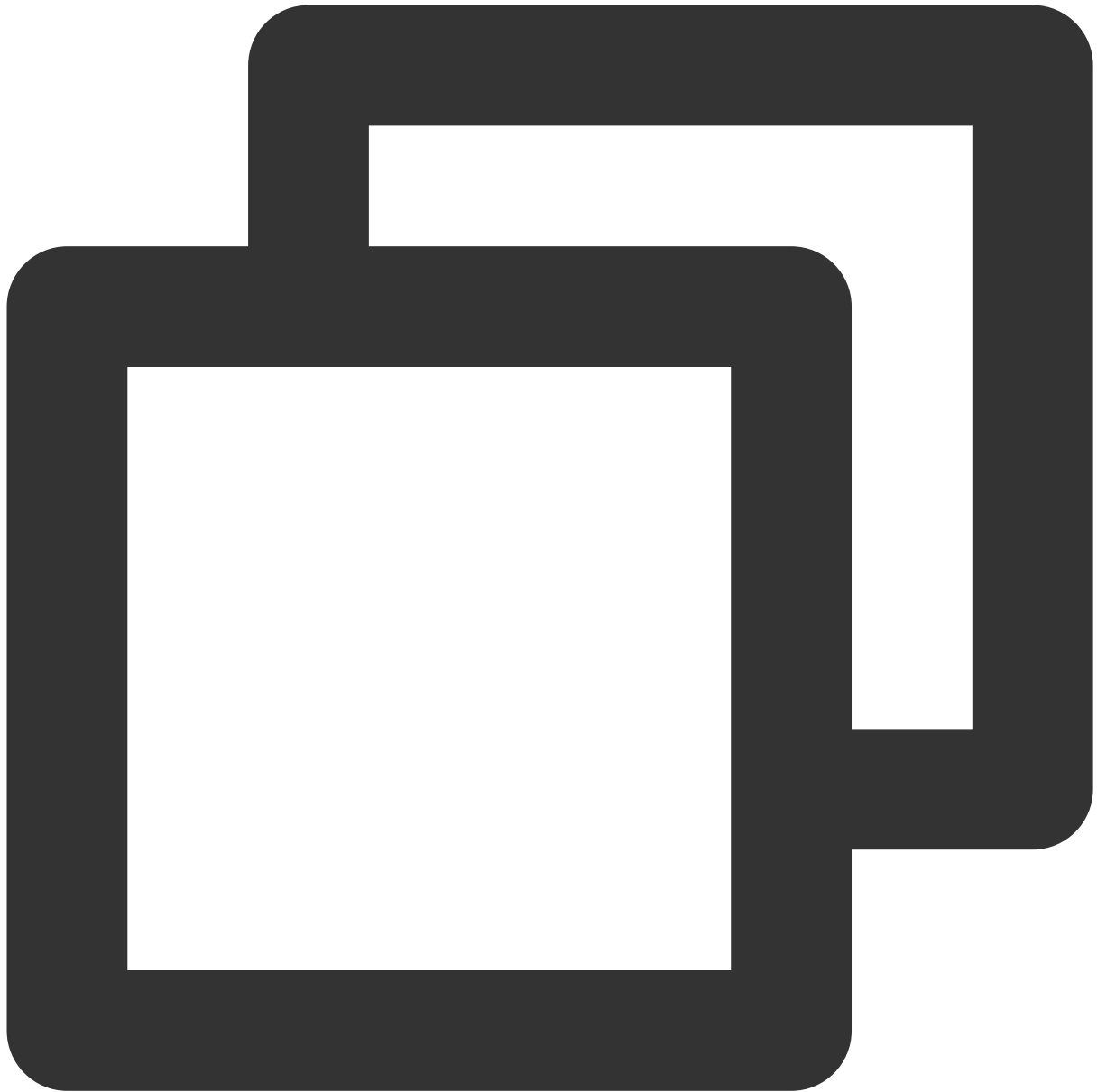
`MINIMAL` ：在 `ROW` 模式下，当表没有主键或唯一键时，前镜像记录所有列，后镜像记录被修改的列；如果存在主键或唯一键，不管是前镜像还是后镜像，都只记录有影响的列。

综上，`binlog_row_image` 需要配置为 `FULL`，源数据库的 `binlog` 记录全镜像。如发生报错，请参考如下步骤进行修复。

说明

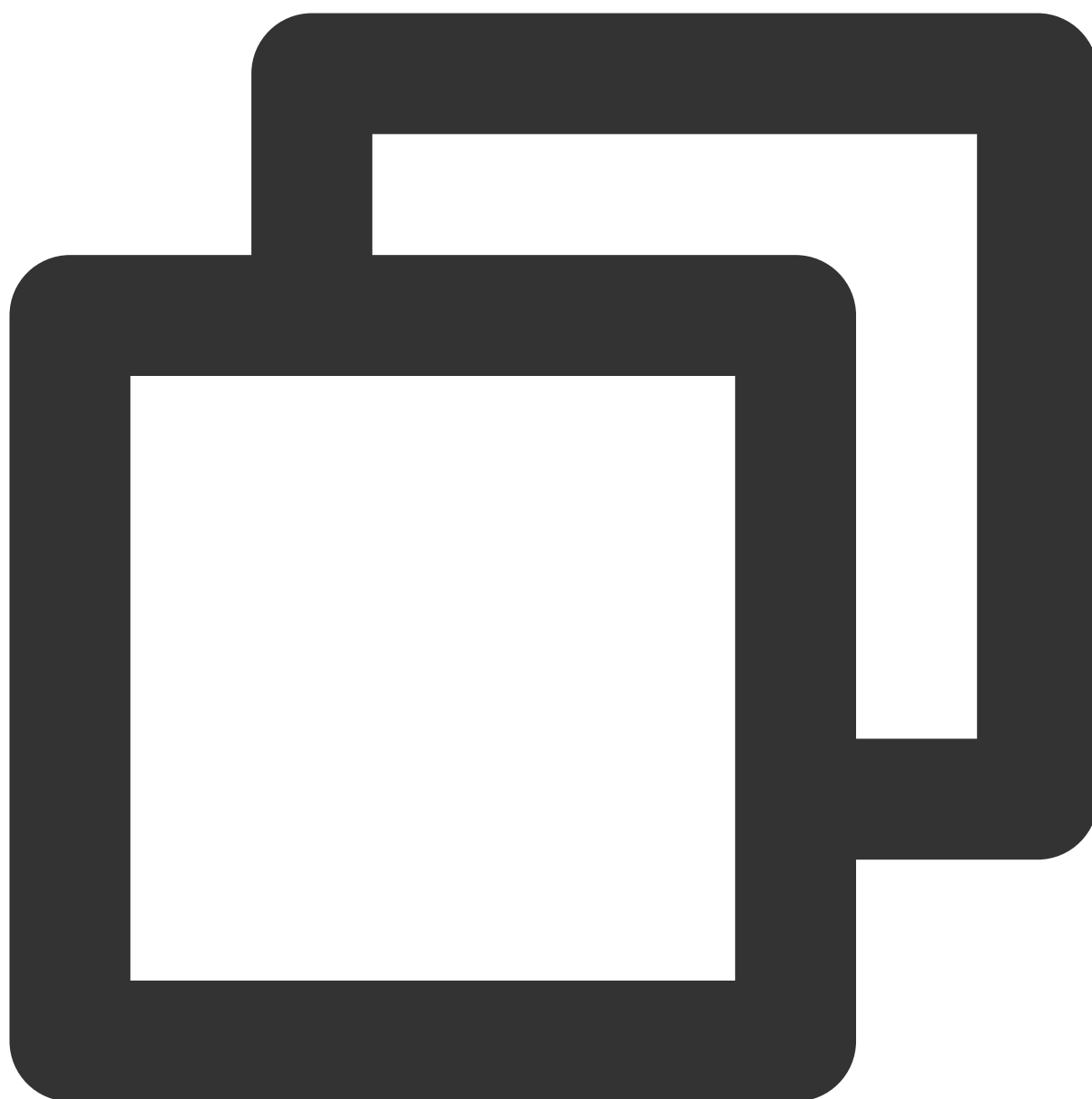
该参数修改需要重置数据库上的所有连接才能生效，当源库为从库时，还需重启主从同步 `SQL` 线程，避免当前业务连接继续使用修改前的模式写入。

1. 登录源数据库。
2. 参考如下内容修改 `binlog_row_image`。



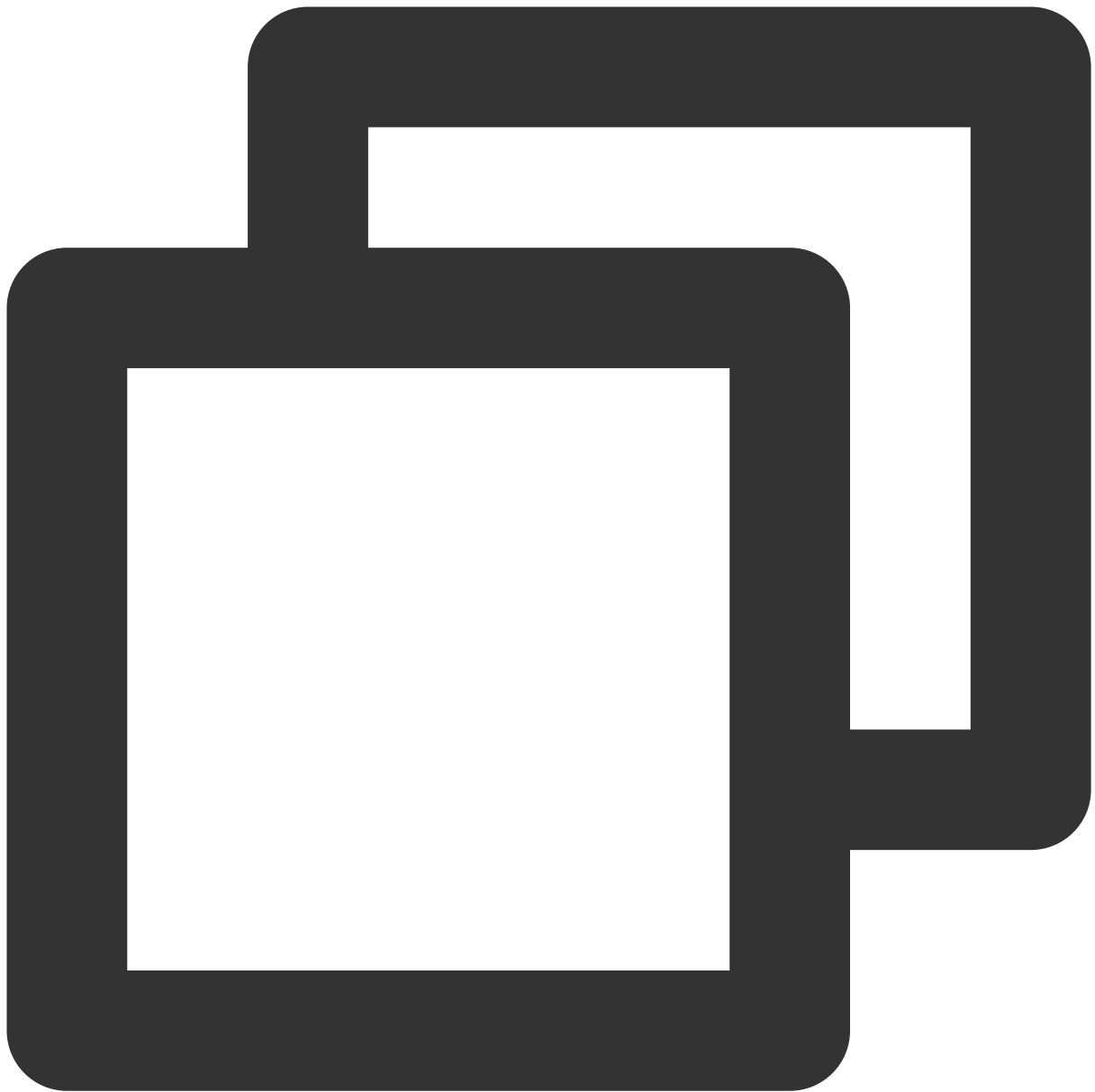
```
set global binlog_row_image = FULL;
```

3. 重启线程使配置生效，然后通过如下命令查看参数修改是否生效。



```
show variables like '%binlog_row_image%';
```

系统显示结果类似如下：



```
mysql> show variables like '%binlog_row_image%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| binlog_row_image | FULL |
+-----+-----+
1 row in set (0.00 sec)
```

4. 重新执行校验任务。

修改 `gtid_mode` 参数

GTID (Global Transaction Identifier, 全局事务标识), 用于在 binlog 中唯一标识一个事务, 使用 GTID 可以避免事务重复执行导致数据混乱或者主从不一致。

GTID 是 MySQL 5.6 的新特性, 所以 MySQL 5.6 及之后版本存在此问题。DTS 只支持 `gtid_mode` 设置为 `ON` 和 `OFF`, 建议将 `gtid_mode` 设置为 `ON`, 否则校验时会报警告。

警告不影响迁移或同步任务进行, 但是会对业务造成一定的影响: 设置 GTID 后, 在增量数据同步阶段, 如果源实例发生 HA 切换, DTS 服务切换重连, 任务几乎无感知; 反之, 任务会中断后失败, 且不可恢复。

`gtid_mode` 的取值如下, 在修改 `gtid_mode` 的值时, 只能从以下四个值逐级修改, 例如, 需要从 `OFF` 修改为 `ON`, 则 `gtid_mode` 修改顺序为 `OFF <-> OFF_PERMISSIVE <-> ON_PERMISSIVE <-> ON`。

`OFF`: 主库所有新启的事务以及从库的事务都要求是匿名事务。

`OFF_PERMISSIVE`: 主库新启的事务是匿名事务, 但从库事务允许是匿名的或者是 GTID 事务, 但不允许只是 GTID 模式。

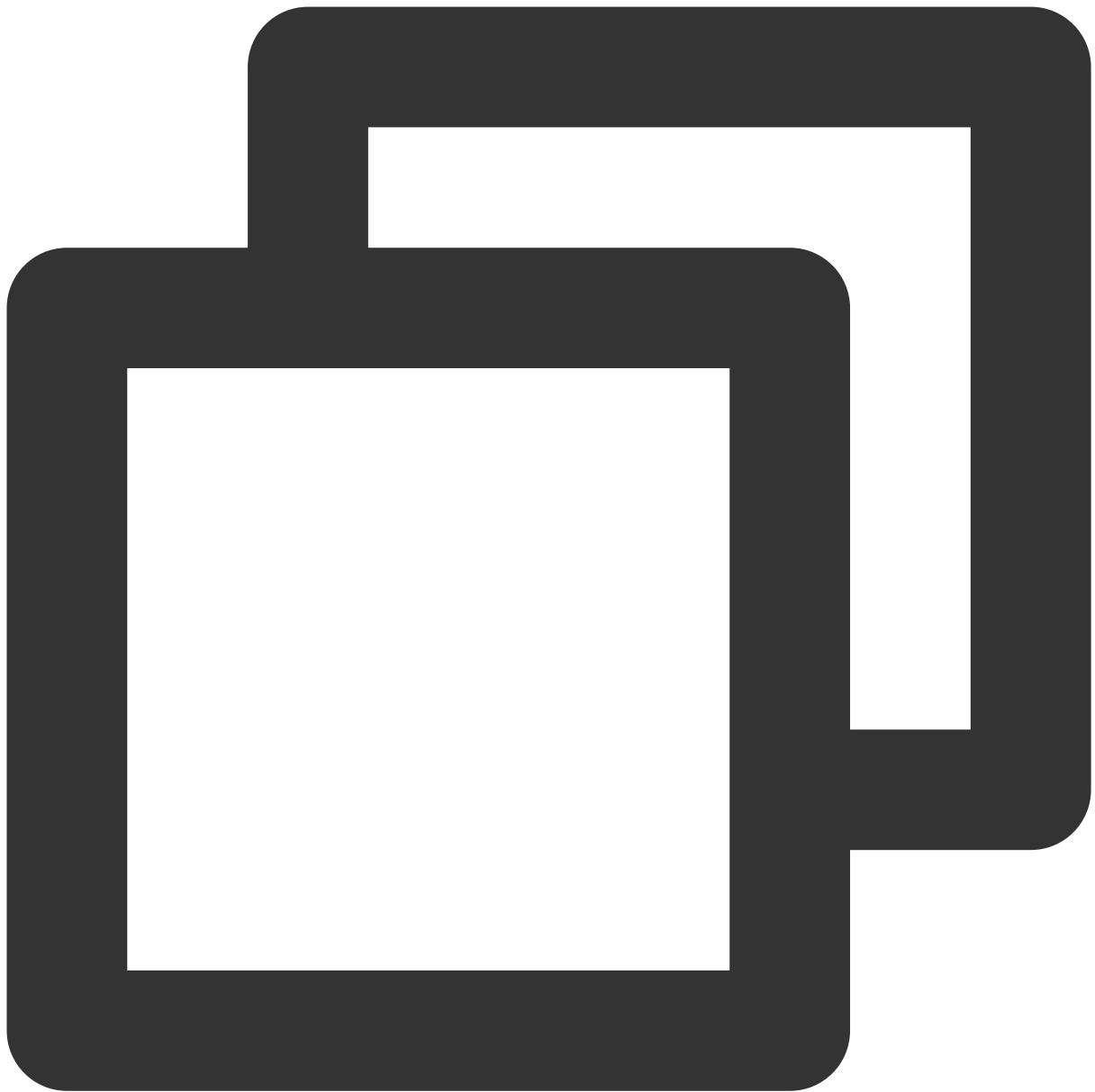
`ON_PERMISSIVE`: 主库新启的事务是 GTID 事务, 从库事务允许是匿名的或者是 GTID 事务。

`ON`: 主库新启的事务是 GTID 事务, 从库的事务也要求是 GTID 事务。

如果发生类似警告, 请按照如下指导进行修复:

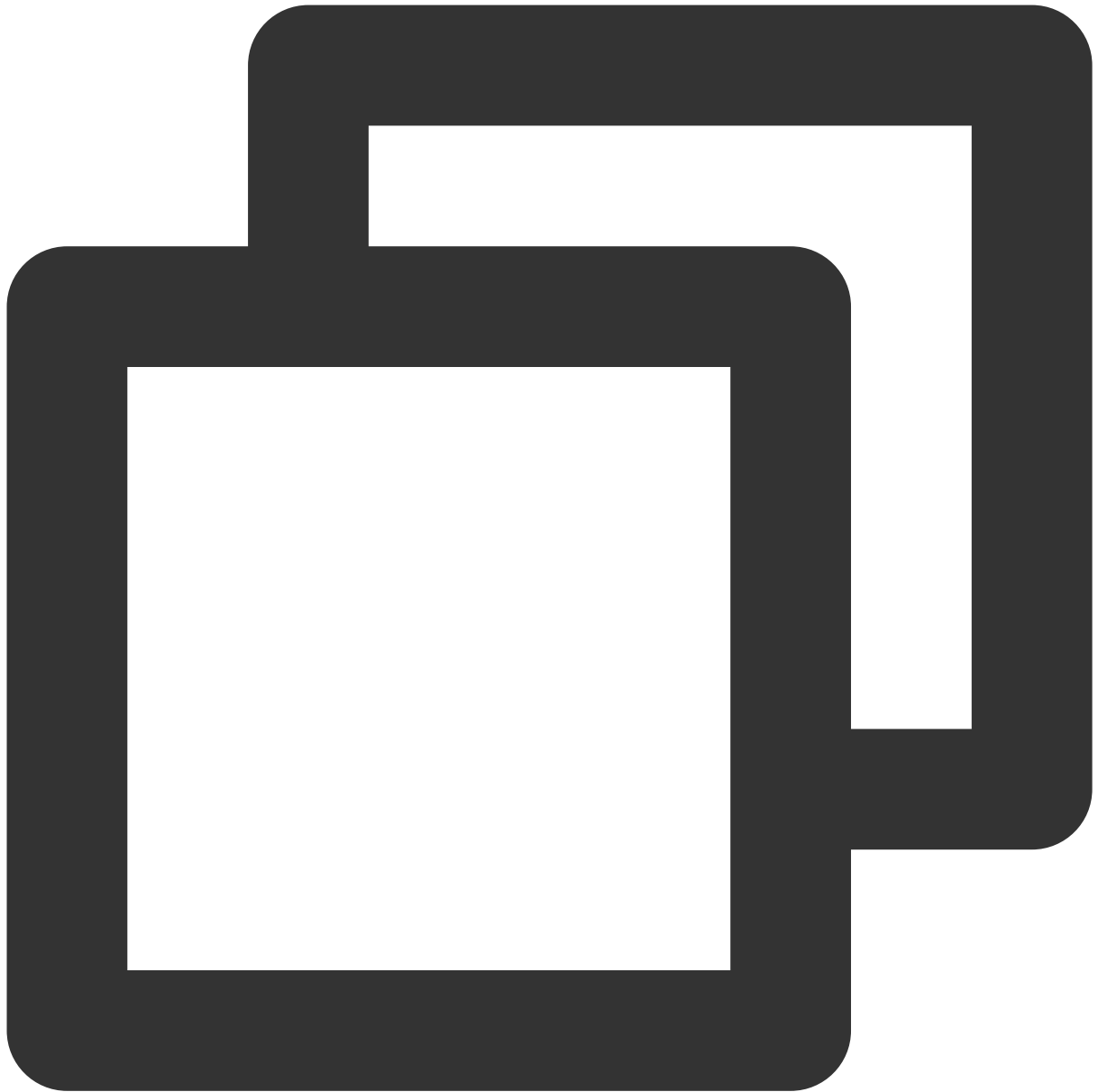
1. 登录源数据库。
2. 在主从复制结构两边都设置 `gtid_mode = OFF_PERMISSIVE`。

MySQL 5.7.6 之前的版本需要在 `my.cnf` 配置文件中修改并重启数据库才能生效, 5.7.6 及之后的版本通过全局命名修改, 不需要重启数据库, 但是需要重置所有业务连接。



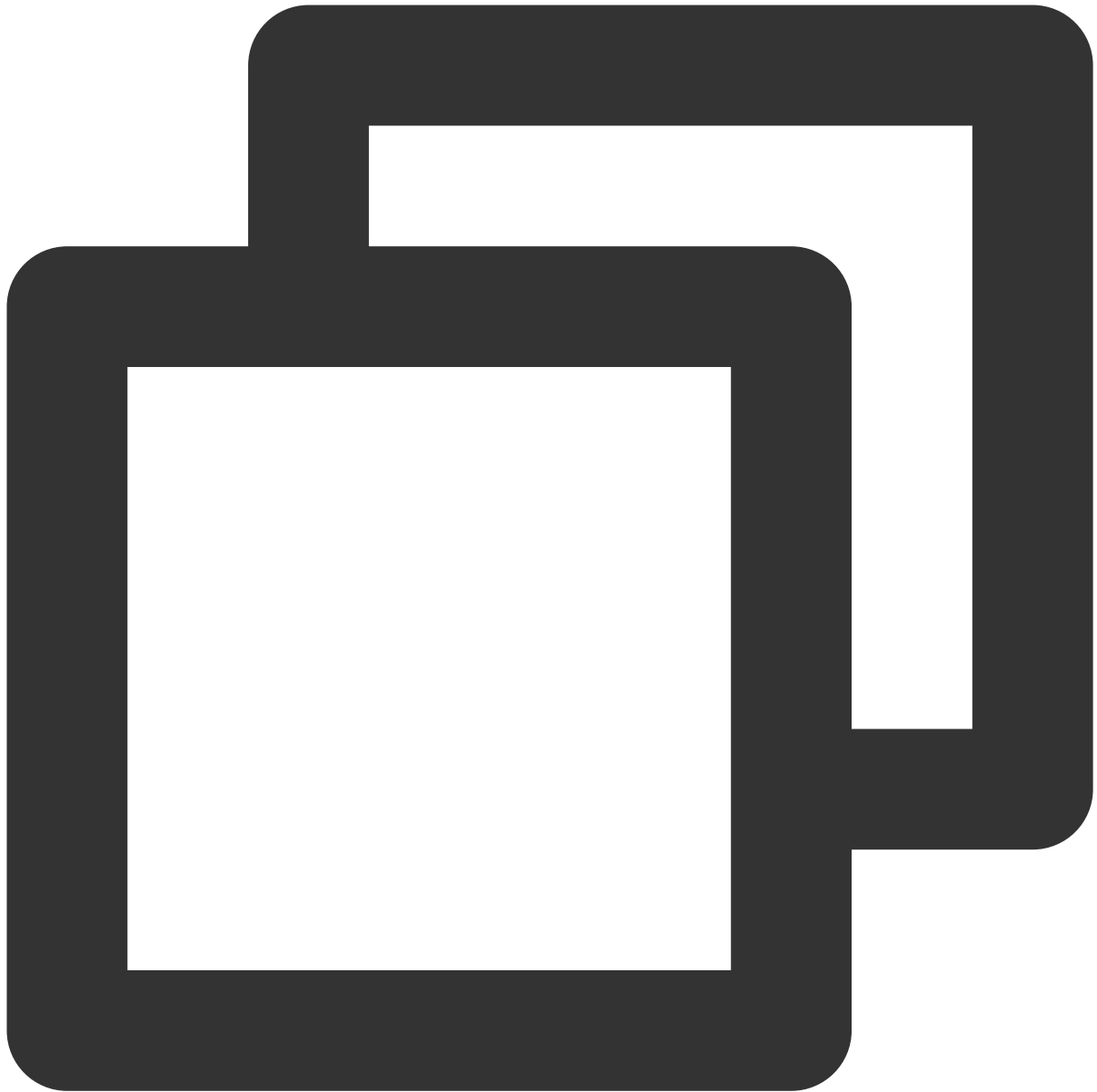
```
set global gtid_mode = OFF_PERMISSIVE;
```

3. 在主从复制结构两边都设置 `gtid_mode = ON_PERMISSIVE` 。



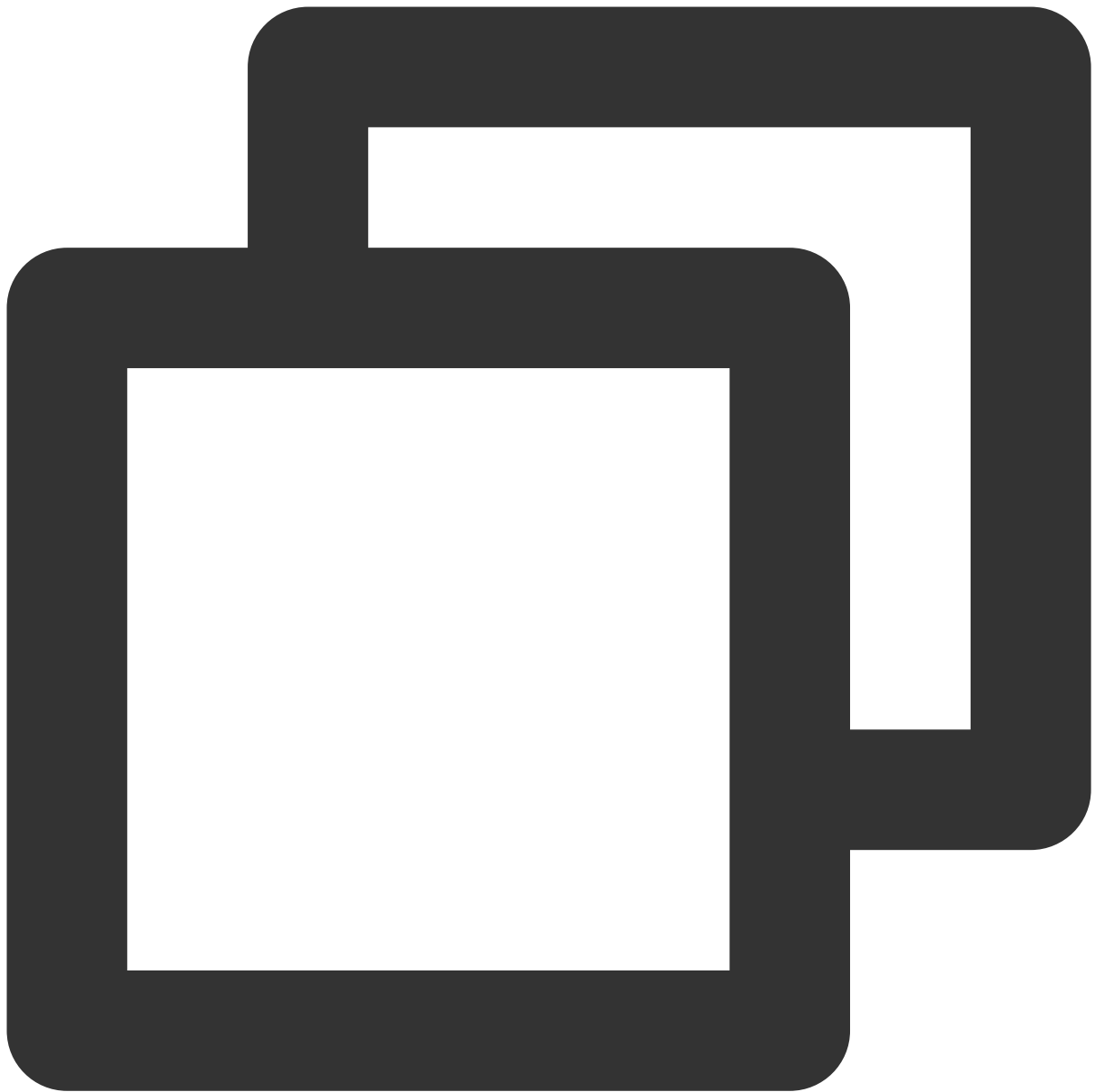
```
set global gtid_mode = ON_PERMISSIVE;
```

4. 在各个实例节点上执行如下命令，检查匿名事务是否消耗完毕，参数值为 0 则代表消耗完毕。



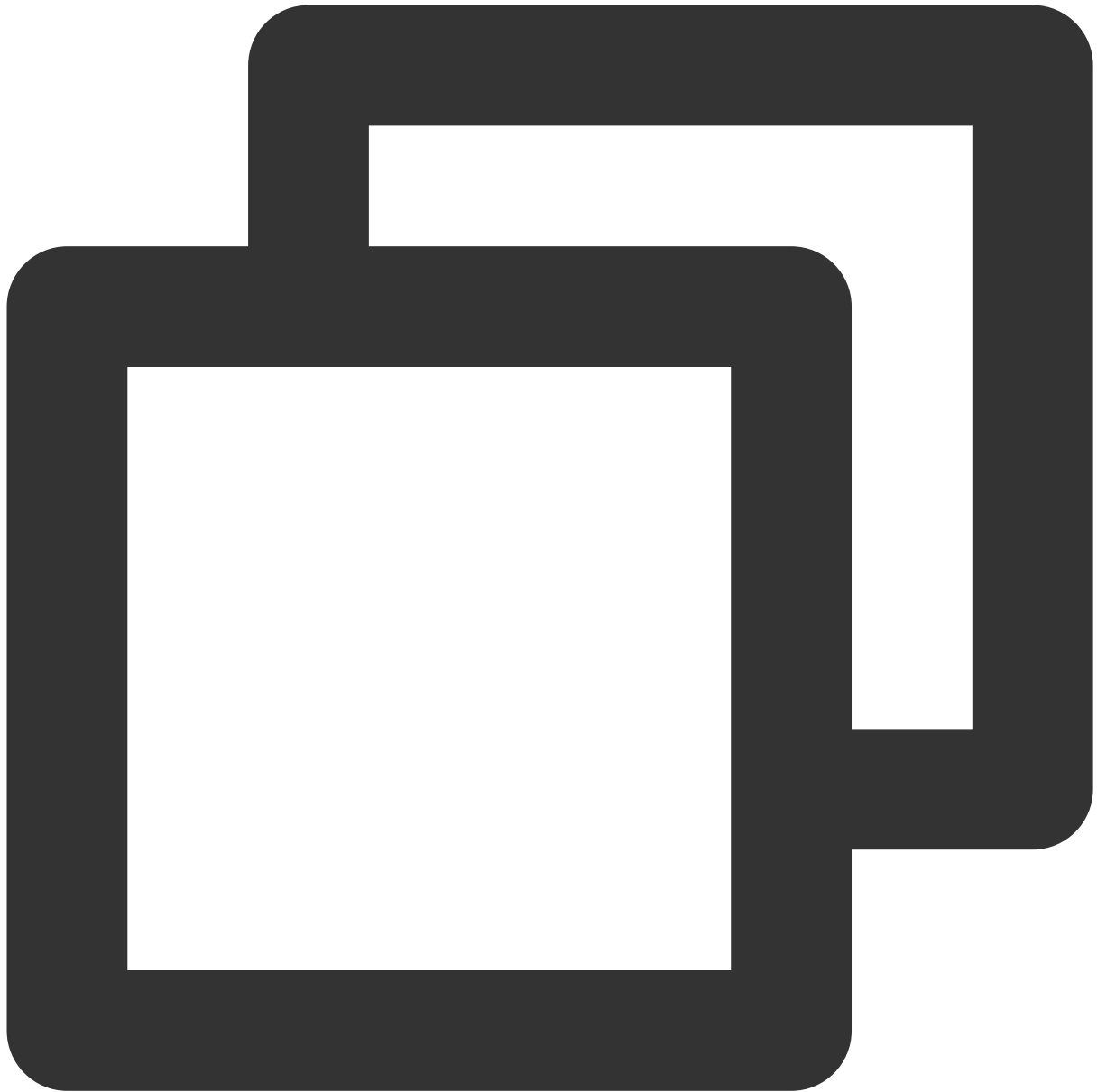
```
show variables like '%ONGOING_ANONYMOUS_TRANSACTION_COUNT%';
```

系统显示结果类似如下：



```
mysql> show variables like '%ONGOING_ANONYMOUS_TRANSACTION_COUNT%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Ongoing_anonymous_transaction_count | 0 |
+-----+-----+
1 row in set (0.00 sec)
```

5. 在主从复制结构两边都设置 `gtid_mode = ON`。

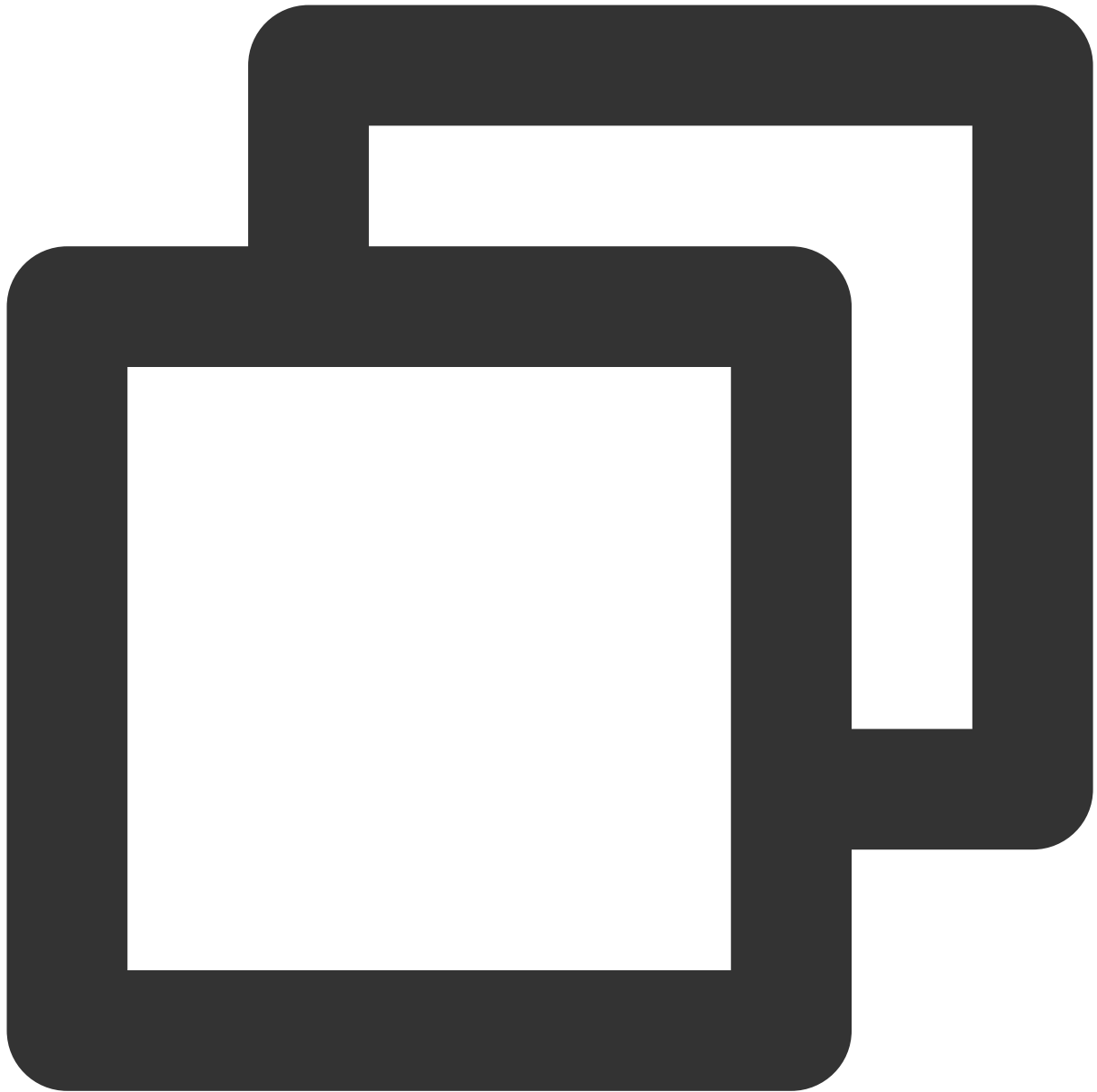


```
set global gtid_mode = ON;
```

6. 在 `my.cnf` 文件中添加如下内容，后续重启数据库后使初始值生效。

说明

`my.cnf` 配置文件的默认路径为 `/etc/my.cnf`，现场以实际情况为准。



```
gtid_mode = on
enforce_gtid_consistency = on
```

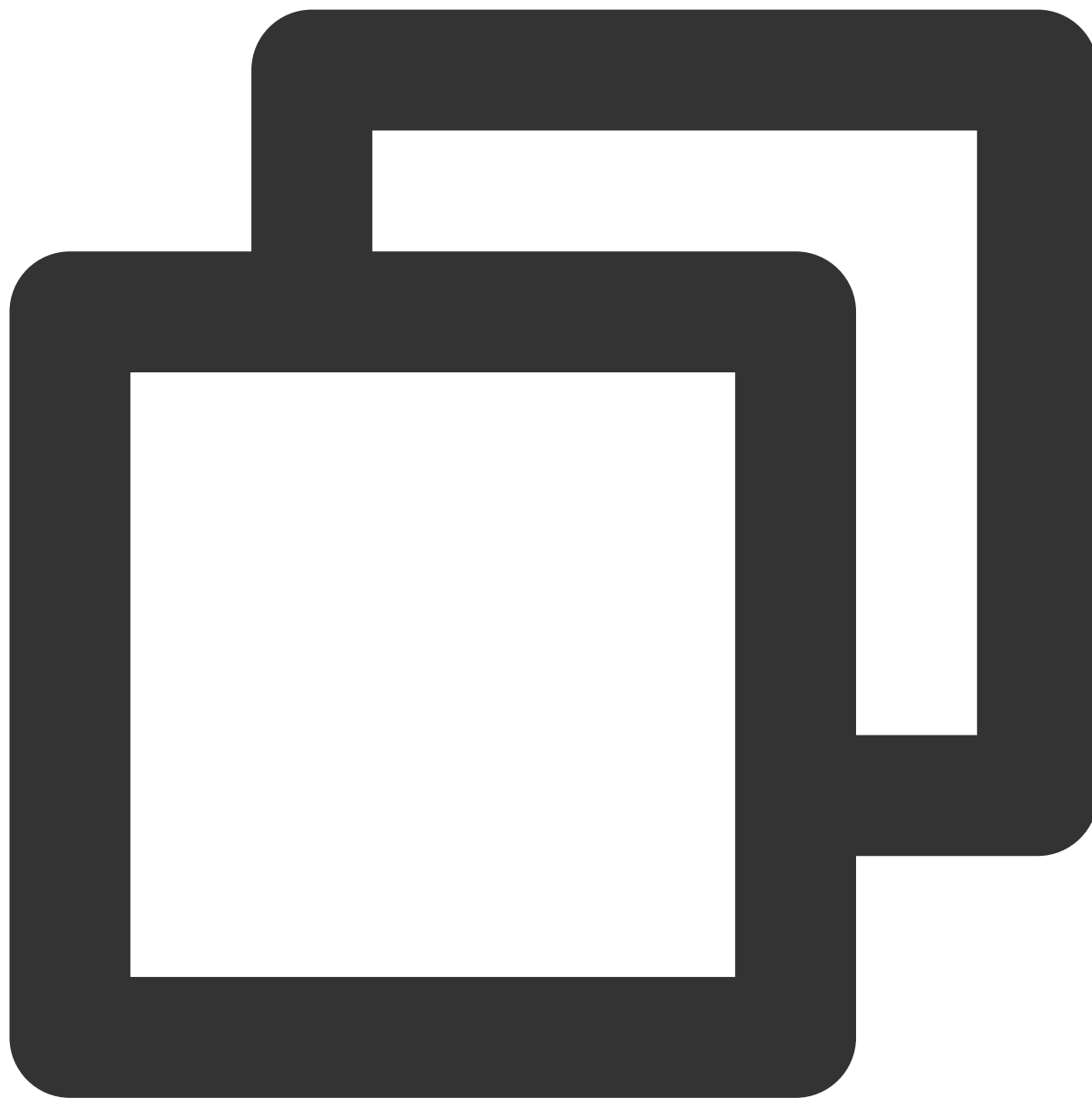
7. 重新执行校验任务。

修改 `server_id` 参数

`server_id` 参数需要手动设置，且值不能设置为0。该参数系统预设值为 `1`，如果查询该参数显示为 `1` 不一定正确，需要手动进行配置。

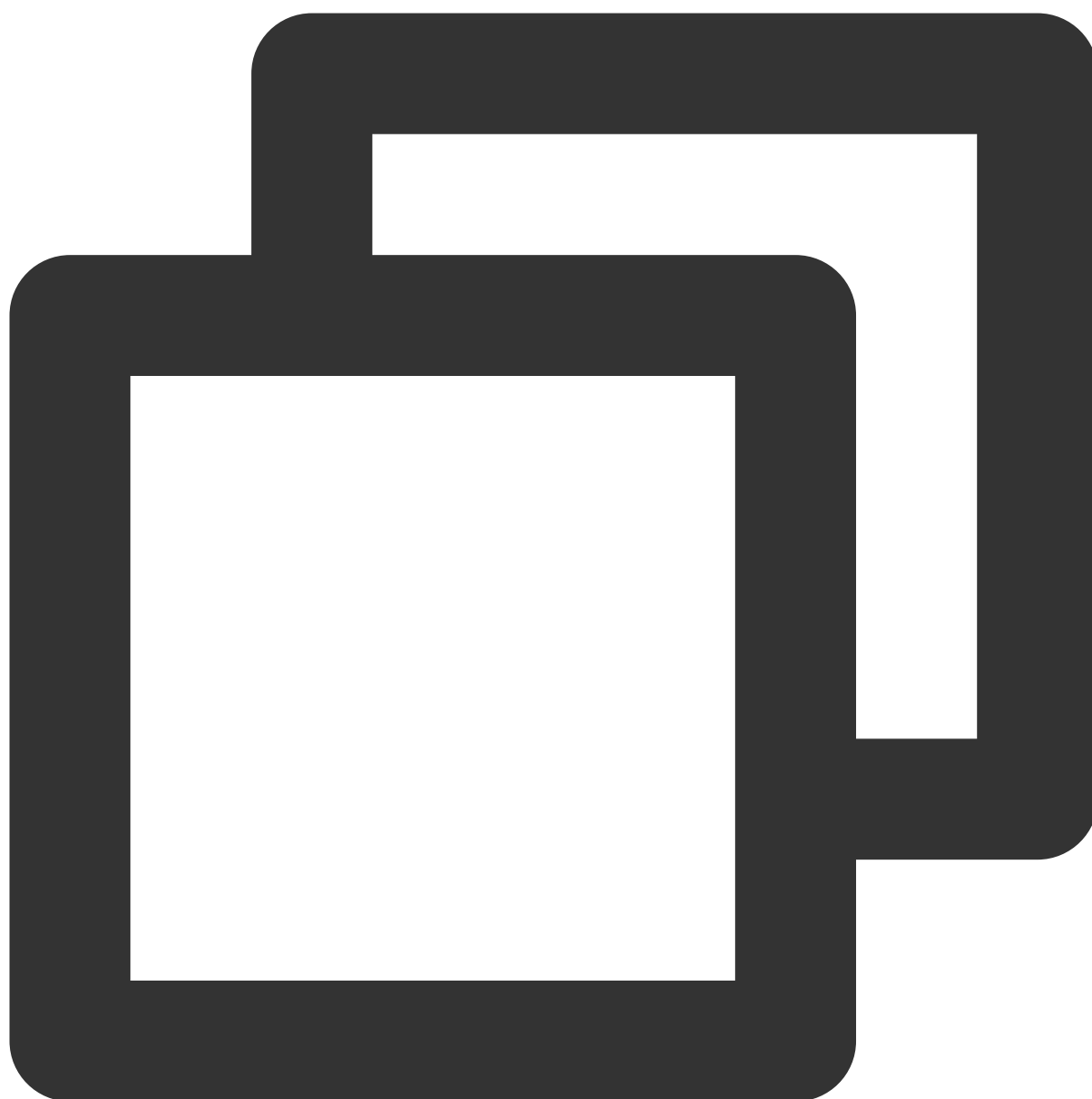
1. 登录源数据库。

2. 参考如下内容修改 `server_id` 。



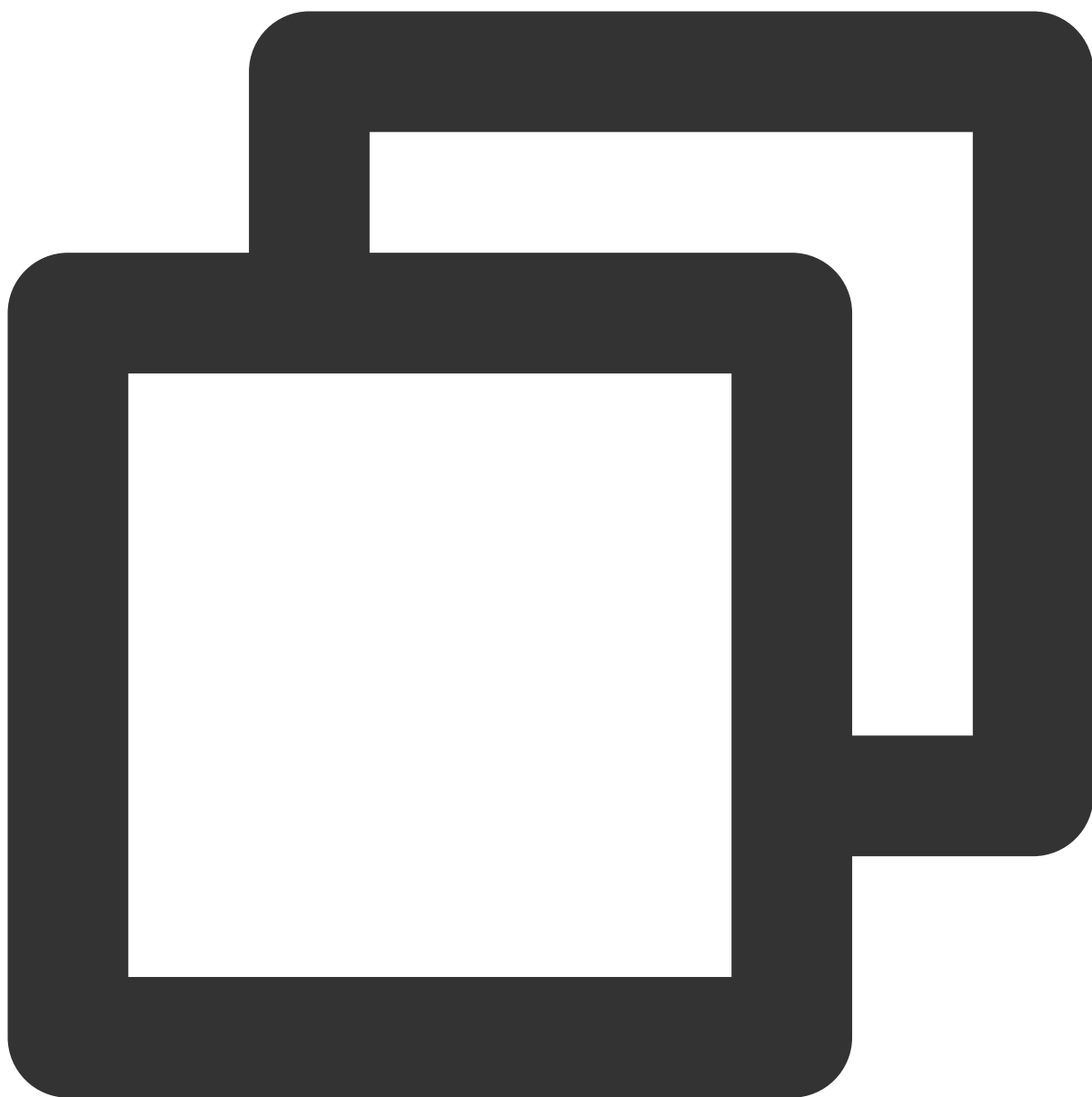
```
set global server_id = 2;  //建议设为大于1的整数，此处仅为示例
```

3. 通过如下命令查看参数修改是否生效。



```
show global variables like '%server_id%';
```

系统显示结果类似如下：



```
mysql> show global variables like '%server_id%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| server_id     | 2     |
+-----+-----+
1 row in set (0.00 sec)
```

4. 重新执行校验任务。

删除 do_db, ignore_db 设置

binlog 会记录数据库所有执行的 DDL 和 DML 语句，而 `do_db`，`ignore_db` 则是设置 binlog 记录的过滤条件。

`binlog_do_db` ：只记录指定数据库的二进制日志，默认全部记录。

`binlog_ignore_db` ：不记录指定的数据库的二进制日志。

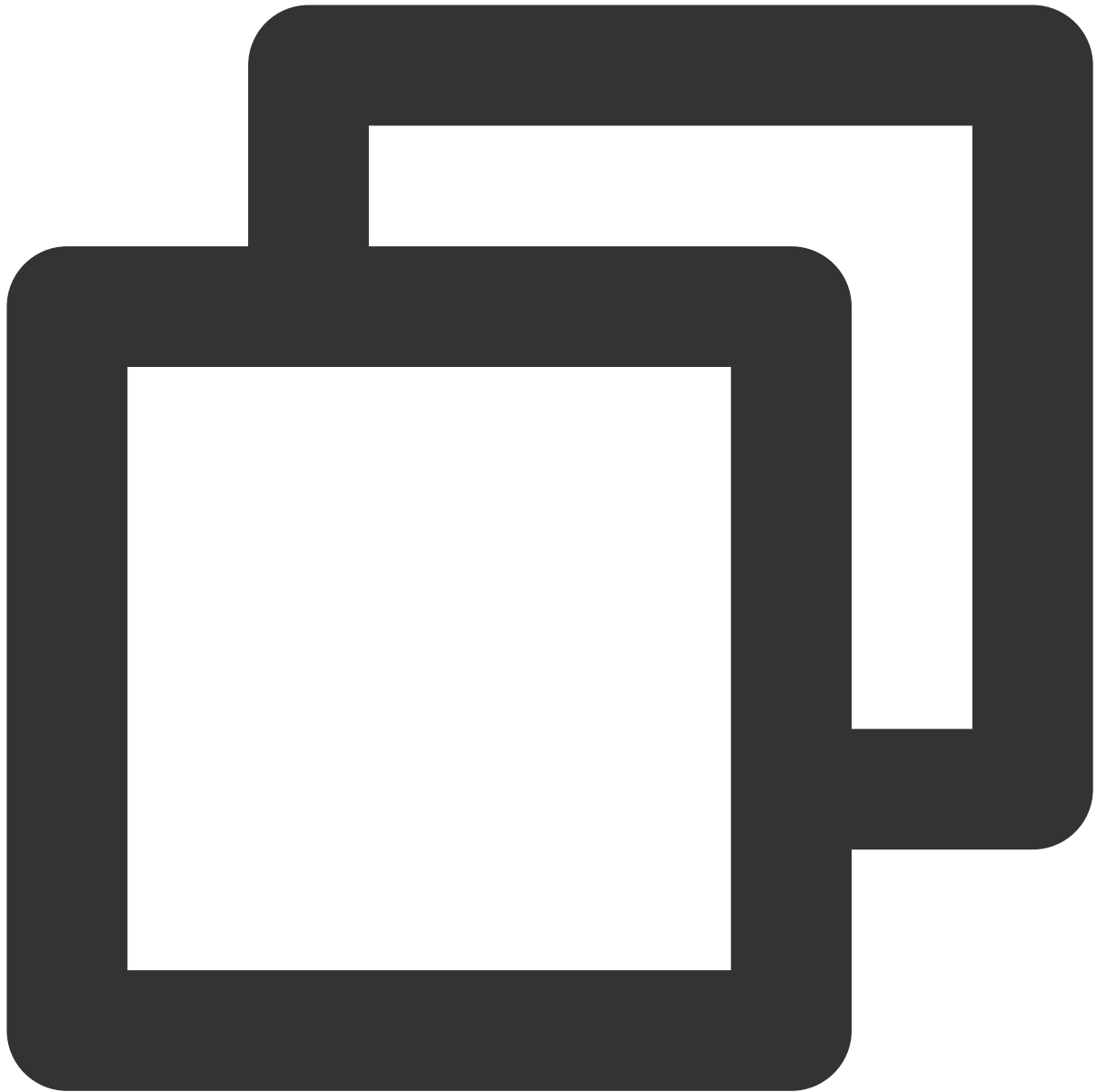
设置 `do_db`，`ignore_db` 后，会导致一些跨库操作 binlog 记录不全，主从复制出现异常，因此不建议设置。如发生类似报错，请参考如下指导进行修复：

1. 登录源数据库。
2. 修改源数据库的配置文件 `my.cnf`，删除 `do_db`，`ignore_db` 相关设置。

说明

`my.cnf` 配置文件的默认路径为 `/etc/my.cnf`，现场以实际情况为准。

3. 参考如下命令重启源数据库。

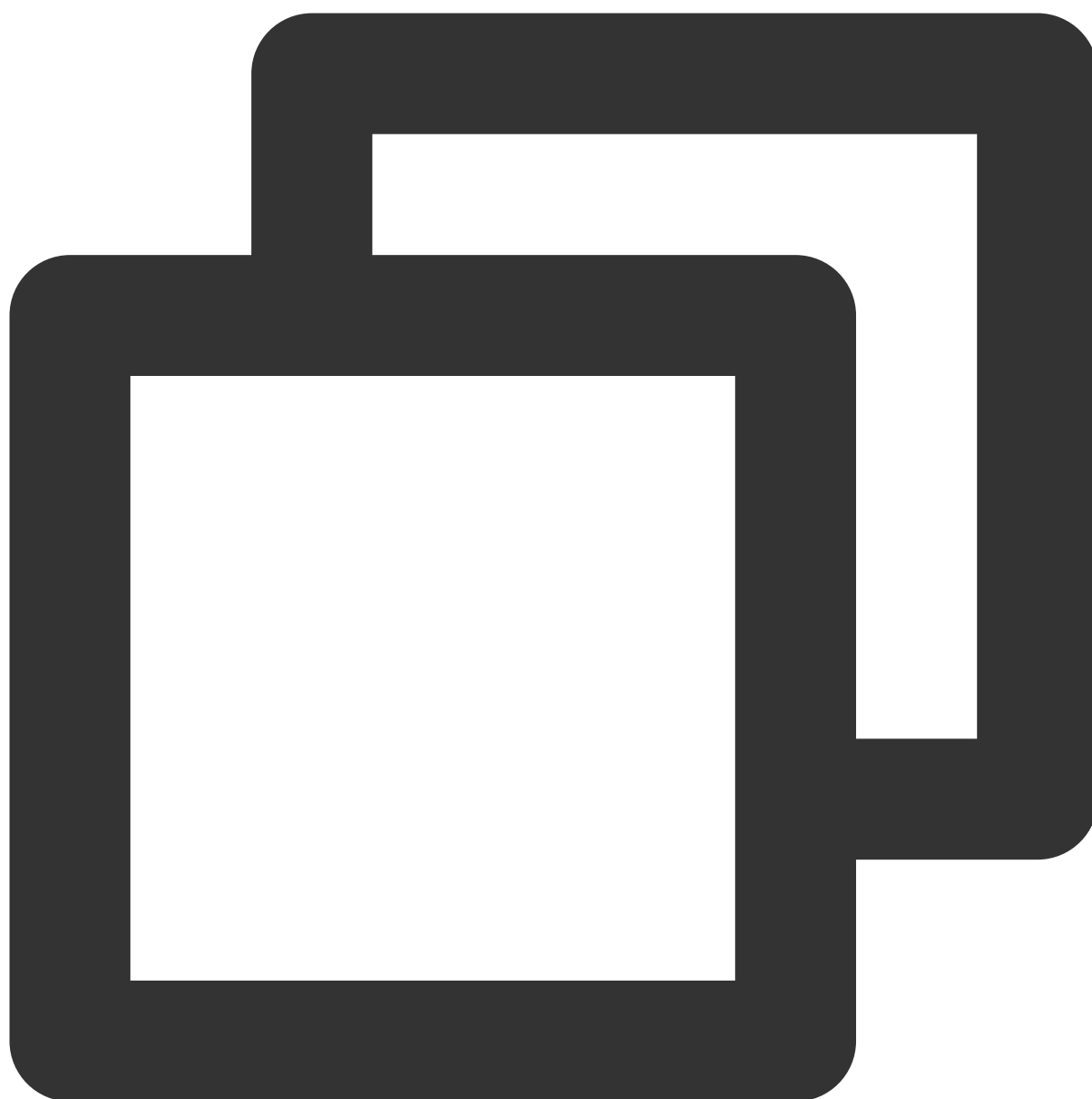


```
[ $Mysql_Dir ]/bin/mysqladmin -u root -p shutdown  
[ $Mysql_Dir ]/bin/safe_mysqld &
```

说明

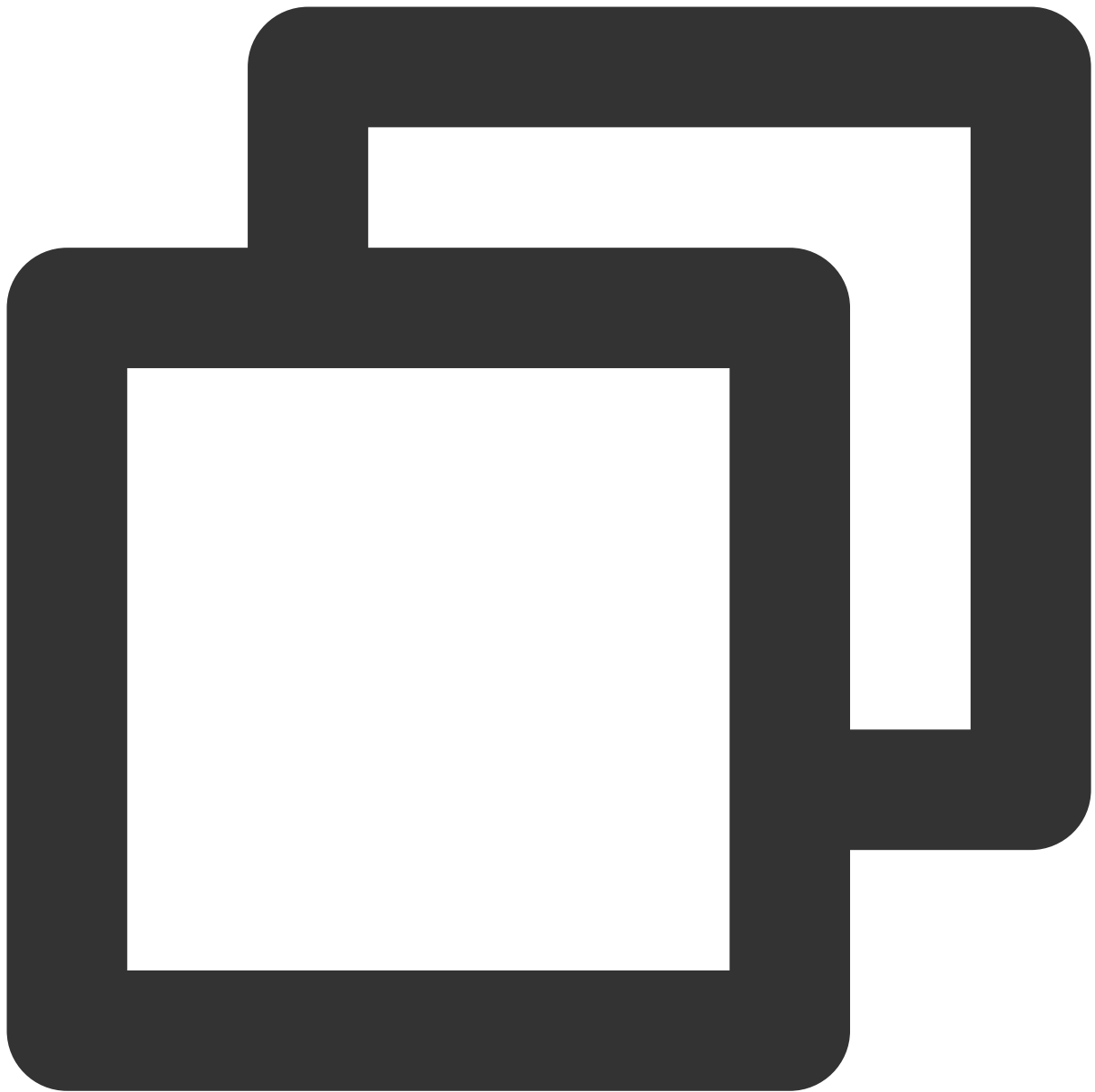
[\$Mysql_Dir] 指源数据库的安装路径，请替换为实际的源数据库安装目录。

4. 确认参数修改是否生效。



```
show master status;
```

系统显示结果类似如下：



```
mysql> show master status;
```

File	Position	Binlog_Do_DB	Binlog_Ignore_DB	Executed_Gtid_Set
binlog.000011	154			

5. 重新执行校验任务。

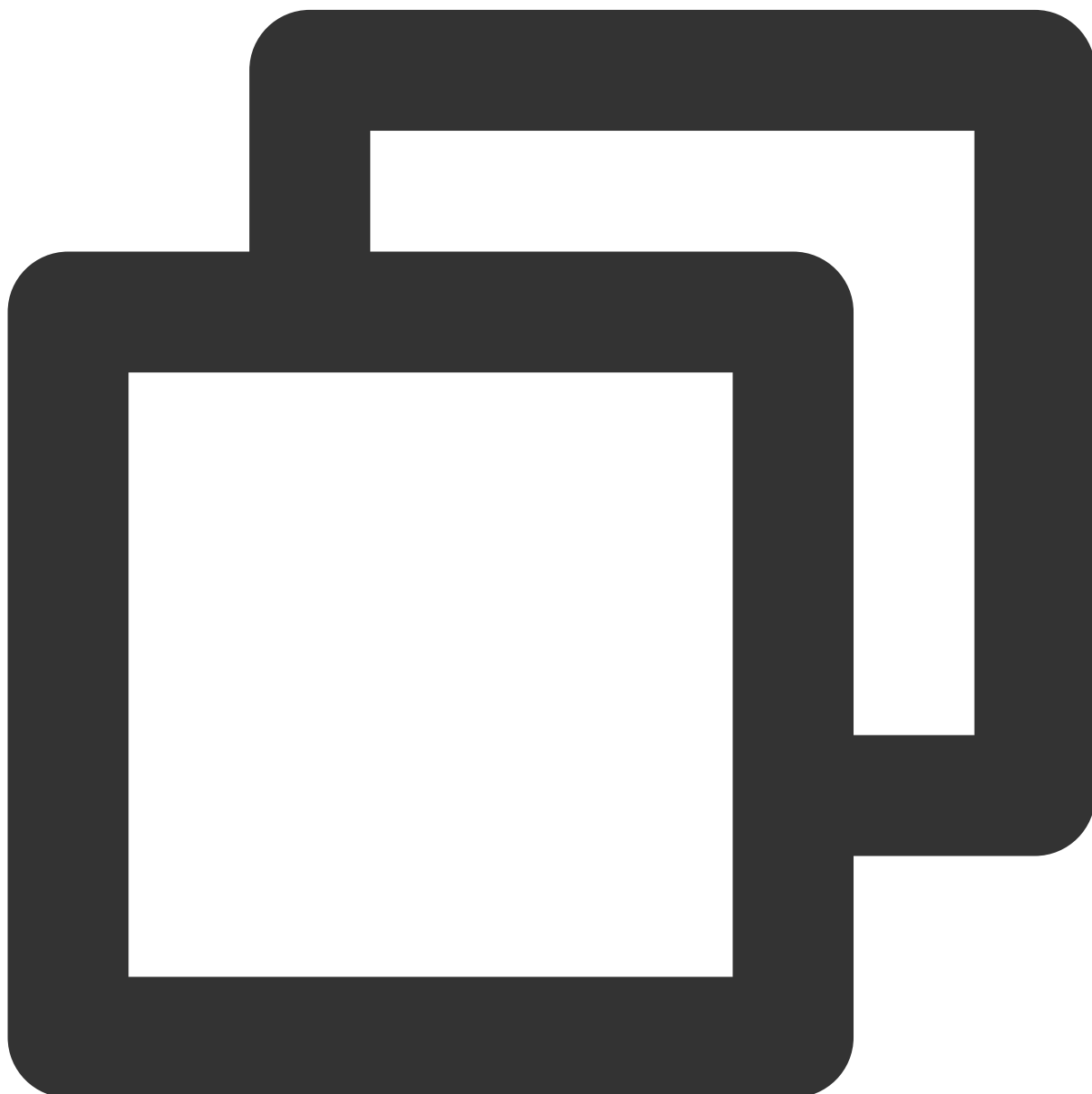
修改 log_slave_updates 参数

在主从复用结构中，从库开启 `log-bin` 参数，直接在从库操作数据时，可以记录在 `binlog` 中，但是从库从主库上复制数据时，不能记录在 `binlog` 中，所以从库作为其他从库的主库时，需要打开 `log_slave_updates` 参数。

1. 登录源数据库。
2. 在源数据库的配置文件 `my.cnf` 中增加如下内容。

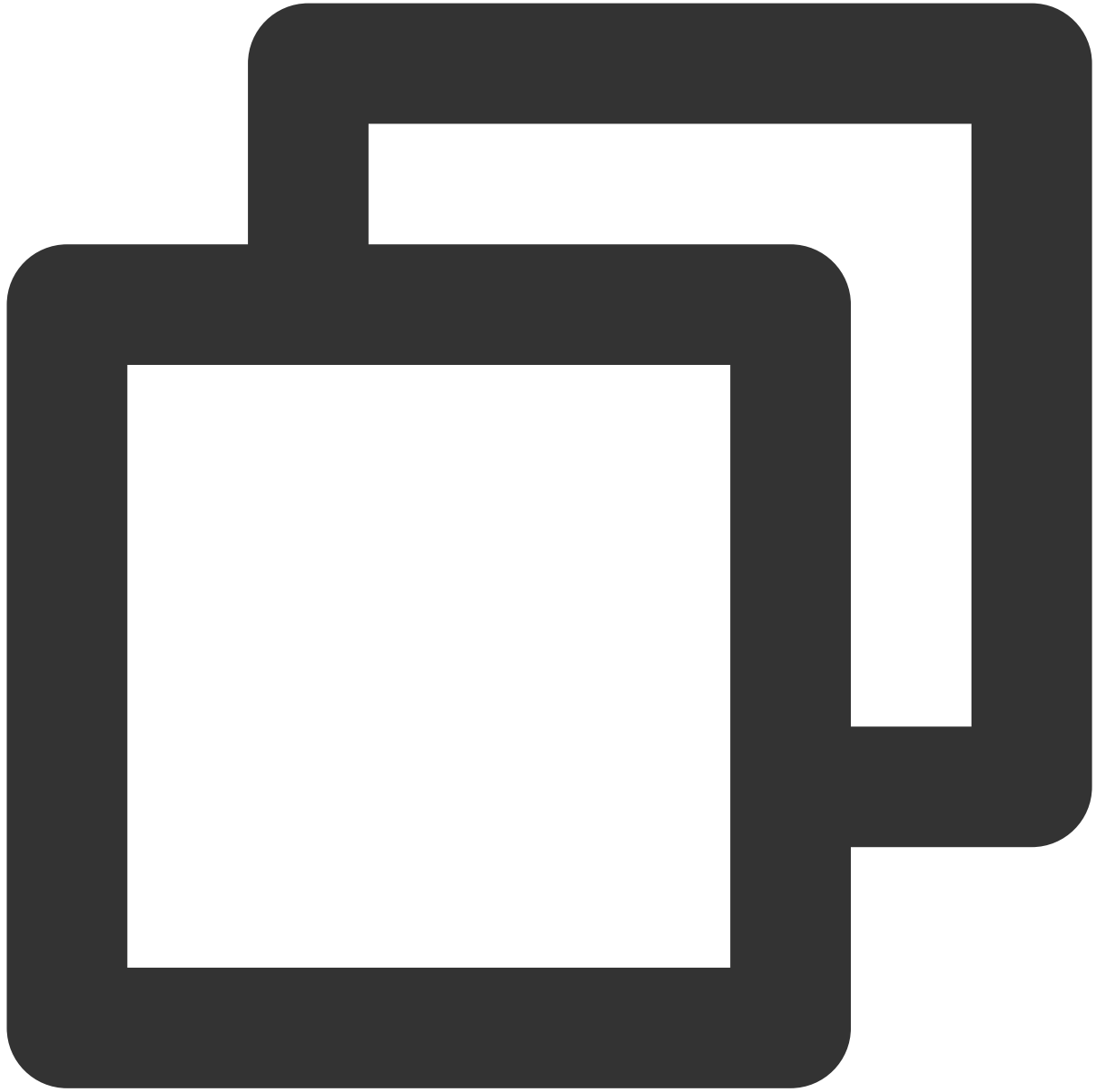
说明

`my.cnf` 配置文件的默认路径为 `/etc/my.cnf`，现场以实际情况为准。



```
log_slave_updates = ON
```

3. 参考如下命令重启源数据库。

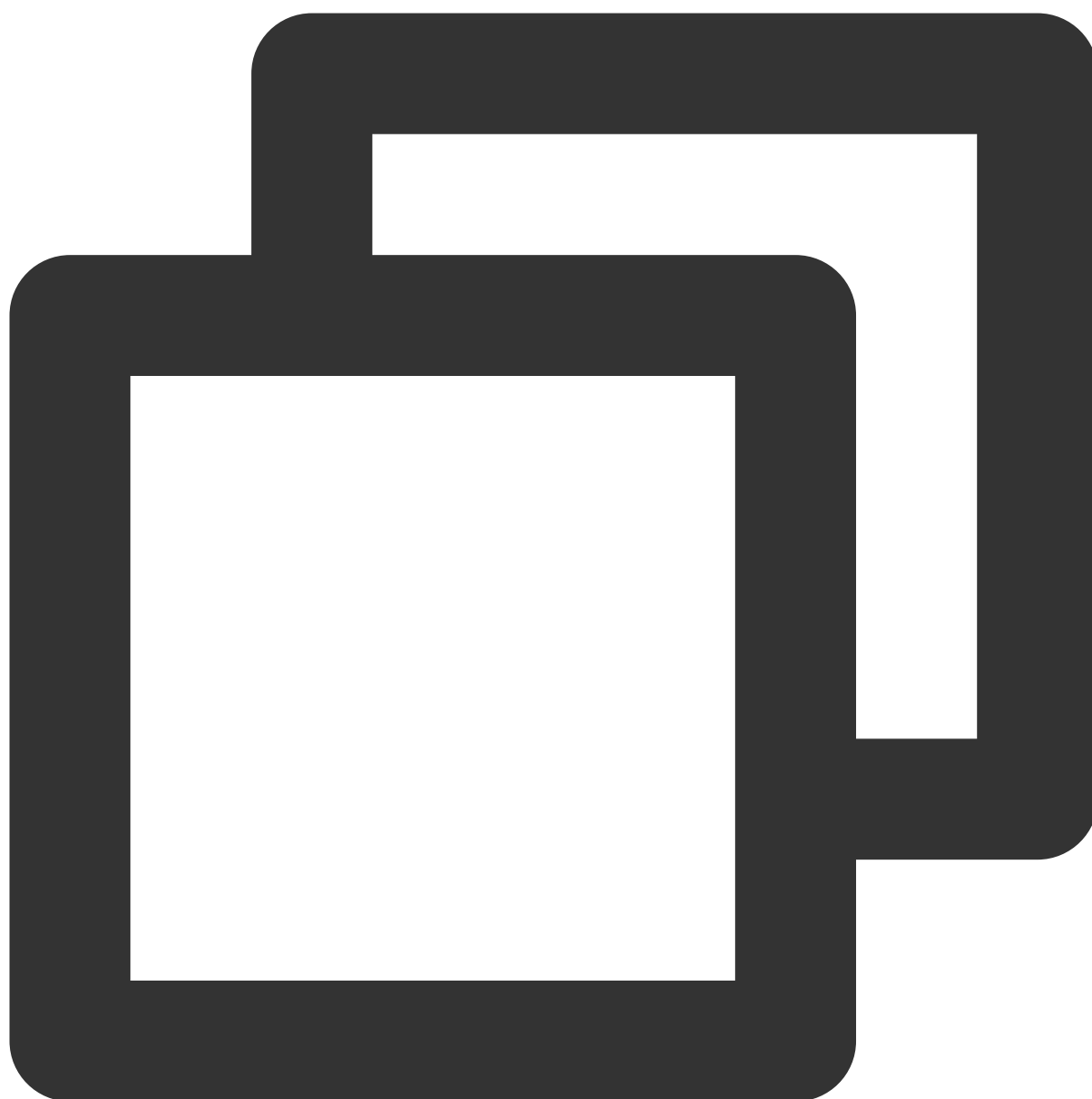


```
[${Mysql_Dir}]/bin/mysqladmin -u root -p shutdown  
[${Mysql_Dir}]/bin/safe_mysqld &
```

说明

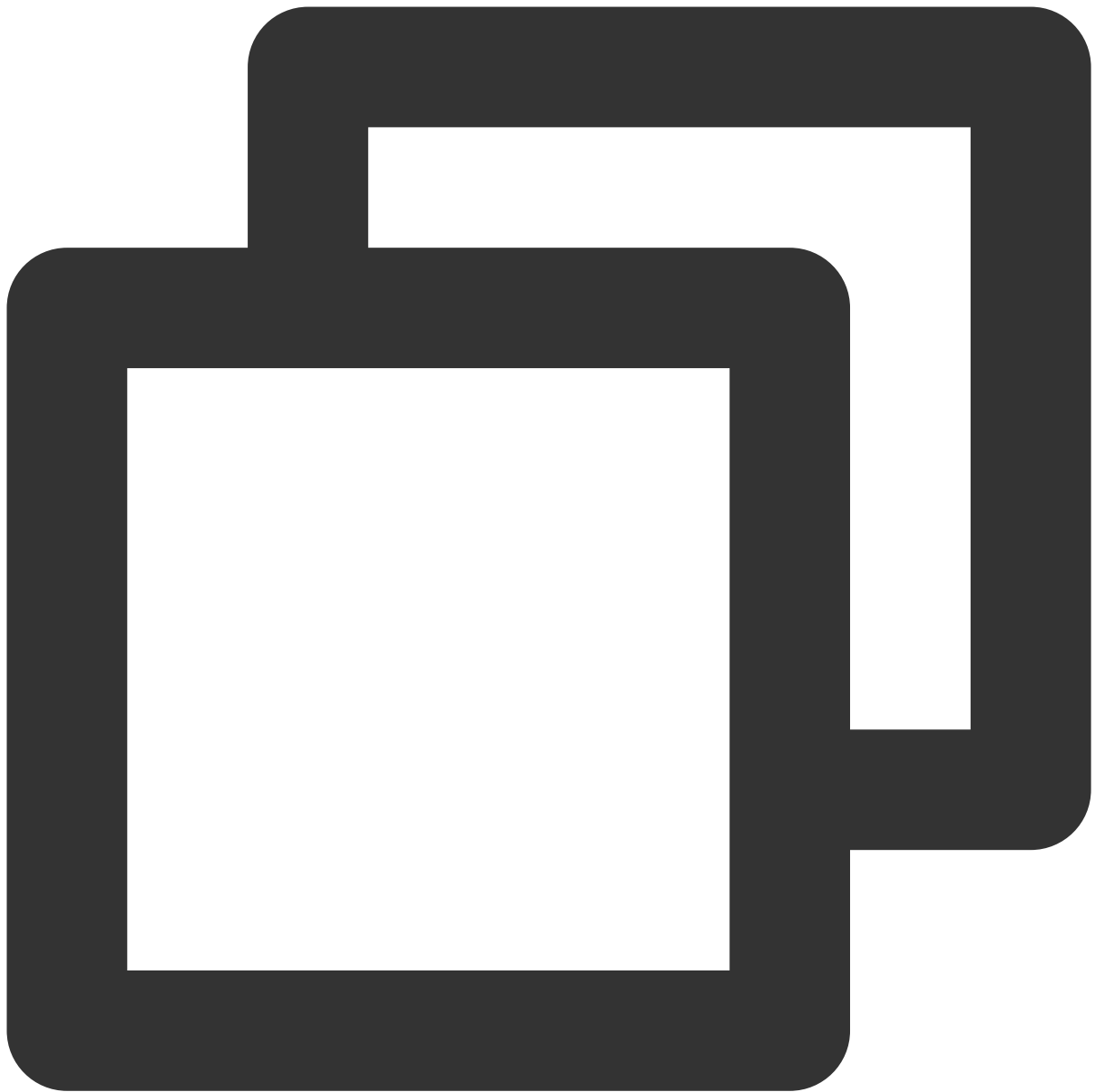
[\${Mysql_Dir}] 指源数据库的安装路径，请替换为实际的源数据库安装目录。

4. 查看配置是否生效。



```
show variables like '%log_slave_updates%';
```

系统显示结果类似如下：



```
mysql> show variables like '%log_slave_updates%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| log_slave_updates | ON    |
+-----+-----+
1 row in set (0.00 sec)
```

5. 重新执行校验任务。

外键依赖检查

最近更新时间：2024-07-08 17:51:43

检查详情

MySQL/MariaDB/Percona/TDSQL-C MySQL 之间的数据迁移：外键依赖支持 CASCADE、SET NULL、RESTRICT、NO ACTION、SET DEFAULT，需要在校验项中开启外键依赖开关。

MySQL/MariaDB/Percona/TDSQL-C MySQL 之间的数据同步：外键依赖只能设置为 NO ACTION、RESTRICT，设置为其他校验将报错，但用户可选择修复报错或者忽略报错继续任务。

TDSQL MySQL 的迁移和同步：外键依赖只能设置为 NO ACTION、RESTRICT，设置为其他校验将报错，但用户可选择修复报错或者忽略报错继续任务。

TDSQL TDSore 数据迁移：不支持外键依赖数据，如果源库中有外键数据校验会报错。
部分库表迁移时，有外键依赖的表必须齐全。

外键依赖参数说明

MySQL 系类型的数据库在设置外键时，删除和更新有四个值可以选择。

- CASCADE**：父表进行删除或者更新记录时，子表会同步删除或更新关联记录。
- SET NULL**：父表进行删除或者更新记录时，子表会将关联记录的外键字段所在列设为 `null`（子表外键不能设为 `not null`）。
- RESTRICT**：父表执行删除或更新记录时，如果子表中有关联该父表的记录，则拒绝该父表删除请求。
- NO ACTION**：同 RESTRICT，也是首先检查外键。
- SET DEFAULT**：父表进行删除或者更新记录时，子表将外键列设置成一个默认的值，但 InnoDB 引擎不能识别。

忽略报错或开启外键依赖关系迁移

对于 MySQL/MariaDB/Percona/TDSQL-C MySQL 之间的数据迁移，外键依赖支持 CASCADE、SET NULL、RESTRICT、NO ACTION、SET DEFAULT。

DTS 默认支持 RESTRICT，NO ACTION，如果源数据库迁移的数据中存在其他类型的外键依赖配置，校验系统会报错，需要用户按照如下指导在校验项中开启外键依赖。

- 在数据迁移操作中的**校验任务**页面，源库存在外键依赖规则 CASCADE、SET NULL、SET DEFAULT，校验项报错，查看校验项详情。

创建校验任务		
查询校验结果		
连接DB检查		通过
周边检查		通过
版本检查		通过
源实例权限检查		通过
部分实例参数检查		通过
目标实例权限检查		通过
目标实例内容冲突检查		通过
目标实例空间检查		通过
binlog参数检查		通过
外键依赖检查		失败 查看详情
外键部分库表依赖检查		通过
视图检查		通过
警告项检查		警告 查看详情
高级对象检查		通过

2. 根据情况选择忽略报错，或者迁移外键依赖关系。确认无误后勾线风险知会，单击**确认**。

选项	说明
屏蔽外键依赖检查报错	勾选后忽略已有的校验报错，继续迁移任务。 源库中存在外键规则 CASCADE 、 SET NULL 、 SET DEFAULT ，仍可以进行迁移，但屏蔽报错后可能出现因为父表进行删除或者更新操作，导致目标库和源库的数据结果不一致。
迁移外键依赖关系	勾选后支持迁移 CASCADE 、 SET NULL 、 SET DEFAULT 类型的外键依赖关系，但 DTS 仅在任务发起时检查外键依赖关系，保证存量数据迁移到目标库的一致性。 1. 任务迁移过程中不要修改源库中的外键依赖规则，否则将导致目标库数据与源库不一致。 如在迁移过程中，将源库外键规则从 CASCADE 修改为 NO ACTION ，目标库中父表的更新及删除操作仍会影响子表。 2. 在外键规则为 CASCADE 或 SET NULL 时，勾选 迁移外键依赖关系 后，在全量迁移阶段会对源库短暂加表锁，影响时间约 5-10 秒。 3. 为了保证数据的一致性，在增量同步阶段，对外键相关表的数据传输将从行级同步降级为表级同步，性能会有所下降。

校验详情

校验结果

失败

错误详情

只支持外键规则：RESTRICT，NO ACTION。subs_test.rooms 依赖 subs_test.buildings 的update_rule:NO ACTION, delete_rule:CASCADE [帮助文档](#)

风险警告

打开外键依赖或屏蔽外键依赖迁移可能带来的风险

建议

方式一：屏蔽外键依赖检查报错

风险提示：屏蔽失败可能会因为父表进行删除或者更新时导致最终数据不一致

方式二：打开外键依赖关系迁移

风险提示：

1. DTS 仅在任务发起时检查外键依赖关系，若在迁移过程中修改外键依赖关系，将导致目标库数据与源库不一致

比如在迁移过程中，将源库外键规则从 CASCADE 修改为 NO ACTION，父表的 UPDATE 及 DELETE 操作仍会影响子表

2. 在外键规则为 CASCADE 或 SET NULL 时，开启外键依赖后，在全量迁移阶段会对源库短暂加表锁，影响时间约 5-10 秒

3. 为了保证数据的一致性，在增量同步阶段，对外键相关表的数据传输将从行级同步降级为表级同步，性能会有所下降

外键依赖

屏蔽外键依赖检查报错

迁移外键依赖关系

☒ 我已知晓并同意：打开外键依赖关系迁移可能会带来的风险

DTS 默认支持 RESTRICT、NO ACTION 外键规则。选择迁移外键依赖关系后可

关闭

确认

3. 重新执行校验任务。

修复报错

MySQL/MariaDB/Percona/TDSQL-C MySQL 之间的数据同步：外键依赖只能设置为 NO ACTION、RESTRICT，设置为其他校验将报错，但用户可选择修复报错或者忽略报错继续任务，如需修复报错请如下指导操作。

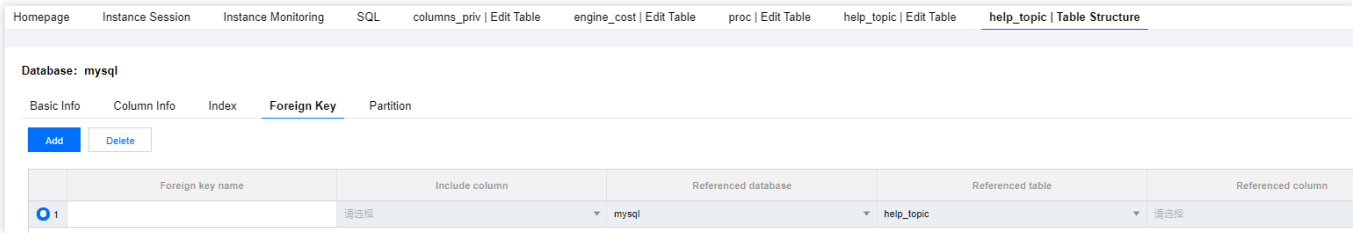
TDSQL MySQL 的迁移和同步：外键依赖只能设置为 NO ACTION、RESTRICT，设置为其他校验将报错，但用户可选择修复报错或者忽略报错继续任务，如需修复报错请如下指导操作。

TDSQL MySQL（TDStore）：不支持外键依赖数据，请删除对应的外键参数内容才能继续任务。

修改外键规则

Windows 操作指导

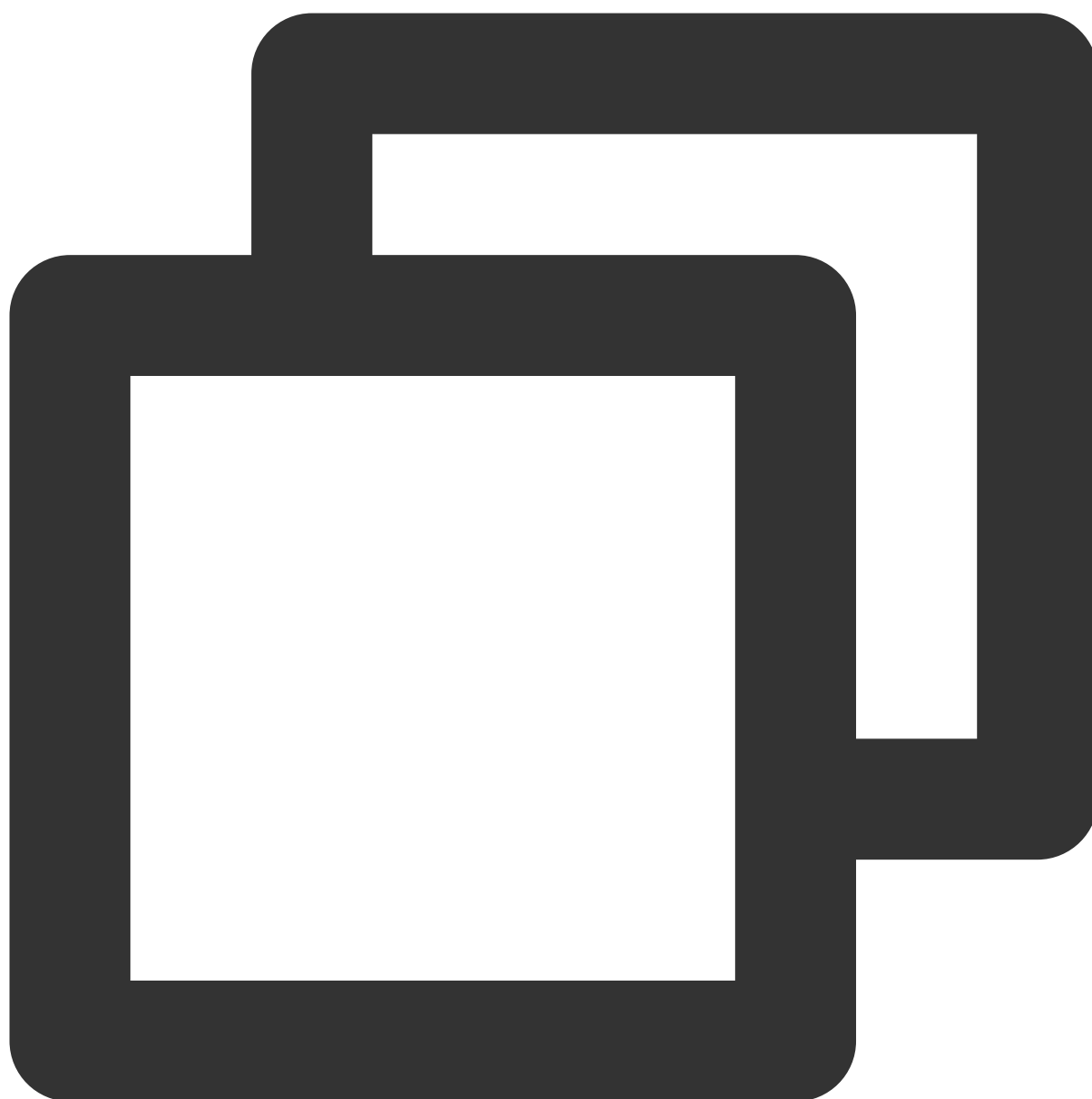
1. 登录源数据库 DMC 平台。
2. 在左侧目标树上选中要修改的表，在打开的表编辑界面上，单击**外键**页签，修改外键参数，如下图所示。



3. 修改完成后，单击保存。
4. 重新执行校验任务。

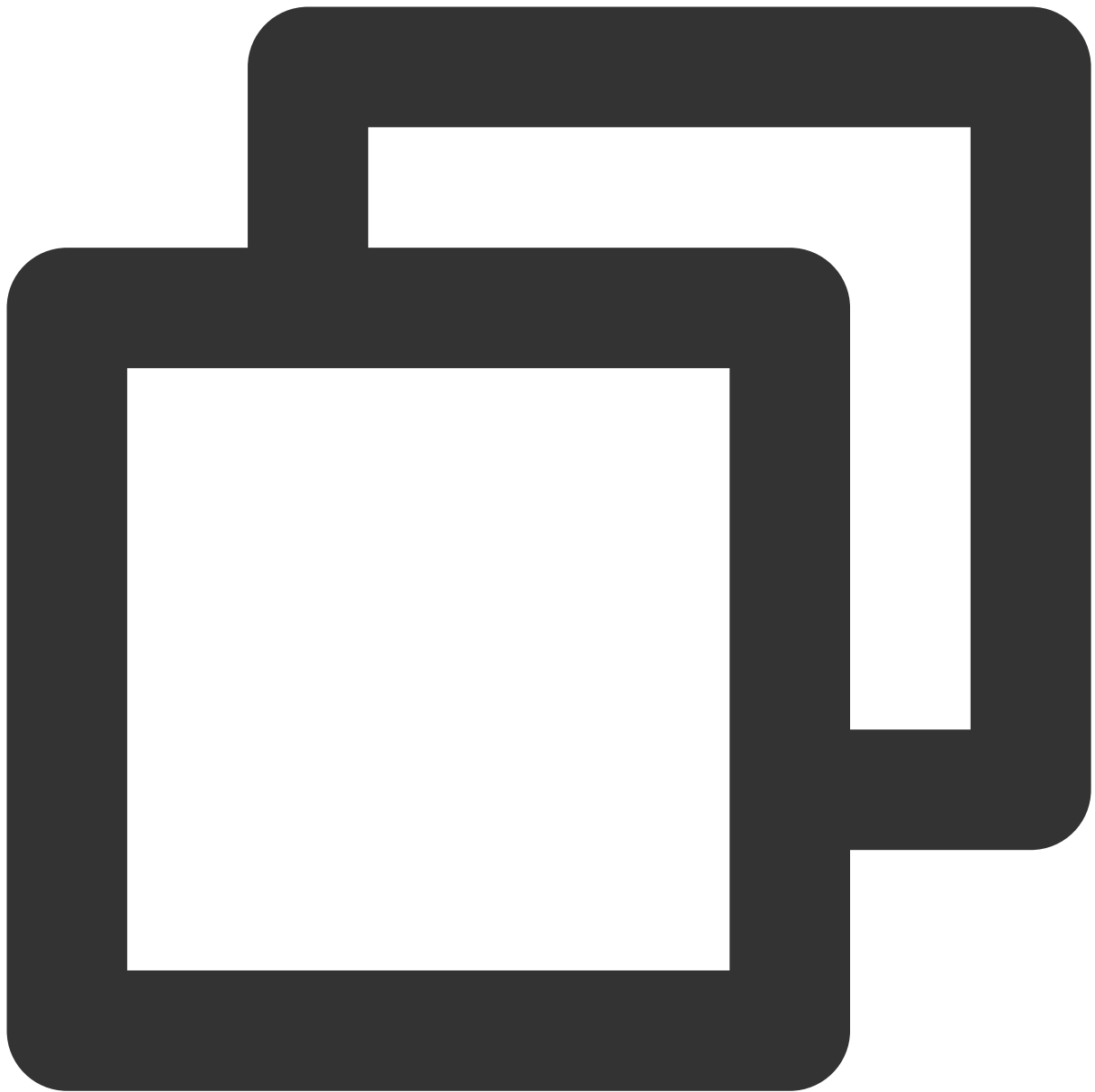
Linux 操作指导

1. [登录源数据库](#)。
2. 删除原来的外键设置。



```
alter table `表名称1` drop foreign key `外键名称1`;
```

3. 重新添加外键设置。



```
alter table `表名称1` add constraint `外键名称2` foreign key `表名称1`(`列名1`) referen  
on update no action on delete no action;
```

4. 重新执行校验任务。

完善迁移对象

修改迁移任务配置，在迁移对象中勾选具有关联关系的对象。

1. 在 [DTS 控制台](#)，选择对应的迁移任务，在**操作**列选择**更多 > 修改**。
2. 在**迁移对象**中勾选具有关联关系的对象。

3. 重新执行校验任务。

视图检查

最近更新时间：2024-07-08 17:51:43

MySQL/TDSQL-C 检查详情

检查要求：在导出视图结构时，DTS 会检查源库中 `DEFINER` 对应的 `user1`（`[DEFINER = user1]`）和迁移目标的 `user2` 是否一致。

如果一致则迁移后不做改动。

如果不一致，则迁移后修改 `user1` 在目标库中的 `SQL SECURITY` 属性，由 `DEFINER` 转换为 `INVOKER`（`[INVOKER = user1]`），同时设置目标库中 `DEFINER` 为迁移目标的 `user2`（`[DEFINER = 迁移目标user2]`）。

检查说明：`SQL SECURITY` 参数用来表示用户访问指定视图时，系统按照谁的权限来执行。

`DEFINER`：表示只有定义者才能执行。

`INVOKER`：表示拥有权限的调用者可以执行。

默认情况下，系统指定为 `DEFINER`。

TDSQL MySQL 检查详情

只允许和迁移目标 `user@host` 相同的 `definer`，即在导出视图结构时，DTS 检查源库中 `definer` 对应的 `user1`（`[DEFINER = user1]`）和迁移目标的 `user2`（`user@host`）是否一致，一致则支持迁移视图，否则不支持。

对于和迁移目标 `user@host` 不同的 `definer`，如果需要迁移，则要修改源数据库视图的 `definer`（修改为迁移目标用户），或者在迁移/同步任务中不勾选，等任务结束后手动同步视图。

高级对象检查

最近更新时间：2024-07-08 17:51:43

MySQL/MariaDB/Percona 检查详情

选择迁移/同步高级对象时，DTS 会对如下内容进行校验。报错项必须要处理才能继续任务，警告项用户评估业务风险后可忽略，继续任务。

报错项：目标实例参数 `log_bin_trust_function_creators` 必须为 ON。

警告项：

迁移/同步高级对象与库表重命名功能冲突，选择高级对象后需要取消库表重命名。

选择高级对象的函数，存储过程时，DTS 会检查源库中 `DEFINER` 对应的 `user1`（`[DEFINER = user1]`）和执行任务账号 `user2` 是否一致。

如果一致则迁移/同步后不做改动。

如果不一致，则迁移/同步后修改 `user1` 在目标库中的 `SQL SECURITY` 属性，由 `DEFINER` 转换为 `INVOKER`（`[INVOKER = user1]`），同时设置目标库中 `DEFINER` 为执行任务账号的 `user2`（`[DEFINER = 执行任务账号user2]`）。

高级对象的迁移/同步时间：

存储过程和函数，在“源库导出”阶段进行迁移/同步。

触发器和事件，没有增量任务，在任务结束时进行迁移/同步；有增量任务，在用户单击**完成**操作后开始迁移/同步，所以单击**完成**后任务的过渡时间会长一些。

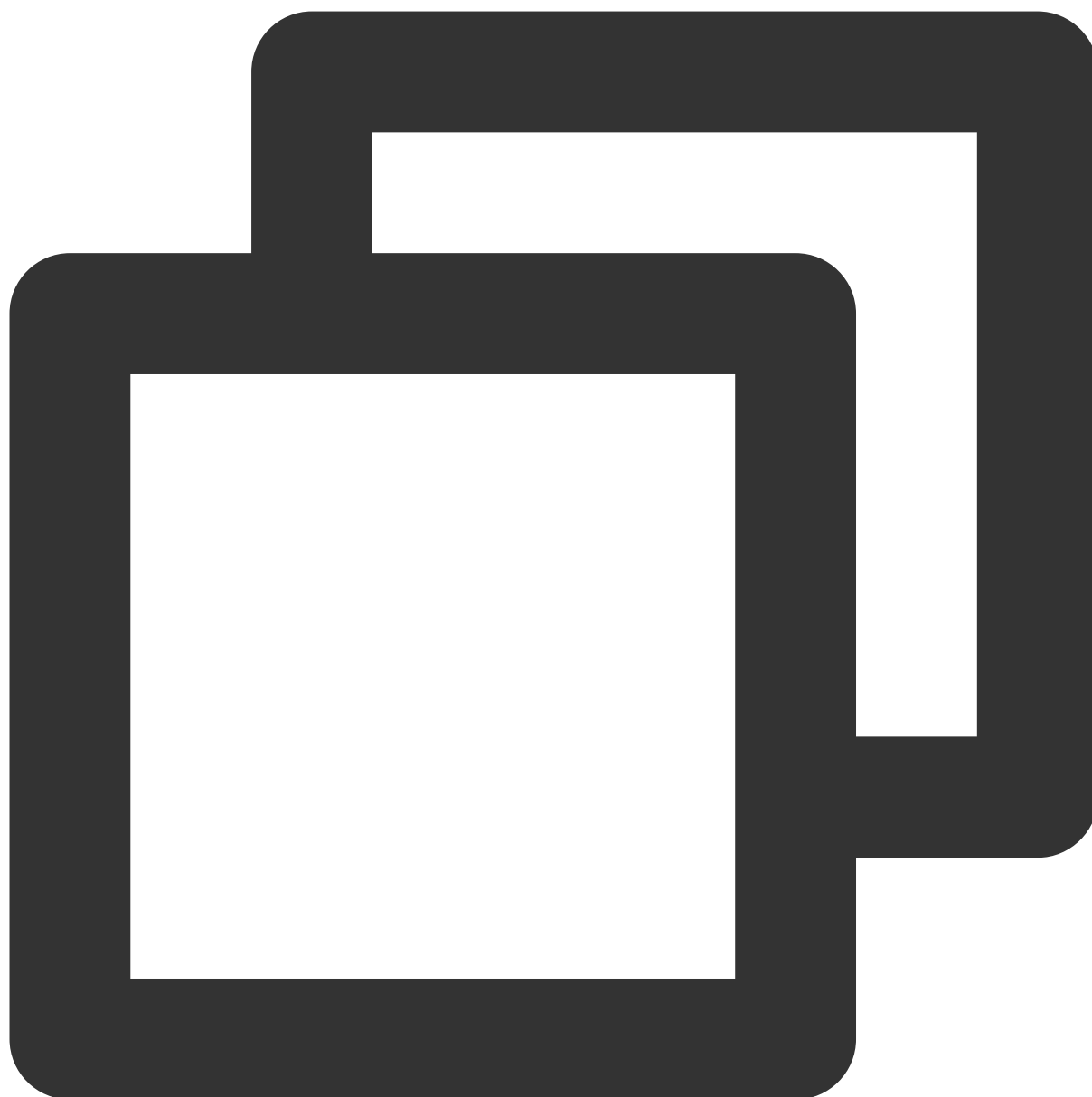
修复方法

修改 `log_bin_trust_function_creators` 参数。

`log_bin_trust_function_creators` 用于控制是否信任用户将存储函数写入 binlog 日志中。设置为 OFF，仅 SUPER 权限的用户可将创建的存储函数操作写入 binlog 日志，设置为 ON，非 SUPER 权限的用户也可将创建的存储函数操作写入 binlog 日志中。

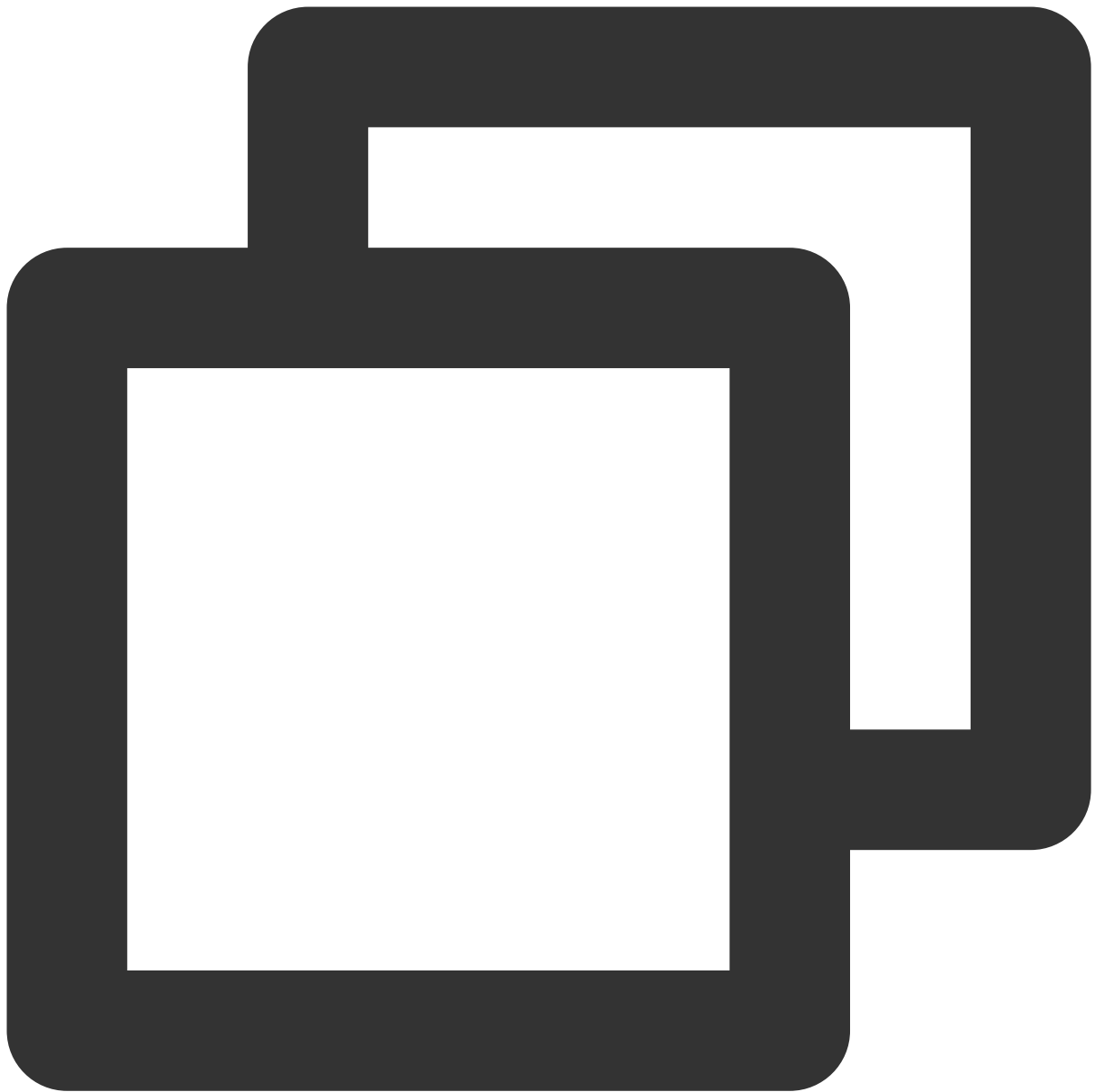
发生报错时，请参考如下步骤进行修改。

1. 登录源数据库。
2. 参考如下内容修改 `log_bin_trust_function_creators` 参数。



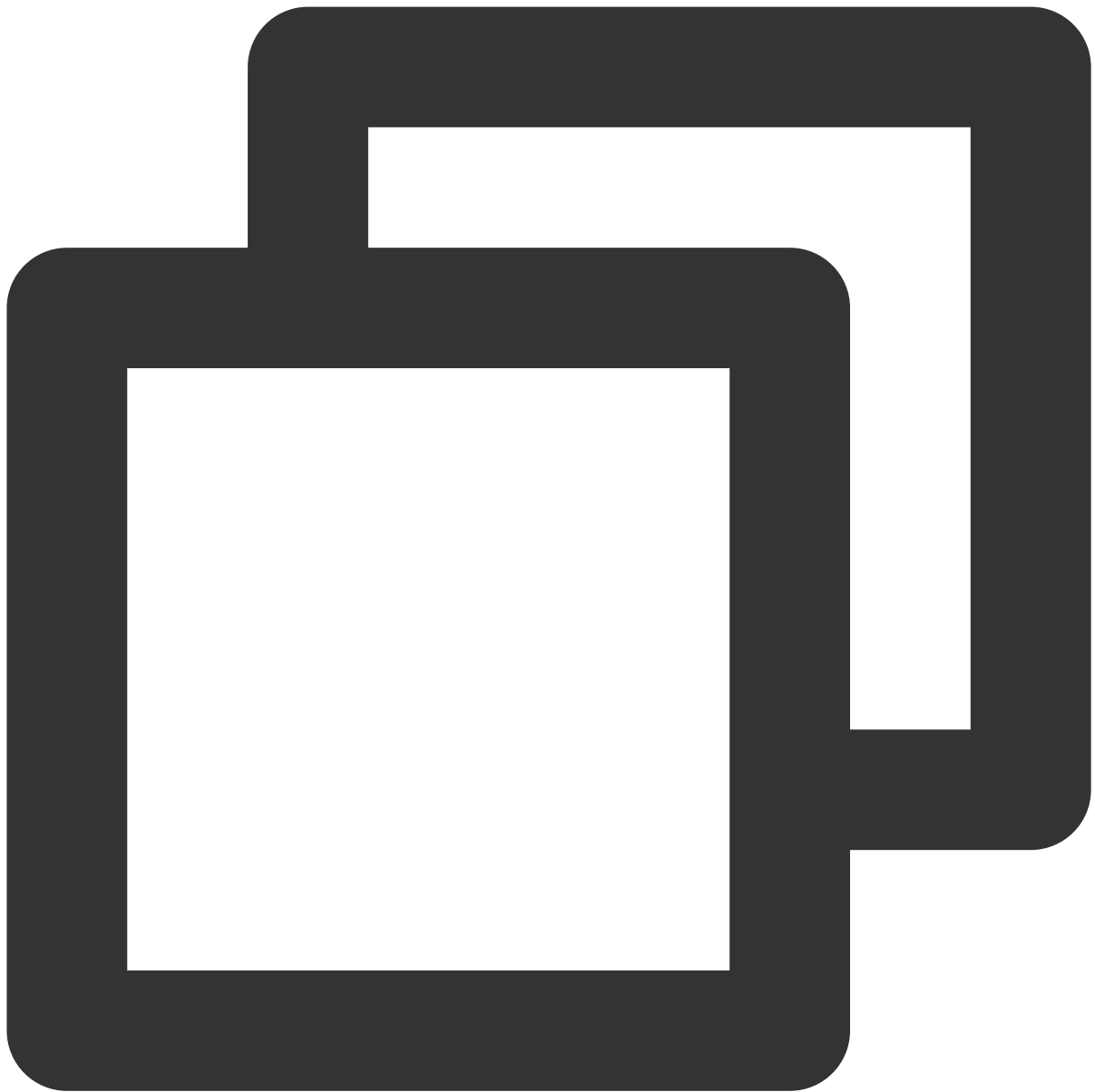
```
set global log_bin_trust_function_creators = ON;
```

3. 通过如下命令查看参数修改是否生效。



```
show variables like '%log_bin_trust_function_creators%';
```

系统显示结果类似如下：



```
mysql> show variables like '%log_bin_trust_function_creators%';
```

Variable_name	Value
log_bin_trust_function_creators	ON

4. 重新执行校验任务。

警告项检查

最近更新时间：2024-07-08 17:51:43

MySQL/TDSQL-C/MariaDB/Percona/TDSQL MySQL 检查详情

如下参数需要按照要求配置，否则校验时系统会发出警告，警告不影响迁移任务的进行，但是会对业务造成一定的影响，请用户评估后自行决定是否修改。

建议目标库 `max_allowed_packet` 的取值大于源库。

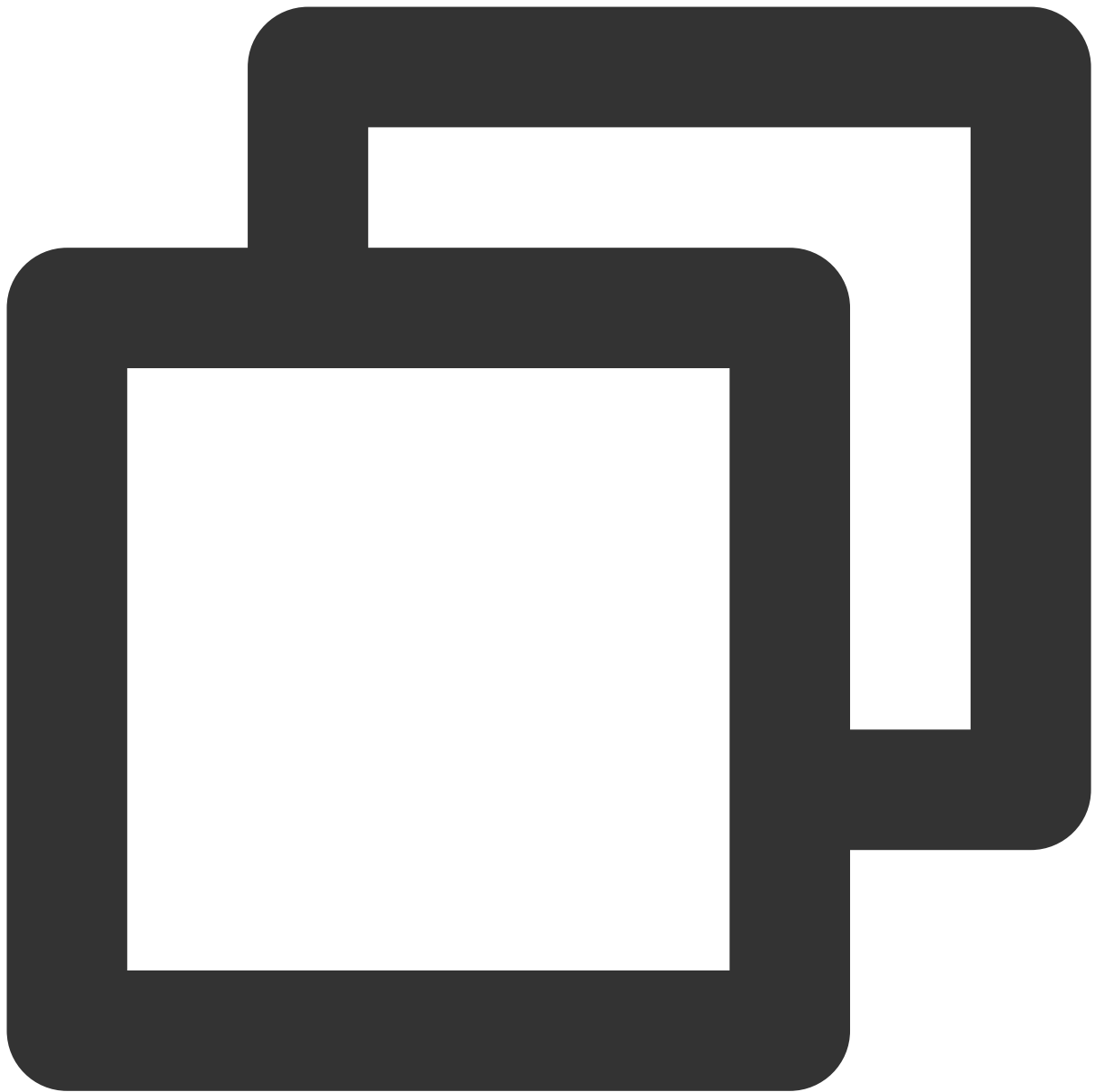
业务影响：目标库的 `max_allowed_packet` 参数设置小于源库，会导致目标库数据无法写入，从而造成全量迁移失败。

处理建议：修改目标库的 `max_allowed_packet` 参数，大于源库取值。

建议目标库的 `max_allowed_packet` 设置大于1GB。

业务影响：`max_allowed_packet` 设置太大，会使用更多内存导致丢包，无法捕捉异常大事物包SQL；设置太小，可能会导致程序报错，备份失败，也会导致频繁的收发网络报，影响系统性能。

处理建议：参考如下命令修改 `max_allowed_packet` 参数。

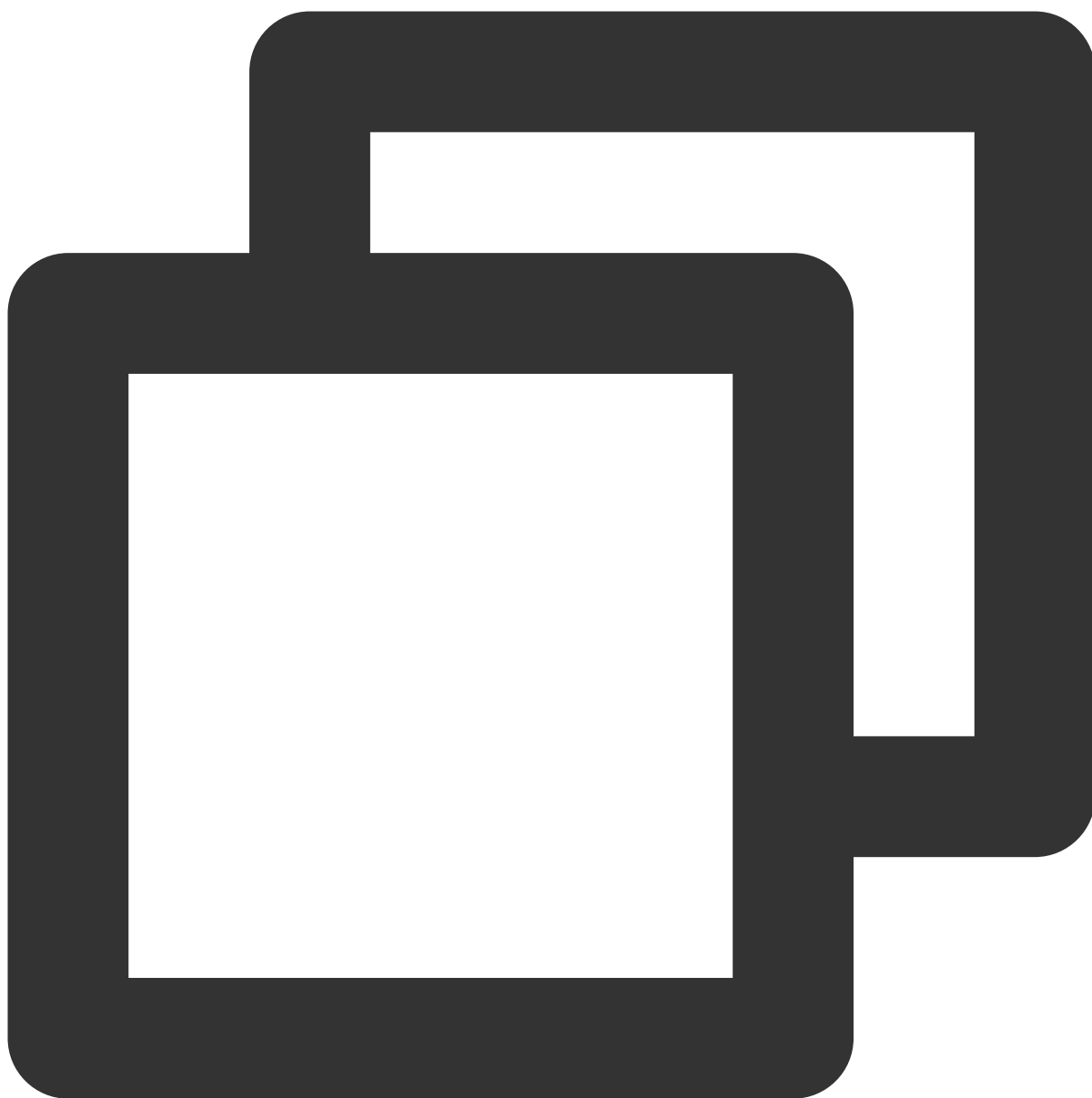


```
set global max_allowed_packet = 1GB
```

建议源库和目标库的字符集保持一致。

业务影响：源库和目标库的字符集不一致可能会导致乱码。

处理建议：参考如下命令将源库和目标库字符集修改为一致。



```
set character_set_server = 'utf8';
```

建议使用2CPU，4G Mem以上规格的实例。

如果仅执行全量数据迁移，请勿在迁移过程中向源实例中写入新的数据，否则会导致源和目标数据不一致。针对有数据写入的场景，为实时保持数据一致性，建议选择全量+增量数据迁移。

有锁导出时：源实例需要使用 Flush Table With Read Lock 短暂加锁，其中的 MyISAM 表会锁定到全量数据导出完成。当前等待加锁超时时间设置为60秒，该时间内无法获取锁将导致任务失败。

无锁导出时：仅对没有主键的表会加读锁，其他不加锁。

对于既没有主键、也没有非空唯一键的表，有数据重复的风险，建议用户对迁移的表设置主键，或者非空唯一键。

对于 TDSQL MySQL 的数据迁移，当源数据库实例为分布式数据库时，需要提前在目标库建立分表，否则这些表被迁移后都将是单表。

目标库为 MySQL/MariaDB/Percona/TDSQL-C MySQL/TDSQL TDSore 时，需要检查源库和目标库

`explicit_defaults_for_timestamp` 的设置。源库为 `OFF`，或者源和目标都为 `ON` 时，任务警告，提醒用户任务运行期间不要随便修改该参数。

在全量导出库表结构时检查表的 `COLUMN_DEFAULT`，`IS_NULLABLE` 属性，如果源库表的 `COLUMN_DEFAULT` 为 `NULL`，`IS_NULLABLE` 为 `NOT NULL`，则该表结构不进行迁移/同步，避免数据同步到目标库后，可能被 MySQL 系统自动添加为 `DEFAULT TIMESTAMP` 的表。

TDSQL Proxy 检查

最近更新时间：2024-07-08 17:51:43

检查详情

目标库 TDSQL MySQL 使用 Proxy 方式连接的场景中，DTS 在目标库执行 SQL 的时候，是将多条语句放到一个 SQL 里去执行，需要将 gateway.mode.multi_query.open 参数设置为1，以支持多个 SQL 合并执行。参数如未打开，预校验任务会报错。

修复方法

用户无法自行修改，请 [提交工单](#) 进行修改。

TDSQL 内核检查

最近更新时间：2024-07-08 17:51:43

检查详情

源端为 TDSQL MySQL 时，检查源端的内核版本，不支持 mariadb 内核。如果是 mariadb 内核，预校验任务会报错。

修复方法

无法同步 TDSQL MySQL（mariadb 内核），请选择其他版本进行同步。

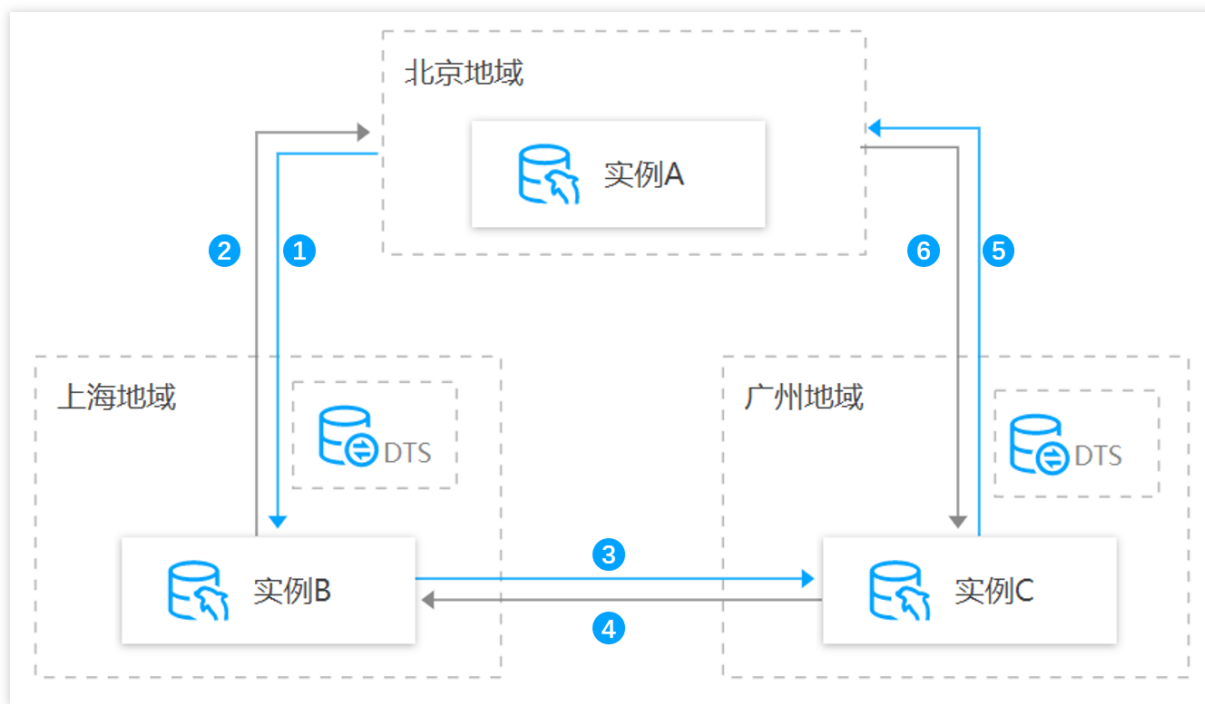
同库表对象 DDL 环形链路同步检测

最近更新时间：2024-07-08 17:51:43

检查详情

在双向同步、多对一同步、多活等需要配置多个同步任务的场景中，DDL 的配置不能形成环形链路，否则可能造成 DDL 语句在系统中循环，进而引发错误。

示例：下图中蓝色线条1、3、5三个同步任务中，最多只能在两个同步任务中选择 DDL，如果选择三个就构成环形链路了。



修复方法

修改同步任务配置，在设置同步选项和同步对象 > 数据同步选项 > 同步操作类型中，修改 DDL 参数配置，避免形成环形链路。

同目标同库表对象 DDL 同步冲突检测

最近更新时间：2024-07-08 17:51:43

检查详情

在多对一同步、多活等需要配置多个同步任务的场景中，同一个库表对象不能接收多个数据中心的 DDL 同步，否则多个数据中心的 DDL 可能在目标端形成冲突，进而引发错误。

修复方法

修改同步任务配置，在**设置同步选项和同步对象 > 数据同步选项 > 同步操作类型**中，修改 DDL 参数配置，避免同一个库表对象接收多个数据中心的 DDL 同步。

同步至 PostgreSQL

PostgreSQL 同步至 PostgreSQL

最近更新时间：2024-04-19 09:56:15

本文为您介绍使用数据传输服务 DTS 从 PostgreSQL 数据库同步数据至腾讯云数据库 PostgreSQL 的过程。

源数据库支持的部署类型如下：

自建 PostgreSQL。

第三方云厂商 PostgreSQL。

腾讯云数据库 PostgreSQL。

前提条件

源数据库和目标数据库符合同步功能和版本要求，请参考 [数据同步支持的数据库](#) 进行核对。

执行 DTS 同步任务的账号需要具备源数据库的权限：Event Trigger、Replication、CREATE、DROP、ALTER、SELECT、UPDATE、INSERT、DELETE。

执行 DTS 同步任务的账号需要具备目标数据库的权限：CREATE、DROP、ALTER、SELECT、UPDATE、INSERT、DELETE。

注意事项

DTS 在执行全量数据同步时，会占用一定的源端实例资源，可能会导致源实例负载上升，增加数据库自身压力。如果您数据库配置过低，建议您在业务低峰期进行操作。

默认采用无锁方式，同步过程中对源库不加全局锁。

数据同步时，DTS 会使用执行同步任务的账号在源库中建立 postgres.public.identify、

{database}.__tencentdb__sync_*. 来记录任务状态、配置信息，目标实例会使用 {database}.__tencentdb__ schema 记录同步状态信息。postgres.public.identify、{database}.__tencentdb__sync_*.、{database}.__tencentdb__ schema 占用空间非常小，约为源库存储空间的千分之一到万分之一，所以对源库的性能无影响，也不会抢占资源。

应用限制

1. 不支持如下对象的同步：ROLE、PROCEDURE、TRIGGER、FUNCTION、RULE 等。
2. 不支持公共对象和权限变更同步。
3. 全量阶段结构初始化不支持外键导入，会自动移除外键约束后导入。

4. 数据同步不支持 FOREIGN TABLE 表类型，全量阶段导出导入不支持 TEMPORARY TABLE，增量同步阶段仅支持 UNLOGGED TABLE 数据结构同步。
5. 当数据同步任务启动前的校验阶段进行插件检测时，检查到第一个不匹配的插件发出告警后，不再继续检测剩余的插件。
6. 因同步任务依赖 max_replication_slots 参数（每个数据库配置同步任务时占用一个 replication_slots），可通过 SQL 语句 `select * from pg_replication_slots` 查看已使用情况，如果已配置同步库数量 + replication_slots > max_replication_slots，新建同步任务会失败，需要用户主动结束任务后系统才会释放 replication_slots（失败任务未结束不会释放 replication_slots）。
7. 源实例带时区的字段（timestamp with time zone）同步到目标实例后，会自动转为目标实例时区的对应值。
8. 源库发生 HA 等操作导致源实例主节点发生迁移会引起 WAL 日志变化，原同步任务会失败，需要清理目标库已同步的数据，然后重新建立同步任务。
9. 源库在增量阶段对 SEQUENCE 对象更新，同步到目标库后不支持对 SEQUENCE 的 last_value 的刷新，可能需要用户自行检查同步结果，并手动修改目标库上的取值。
10. 源库的对象同步目标端后，对象 owner 会改为目标库账号。例如之前表1的 owner 为 A，目标库使用 B 账号进行同步，则写入到目标库后表1的 owner 改为 B。

操作限制

同步过程中请勿进行如下操作，否则会导致同步任务失败。

请勿修改、删除源数据库和目标数据库中的用户信息（包括用户名、密码和权限）和端口号。

增量同步阶段同时高并发的 DDL 和 DML 操作可能会存在数据冲突，建议减少增量阶段的 DDL。

请勿在源库上执行清除 WAL 的操作。

在同步增量阶段，请勿删除系统库和 schema postgres.public.identify、__tencentdb__sync_*.、{database}.__tencentdb__ schema。

源和目标数据库类型为腾讯云数据库时，请勿修改数据库实例的网络，如发生源实例和目标实例网络变更，导致原 VIP 被删除，同步任务失败、或保持运行中状态不变（此时同步已不再工作），这种情况，同步任务无法重试，需要手动结束该任务后重新建立新的同步任务。

支持同步的 SQL 操作

操作类型	SQL 操作语句
DML	INSERT、UPDATE、DELETE
DDL	CREATE DATABASE、DROP DATABASE、ALTER DATABASE、CREATE SCHEMA、DROP SCHEMA、CREATE TABLE、ALTER TABLE、DROP TABLE、CREATE VIEW、DROP VIEW、CREATE INDEX、DROP INDEX、CREATE SEQUENCE、DROP SEQUENCE

说明：

暂不支持 CREATE TABLE 表名 AS SELECT 语句。

暂不支持 Where 条件过滤的 SQL 过滤策略。

同步支持的数据类型

Aclitem、Bit、Bool、Box、Bpchar、Bytea、Char、Cid、Cidr、Circle、Date、Float4、Float8、Inet、Int2、Int4、Int8、Interval、Json、Jsonb、Jsonpath、Line、Lseg、Macaddr、Name、Numeric、Oid、Path、Point、Polygon、Record、Text、Tid、Time、Timestamp、Timestamptz、Unknown、Uuid、Varbit、Varchar、XID、Daterange、Int4range、Int8range、Numrange、Tsrage、Tstzrange、Datemultirange、Int4multirange、Int8multirange、Nummultirange、Tsmultirange、Tstzmultirange、XML、Money、Time with time zone、Tsvector、Tsquery、pg_lsn、txid_snapshot。

环境要求

类型	环境要求
源数据库要求	源库和目标库网络能够连通。 源库参数要求如下： wal_level 参数设置为 logical。 源库需满足：待同步库数量 + replication_slots <= max_replication_slots。 源库对象需要设置不区分大小写。 源库中的表必须含有主键或者联合主键。
目标数据库要求	目标库的版本必须大于等于源库的版本。 目标库需要有足够的存储空间，如果初始类型选择“全量数据初始化”，则目标库的空间大小须是源库待同步库表空间的1.2倍以上。 目标库不能有和源库同名的表、视图等同步对象。

操作步骤

1. 登录 [数据同步购买页](#)，选择相应配置，单击**立即购买**。

参数	描述
计费模式	支持包年包月和按量计费。

源实例类型	选择 PostgreSQL，购买后不可修改。
源实例地域	选择源实例所在地域，购买后不可修改。
目标实例类型	选择 PostgreSQL，购买后不可修改。
目标实例地域	选择目的实例所在地域，购买后不可修改。
规格	请根据业务诉求选择规格，规格越高，性能越好。详情请参考 计费概述 。

2. 购买完成后，返回数据同步列表，可看到刚创建的数据同步任务，刚创建的同步任务需要进行配置后才可以使
用。
3. 在数据同步列表，单击**操作**列的**配置**，进入配置同步任务页面。
4. 在配置同步任务页面，配置源端实例、账号密码，配置目标端实例、账号和密码，测试连通性后，单击**下一步**。
因源数据库部署形态和接入类型的交叉场景较多，各场景同步操作步骤类似，如下仅提供典型场景的配置示例，其
他场景请用户参考配置。

示例一：本地自建数据库通过专线/VPN 方式同步至腾讯云数据库。

设置项	参数	描述
任务设置	任务名称	DTS 会自动生成一个任务名称，用户可以根据实际情况进行设置。
	运行模式	支持立即执行和定时执行两种模式。
源实例设置	源实例类型	购买时所选择的源实例类型，不可修改。
	源实例地域	购买时选择的源实例所在地域，不可修改。
	接入类型	请根据您的场景选择，本场景选择“专线接入”或“VPN接入”，该场景需要 配置 VPN 和 IDC 之间的互通 ，其他接入类型的准备工作请参考 准备工作概述 。 公网：源数据库可以通过公网 IP 访问。 云主机自建：源数据库部署在 腾讯云服务器 CVM 上。 专线接入：源数据库可以通过 专线接入 方式与腾讯云私有网络打通。 VPN接入：源数据库可以通过 VPN 连接 方式与腾讯云私有网络打通。 云数据库：源数据库属于腾讯云数据库实例。 云联网：源数据库可以通过 云联网 与腾讯云私有网络打通。
	私有网络专线网关/VPN 网关	专线接入时只支持私有网络专线网关，请确认网关关联网络类型。 VPN 网关，请选择通过 VPN 网关接入的 VPN 网关实例。
	私有网络	选择私有网络专线网关和 VPN 网关关联的私有网络和子网。
	主机地址	源实例 MySQL 访问 IP 地址或域名。
	端口	源实例 MySQL 访问端口。

	账号	源实例账号，账号权限需要满足要求。
	密码	源实例账号的密码。
目标实例设置	目标实例类型	购买时选择的目标实例类型，不可修改。
	目标实例地域	购买时选择的目标实例地域，不可修改。
	接入类型	根据您的场景选择，本场景选择“云数据库”。
	实例 ID	选择目标实例 ID。
	账号	目标实例账号，账号权限需要满足要求。
	密码	目标实例账号的密码。

示例二：腾讯云数据库同步至腾讯云数据库

设置项	参数	描述
任务设置	任务名称	DTS 会自动生成一个任务名称，用户可以根据实际情况进行设置。
	运行模式	支持立即执行和定时执行两种模式。
源实例设置	源实例类型	购买时所选择的源实例类型，不可修改。
	源实例地域	购买时选择的源实例所在地域，不可修改。
	接入类型	请根据您的场景选择，本场景选择“云数据库”，不同接入类型的准备工作请参考准备工作概述。 公网：源数据库可以通过公网 IP 访问。 云主机自建：源数据库部署在 腾讯云服务器 CVM 上。 专线接入：源数据库可以通过 专线接入 方式与腾讯云私有网络打通。 VPN 接入：源数据库可以通过 VPN 连接 方式与腾讯云私有网络打通。 云数据库：源数据库属于腾讯云数据库实例。 云联网：源数据库可以通过 云联网 与腾讯云私有网络打通。
	实例 ID	源实例 ID。可在 实例列表 查看源实例信息。
	账号	源实例账号，账号权限需要满足要求。
	密码	源实例账号的密码。
目标实例设置	目标实例类型	购买时选择的目标实例类型，不可修改。
	目标实例地域	购买时选择的目标实例地域，不可修改。

	接入类型	根据您的场景选择，本场景选择“云数据库”。
	实例 ID	选择目标实例 ID。
	账号	目标实例账号，账号权限需要满足要求。
	密码	目标实例账号的密码。

示例三：将第三方云厂商 PostgreSQL 通过公网方式同步至腾讯云数据库。

设置项	参数	描述
任务设置	任务名称	DTS 会自动生成一个任务名称，用户可以根据实际情况进行设置。
	运行模式	支持立即执行和定时执行两种模式。
源实例设置	源实例类型	购买时所选择的源实例类型，不可修改。
	源实例地域	购买时选择的源实例所在地域，不可修改。
	接入类型	对于第三方云厂商数据库，一般可以选择公网方式，也可以选择 VPN 接入，专线或者云联网的方式，需要根据实际的网络情况选择。 本场景选择“公网”，不同接入类型的准备工作请参考 准备工作概述 。 公网：源数据库可以通过公网 IP 访问。 云主机自建：源数据库部署在 腾讯云服务器 CVM 上。 专线接入：源数据库可以通过 专线接入 方式与腾讯云私有网络打通。 VPN 接入：源数据库可以通过 VPN 连接 方式与腾讯云私有网络打通。 云数据库：源数据库属于腾讯云数据库实例。 云联网：源数据库可以通过 云联网 与腾讯云私有网络打通。
	主机地址	源实例访问 IP 地址或域名。
	端口	源实例端口。
	账号	源实例账号，账号权限需要满足要求。
	密码	源实例账号的密码。
目标实例设置	目标实例类型	购买时选择的目标实例类型，不可修改。
	目标实例地域	购买时选择的目标实例地域，不可修改。
	接入类型	根据您的场景选择，本场景选择“云数据库”。
	实例 ID	选择目标实例 ID。

	账号	目标实例账号，账号权限需要满足要求。
	密码	目标实例账号的密码。

5. 在设置同步选项和同步对象页面，将对数据初始化选项、数据同步选项、同步对象选项进行设置，在设置完成后单击**保存并下一步**。

说明：
当**初始化类型**仅选择**全量数据初始化**，系统默认用户在目标库已经创建了表结构，不会进行表结构同步，也不会校验源库和目标库是否有同名表，所以当用户同时在**已存在同名表中**选择**前置校验并报错**，则校验并报错功能不生效。

如果用户在同步过程中确定会对某张表使用 `rename` 操作（例如将 `table A` rename 为 `table B`），则**同步对象**需要选择 `table A` 所在的整个库（或者整个实例），不能仅选择 `table A`，否则 `rename` 操作后，`table B` 的数据不会同步到目标库。

设置项	参数	描述
数据初始化选项	初始化类型	结构初始化：同步任务执行时会先将源实例中表结构初始化到目标实例中。 全量数据初始化：同步任务执行时会先将源实例中数据初始化到目标实例中。 仅选择全量数据初始化的场景，用户需要提前在目标库创建好表结构。 默认两者都勾选，可根据实际情况取消。
	已存在同名表	前置校验并报错：存在同名表则报错，流程不再继续。 忽略并继续执行：全量数据和增量数据直接追加目标实例的表中。
数据同步选项	冲突处理机制	冲突报错：在同步时发现表主键冲突，报错并暂停数据同步任务。 冲突忽略：在同步时发现表主键冲突，保留目标库主键记录。 冲突覆盖：在同步时发现表主键冲突，用源库主键记录覆盖目标库主键记录。
	同步操作类型	支持操作： <code>Insert</code> 、 <code>Update</code> 、 <code>Delete</code> 、 <code>DDL</code> 。当前不支持“DDL 自定义”。
同步对象选项	源实例库表对象	选择待同步的对象，支持基础库表、视图。高级对象仅支持 <code>SEQUENCE</code> 。

6. 在校验任务页面，完成校验并全部校验项通过后，单击**启动任务**。
如果校验任务不通过，可以参考 [校验不通过处理方法](#) 修复问题后重新发起校验任务。

失败：表示校验项检查未通过，任务阻断，需要修复问题后重新执行校验任务。
警告：表示校验项检查不完全符合要求，可以继续任务，但对业务有一定的影响，用户需要根据提示自行评估是忽略警告项还是修复问题再继续。

7. 返回数据同步任务列表，任务开始进入**运行中**状态。
说明：
选择**操作列**的**更多 > 结束**可关闭同步任务，请您确保数据同步完成后再关闭任务。

8. （可选）您可以单击**任务名**，进入任务详情页，查看任务初始化状态和监控数据。

PostgreSQL 前置校验不通过处理

连接 DB 检查

最近更新时间：2024-07-08 18:39:54

检查详情

源数据库和目标数据库需要能正常连通，如果未连通，会报连接失败。

问题原因

[源数据库所在网络或服务器设置了安全组或防火墙。](#)

[源数据库对来源 IP 地址进行了限制。](#)

[网络端口未放通。](#)

数据库账号或密码不正确。

修复方法

请按照问题原因中的对应链接进行处理。

对象依赖检查

最近更新时间：2024-07-08 18:39:54

检查详情

1. DTS 任务配置界面选择的是“全量迁移”或者“全量+增量迁移”，但迁移对象中除了公共对象外，没有勾选其他具体要迁移的表，则校验任务会报错，请勾选表对象。
2. 如果待迁移对象勾选了视图，但未勾选视图依赖的表，则校验任务报错，相互关联的对象需要一并迁移。

修复方法

重新配置 DTS 任务，相互关联的对象需要一并迁移。

版本检查

最近更新时间：2024-07-08 18:39:54

检查详情

迁移场景

PostgreSQL 10.x 以前的版本（如 9.x）不支持“全量+增量迁移”，若源库配置了增量迁移任务，则会导致校验失败。当版本不一致时，会存在部分特殊的兼容性问题，任务会发出警告。请自行阅读各版本的兼容性报告，来确认业务是否有使用到不兼容的特性。

同步场景

PostgreSQL 10 之前的版本不支持同步，数据同步时目标实例版本必须大于等于源实例版本。当版本不一致时，会存在部分特殊的兼容性问题，任务会发出警告。请自行阅读各版本的兼容性报告，来确认业务是否有使用到不兼容的特性。

修复方法

请按 [数据迁移支持的数据库](#) 和 [数据同步支持的数据库](#) 中的版本要求检查源库和目标库，如果源库或者目标库版本不支持，请升级目标实例版本或者使用更高版本的数据库实例。

源实例或目标实例权限检查

最近更新时间：2024-07-08 18:39:54

检查详情

检查用户是否具备对数据库的操作权限，请参考操作指导中的对应授权内容。

修复方法

用户可能不具备操作权限，请按照检查要求中的对应权限要求对用户进行授权，然后重新执行校验任务。

增量迁移预检查

最近更新时间：2024-07-08 18:39:54

检查详情

当迁移类型选择增量迁移时，需要对如下条件进行检查，否则校验失败。

源和目标库的主版本号需要为 PostgreSQL 10.x 之前。

源实例的 `wal_level` 必须为 `logical`。

目标库 `max_replication_slots` 和 `max_wal_senders` 参数需要大于待迁移的数据库总数。

目标实例的 `max_worker_processes` 必须大于 `max_logical_replication_workers` 的值。

待迁移表中不能存在 `unlogged table`，否则无法迁移。

修复方法

如果版本不符合要求，请升级版本。修改参数

`wal_level`，`max_replication_slots`，`max_worker_processes` 和 `max_wal_senders` 的方法如下。

1. 登录源数据库。

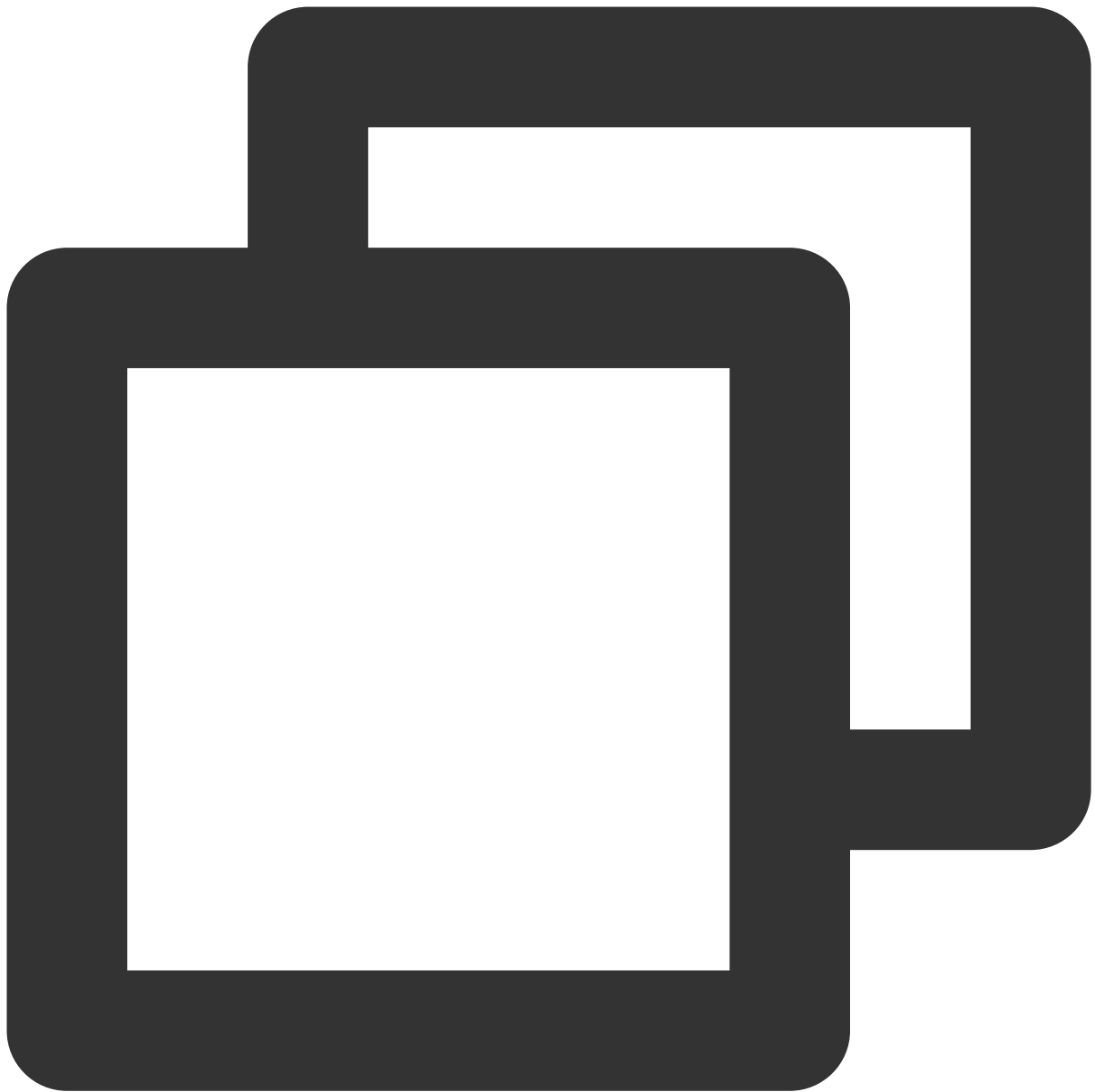
说明：

如源数据库为自建数据库，需要登录至数据库的运行服务器上，进入数据库数据主目录中，一般为 `$PGDATA`。

如源数据库为其他云数据库，请使用相关云平台的参数修改方法。

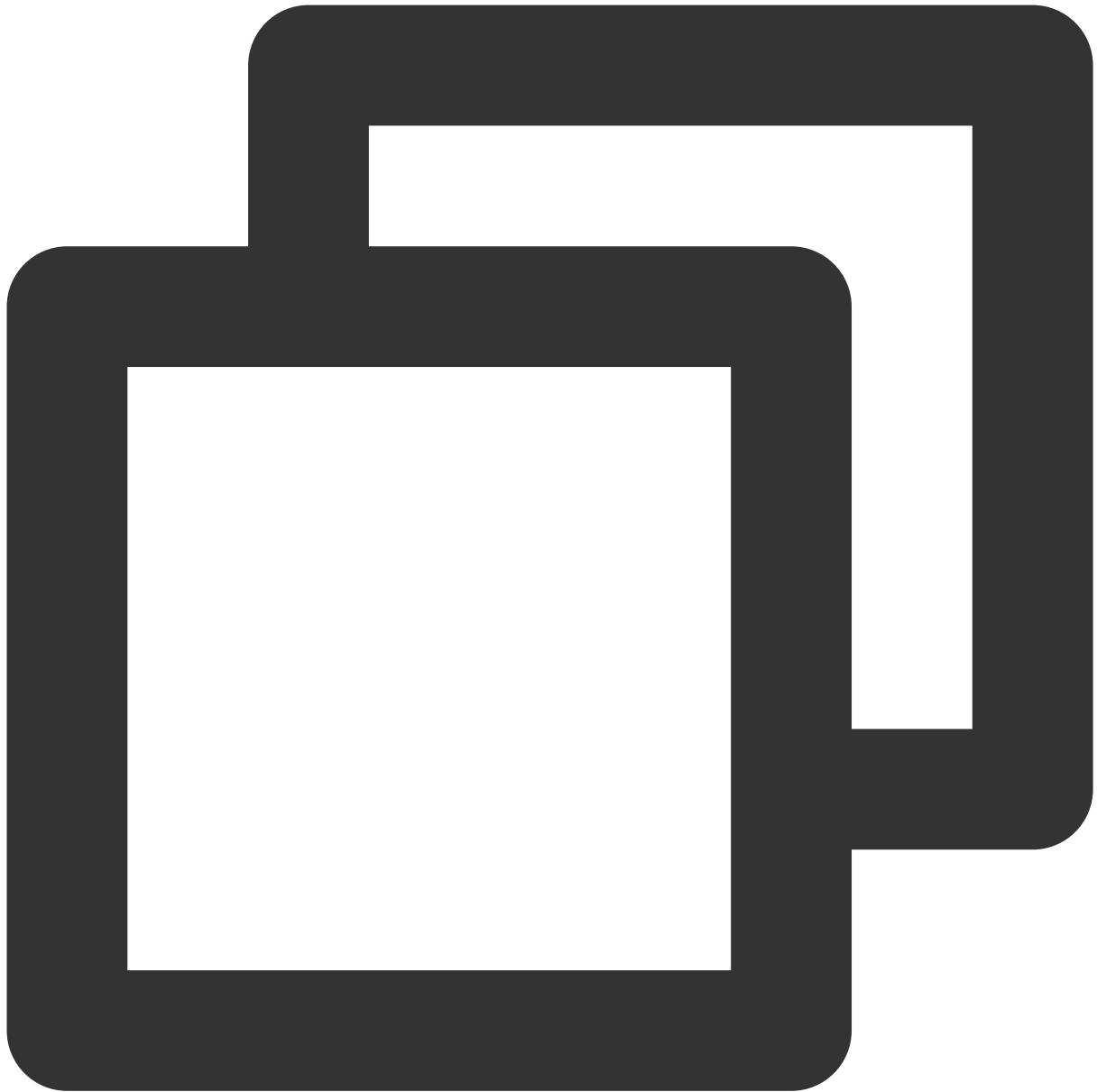
如需要修改目标实例的参数，请通过 [提交工单](#)处理。

2. 找到 `postgresql.conf` 文件，打开此文件，修改 `wal_level`。



```
wal_level = logical
```

3. 修改完成后，重启数据库实例。
4. 登录至数据库实例中，使用以下命令查看参数值是否正确：



```
postgres=> select name,setting from pg_settings where name='wal_level';
name      | setting 
-----+-----
 wal_level | logical
(1 row)
postgres=> select name,setting from pg_settings where name='max_replication_slots';
          name      | setting 
-----+-----
max_replication_slots | 10
(1 row)
postgres=> select name,setting from pg_settings where name='max_wal_senders';
```

```
name          | setting
-----+-----
max_wal_senders | 10
(1 row)
```

5. 重新执行校验任务。

结构兼容性检查

最近更新时间：2024-07-12 09:43:00

检查详情

如果目标库版本在12及以上，且待迁移的表包含 `abstime`、`reltime`、`tinterval` 三种数据类型，则校验任务报错。

`abstime`、`reltime`、`tinterval` 数据类型是一些较旧的时间和日期类型，在新的 PostgreSQL 版本中已被废弃，建议使用以下类型进行替代：

`abstime`：可以使用 `timestamp` 或 `timestamp with time zone` 类型来替代。这些类型提供了更广泛的日期和时间表示能力。

`reltime`：可以使用 `interval` 类型来替代。`interval` 类型用于表示时间间隔，可以包含年、月、日、小时、分钟、秒等单位。

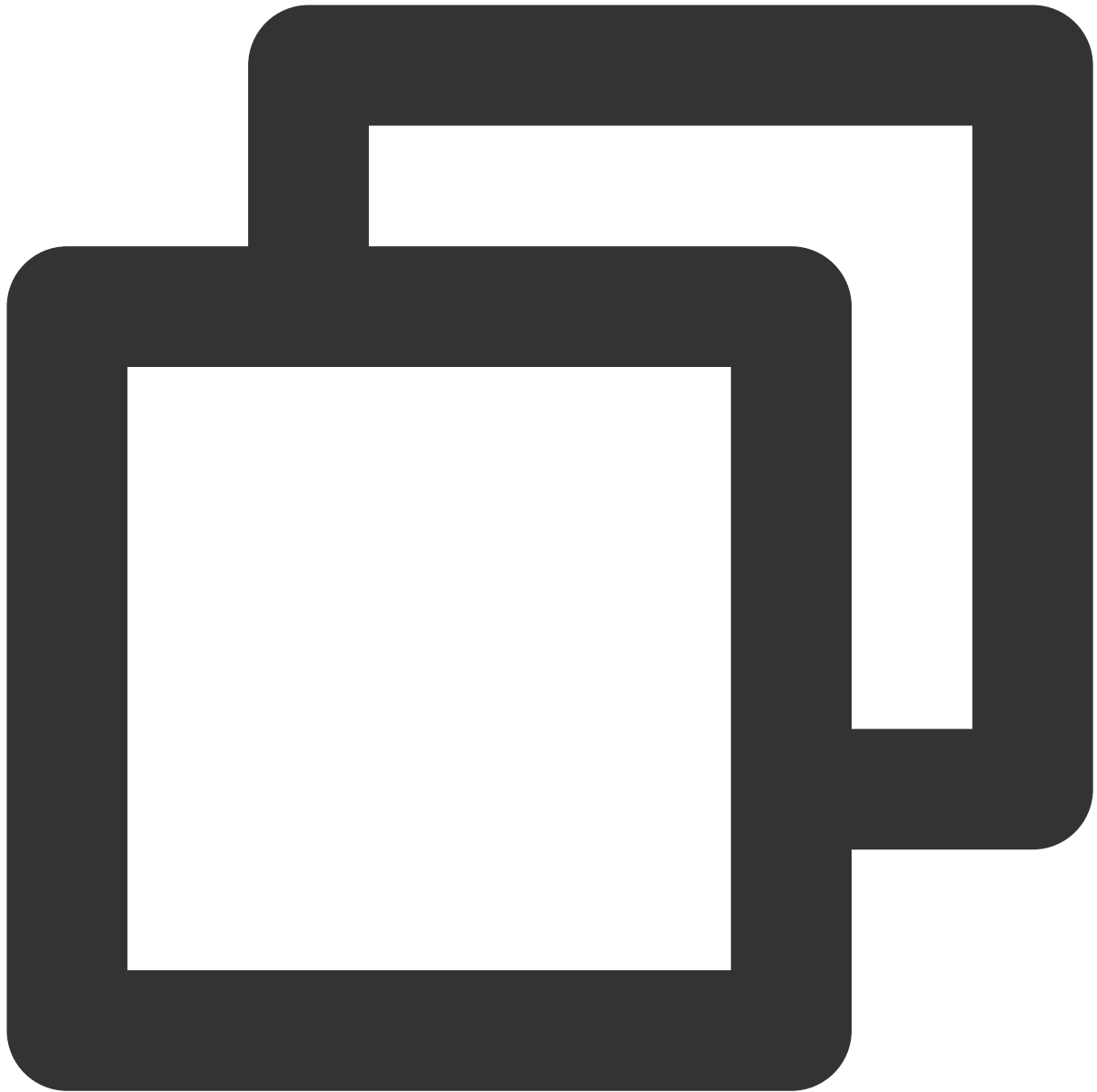
`tinterval`：可以使用 `tsrange` 或 `tstzrange` 类型来替代。这些类型用于表示时间范围，可以包含起始时间和结束时间。

修复方法

`abstime`、`reltime`、`tinterval` 三种数据类型的修改方法类似，都是先新建一个列，将旧版数据格式转化为新版数据格式并存储在新列中，验证新列数据无误后删除旧的列，最后根据需要，将新的列重命名为旧的列的名称。

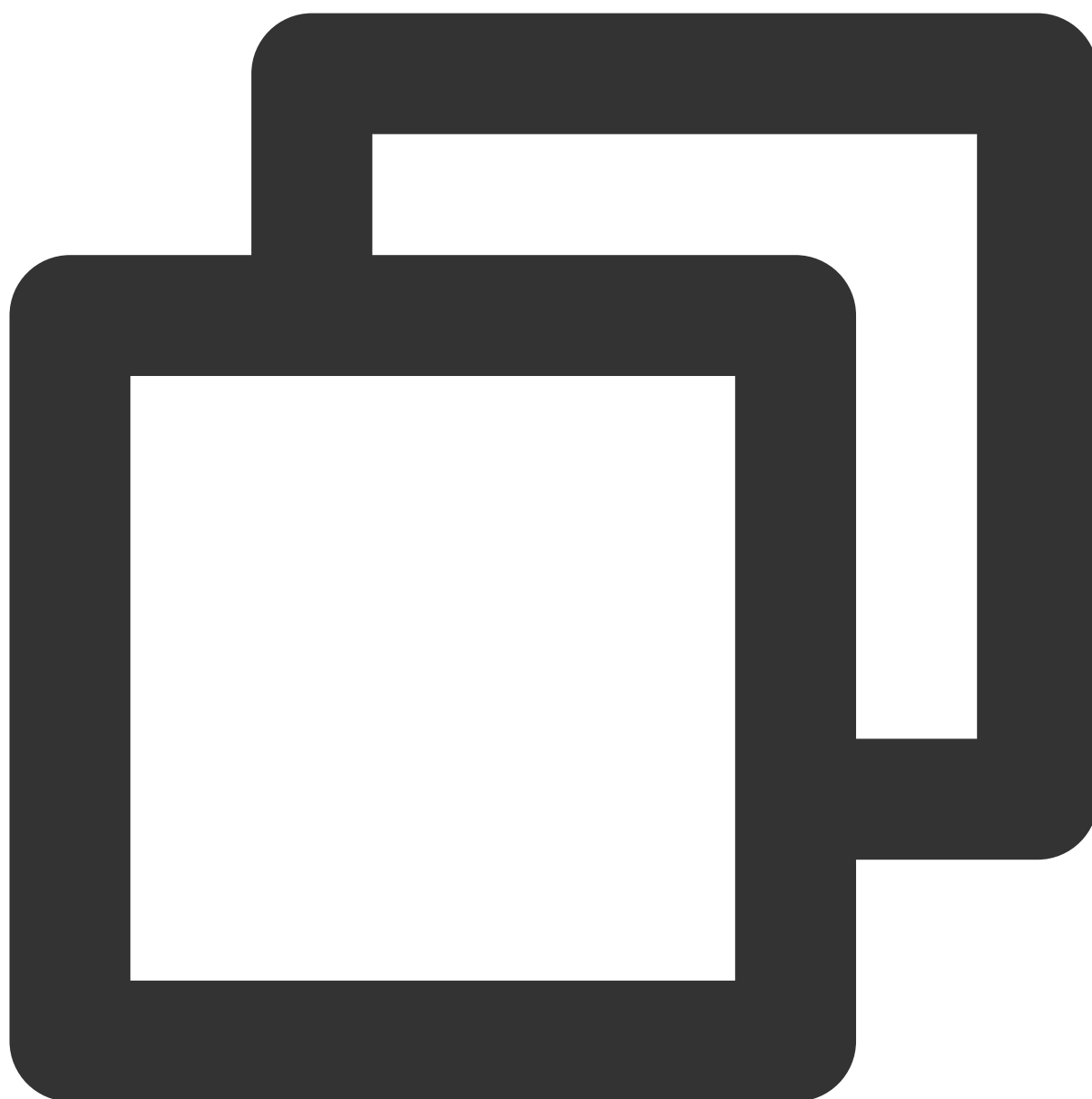
如下以将 `abstime` 修改为 `timestamp` 为例进行介绍。

1. 在校验任务提示报错的表中，新增一个 `timestamp` 类型的列。



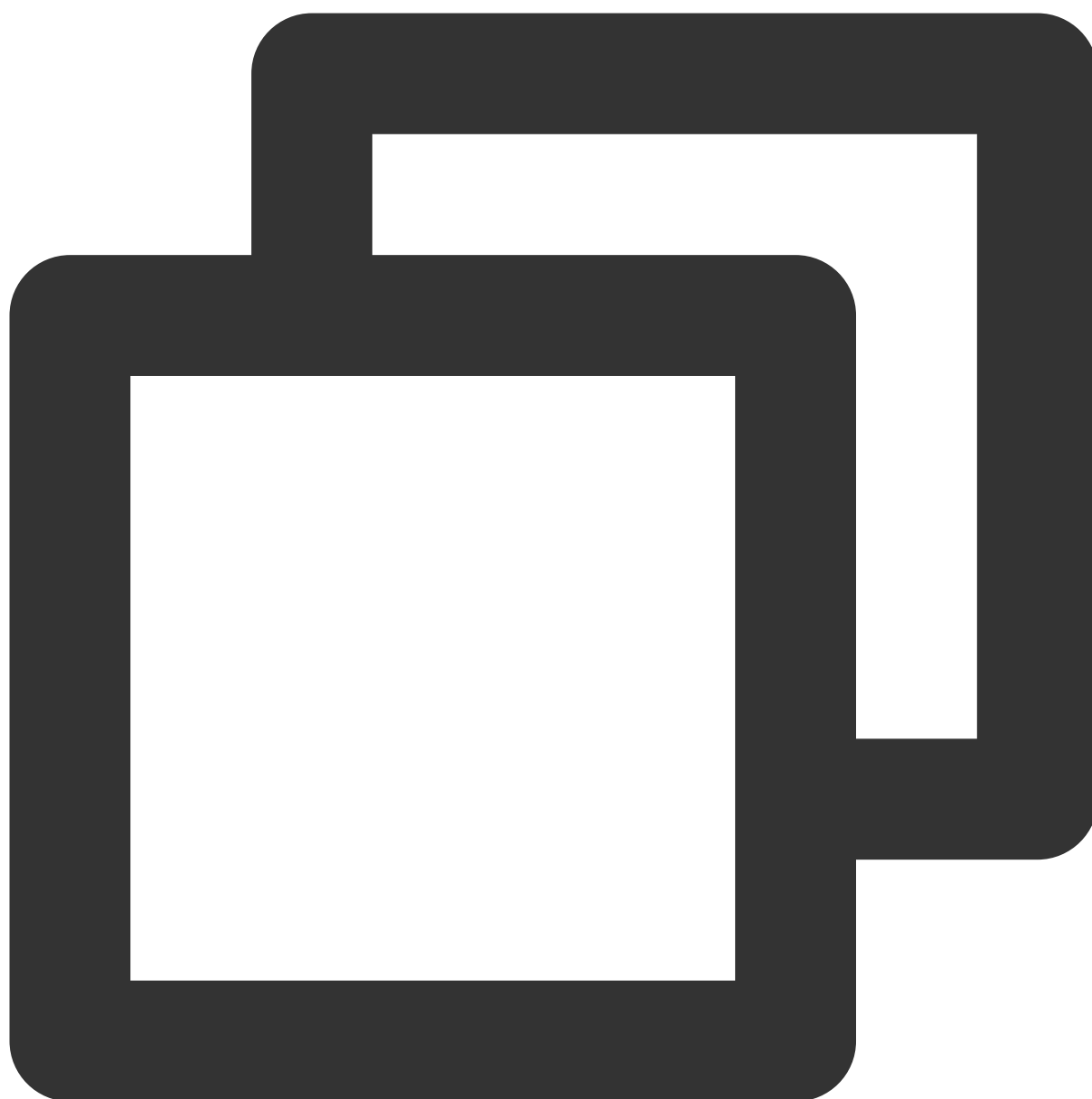
```
ALTER TABLE your_table ADD COLUMN new_column TIMESTAMP;
```

2. 将 `abstime` 列的数据转换为 `timestamp` 类型，并将结果存储在新的列中。



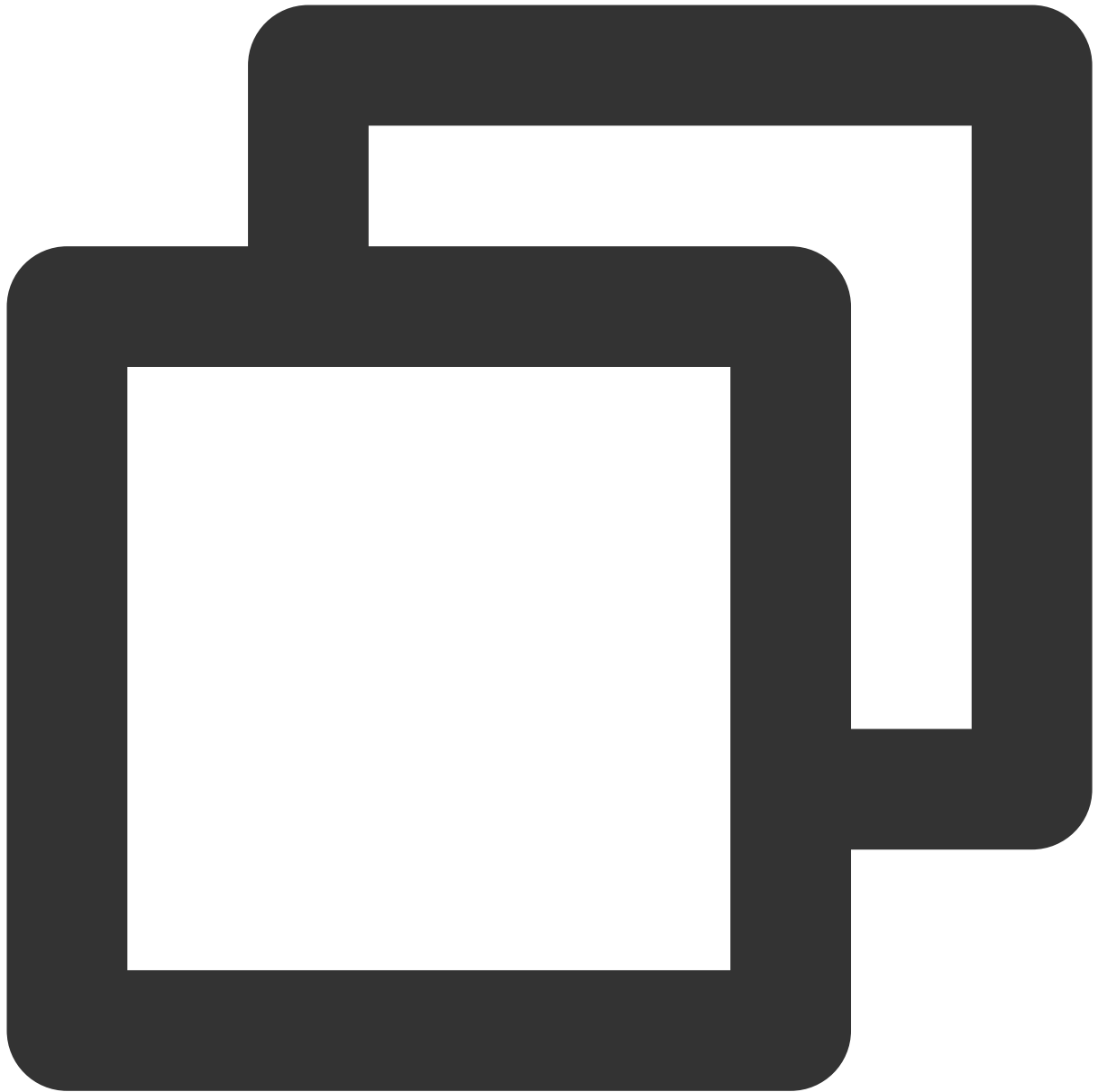
```
UPDATE your_table SET new_column = your_abstime_column::TIMESTAMP;
```

3. 验证新的列中的数据是否正确。



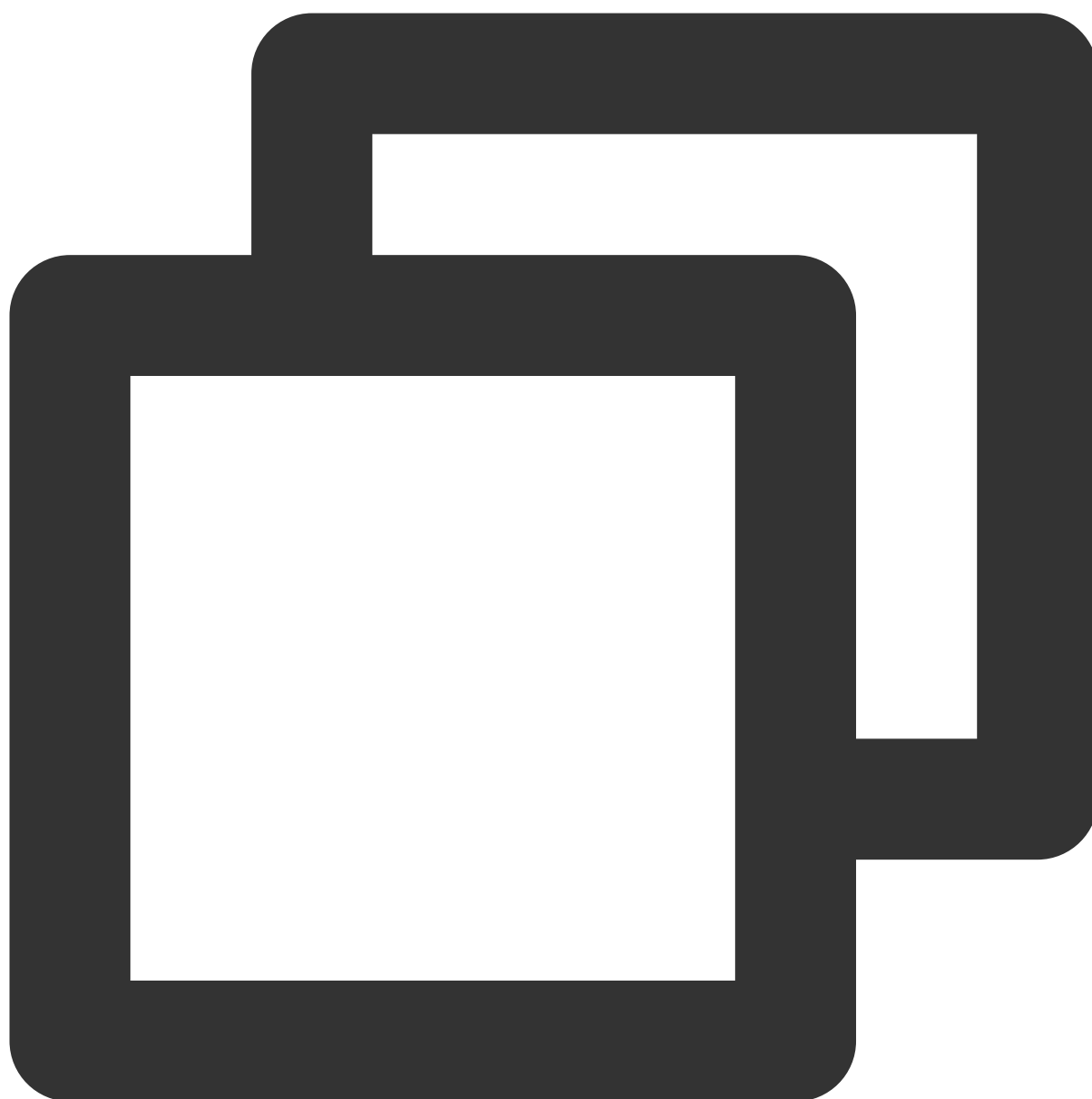
```
SELECT * FROM your_table;
```

4. 如果确认新的列中数据正确，可以删除旧的 `abstime` 列。



```
ALTER TABLE your_table DROP COLUMN your_abstime_column;
```

5. 如果需要，可以将新的列重命名为旧的列的名称。



```
ALTER TABLE your_table RENAME COLUMN new_column TO your_abstime_column;
```

结构冲突检查

最近更新时间：2024-07-08 18:39:54

检查要求

PostgreSQL 迁移场景中，目标实例不能有和源库同名的对象。

PostgreSQL 整个实例迁移时，目标实例必须为空。如果存在冲突报错，需要删除实例内容。

修复方法

如果检测到冲突，请删除冲突内容，然后重新校验。

账号冲突检查

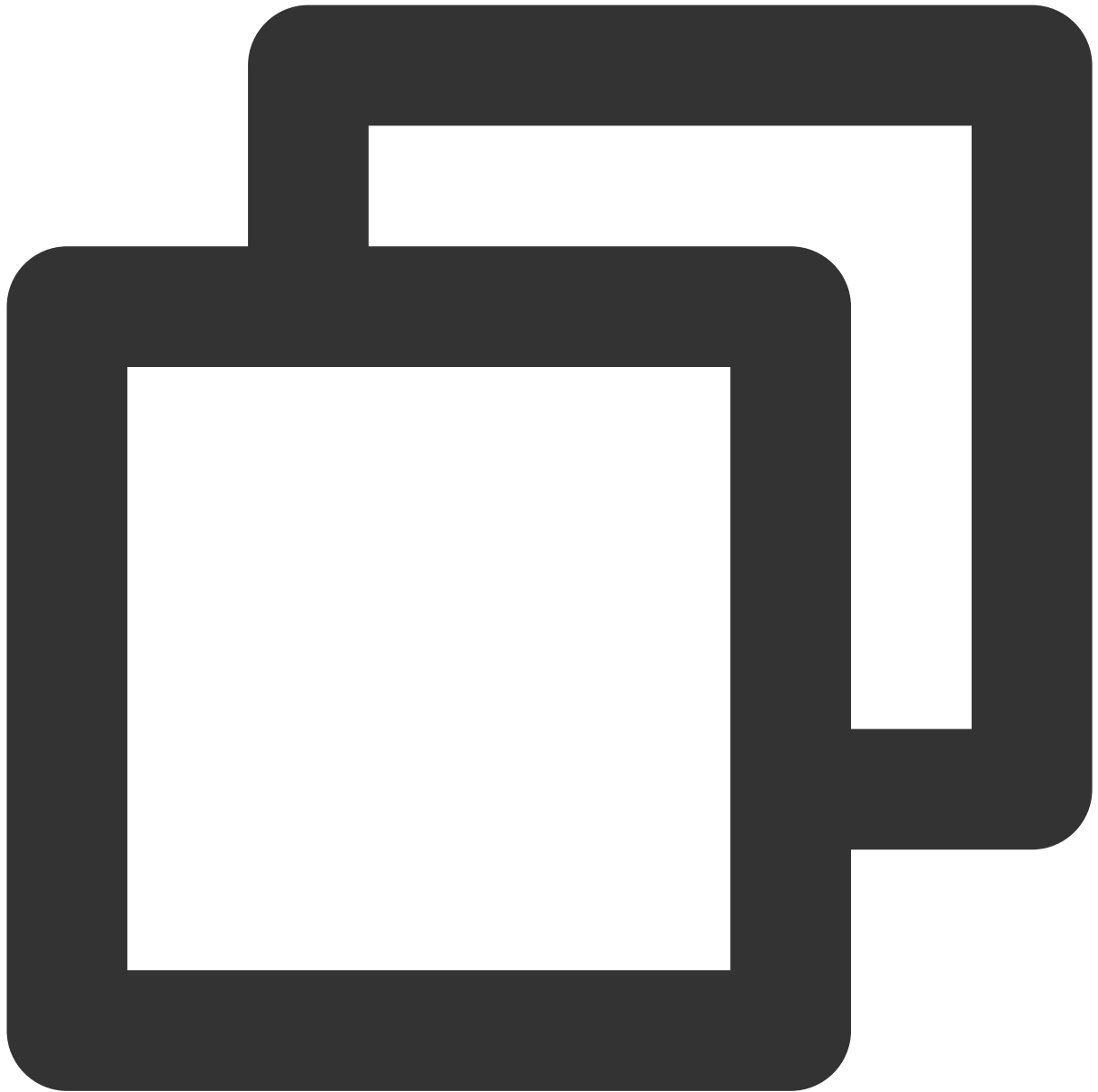
最近更新时间：2024-07-08 18:39:54

检查详情

检查目标库用户是否与源数据库的用户重复。

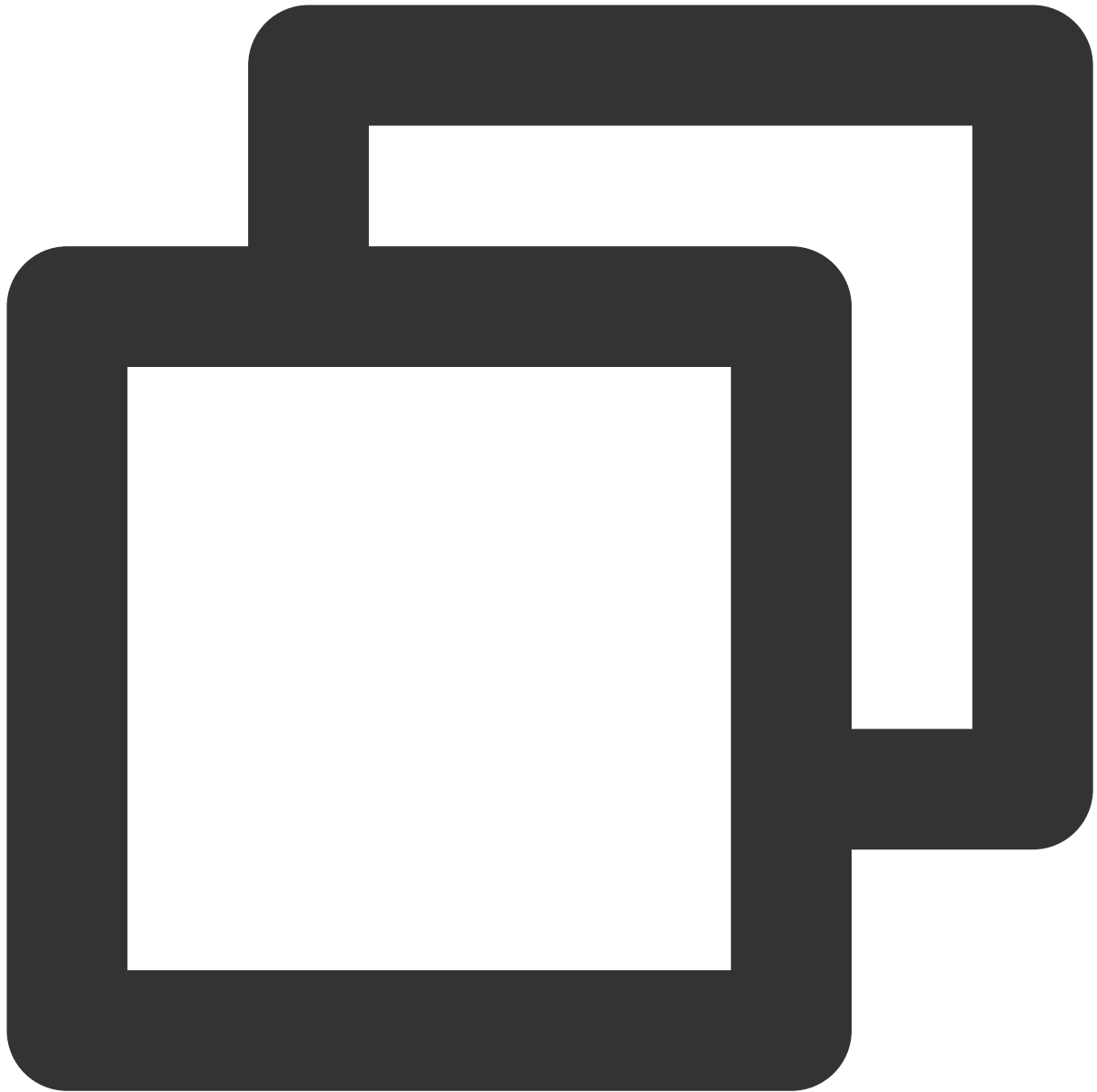
修复方法

在整实例迁移的场景下，如果目标实例中存在和源实例一模一样的帐号，需要将目标实例中的同名帐号删除。
如果目标库中的帐号为初始化帐号，请使用初始化帐号登录至数据库中，执行以下语句。



```
create user 新用户 with password '密码';  
grant pg_tencentdb_superuser to 新用户名;  
alter user 新用户 with CREATEDB;  
alter user 新用户 with CREATEROLE;
```

如果目标库中的帐号为新增用户，则使用新创建的用户登录至数据库中，删除冲突用户。



```
drop user 冲突用户;  
# 如果冲突用户存在资源依赖, 则请先修改依赖对象的owner, 如表的owner修改语句为:  
alter table 表名 owner to 新用户;
```

当冲突用户删除完成后, 请重新执行校验任务。

参数配置冲突检查

最近更新时间：2024-07-08 18:39:54

检查详情

检查要求：对源和目标数据库的如下参数值进行检查，如果源和目标库的参数值不一致，则会提示校验警告。警告不会阻塞迁移任务，但是会对业务有一定的影响，请评估后自行决定是否修改。

TimeZone, lc_monetary, lc_numeric, array_nulls, server_encoding, DateStyle, extra_float_digits, gin_fuzzy_search_limit, xmlbinary, constraint_exclusion。

业务影响：如果参数未设置一致，可能会导致源库和目标库的数据不一致，具体影响如下。

TimeZone：设置实例的时区，此参数值如果不一致，可能会导致迁移后数据错误。

lc_monetary：设置实例货币模式，此参数值如果不一致，可能会导致迁移后货币数字错误。

lc_numeric：设置实例数字模式，此参数值不一致，可能会导致迁移后数据错误。

array_nulls：设置数组是否允许为空，此参数值不一致，可能会导致迁移数据不一致，某一些数据无法迁移成功。

server_encoding：设置实例的字符集，此参数值不一致，可能会导致数据保存乱码。

DateStyle：设置日期显示格式，此参数值不一致，可能会导致数据无法迁移成功。

extra_float_digits：设置浮点值的输出精度，此参数值不一致，会影响数据精度问题，在高精度的数据库使用场景，会导致迁移后的数据不一致问题。

gin_fuzzy_search_limit：设置 GIN 索引返回的集合尺寸的软上限，此参数值不一致，会导致迁移后数据显示结果不一致的问题。

xmlbinary：设置 xml 函数转换的结果问题，此参数值不一致，可能会导致在目标库中执行相应函数时与源库的行为不一致的问题。

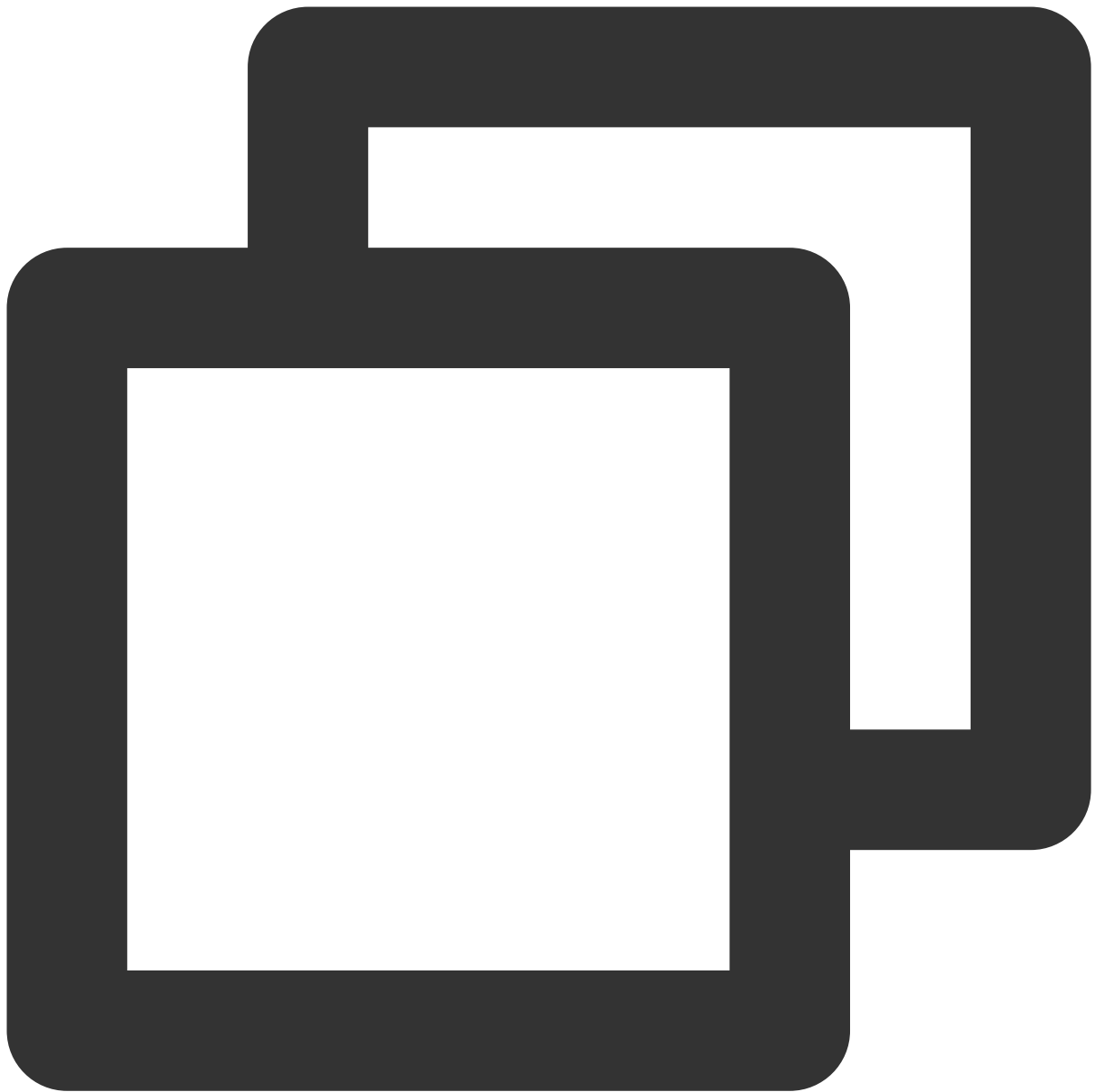
constraint_exclusion：设置约束是否生效，此参数值不一致，可能会导致迁移后数据不一致的问题。

修复方法

1. 使用 superuser 账号登录源数据库。
2. 执行下列示例语句修改对应的参数：

用户可先选择源库的参数修改，如源数据库对应参数不能修改，则需要修改目标库对应参数，目标库的修改请通过[提交工单](#)处理。

server_encoding 参数无法在源库修改，如果该参数异常，请检查该参数在目标库是否已经创建，如果已经创建，且和源库不一致，则需要申请新的实例，如果未创建，则参考如下方法修改（当前云数据库实例仅支持 UTF8 与 LATIN 两种字符集）。



```
alter system set timezone='参数值';  
alter system set lc_monetary='参数值';  
alter system set lc_numeric='参数值';
```

插件兼容性检查

最近更新时间：2024-07-08 18:39:54

检查详情

检查源库已存在的插件，目标库是否也同样存在。

如果源端有插件迁移，目标端无创建插件权限，校验任务报错。

如果源端有插件迁移，目标端未安装对应插件，或者版本号不一致，校验任务报错。

修复方法

请在目标库上安装与源库相同的插件版本，同时在目标库上授权插件的对应权限，具体可参考 [PostgreSQL 官方指导](#)。

主键依赖检查

最近更新时间：2024-07-08 18:41:31

检查详情

PostgreSQL 同步场景，检查待同步表格是否有主键，若没有主键，任务校验会告警，会自动设置 `ALTER TABLE XXX REPLICA IDENTITY`。

修复方法

请按照界面提示，设置主键，然后重新执行校验任务。

实例关键参数检查

最近更新时间：2024-07-08 18:41:50

检查详情

PostgreSQL 同步场景会检查复制参数 `wal_level` 是否等于 `logical`，同时 `replication_slots` 需满足选择的待同步库数量+`replication_slots` $\leq$ `max_wal_senders`。

修复方法

设置实例参数 `wal_level` 为 `logical`。

减少待同步的库数量。

多任务冲突检测

最近更新时间：2024-07-08 18:39:54

检查详情

PostgreSQL 同步任务中，在配置任务时不支持环形链路，如果为环形链路则任务校验会报错。如果源端同步对象已经存在多个任务，则会告警，请检查任务配置是否合理。

修复方法

请按照界面提示，取消已经存在回环或者重复配置同步的任务，然后重新执行校验任务。

同步至 Kafka

数据同步至 Kafka 与数据订阅的区别

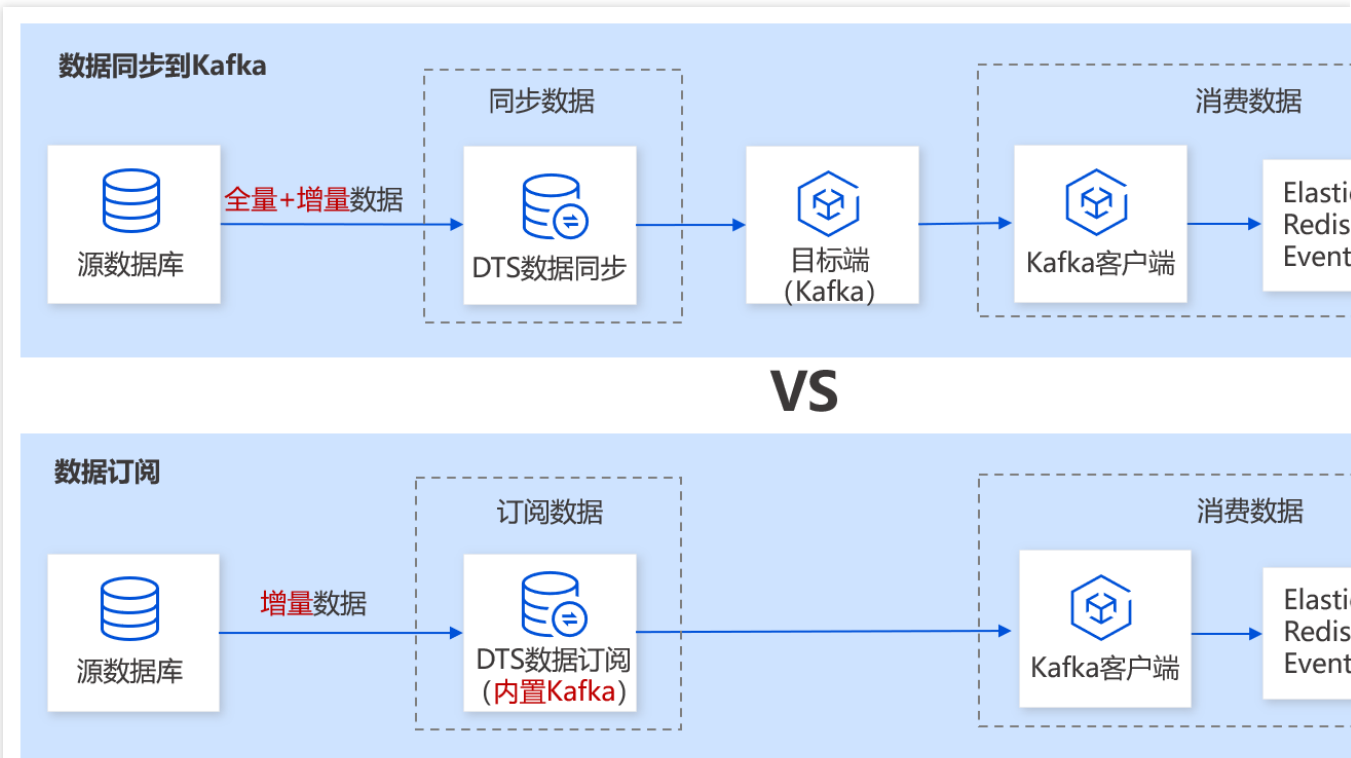
最近更新时间：2023-12-11 11:30:40

同步至 Kafka 与数据订阅实现原理类似，两者都可实时获取源库的数据变更，可应用于数据归档、数据分析等场景中。

如果用户已有自己的 Kafka（腾讯云 Ckafka），或者需要对源库的全量+增量数据进行同步，建议选择 [同步至 Kafka](#) 的功能。

如果用户没有自己的 Kafka（腾讯云 Ckafka），并仅需要获取源库的库、表的增量变更，建议选择 [数据订阅（Kafka 版）](#) 功能。

如果两个方案都可以满足业务使用，建议对比两个方案的计费金额，选择最便宜的方案，计费详情请参考 [计费概述](#)。



详细的功能差别如下：

功能	数据同步至 Kafka	数据订阅
支持的链路	MySQL/MariaDB/Percona/TDSQL-C MySQL/TDSQL MySQL 数据同步至 Ckafka	源端为 MySQL/MariaDB/Percona/TDSQL-C MySQL/TDSQL MySQL/TDSQL PostgreSQL/MongoDB
数据格式	Avro、JSON	Protobuf、Avro、JSON

目标端 Kafka	用户自己的 Kafka（当前仅支持目标端为腾讯云 Ckafka），可自行设置 Topic 名称。	DTS 数据订阅内置 Kafka，用户不可设置 Topic 名称。
数据类型	全量+增量数据。	仅增量数据。
同步对象	同步源库的基础库、表、视图、函数、存储过程。	订阅源库的基础库、表对象。
SQL 过滤	支持 SQL 过滤，可筛选 INSERT、UPDATE、DELETE 和具体的 DDL，可进行 where 条件过滤。	仅可选择结构更新，或者数据更新，不能进行 SQL 过滤。

数据投递到 Kafka 策略设置

最近更新时间：2024-07-08 18:44:49

操作场景

源端数据同步到 Kafka 时，支持灵活的投递策略，可以将不同表的数据投递到不同的 Topic 中，也可以集中投递到单 Topic 中。不同策略的说明如下：

策略类别	单 Topic 分区策略	功能场景
自定义 Topic 名称	不涉及	可将不同的库、表数据投递到不同的 Topic 中。
集中投递到单 Topic	Topic 分区策略（默认分区策略）	全部投递至 Partition0：全部投递到单 Topic 中的第一个分区。 按表名分区：将同一个表的数据投递到同一个分区。 按表名 + 主键分区：将同一个表中相同主键值的数据投递到同一个分区。适用于热点数据，设置后热点数据的表，可以分散投递到不同分区。
	Topic 分区策略（默认分区策略）+ 自定义分区策略	1. 先将匹配到的库、表数据，按照自定义分区策略（支持按表名分区、按表名+主键分区、按列分区）进行投递。 2. 再将剩余未匹配到的库、表数据，按照 Topic 分区策略（默认分区策略），进行投递。

自定义 Topic 名称

功能说明

用户自行设置投递的 Topic 名称，DTS 按照填入的 Topic 名称往目标端 Kafka 写入：

如果目标端有该 Topic，或者没有该 Topic 但是 `auto.create.topics.enable = ture`，DTS 会写入成功。

如果目标端没有该 Topic，并且 `auto.create.topics.enable = false`，则 DTS 写入失败，同时同步任务会报错。

`auto.create.topics.enable` 为 Kafka 中的配置参数，用于控制是否允许自动创建 Topic，通常在 Kafka 的配置文件 `server.properties` 中修改。当设置为 `true` 时，Kafka 在收到不存在的 Topic 的请求时会自动创建该 Topic。当设置为 `false` 时，不会自动创建 Topic。

配置规则

选择自定义 Topic 名称，用户设置多条规则时，DTS 会从上到下逐条匹配。

如果匹配到设置的规则（源端的库名和表名都符合设置的规则），则会投递到该条规则对应的 Topic 中。

如果匹配到多条规则，则会投递到所有匹配规则的 Topic 中。

剩余没有匹配到设置规则的数据，则会投递到最后一条规则的 Topic 中。

匹配规则对库名、表名大小写敏感。

如果源库设置 lower_case_table_names = 0，则设置的匹配规则中库表名需要与源库中的名称大小写严格保持一致。

如果源库设置 lower_case_table_names = 1，则库表名统一转换为小写，设置的匹配规则中库表名统一输入小写。

库表、表名的匹配规则支持 RE2 正则表达式，具体语法请参考 [语法说明](#)。如果需要精确匹配，则需加开始符“^”和结束符“\$”，如精准匹配“test”表应该为“^test\$”。

配置示例

数据库实例 X 中，Users 库里有表 Teacher、Student、Student1、Student2、Student3。

示例一：将 Users 库中的数据都投递到 Topic_A 中，剩余的数据投递到 Topic_default 中。

同步 Topic 策略 *

自定义 Topic 名称

集中投递到单 Topic

随机投递到不同的 Partition 中

Topic *

自定义 Topic 匹配规则从上到下逐条匹配，如果没有匹配到库表匹配规则，则会投递到最后一条规则的Topic中；如果匹配成功，则不再继续匹配。

举例：数据库实例X中，Users库里有Student、Teacher表，需要将Users库里的数据都投递到“users”的Topic中

配置规则设置：Topic 名称 = users，库名匹配规则 = ^Users\$，表名匹配规则 = .*

Topic 名称	库名匹配模式	表名匹配模式
Topic_A	^Users\$	.*
Topic_default	不符合匹配规则的剩余库	不符合匹配规则的剩余表

添加

Topic 名称	库名匹配模式	表名匹配模式	说明
Topic_A	^Users\$	.*	“^Users\$”表示精准匹配Users库，“^”为开始符，“\$”为结束符；仅填入“Users”表示匹配所有包含Users的库名，如表X_Users_1会匹配上。 “.*”表示匹配所有的表名。
Topic_default	不符合匹配规则的剩余库	不符合匹配规则的剩余表	-

示例二：将“Teacher”表的数据投递到“Topic_A”，“Student”表的数据投递到“Topic_B”，剩余数据投递到“Topic_default”。

同步 Topic 策略

自定义 Topic 名称

集中投递到单 Topic

随机投递到不同的 Partition 中

Topic

自定义 Topic 匹配规则从上到下逐条匹配，如果没有匹配到库表匹配规则，则会投递到最后一条规则的Topic中；如果匹配到多条库表匹配规则，则会投递到所有匹配规则的
举例：数据库实例X中，Users库里有Student、Teacher表，需要将Users库里的数据都投递到“users”的Topic中
配置规则设置：Topic 名称 = users，库名匹配规则 = ^Users\$，表名匹配规则 = .*

Topic 名称	库名匹配模式	表名匹配模式	操作
Topic_A	^Users\$	^Teacher\$	删除
Topic_B	^Users\$	^Student\$	删除
Topic_default	不符合匹配规则的剩余库	不符合匹配规则的剩余表	删除

添加

Topic 名称	库名匹配模式	表名匹配模式	说明
Topic_A	^Users\$	^Teacher\$	“^Teacher\$”表示精准匹配表名Teacher，“^”为开始符，“\$”为结束符；如果仅填入“Teacher”，则会匹配包含Teacher的所有表，如F_Teacher_1会匹配到。
Topic_A	^Users\$	^Student\$	-
Topic_default	不符合匹配规则的剩余库	不符合匹配规则的剩余表	-

示例三：将“Teacher”表数据投递到“Topic_A ”中，前缀为“Student”的表（即表Student、Student1、Student2、Student3）都投递到“Topic_B”，剩余数据投递到“Topic_default”。

同步 Topic 策略 *

自定义 Topic 名称

集中投递到单 Topic

随机投递到不同的 Partition 中

Topic *

自定义 Topic 匹配规则从上到下逐条匹配，如果没有匹配到库表匹配规则，则会投递到最后一条规则的Topic中；如果匹配到多条库表匹配规则，则会投递到所有匹配规则的Topic
举例：数据库实例X中，Users库里有Student、Teacher表，需要将Users库里的数据都投递到“users”的Topic中
配置规则设置：Topic 名称 = users，库名匹配规则 = ^Users\$，表名匹配规则 = .*

Topic 名称	库名匹配模式	表名匹配模式	操作
Topic_A	^Users\$	^Teacher\$	删除
Topic_B	^Users\$	^Student	删除
Topic_default	不符合匹配规则的剩余库	不符合匹配规则的剩余表	删除

添加

Topic 名称	库名匹配模式	表名匹配模式	说明
Topic_A	^Users\$	^Teacher\$	-
Topic_A	^Users\$	^Student	“^Student”表示匹配所有前缀为Student 的表，即 Student、Student1、Student2、Student3都匹配。
Topic_default	不符合匹配规则的剩余库	不符合匹配规则的剩余表	-

集中投递到单 Topic（默认分区策略）

选择一个目标端已有的 Topic，然后按照多种分区策略投递，支持“单分区”、“按表名分区”、“按表名 + 主键分区”。

数据同步到 Kafka 策略

⚠ 请确认前往Ckafka控制台设置 Topic 的消息体大小上限，若某条消息超出限制，则会导致任务失败，点击查看 [配置消息大小](#)。

同步 Topic 策略 *

自定义 Topic 名称

集中投递到单 Topic

支持多种分区策略，支持单分区或者按表名或者按表名+主键划分分区，投递到不同的 Partition

Topic *

Topic 分区策略 *

全部投递至 Partition0

按表名分区

按表名 + 主键分区

存储 DDL 的 Topic

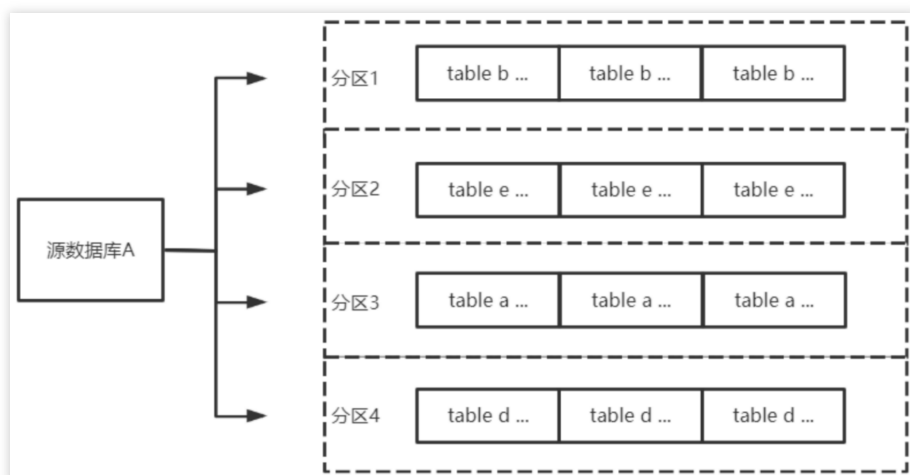
推荐选择一个单独的 Topic，用以将 DDL 操作投递到所选 Topic，默认投递到 Partition0

全部投递至 Partition0（即单分区）

将源库的同步数据全部投递到单 Topic 中的第一个分区。

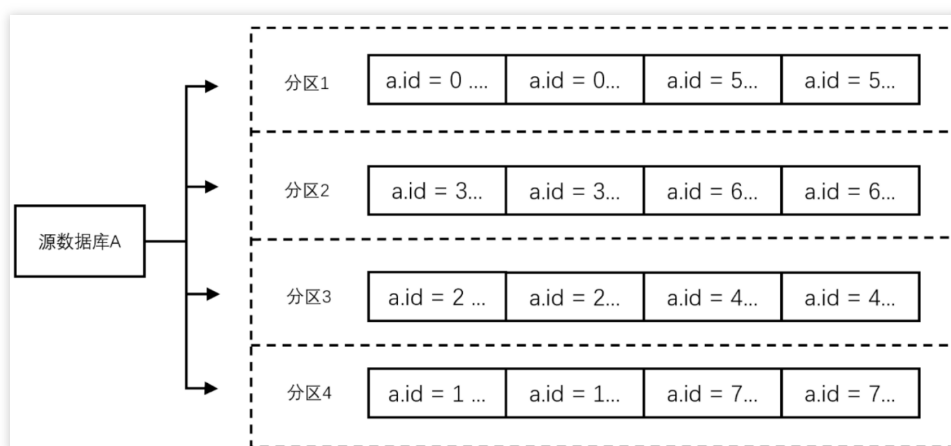
按表名分区

将源库的同步数据按照表名进行分区，设置后相同表名的数据会投递同一个分区中。



按表名 + 主键分区

将源库的同步数据按照表名 + 主键分区，设置后同一个表中主键值相同的数据会投递同一个分区中。该策略适用于热点数据，设置后热点数据的表，将分散到不同分区中，提升并发消费效率。



自定义分区策略

当 Topic 分区策略（默认分区策略）选择按“表名分区”、“按表名 + 主键分区”时，支持勾选自定义分区策略。

自定义分区策略是通过正则表达式对库名和表名进行匹配，将匹配到的数据按照表名、表名 + 主键、列进行分区，剩余未匹配到的数据再按照 Topic 分区策略（默认分区策略）的设置进行分区。

数据同步到 Kafka 策略

① 请确认前往Ckafka控制台设置 Topic 的消息体大小上限，若某条消息超出限制，则会导致任务失败，点击查看 [配置消息大小](#)。

同步 Topic 策略

自定义 Topic 名称

集中投递到单 Topic

支持多种分区策略，支持单分区或者按表名或者按表名+主键划分分区，投递到不同的 Partition

Topic

test1

Topic 分区策略

全部投递至 Partition0

按表名分区

按表名+主键分区

使用自定义分区策略

自定义分区策略

满足下列库表规则的对象，将按照自定义分区规则分区，库表规则需满足 RE2 正则表达式语法，[语法说明](#)

填写示例：
分区匹配规则：匹配规则从上到下逐条匹配，如果匹配到某条库表匹配规则，按照规则投递到对应分区；**如果某条规则已经匹配到，下方的规则不会继续匹配**；如果没有匹配到任意一条库表匹配规则，则会投递到最后一条规则中。
按表名分区：匹配正则表达式中的库名及表名的数据表，保证相同表名的消息在同一个分区中，保证一个表内的数据变更，总是顺序获得
按表名+主键分区：匹配正则表达式中的库名及表名的数据表，通过表名+主键值将数据分散到不同分区中；一张表内的数据会按照主键值被分散到各个分区中
按列分区：匹配正则表达式中的库名及表名的数据表，通过指定列的值将数据分散到不同分区中；一张表内的数据会按照指定列的值被分散到各个分区中

库名匹配模式	表名匹配模式	分区策略	自定义分区列	操作
^A\$	^test\$	按表名分区	—	删除
不符合匹配规则的剩余库	不符合匹配规则的剩余表	按表名分区 ①	—	删除

添加

请添加自定义分区策略

策略组合结果

开启自定义分区策略，将优先匹配自定义策略，其次匹配 Topic 分区策略。
库名匹配正则表达式 ^A\$ 的数据，表名匹配正则表达式 ^test\$ 的数据，将按照‘表名’进行分区，路由至 Kafka 分区
对于不满足上述自定义分区策略的库表，按照默认策略：‘按表名分区’路由至 Kafka 分区

存储 DDL 的 Topic

请输入 Topic 名称

推荐选择一个单独的 Topic，用以将 DDL 操作投递到所选 Topic，默认投递到 Partition0

匹配规则

库表、表名的匹配规则支持 RE2 正则表达式，具体语法请参考 [语法说明](#)。

库名的匹配规则，按照正则表达式来匹配**库名**。表名的匹配规则，按照正则表达式来匹配**表名的数据**。如果库名和表名需要精确匹配，需要加开始和结束符，如 `test` 表应该为 `^test$`。

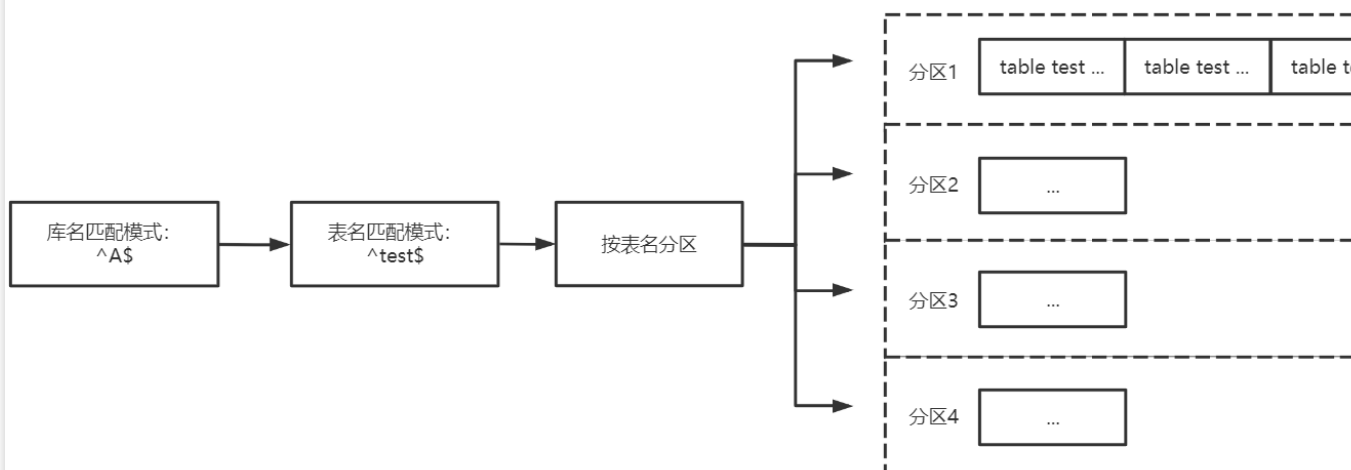
列名的匹配规则，按照等值 `==` 来匹配，大小写不敏感。

优先匹配自定义分区策略，当自定义分区策略有多条时，自上向下逐条匹配。自定义分区策略未匹配到的剩余数据，再按照 Topic 分区策略进行投递。

按表名分区

库名匹配模式填入 `^A$`，表名匹配模式填入 `^test$`，选择**按表名分区**后，`A` 库中 `test` 的数据会投递到同一个分区中，`test` 除外其他未匹配到的库表数据会根据 Topic 分区策略（默认分区策略）中设置的策略进行投递。

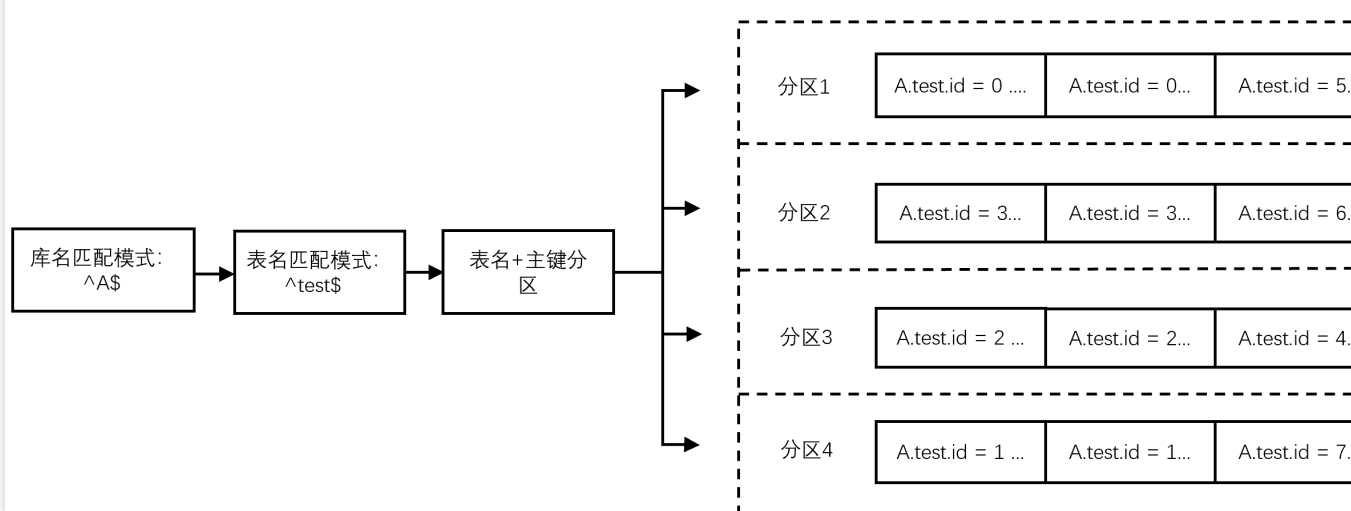
自定义分区策略：按表名（分区=4）



按表名 + 主键分区

库名匹配模式填入 `^A$`，表名匹配模式填入 `^test$`，选择**按表名+主键分区**后，`A` 库中 `test` 的数据会根据主键数据的不同，散列投递到不同的分区中，最终主键数据相同的数据都投递到了同一个分区。`test` 除外其他未匹配到的库表数据会根据 Topic 分区策略（默认分区策略）中设置的进行投递。

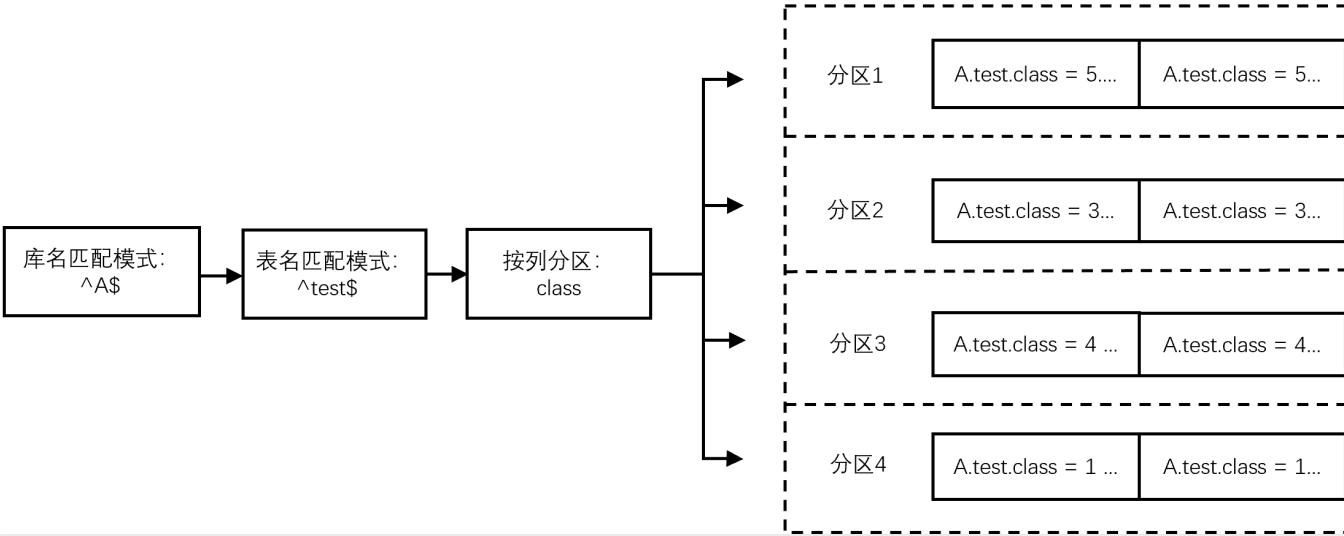
自定义分区策略：按表名+主键分区（分区=4）



按列分区

库名匹配模式填入 `^A$`，表名匹配模式填入 `^test$`，自定义分区列填入 `class`，选择**按列名分区**后，`A` 库中表 `test` 列名为 `class` 的数据将被散列到不同分区中，最终同一列的数据更新都投递到了一个分区中。`test` 除外其他未匹配到的库表数据会根据 Topic 分区策略（默认分区策略）中设置进行投递。

自定义分区策略：按列名分区（分区=4）



MySQL/MariaDB/Percona/TDSQL-C MySQL

数据库同步至 Kafka

支持能力

最近更新时间：2024-07-08 18:59:27

功能大类	功能子项或说明	支持能力
同步对象	-	普通对象表、索引、视图、存储过程和函数
同步类型	-	结构同步 全量同步（全量结束后自动接续增量） 增量同步
断点续传	-	全量导出部分支持（结构导出、无主键表导出阶段不支持续传）； 全量导入、增量支持。
任务通道关键操作	重试	支持
	暂停、恢复	支持
	修改同步配置（动态增减表对象）	支持
	创建类似任务	支持
	限速	支持
同步方向	多对一（多实例合一/多表合一）	支持
	一对多	支持
结构同步	库表过滤	支持
	列过滤	不支持
	跳过同步表及订正	不支持
全量同步	Where 条件过滤	支持
增量同步	DML 同步 （INSERT/UPDATE/DELETE）	支持
	DML 过滤	支持

	DDL 同步	CREATE DATABASE、DROP DATABASE、ALTER DATABASE CREATE TABLE、ALTER TABLE、DROP TABLE、TRUNCATE TABLE、RENAME TABLE CREATE VIEW、DROP VIEW CREATE INDEX、DROP INDEX 暂不支持 CREATE TABLE 表名 AS SELECT 语句。
	DDL 过滤	支持
	Where 条件过滤	支持
	指定时间点同步	支持
	Online DDL 同步	支持
数据订阅（同步到 Kafka 链路特有）	数据格式	Avro、JSON、Canal Protobuf、Canal JSON
	指定多 Topic 投递	支持
	指定单 Topic 多 partition 投递	支持
	指定单 Topic + 自定义分区策略	支持
	过滤事务 BEGIN/COMMIT	支持
	过滤 CHECKPOINT 消息	支持

数据同步操作指导

最近更新时间：2024-07-08 18:59:27

DTS 支持将源数据库的全量、增量数据同步到 Ckafka 中，方便用户快速获取业务变更数据并进行分析应用。本文为您介绍使用数据传输服务 DTS 将 MySQL/MariaDB/Percona/TDSQL-C MySQL 数据同步至腾讯云数据库 Ckafka 的过程。

源数据库支持的部署类型如下：

自建 MySQL、第三方云厂商 MySQL、腾讯云数据库 MySQL。

自建 MariaDB、腾讯云数据库 MariaDB。

自建 Percona。

腾讯云数据库 TDSQL-C MySQL。

注意事项

同一个任务可以关联多个 Topic，但同一个 Topic 不能同时被多个同步任务使用，否则会导致数据错乱，消费到的数据是多个不同任务的数据，也可能导致在任务重试等场景中数据处理异常，任务报错。

默认采用有锁同步方式，全量数据导出阶段会对源库加全局锁（FTWRL），锁表过程中会短暂（秒级）阻塞写入操作。

如果用户在同步过程中确定会对某张表使用 rename 操作，投递到目标 Ckafka 中，分区规则会按照新的表名匹配 topic 和 partition。

在全量同步阶段，每同步10万条数据，会在目标端 Kafka 插入一条 checkpoint 消息，标识当前的数据同步位点。

在同步任务前，源库中已经存在的全量数据，因为没有数据写入的准确时间，在消费端 Demo 中 happenAt 字段会显示为1970-01-01 08:00:00，该时间字段无需关注。消费增量数据时，时间字段可以正确显示。

DTS 在全量数据同步阶段，将源库数据导出并导入到目标端 Ckafka，均使用 utf8mb4 字符集，避免因字符集不识别导致乱码问题。

建议目标 Topic 为空，同时在同步任务中，不要在目标 Topic 中进行写入，否则可能会导致消息错乱，任务报错。

DTS 同步到目标端 Kafka 的单条消息存在性能上限，建议用户源数据库中的单行数据不要超过8MB，否则在同步过程中可能会报错。

数据同步时，DTS 会使用执行同步任务的账号在源库中写入系统库 __tencentdb__，用于记录事务标记 ID 等元信息，需要确保源库对 __tencentdb__ 的读写权限。

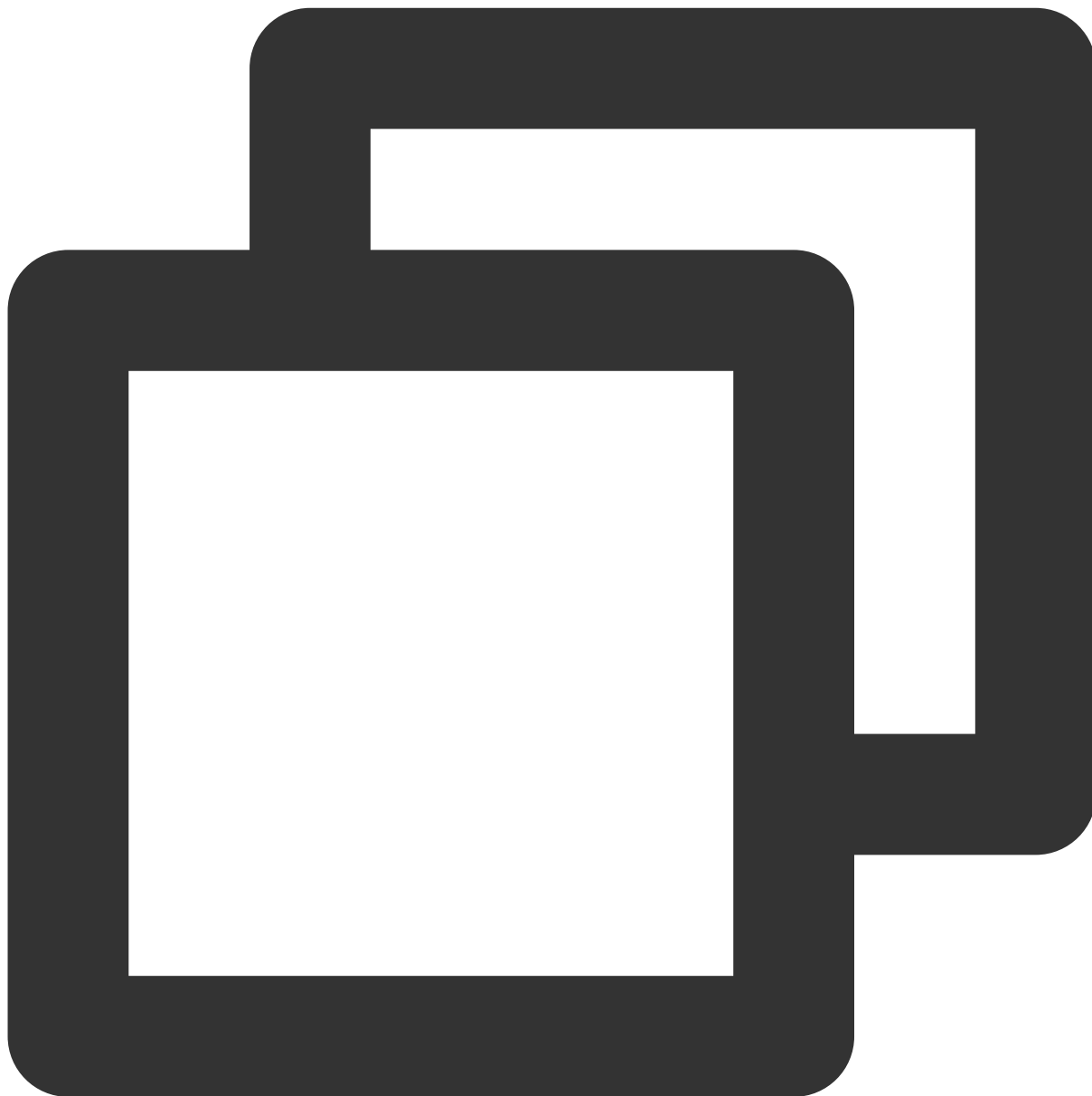
为保证后续数据对比问题可定位，同步任务结束后不会删除源库中的 __tencentdb__。

__tencentdb__ 系统库占用空间非常小，约为源库存储空间的千分之一到万分之一（例如源库为50GB，则 __tencentdb__ 系统库约为5MB-50MB），并且采用单线程，等待连接机制，所以对源库的性能几乎无影响，也不会抢占资源。

前提条件

源数据库和目标数据库符合同步功能和版本要求，请参考 [数据同步支持的数据库](#) 进行核对。

同步任务账号需要具备源数据库的权限如下：



```
GRANT RELOAD,LOCK TABLES,REPLICATION CLIENT,REPLICATION SLAVE,SHOW VIEW,PROCESS,SEL  
GRANT ALL PRIVILEGES ON `__tencentdb__`.* TO '账号'@'%';  
FLUSH PRIVILEGES;
```

在目标 Ckafka 中修改消息保留时间和消息大小上限。

消息保存时间建议设置为3天，超过保存时间的数据会被清除，请用户在设置的时间内及时消费；消息大小上限，即 Ckafka 可以接收的单个消息内存的大小，设置时需要大于源库表中单行数据的最大值，以确保源库的数据都可以正常投递到 CKafka 中。

应用限制

只支持同步基础表、视图、存储过程和函数。

源端如果是非 GTID 实例，DTS 不支持源端 HA 切换，一旦源端 MySQL 发生切换可能会导致 DTS 增量同步中断。

只支持同步 InnoDB、MyISAM、TokuDB 三种数据库引擎，如果存在这三种以外的数据引擎表则默认跳过不进行同步。

相互关联的数据对象需要一起同步，否则会导致同步失败。如主外键关联表等。

不支持同时包含 DML 和 DDL 语句在一个事务的场景，遇到该情况任务会报错。

不支持 Geometry 相关的数据类型，遇到该类型数据任务报错。

不支持 ALTER VIEW 语句，遇到该语句任务跳过不同步。

为了保证数据一致性，同步到 Kafka 默认采用有锁同步，但源库为阿里云 MySQL 5.5/5.6、AWS MySQL 5.7.18 及之前的版本时无法加上锁，可能会造成同步到目标 Ckafka 的部分数据重复。用户如果对重复数据比较关注，请在消费数据时设置去重逻辑。

同步任务过程中，如果发生任务重启，可能会导致同步到目标端 Ckafka 的数据出现重复。

DTS 是按最小数据单元进行同步的（全量阶段单个表对象的一块数据即为最小数据单元，增量阶段每标记一个位点就是一个数据单元），如果重启时，刚好一个数据单元同步已完成，则不会导致数据重复；如果重启时，一个数据单元还正在同步中，那么再次启动后需要重新同步这个数据单元，这样就会导致数据重复。

修改同步对象、任务错误时手动点操作列的重试按钮、变配、暂停任务再恢复，隔离后再解隔离，这些场景都会发生任务重启。

用户如果对重复数据比较关注，请在消费数据时设置去重逻辑。

目标端 Kafka 的版本，当前仅支持在腾讯云上可购买的[专业版](#)和[标准版](#)，并且为 1.1.1 版本及以后；历史已经下线的物理机标准版不支持作为目标端。

操作限制

同步过程中请勿进行如下操作，否则会导致同步任务失败。

请勿修改、删除源数据库和目标 Ckafka 中用户信息（包括用户名、密码和权限）和端口号。

请勿在源库写入 Binlog 格式为 STATEMENT 的数据。

请勿在源库上执行清除 Binlog 的操作。

请勿在目标端已选择同步的 Topic 中写入数据。

支持同步的 SQL 操作

操作类型	SQL 操作语句
DML	INSERT、UPDATE、DELETE
DDL	CREATE DATABASE、DROP DATABASE、ALTER DATABASE、CREATE TABLE、ALTER TABLE、DROP TABLE、TRUNCATE TABLE、CREATE VIEW、DROP VIEW、CREATE INDEX、DROP INDEX

说明：

暂不支持 CREATE TABLE 表名 AS SELECT 语句。

环境要求

类型	环境要求
源数据库要求	源库和目标库网络能够连通。 实例参数要求： 源库 <code>server_id</code> 参数需要手动设置，且值不能设置为0。 源库表的 <code>row_format</code> 不能设置为 <code>FIXED</code>。 源库变量 <code>connect_timeout</code> 设置数值必须大于10。 Binlog 参数要求： 源端 <code>log_bin</code> 变量必须设置为 <code>ON</code>。 源端 <code>binlog_format</code> 变量必须设置为 <code>ROW</code>。 源端 <code>binlog_row_image</code> 变量必须设置为 <code>FULL</code>。 MySQL 5.6 及以上版本 <code>gtid_mode</code> 变量不为 <code>ON</code> 时会报警告，建议打开 <code>gtid_mode</code>。 不允许设置 <code>do_db</code>，<code>ignore_db</code>。 源实例为从库时，<code>log_slave_updates</code> 变量必须设置为 <code>ON</code>。 建议源库 Binlog 日志至少保留3天及以上，否则可能会因任务暂停/中断时间大于 Binlog 日志保留时间，造成任务无法续传，进而导致任务失败。 外键依赖： 外键依赖建议设置为 <code>NO ACTION</code>，<code>RESTRICT</code> 两种类型，其他类型可能会影响数据一致性结果。 部分库表同步时，有外键依赖的表必须齐全。 环境变量 <code>innodb_stats_on_metadata</code> 必须设置为 <code>OFF</code>。
目标 Ckafka 要求	目标 Ckafka 可以正常连通。 目标 Ckafka 中消息大小设置的上限需要大于源库表中单行数据的最大值。

操作步骤

因为 MySQL/MariaDB/Percona/TDSQL-C MySQL 同步至腾讯云数据库 Ckafka，各场景的同步要求和操作步骤基本一致，本章节仅以 MySQL 到 Ckafka 的数据同步为例进行介绍，其他场景请参考相关内容。

1. 登录 [数据同步购买页](#)，选择相应配置，单击**立即购买**。

参数	描述
计费模式	支持包年包月和按量计费。
源实例类型	选择 MySQL，购买后不可修改。
源实例地域	选择源实例所在地域，购买后不可修改。如果源数据库为自建数据库，选择离自建地域最近的一个地域即可。
目标实例类型	选择 Kafka，购买后不可修改。
目标实例地域	选择目的实例所在地域，购买后不可修改。
规格	请根据业务诉求选择规格，规格越高，性能越好。详情请参考 计费概述 。

2. 购买完成后，返回 [数据同步列表](#)，可看到刚创建的数据同步任务，单击**操作**列的**配置**，进入配置同步任务页面。

3. 在配置同步任务页面，配置源端实例、账号密码，配置目标端实例、账号和密码，测试连通性后，单击**下一步**。

设置项	参数	描述
任务设置	任务名称	DTS 会自动生成一个任务名称，用户可以根据实际情况进行设置。
	运行模式	支持立即执行和定时执行两种模式。
源实例设置	源实例类型	购买时所选择的源实例类型，不可修改。
	源实例地域	购买时选择的源实例所在地域，不可修改。
	接入类型	请根据您的场景选择，不同接入类型的准备工作请参考 准备工作概述 。 公网：源数据库可以通过公网 IP 访问。 云主机自建：源数据库部署在 腾讯云服务器 CVM 上。 专线接入：源数据库可以通过 专线接入 方式与腾讯云私有网络打通。 VPN 接入：源数据库可以通过 VPN 连接 方式与腾讯云私有网络打通。 云数据库：源数据库属于腾讯云数据库实例。 云联网：源数据库可以通过 云联网 与腾讯云私有网络打通。
	公网	主机地址：源数据库 IP 地址或域名。 端口：源数据库使用的端口。

	云主机自建	云主机实例：云服务器 CVM 的实例 ID。 端口：源数据库使用的端口。
	专线接入	私有网络专线网关/：专线接入时只支持私有网络专线网关，请确认网关关联网络类型。 私有网络：选择私有网络专线网关和 VPN 网关关联的私有网络和子网。 主机地址：源数据库 IP 地址。 端口：源数据库使用的端口。
	VPN 接入	VPN 网关：VPN 网关，请选择通过 VPN 网关接入的 VPN 网关实例。 私有网络：选择私有网络专线网关和 VPN 网关关联的私有网络和子网。 主机地址：源数据库 IP 地址。 端口：源数据库使用的端口。
	云数据库	云数据库实例：源数据库的实例 ID。
	云联网	主机地址：源数据库的主机 IP 地址。 端口：源数据库使用的端口。 私有网络云联网：云联网实例名称。 接入 VPC：接入 VPC 指的是云联网中接入订阅链路的 VPC。请在云联网关联的所有 VPC 中，选择除了源数据库所属 VPC 外的其他 VPC。 例如，广州地域数据库作为源数据库，则接入 VPC 选择其他地域，如成都 VPC 或者上海 VPC。 子网：已选择 VPC 网络的子网名称。 接入 VPC 地域：购买任务时选择的源数据库地域与接入 VPC 地域需要保持一致，如果不一致，DTS 会将购买任务中选择的源数据库地域，改为接入 VPC 地域。
	账号/密码	账号/密码：源数据库的账号、密码。
目标实例设置	目标实例类型	购买时选择的目标实例类型，不可修改。
	目标实例地域	购买时选择的目标实例地域，不可修改。
	接入类型	根据您的场景选择，本场景选择“Ckafka 实例”。
	实例 ID	选择目标实例 ID。

4. 在设置同步选项和同步对象页面，将对数据初始化选项、数据同步到 Kafka 策略、数据同步选项、同步对象选项进行设置，在设置完成后单击**保存并下一步**。

自定义 Topic 名称

集中投递到单 Topic

设置项	参数	描述

数据初始化选项	初始化类型	结构初始化：同步任务执行时会先将源实例中表结构初始化到目标实例中。 全量数据初始化：同步任务执行时会先将源实例中数据初始化到目标实例中。 仅选择全量数据初始化的场景，用户需要提前在目标库创建好表结构。 默认两者都勾选上，可根据实际情况取消。
	投递到 Kafka 数据格式	Avro 为二进制格式，消费效率更高，JSON 为轻量级的文本格式，更加简单易用。
数据同步到 Kafka 策略	同步 Topic 策略	自定义 Topic 名称：自行设置投递的 Topic 名称，设置后目标端 Kafka 会自动创建该 Topic。实现效果，将同步数据随机投递到该 Topic 下的不同分区中。 如果目标端 Kafka 创建 Topic 没建成功，任务会报错。 集中投递到单 Topic：选择一个目标端已有的 Topic，然后按照多种分区策略投递，支持单分区或者按表名或者按表名+主键划分分区，投递到不同分区中。
	自定义 Topic 名称规则	用户添加多条规则，自定义 Topic 匹配规则会从上到下逐条匹配。设置的库名和表名规则都匹配上，才会投递到该条规则对应的 Topic 中。如果没有匹配到库表匹配规则，则会投递到最后一规则匹配的 Topic 中；如果匹配到多条库表匹配规则，则会投递到所有匹配规则的 Topic 中。 说明： 匹配规则对库表名大小写敏感。 如果源库设置 lower_case_table_names=0，则库表匹配规则中，库表名需要与源库中的名称大小写严格保持一致。 如果源库设置 lower_case_table_names=1，则库表名统一转换为小写，库表匹配规则中库表名统一输入小写。 举例一：数据库实例 X 中，Users 库里有 Student、Teacher 表，需要将 Users 库里的数据都投递到 Topic_A 中。配置规则如下： Topic 名称填入 Topic_A，库名匹配模式填入 ^Users\$，表名匹配模式填入 .* Topic 名称填入 Topic_default，库名匹配模式（不符合匹配规则的剩余库），表名匹配模式（不符合匹配规则的剩余表）。 示例二：数据库实例 X 中，Users 库里有 Student、Teacher 表，需要将 Student 表数据投递到 Topic_A 中，Teacher 表数据投递到 Topic_default 中。配置规则如下： Topic 名称填入 Topic_A，库名匹配模式填入 ^Users\$，表名匹配模式填入 ^Student\$。 Topic 名称填入 Topic_default，库名匹配模式（不符合匹配规则的剩余库），表名匹配模式（不符合匹配规则的剩余表）。
	集中投递到单 Topic 规则	选择指定的 Topic 后，系统会按照如下指定策略进行分区。 全部投递至 Partition0：将源库的同步数据全部投递到第一个分区。 按表名分区：将源库的同步数据按照表名进行分区，设置后相同表名的数据会写入同一个分区中。 按表名+主键分区：将源库的同步数据按照表名+主键分区，适用于热点数据，设置后热点数据的表，通过表名+主键的方式将数据分散到不同分区中，提升并发消费效率。

	存储 DDL 的 Topic	可选，如果用户需要将源库的 DDL 操作单独投递到指定 Topic 中，可以在此处选择设置。 设置后默认投递到已选 Topic 的 Partition0；如果没设置会根据上面选择的 Topic 规则进行投递。
--	----------------	---

数据同步选项

设置项	参数	描述
数据同步选项	同步操作类型	支持操作：Insert、Update、Delete、DDL。勾选“DDL 自定义”，可以根据需要选择不同的 DDL 同步策略。详情请参考 设置 SQL 过滤策略 。
同步对象选项	源实例库表对象	选择待同步的对象，支持基础库表、视图、存储过程和函数。高级对象的同步是一次性动作，仅支持同步在任务启动前源库中已有的高级对象，在任务启动后，新增的高级对象不会同步到目标库中。更多详情，请参考 同步高级对象 。
	已选对象	在左侧选择同步对象后，单击  ，即可在右侧看到已勾选的对象。
	是否同步 Online DDL 临时表	如果使用 gh-ost、pt-osc 工具对源库中的表执行 Online DDL 操作，DTS 支持将 Online DDL 变更产生的临时表迁移到目标库。 勾选 gh-ost，DTS 会将 gh-ost 工具产生的临时表名（`_表名_ghc`、`_表名_gho`、`_表名_del`）迁移到目标库。 勾选 pt-osc，DTS 会将 pt-osc 工具产生的临时表名（`_表名_new`、`_表名_old`）迁移到目标库。 更多详情请参考 同步 Online DDL 临时表 。

5. 在校验任务页面，完成校验并全部校验项通过后，单击**启动任务**。

如果校验任务不通过，可以参考 [校验不通过处理方法](#) 修复问题后重新发起校验任务。

失败：表示校验项检查未通过，任务阻断，需要修复问题后重新执行校验任务。

警告：表示校验项检查不完全符合要求，可以继续任务，但对业务有一定的影响，用户需要根据提示自行评估是忽略警告项还是修复问题再继续。

6. 返回数据同步任务列表，任务开始进入**运行中**状态。

说明

选择**操作**列的**更多**>**结束**可关闭同步任务，请您确保数据同步完成后再关闭任务。

7. （可选）您可以单击任务名，进入任务详情页，查看任务初始化状态和监控数据。

常见问题

增量同步中，源库同步对象长时间无写入，监控显示延迟大。

请用户先确认源库中，选择同步的对象是否长期无数据写入（其他未同步库表对象有数据写入），如果确认没有写入，则忽略此问题，这种场景监控指标失真，请参见如下解释；如果有写入，则 [提交工单](#) 处理。

MySQL 系列数据库同步到 Kafka 链路中，DTS 需要先拉取源库的 Binlog，解析后识别出勾选对象的相关数据，同步到目标 Kafka。源库选择同步的库表对象长期无数据写入，其他未同步库表对象有数据写入，这种场景，DTS 拉取 Binlog 并解析后发现，不需要同步到目标端，目标端的数据一直没有更新，所以就显示了延迟。用户可以尝试在源库选择同步的对象中更新数据来刷新指标，进而得到真实的指标数据。

后续操作

数据同步到目标端 Kafka 后，可进行数据消费，我们为您提供了消费示例 Demo，方便您快速测试消费数据的流程，了解数据格式解析的方法。具体请参考 [数据消费操作指导](#)。

数据消费操作指导

最近更新时间：2024-07-15 15:27:18

操作场景

数据同步到 Kafka 后，您可以通过0.11版本及以上的 [Kafka 客户端](#) 进行消费订阅数据，本文为您提供了 Java、Go、Python 语言的客户端消费 Demo 示例，方便您快速测试消费数据的流程，了解数据格式解析的方法。

注意事项

- 1. Demo 并不包含消费数据的用法演示，仅对数据做了打印处理，您需要在此基础上自行编写数据处理逻辑，您也可以使用其他语言的 Kafka 客户端消费并解析数据。**
2. 写入到 Kafka 中的数据支持兼容开源工具 Canal 的格式，采用 Protobuf 或者 JSON 的序列化协议，您可以在配置同步任务的过程中选择数据格式 Canal ProtoBuf 或者 Canal JSON。
3. 目标 Ckafka 中消息大小设置的上限需要大于源库表中单行数据的最大值，以便数据可以正常同步到目标端。
4. 为了保证数据可重入，DTS 引入 CHECKPOINT 机制，在事务之间插入 CHECKPOINT 消息用来标识数据同步的位点，任务中断重启后可实现断点续传。同时，消费端遇到 CHECKPOINT 消息会做一次 Kafka 消费位点提交，以便及时更新消费位点。
如果用户对 CHECKPOINT 比较在意，不希望在消费的消息中出现 CHECKPOINT 消息，可以在配置任务过程中，设置过滤 CHECKPOINT 消息。过滤后 DTS 仍可以实现任务重启后的断点续传，仅在写入目标端 Kafka 时不插入 CHECKPOINT 消息。
5. 源库中 Timestamp 类型的字段同步到目标端会转换为0时区（如"2021-05-17 07:22:42 +00:00"），在消费数据时请根据您的场景评估，是否要对该字段进行格式转换。

消费 Demo 下载

在配置同步任务中，您可以选择不同的数据格式，具体如下。配置任务时选择了哪种格式，这里的 Demo 下载也需要下载对应格式的 Demo。

Avro：二进制格式，消费效率更高。

JSON：为轻量级的文本格式，更加简单易用。

Canal ProtBuf：适配 Canal 的 ProtoBuf 数据格式。

Canal JSON：适配 Canal 的 JSON 数据格式。

如下提供的 Demo 示例，均已包含对应的 Avro/JSON/ProtoBuf 协议文件，无需另外下载。Demo 中的逻辑讲解及关键参数说明，请参考 [Demo 说明](#)。

--	--	--	--

Demo 语言	Avro 格式	JSON 格式	Canal ProtoBuf/Canal JSON 格式
Go	地址	地址	地址
Java	地址	地址	地址
Python	地址	地址	地址

Java Demo 使用说明

编译环境：**Maven** 或者 **Gradle** 包管理工具，**JDK8**。用户可自行选择打包工具，如下以 **Maven** 为例进行介绍。
操作步骤：

1. 下载 **Java Demo**，然后解压该文件。
2. 进入解压后的目录，为方便使用，目录下分别放置了 **Maven** 模型文件、**pom.xml** 文件，用户根据需要选用。
使用 **Maven** 进行打包：**mvn clean package**。
3. 运行 **Demo**。

使用 **Maven** 打包后，进入目标文件夹 **target**。

数据格式选择 **Avro** 格式，运行如下代码：

```
java -jar consumerDemo.jar --brokers xxx --topic xxx --group xxx --trans2sql
```

数据格式选择 **JSON** 格式，运行如下代码：

```
java -jar consumerDemo.jar --brokers xxx --topic xxx --group xxx --trans2sql
```

数据格式选择 **Canal ProtoBuf** 格式，运行如下代码：

```
java -jar consumerDemo.jar --brokers xxx --topic xxx --group xxx
```

数据格式选择 **Canal JSON** 格式，运行如下代码：

```
java -jar consumerDemo.jar --brokers xxx --topic xxx --group xxx --flatMessage
```

各参数详细说明如下：

参数	说明
broker	CKafka 的访问地址。
topic	同步任务中设置的 topic 名称。
group	消费组名称，用户可提前在 Ckafka 中创建消费组，也可以在此处自定义输入。
trans2sql	表示是否转换为 SQL 语句，携带该参数表示转换为 SQL 语句，不携带则不转换。 选择 Avro、JSON 格式时需要配置。
flatMessage	仅选择 Canal JSON 数据格式时需要携带。

4. 观察消费情况。

```
BEGIN
-->[partition: 0, offset: 87008, partitionSeq: 87009] [mysql-bin.000004:24574], happenedAt: 2021-03-01T
17:49:14
INSERT INTO `kafka-subscribe`.`table1` VALUES (_binary'subscribe-kafka', 61)
-->[partition: 0, offset: 87008, partitionSeq: 87009] [mysql-bin.000004:24605], happenedAt: 2021-03-01T
17:49:14
COMMIT
```

Golang Demo 使用说明

编译环境：Golang 1.12 及以上版本，配置好 Go Module 环境。

操作步骤：

1. 下载 Golang Demo，然后解压该文件。
2. 进入解压后的目录，运行 `go build -o consumer ./main/main.go`，生成可执行文件 `consumer`。
3. 运行命令。

数据格式选择 Avro 格式，运行如下代码：

```
./consumer --brokers=xxx --topic=xxx --group=xxx --trans2sql=true
```

数据格式选择 JSON 格式，运行如下代码：

```
./consumer --brokers=xxx --topic=xxx --group=xxx --trans2sql=true
```

数据格式选择 Canal ProtoBuf 格式，运行如下代码：

```
./consumer --brokers=xxx --topic=xxx --group=xxx
```

数据格式选择 Canal JSON 格式，运行如下代码：

```
./consumer --brokers=xxx --topic=xxx --group=xxx --flatMessage=true
```

各参数详细说明如下：

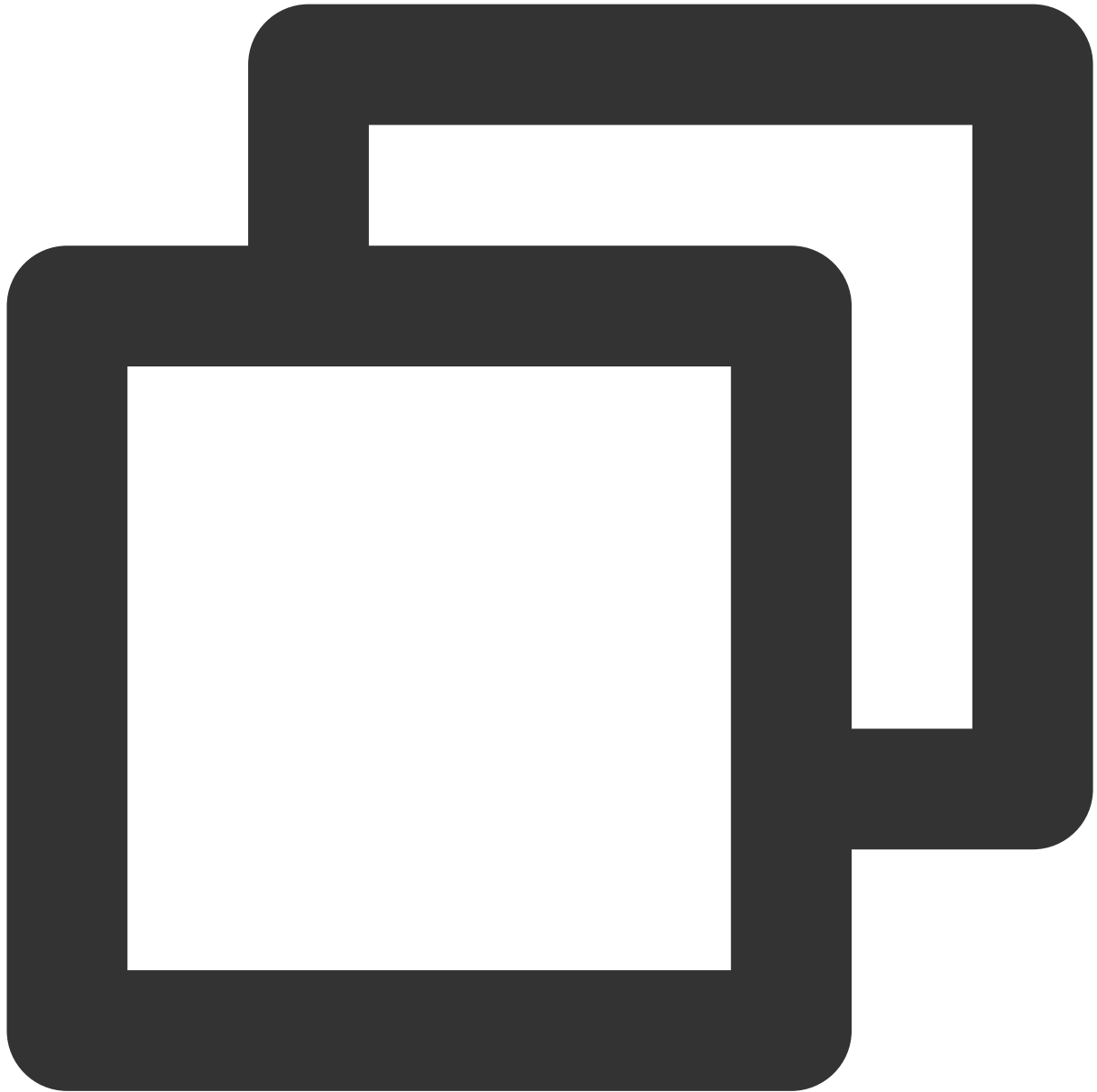
参数	说明
broker	CKafka 的访问地址。
topic	同步任务中设置的 topic 名称。
group	消费组名称，用户可提前在 Ckafka 中创建消费组，也可以在此处自定义输入。
trans2sql	表示是否转换为 SQL 语句。选择 Avro、JSON 格式时需要配置。
flatMessage	仅选择 Canal JSON 数据格式时需要携带。

4. 观察消费情况。

Python3 Demo 使用说明

编译运行环境：安装 Python3, pip3（用于依赖包安装）。

使用 pip3 安装依赖包：



```
pip install flag
pip install kafka-python
pip install avro
```

操作步骤：

1. 下载 Python3 Demo，然后解压该文件。
2. 运行 `main.py`。

数据格式选择 Avro 格式，运行如下代码：

```
python main.py --brokers=xxx --topic=xxx --group=xxx --trans2sql=1
```

数据格式选择 JSON 格式，运行如下代码：

```
python main.py --brokers=xxx --topic=xxx --group=xxx --trans2sql=1
```

数据格式选择 Canal ProtoBuf 格式，运行如下代码：

```
python main.py --brokers=xxx --topic=xxx --group=xxx
```

数据格式选择 Canal JSON 格式，运行如下代码：

```
python main.py --brokers=xxx --topic=xxx --group=xxx --flatMessage=1
```

各参数详细说明如下：

参数	说明
broker	CKafka 的访问地址。
topic	同步任务中设置的 topic 名称。
group	消费组名称，用户可提前在 Ckafka 中创建消费组，也可以在此处自定义输入。
trans2sql	表示是否转换为 SQL 语句，设置为1表示转换为 SQL 语句，设置为0则不转换。选择 Avro、JSON 格式时需要配置。
flatMessage	仅选择 Canal JSON 数据格式时需要携带。

3. 观察消费情况。

Demo 说明

Avro Demo 说明

最近更新时间：2024-07-08 18:59:27

Demo 关键逻辑讲解

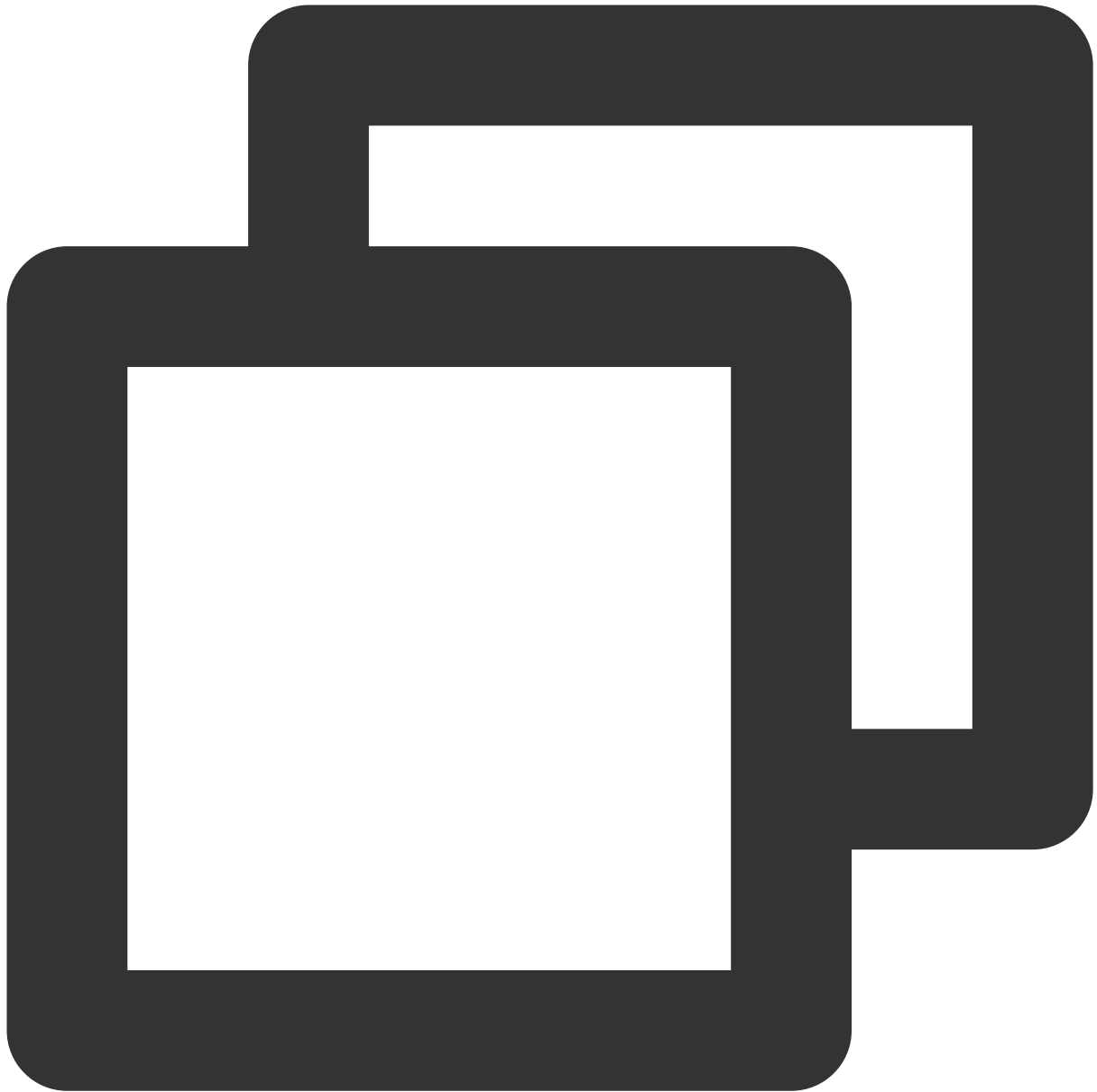
Demo 中的文件说明如下，以 Java Demo 为例进行介绍。

`consumerDemo-avro-java\src\main\resources\avro-tools-1.8.2.jar`：用来生成 Avro 协议相关代码的工具。

`consumerDemo-avro-java\src\main\java\com\tencent\subscribe\avro`：Avro 工具生成代码的目录。

`consumerDemo-avro-java\src\main\resources\Record.avsc`：协议定义文件。

Record.avsc 中我们定义了14个结构（Avro 中叫做 schema），其中主要的数据结构为 Record，用于表示 binlog 中的一条数据，Record 的结构如下，其他数据结构可以在 Record.avsc 中查看：



```
{
  "namespace": "com.tencent.subscribe.avro",    //Record.avsc 中的最后1个 schema, "
  "type": "record",
  "name": "Record",    // "name" 显示为 "Record", 表示从 kafka 中消费的数据格式
  "fields": [
    {
      "name": "id",    //id 表示全局递增 ID, 更多 record 取值解释如下表
      "type": "long",
      "doc": "unique id of this record in the whole stream"
    },
    {
```

```
    "name": "version",    //version 表示协议版本
    "type": "int",
    "doc": "protocol version"
  },
  {
    "name": "messageType",    //消息类型
    "aliases": [
      "operation"
    ],
    "type": {
      "namespace": "com.tencent.subscribe.avro",
      "name": "MessageType",
      "type": "enum",
      "symbols": [
        "INSERT",
        "UPDATE",
        "DELETE",
        "DDL",
        "BEGIN",
        "COMMIT",
        "HEARTBEAT",
        "CHECKPOINT",
        "ROLLBACK",
        "STATEMENT",
        "INIT_DDL",
        "INIT_INSERT"
      ]
    }
  },
  {
    .....
  },
}
```

Record 中的字段类型解释如下：

Record 中的字段名称	说明
id	全局递增 ID。
version	协议版本，当前版本为1。
messageType	消息类型，枚举 值："INSERT", "UPDATE", "DELETE", "DDL", "BEGIN", "COMMIT", "HEARTBEAT" 其中，"INIT_DDL"表示全量数据的 DDL，"DDL"表示增量数据的 DDL，"INIT_INSERT"表
fileName	当前 record 所在的 binlog 文件名。

position	当前 record 的在 binlog 中结束的偏移量，格式为 End_log_pos@binlog 文件编号。例如，其值为"2196@4"。
safePosition	当前事务在 binlog 中开始的偏移量，格式同上。
timestamp	写 binlog 的时间，unix 时间戳，秒级。
gtid	当前的 gtid，如：c7c98333-6006-11ed-bfc9-b8cef6e1a231:9。
transactionId	事务 ID，只有 commit 事件才会生成事务 ID。
serverId	源库 serverId，查看源库 server_id 参考 SHOW VARIABLES LIKE 'server_id'。
threadId	提交当前事务的会话 ID，参考 SHOW processlist;。
sourceType	源库的数据库类型，当前版本只有 MySQL。
sourceVersion	源库版本，查看源库版本参考 <code>select version();</code> 。
schemaName	库名。
tableName	表名。
objectName	格式为：库名.表名。
columns	表中各列的定义。
oldColumns	DML 执行前该行的数据，如果是 insert 消息，该数组为 null。数组中元素共有12种类型：TimestampWithTimeZone, BinaryGeometry, TextGeometry, BinaryObject, TextObject
newColumns	DML 执行后该行的数据，如果是 delete 消息，该数组为 null。数组中元素共有12种类型：TimestampWithTimeZone, BinaryGeometry, TextGeometry, BinaryObject, TextObject
sql	DDL 的 SQL 语句。
executionTime	DDL 执行时长，单位为秒。
heartbeatTimestamp	心跳消息的时间戳，秒级。只有 heartbeat 消息才有该字段。
syncedGtid	DTS 已解析 GTID 集合，格式形如：c7c98333-6006-11ed-bfc9-b8cef6e1a231:1-13。
fakeGtid	是否为构造的假 GTID，如未开启 gtid_mode，则 DTS 会构造一个 GTID。
pkNames	如果源库的表设有主键，则 DML 消息中会携带该参数，否则不会携带。
readerTimestamp	DTS 处理这条数据的时间，unix 时间戳，单位为毫秒数。
tags	QueryEvent 中的 status_vars，详细参考 QueryEvent 。

total	如果消息分片，记录分片总数。当前版本 (version=1) 无意义，预留扩展。
index	如果消息分片，记录当前分片的索引。当前版本 (version=1) 无意义，预留扩展。

Record 中描述列属性的字段为 "Field"，包含如下四个属性：

name：列名。

dataTypeName：是 binlog 中记录的数据类型。取值详见 [MySQL](#)。

isKey：是否主键。

originalType：DDL 中定义的类型。

数据库字段映射关系

如下为数据库（如 MySQL）字段类型和 Avro 协议中定义的数据类型之间的映射关系。

MySQL 类型	对应 Avro 中的类型
MYSQL_TYPE_NULL	EmptyObject
MYSQL_TYPE_INT8	Integer
MYSQL_TYPE_INT16	Integer
MYSQL_TYPE_INT24	Integer
MYSQL_TYPE_INT32	Integer
MYSQL_TYPE_INT64	Integer
MYSQL_TYPE_BIT	Integer
MYSQL_TYPE_YEAR	DateTime
MYSQL_TYPE_FLOAT	Float
MYSQL_TYPE_DOUBLE	Float
MYSQL_TYPE_VARCHAR	Character
MYSQL_TYPE_STRING	Character，如果原类型为 binary，则对应 BinaryObject
MYSQL_TYPE_VAR_STRING	Character，如果原类型为 varbinary，则对应 BinaryObject
MYSQL_TYPE_TIMESTAMP	Timestamp
MYSQL_TYPE_DATE	DateTime

MYSQL_TYPE_TIME	DateTime
MYSQL_TYPE_DATETIME	DateTime
MYSQL_TYPE_TIMESTAMP_NEW	Timestamp
MYSQL_TYPE_DATE_NEW	DateTime
MYSQL_TYPE_TIME_NEW	DateTime
MYSQL_TYPE_DATETIME_NEW	DateTime
MYSQL_TYPE_ENUM	TextObject
MYSQL_TYPE_SET	TextObject
MYSQL_TYPE_DECIMAL	Decimal
MYSQL_TYPE_DECIMAL_NEW	Decimal
MYSQL_TYPE_JSON	TextObject
MYSQL_TYPE_BLOB	BinaryObject
MYSQL_TYPE_TINY_BLOB	BinaryObject
MYSQL_TYPE_MEDIUM_BLOB	BinaryObject
MYSQL_TYPE_LONG_BLOB	BinaryObject
MYSQL_TYPE_GEOMETRY	BinaryObject

JSON Demo 说明

最近更新时间：2024-07-08 18:59:27

Demo 中我们采用 JSON 进行序列化，各语言 Demo 中均附带有 Record 定义文件。

Java Demo 中的定义文件路径为：`consumerDemo-json-java\src\main\java\json\FlatRecord.java`。

Record 中字段类型

Record 中的字段名称	说明
id	全局递增 ID。
version	协议版本，当前版本为1。
messageType	消息类型，枚举 值："INSERT", "UPDATE", "DELETE", "DDL", "BEGIN", "COMMIT", "HEARTBEAT"
fileName	当前 record 所在的 binlog 文件名。
position	当前 record 的在 binlog 中结束的偏移量，格式为 End_log_pos@binlog 文件编号。例如，件 mysql-bin.000004 中，结束偏移量为2196，则其值为"2196@4"。
safePosition	当前事务在 binlog 中开始的偏移量，格式同上。
timestamp	写入 binlog 的时间，unix 时间戳，秒级。
gtid	当前的 gtid，如：c7c98333-6006-11ed-bfc9-b8cef6e1a231:9。
transactionId	事务 ID，只有 commit 事件才会生成事务 ID。
serverId	源库 serverId，查看源库 server_id 参考 SHOW VARIABLES LIKE 'server_id'。
threadId	提交当前事务的会话 ID，参考 SHOW processlist;。
sourceType	源库的数据库类型，当前版本只有 MySQL。
sourceVersion	源库版本，查看源库版本参考 <code>select version();</code> 。
schemaName	库名。
tableName	表名。

objectName	格式为：库名.表名。
columns	表中各列的定义。
oldColumns	DML 执行前该行的数据，如果是 insert 消息，该数组为 null。
newColumns	DML 执行后该行的数据，如果是 delete 消息，该数组为 null。
sql	DDL 的 SQL 语句。
executionTime	DDL 执行时长，单位为秒。
heartbeatTimestamp	心跳消息的时间戳，秒级。只有 heartbeat 消息才有该字段。
syncedGtid	DTS 已解析 GTID 集合，格式形如：c7c98333-6006-11ed-bfc9-b8cef6e1a231:1-13。
fakeGtid	是否为构造的假 GTID，如未开启 gtid_mode，则 DTS 会构造一个 GTID。
pkNames	如果源库的表设有主键，则 DML 消息中会携带该参数，否则不会携带。
readerTimestamp	DTS 处理这条数据的时间，unix时间戳，单位为毫秒数。
tags	QueryEvent 中的 status_vars，详细参考 QueryEvent 。
total	如果消息分片，记录分片总数。当前版本 (version=1) 无意义，预留扩展。
index	如果消息分片，记录当前分片的索引。当前版本 (version=1) 无意义，预留扩展。

Record 中 MySQL 列属性

- name：列名。
- dataTypeName：是 binlog 中记录的数据类型。取值详见 [MySQL](#)。
- isKey：是否主键。
- originalType：DDL 中定义的类型。

MySQL 数据类型转换逻辑

- 在 JSON 协议中，将 MySQL 类型全部转换成了字符串。
- 对于 varchar 等字符串类型，全部转成了 UTF8 编码。
- 对于数值类型，全部转成了与值相同的字符串，如 "3.0"。
- 对于时间类型，格式为：yyyy-dd-mm hh:MM:ss.milli。
- 对于时间戳类型，输出为毫秒数。

对于 binary、blob 等二进制类型，输出为与16进制值相同的字符串，如 "0xfff"。

Canal Demo 说明（Canal ProtoBuf/Canal JSON）

最近更新时间：2024-07-08 18:59:27

功能描述

DTS 写入到 Kafka 的同步数据支持兼容开源工具 Canal 格式，采用 ProtoBuf 或者 JSON 的序列化协议，您可以在配置 DTS 同步任务的过程中选择数据格式 Canal Protobuf 或者 Canal JSON，然后使用消费 Demo 进行业务适配，即可得到消费数据。



如果您想了解更多 Canal 信息，请参考 [Canal 详情](#)。

方案对比

功能	DTS 同步到 Kafka 方案	Canal 同步方案
数据类型	全量+增量	仅增量
数据格式	Canal ProtoBuf、Canal JSON	ProtoBuf、JSON
成本	购买云资源，初始配置完成后，后续基本不需要维护	需要客户自己部署和维护

Canal JSON 格式兼容性说明

用户可使用之前 Canal 方案中的消费程序直接消费。对 DTS 方案中 Canal JSON 格式进行数据消费时的字段名称，与 Canal 方案中 JSON 格式的字段名称保持一致，仅需要注意以下差异。

1. 源库中二进制相关类型的字段（包括 binary、varbinary、blob、tinyblob、mediumblob、longblob、geometry）同步到目标端后会转换为 HexString，请用户在消费数据时注意。
2. 源库中 Timestamp 类型的字段同步到目标端会转换为0时区（如"2021-05-17 07:22:42 +00:00"），用户在解析和转换的时候需要考虑时区信息。
3. Canal 方案的 JSON 格式中定义了 sqlType 字段，该字段在 JDBC（Java 数据库连接）中使用，表示 SQL 数据类型。因为 Canal 底层采用 Java 语言，而 DTS 底层采用 Golang 语言实现，所以 DTS 提供的 Canal JSON 格式中这个字段留空处理。

Canal ProtoBuf 格式兼容性说明

对 Canal ProtoBuf 格式的数据消费，需要使用 DTS 提供的协议文件，因为 DTS 协议文件中增加了如全量同步等功能逻辑，该协议文件已包含在消费 Demo 中。所以用户需要使用 [DTS 提供的消费 Demo](#)，并在这个 Demo 基础上适配自身业务逻辑，才能得到消费数据。

消费 DTS 提供的 Canal ProtoBuf 格式数据时的字段名称，与 Canal 方案提供的 ProtoBuf 格式一致，仅需要注意以下差异。

1. 源库中二进制相关类型的字段（包括 binary、varbinary、blob、tinyblob、mediumblob、longblob、geometry）同步到目标端后会转换为 HexString，请用户在消费数据时注意。
2. 源库中 Timestamp 类型的字段同步到目标端会转换为0时区（如"2021-05-17 07:22:42 +00:00"），用户在解析和转换的时候需要考虑时区信息。

TDSQL MySQL 同步至 Kafka

支持能力

最近更新时间：2024-07-08 18:59:27

功能大类	功能子项或说明	支持能力
同步对象	-	普通对象表、索引、视图
同步类型	-	结构同步 全量同步（全量结束后自动接续增量） 增量同步
断点续传	-	全量导出部分支持（结构导出、无主键表导出阶段不支持续传）； 全量导入、增量支持。
任务通道关键操作	重试	支持
	暂停、恢复	支持
	修改同步配置（动态增减表对象）	不支持
	创建类似任务	支持
	限速	支持
同步方向	多对一（多实例合一/多表合一）	支持
	一对多	支持
结构同步	库表过滤	支持
	列过滤	不支持
	跳过同步表及订正	不支持
全量同步	Where 条件过滤	支持
增量同步	DML 同步 (INSERT/UPDATE/DELETE)	支持
	DML 过滤	支持
	DDL 同步	CREATE DATABASE、DROP DATABASE、 ALTER DATABASE

		CREATE TABLE、ALTER TABLE、DROP TABLE、TRUNCATE TABLE、RENAME TABLE CREATE VIEW、DROP VIEW CREATE INDEX、DROP INDEX 暂不支持 CREATE TABLE 表名 AS SELECT 语句。
	DDL 过滤	支持
	Where 条件过滤	支持
	指定时间点同步	支持
	Online DDL 同步	支持
同步到 Kafka 链路特有	数据格式	Avro、JSON
	指定多 Topic 投递	支持
	指定单 Topic 多 partition 投递	支持
	指定单 Topic + 自定义分区策略	支持
	过滤事务 BEGIN/COMMIT	不支持
	过滤 CHECKPOINT 消息	不支持

使用说明

最近更新时间：2024-07-08 18:59:27

类别	说明
同步对象	1. 只支持同步基础表、视图，不支持同步函数、触发器、存储过程等对象。 2. 相互关联的数据对象需要同时同步，否则会导致同步失败。 3. 增量同步阶段，源库的表名如含有“TDSQLagent”、“tdsql_sub”字符可能会被过滤或者引起同步异常，因为这些表名与 TDSQL 系统的临时表名相同，TDSQLagent 为扩容时的临时表，tdsql_sub 表为 hash-lish 和 hash-range 的子表，因此建议源端待同步的表名不要设置为这些类型。
源库影响	1. 数据同步时，DTS 会使用执行同步任务的账号在源库中写入系统库`__tencentdb__`，用于记录事务标记ID等元信息，需要确保源库对`__tencentdb__`的读写权限。 为保证后续数据对比问题可定位，同步任务结束后不会删除源库中的`__tencentdb__`。`__tencentdb__`系统库占用空间非常小，约为源库存储空间的千分之一到万分之一（例如源库为50GB，则`__tencentdb__`系统库约为5MB - 50MB），并且采用单线程，等待连接机制，所以对源库的性能几乎无影响，也不会抢占资源。 2. 默认采用无锁同步方式，全量数据导出阶段不会对源库加全局锁（FTWRL），仅对无主键的表加表锁。
目标端 Kafka 要求	1. 目标端 Kafka 的版本，当前仅支持在腾讯云上可购买的 专业版和标准版，并且为1.1.1版本及以后；历史已经下线的物理机标准版不支持作为目标端。 2. 需要在目标 kafka 中修改消息保留时间和消息大小上限。 消息保存时间建议设置为3天，超过保存时间的数据会被清除，请用户在设置的时间内及时消费；消息大小上限，即 Kafka 可以接收的单个消息内存的大小，设置时需要大于源库表中单行数据的最大值，以确保源库的数据都可以正常投递到 Kafka 中。 3. 建议目标 Topic 为空，同时在同步任务过程中，不要在目标端选择同步的 Topic 中进行数据写入，否则可能会导致消息错乱，任务报错。
数据同步说明	1. 只支持同步 InnoDB、MyISAM、TokuDB 三种数据库引擎，如果存在这三种以外的数据引擎表则默认跳过不进行同步。 2. 同一个任务可以关联多个 Topic，但同一个 Topic 不能同时被多个同步任务使用，否则会导致数据错乱，消费到的数据是多个不同任务的数据，也可能导致在任务重试等场景中数据处理异常，任务报错。 3. 在全量同步阶段，每同步10万条数据，会在目标端 Kafka 插入一条 checkpoint 消息，标识当前的数据同步位点。 4. DTS 同步到目标端 Kafka 的单条消息存在性能上限，建议用户源数据库中的单行数据不要超过8MB，否则在同步过程中可能会报错。 5. 如果用户在同步过程中确定会对某张表使用 rename 操作，投递到目标 kafka 中，分区规则会按照新的表名匹配 topic 和 partition。 6. 增量同步过程中，若源库产生了类型为 STATEMENT 格式的 Binlog 语句，则会导致同步失败。 7. 不支持 ALTER VIEW 语句，遇到该语句任务跳过不同步。

数据消费说明	在同步任务前，源库中已经存在的全量数据，因为没有数据写入的准确时间，在消费端 Demo 中 happenAt 字段会显示为1970-01-01 08:00:00，该时间字段无需关注。消费增量数据时，时间字段可以正确显示。
重启影响	同步任务过程中，如果发生任务重启（如源库发生 HA 切换会重启，后台检查任务异常会重启）， 可能会导致同步到目标端 Kafka 的数据出现重复。 DTS 是按最小数据单元进行同步的（全量阶段单个表对象的一块数据即为最小数据单元，增量阶段每标记一个位点就是一个数据单元），如果重启时，刚好一个数据单元同步已完成，则不会导致数据重复；如果重启时，一个数据单元还正在同步中，那么再次启动后需要重新同步这个数据单元，这样就会导致数据重复。 用户如果对重复数据比较关注，请在消费数据时设置去重逻辑。
操作限制	同步过程中请勿进行如下操作，否则会导致同步任务失败。 1. 在全量导出阶段，请勿在源库上执行库或表结构变更的 DDL 操作。 2. 请勿修改、删除源数据库和目标端中的用户信息（包括用户名、密码和权限）和端口号。 3. 请勿在源库上执行清除 Binlog 的操作。
数据类型	1. TDSQL MySQL 作为源库并使用 proxy 方式连接的场景中，浮点数如果源库使用 float 数据类型，会导致全量阶段的数据精度不准确。若需保留数据精度，推荐使用 double 数据类型。全量阶段的精度问题具体影响如下： 全量阶段和增量阶段数据同步的精度不一致。 使用 float 作为键值，全量阶段和增量阶段操作的主键数据不一致。 2. DTS 在全量数据同步阶段，将源库数据导出并导入到目标端 Kafka，均使用 utf8mb4 字符集，避免因字符集不识别导致乱码问题。 3. 不支持 Geometry 相关的数据类型，遇到该类型数据任务报错。
事务	不支持同时包含 DML 和 DDL 语句在一个事务的场景，遇到该情况任务会报错。
HA 切换和扩容	1. 源库如果是非 GTID 数据库，DTS 不支持源端 HA 切换，一旦源端 TDSQL MySQL 发生切换可能会导致 DTS 增量同步中断。 2. DTS 采用 SET 直连 TDSQL MySQL 的场景，不支持 TDSQL MySQL 扩容。采用 Proxy 连接 TDSQL MySQL 的场景，支持 TDSQL MySQL 逻辑扩容，水平扩容时 DTS 可能会报错。
分区表同步	1. 一级/二级分区表的语法需要符合规范，一级 Hash 分区表仅支持通过 shardkey 方式创建。TDSQL MySQL 创建分区表的关键语法如下，详细语法请参考 TDSQL MySQL 创建表语法示例 。 一级 Hash 分区：shardkey 一级 Range 分区：TDSQL_DISTRIBUTED BY RANGE 一级 List 分区：TDSQL_DISTRIBUTED BY LIST 一级 Hash 分区 + 二级 Range/List 分区：shardkey + PARTITION BY RANGE/LIST 一级 Range 分区 + 二级 Range/List 分区：TDSQL_DISTRIBUTED BY RANGE + PARTITION BY RANGE/LIST 一级 List 分区 + 二级 Range/List 分区：TDSQL_DISTRIBUTED BY LIST + PARTITION BY RANGE/LIST 2. 增量同步阶段，不支持进行并发的 DDL 操作，需要等前一个 DDL 生效后再执行下一个，否则可能会因为 DDL 乱序导致报错。同时，也不支持快速地对同名表进行 create、drop、create 操

	作，否则可能会引起表类型错误。
指定启动 时间点	1. 增量同步任务，从设置的时间点开始同步数据，设置的时间点范围为：之前的14天-当前时间。 2. 如果全量同步和增量同步分开执行，这里请注意，进行全量同步时，DTS 已经同步的全量数据时间点1与设置的时间点2之间，不能存在 DDL 变更数据，否则任务会报错。 3. 从设置的指定时间点1，到增量任务启动点2之间（任务步骤从“寻找指定位点”转化为“同步增量”才算启动），源库操作注意事项如下。 建议源库不要进行主从切换、增加分片、重做备机操作，否则可能会影响 DTS 获取源库的 GTID 位点。 源端可以操作与同步对象无关的 DDL，但需要保证上一条 DDL 同步到目标端后再执行新的 DDL。密集地执行 DDL 可能会导致任务报错。 源端操作与同步对象有关的 DDL时，同步任务会报错。 4. 因为指定位点同步是根据 Binlog 中 context 的时间（SET TIMESTAMP=XXXX）来判断其 GTID，为保证同步数据的正确性，建议用户不要修改该 context。 5. 请确认数据库设置的时区与当前控制台时区（即浏览器时区）一致，或者换算为数据库设置时区所对应的时间，否则可能会导致指定位点同步结果不合预期。 6. 请确认 TDSQL MySQL 各集群节点的时间保持一致，否则可能会导致指定位点同步结果不合预期。 7. 如果设置的时间点在 XA 事务持续时间的范围内，则该XA事务不会被同步。对于这种情况，建议时间点可以往前设置一些（在 XA START 的时间之前）。

数据同步操作指导

最近更新时间：2024-07-08 18:59:27

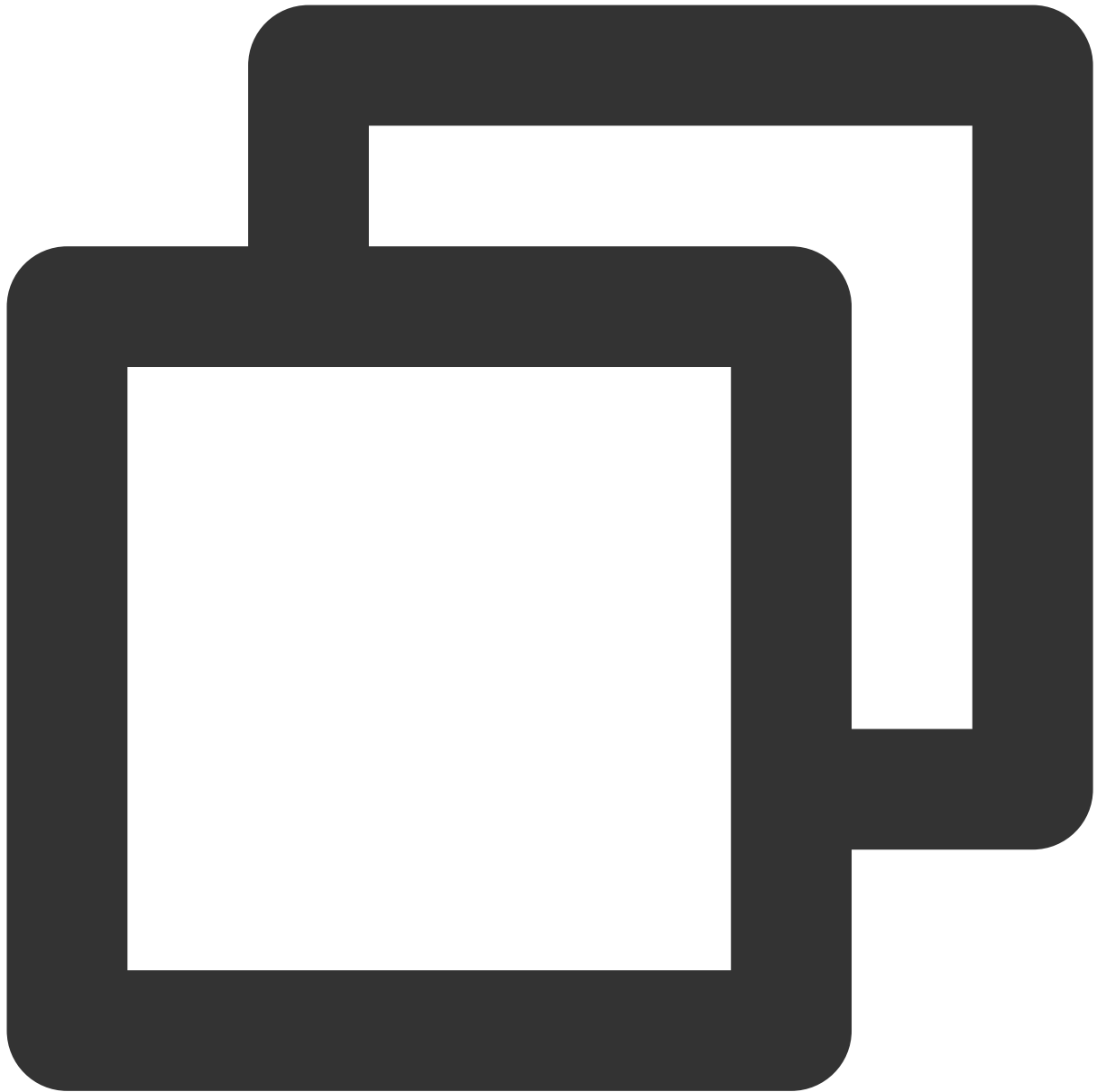
操作场景

DTS 支持将源数据库的全量、增量数据同步至 Ckafka 中，方便用户快速获取业务变更数据并进行分析应用。本文为您介绍使用数据传输服务 DTS 将 TDSQL MySQL 数据同步至腾讯云数据库 Ckafka 的过程。

源数据库类型当前仅支持腾讯云数据库 TDSQL MySQL。

准备工作

1. 准备好源端数据库和目标端 Kafka，版本符合 DTS 要求，请参考 [数据同步支持的数据库](#) 进行核对。
2. 在源数据库上，对同步任务账号进行授权，参考如下。



```
GRANT RELOAD,LOCK TABLES,REPLICATION CLIENT,REPLICATION SLAVE,SHOW VIEW,PROCESS,SEL
GRANT ALL PRIVILEGES ON `__tencentdb__`.* TO '账号'@'%';
FLUSH PRIVILEGES;
```

3. 在目标 **Ckafka** 中修改消息保留时间和消息大小上限。

消息保存时间建议设置为3天，超过保存时间的数据会被清除，请用户在设置的时间内及时消费；消息大小上限，即 **Ckafka** 可以接收的单个消息内存的大小，设置时需要大于源库表中单行数据的最大值，以确保源库的数据都可以正常投递到 **CKafka** 中。

操作步骤

1. 登录 数据同步购买页，选择相应配置，单击**立即购买**。

参数	描述
计费模式	支持包年包月和按量计费。
源实例类型	选择 TDSQL MySQL，购买后不可修改。
源实例地域	选择源实例所在地域，购买后不可修改。
目标实例类型	选择 Kafka，购买后不可修改。
目标实例地域	选择目的实例所在地域，购买后不可修改。
规格	请根据业务诉求选择规格，规格越高，性能越好。详情请参考 计费概述 。

2. 购买完成后，返回 数据同步列表，可看到刚创建的数据同步任务，单击**操作**列的**配置**，进入配置同步任务页面。

3. 在配置同步任务页面，配置源端和目标端帐号密码，**测试连通性**后，单击**下一步**。

设置项	参数	描述
任务设置	任务名称	DTS 会自动生成一个任务名称，用户可以根据实际情况进行设置。
	运行模式	支持立即执行和定时执行两种模式。
源实例设置	源实例类型	购买时所选择的源实例类型，不可修改。
	源实例地域	购买时选择的源实例所在地域，不可修改。
	接入类型	请根据您的场景选择，当前仅支持源端为腾讯云数据库。
	账号/密码	账号/密码：源数据库的账号、密码。
目标实例设置	目标实例类型	购买时选择的目标实例类型，不可修改。
	目标实例地域	购买时选择的目标实例地域，不可修改。
	接入类型	根据您的场景选择，本场景选择“Ckafka 实例”。
	实例 ID	选择目标实例 ID。

4. 在设置同步选项和同步对象页面，完成同步选项和同步对象的设置后，单击**保存并下一步**。

4.1 数据初始化选项

参数	描述

初始化类型	结构初始化：同步任务执行时会先将源实例中表结构初始化到目标实例中。 全量数据初始化：同步任务执行时会先将源实例中数据初始化到目标实例中。仅选择全量数据初始化的场景，用户需要提前提在目标库创建好表结构。 全量同步，两个都勾选；增量同步，两个都不勾选，然后在下面的同步操作类型中勾选 DDL 和 DML。
投递到 Kafka 数据格式	Avro 为二进制格式，消费效率更高，JSON 为轻量级的文本格式，更加简单易用。
指定启动位点	初始化类型中都不勾选，即选择增量同步时，可设置数据同步的起始位点。 1. 如果全量同步和增量同步分开执行，这里请注意，进行全量同步时，DTS 已经同步的全量数据位点1，与增量同步设置的启动位点2之间，不能存在 DDL 变更数据，否则任务会报错。 2. 请确认数据库设置的时区与当前控制台时区（即浏览器时区）是否一致，或者换算为数据库设置时区所对应的时间，否则可能会导致指定位点同步结果不合预期。 3. 请确认源端 TDSQL MySQL 各集群节点的时间保持一致，否则可能会导致指定位点同步结果不合预期。 4. 从发起指定位点同步，到增量任务开始前（即任务步骤从“寻找指定位点”转化为“同步增量”前），源库不能操作 DDL，否则任务会报错。

4.2 数据同步到 Kafka 策略

参数	描述
同步 Topic 策略	自定义 Topic 名称：自行设置投递的 Topic 名称，设置后目标端 Kafka 会自动创建该 Topic。实现效果，将同步数据随机投递到该 Topic 下的不同分区中。如果目标端 Kafka 创建 Topic 失败，任务会报错。 集中投递到单 Topic：选择一个目标端已有的 Topic，然后按照多种分区策略投递，支持单分区或者按表名或者按表名+主键划分分区，投递到不同分区中。
自定义 Topic 名称规则	用户添加多条规则，自定义 Topic 匹配规则会从上到下逐条匹配。设置的库名和表名规则都匹配上，才会投递到该条规则对应的 Topic 中。如果没有匹配到库表匹配规则，则会投递到最后一条规则的 Topic 中；如果匹配到多条库表匹配规则，则会投递到所有匹配规则的 Topic 中。 设置示例请参考 数据投递到 Kafka 策略设置示例。 说明： 匹配规则对库表名大小写敏感。 如果源库设置 lower_case_table_names=0，则库表匹配规则中库表名需要与源库中的名称大小写严格保持一致。 如果源库设置 lower_case_table_names=1，则库表名统一转换为小写，库表匹配规则中库表名统一输入小写。
集中投递到单 Topic 规则	选择指定的 Topic 后，系统会按照如下指定策略进行分区。 全部投递至 Partition0：将源库的同步数据全部投递到第一个分区。 按表名分区：将源库的同步数据按照表名进行分区，设置后相同表名的数据会写入同一个分区中。

	按表名+主键分区：将源库的同步数据按照表名+主键分区，适用于热点数据，设置后热点数据的表，通过表名+主键的方式将数据分散到不同分区中，提升并发消费效率。
存储 DDL 的 Topic	可选，如果用户需要将源库的 DDL 操作单独投递到指定 Topic 中，可以在此处选择设置。设置后默认投递到已选 Topic 的 Partition0；如果没设置会根据上面选择的 Topic 规则进行投递。

4.3 数据同步选项

设置项	参数	描述
数据同步选项	同步操作类型	支持操作：Insert、Update、Delete、DDL。
同步对象选项	源实例库表对象	仅支持同步的基础的库、表对象。

5. 在校验任务页面，完成校验并全部校验项通过后，单击**启动任务**。

如果校验任务不通过，可以参考 [校验不通过处理方法](#) 修复问题后重新发起校验任务。

失败：表示校验项检查未通过，任务阻断，需要修复问题后重新执行校验任务。

警告：表示检验项检查不完全符合要求，可以继续任务，但对业务有一定的影响，用户需要根据提示自行评估是忽略警告项还是修复问题再继续。

6. 返回数据同步任务列表，任务开始进入**运行中**状态。

说明：

选择**操作列**的**更多 > 结束**可关闭同步任务，请您确保数据同步完成后再关闭任务。

7. （可选）您可以单击任务名，进入任务详情页，查看任务初始化状态和监控数据。

后续操作

数据同步到目标端 **Kafka** 后，可进行数据消费，我们为您提供了消费示例 **Demo**，方便您快速测试消费数据的流程，了解数据格式解析的方法。具体请参考 [数据消费操作指导](#)。

数据消费操作指导

最近更新时间：2024-07-08 18:59:27

操作场景

数据同步到 Kafka 后，您可以通过0.11版本及以上的 [Kafka 客户端](#) 进行消费订阅数据，本文为您提供了 Java、Go、Python 语言的客户端消费 Demo 示例，方便您快速测试消费数据的流程，了解数据格式解析的方法。

注意事项

- Demo 并不包含消费数据的用法演示，仅对数据做了打印处理，您需要在此基础上自行编写数据处理逻辑，您也可以使用其他语言的 Kafka 客户端消费并解析数据。**
- 目标 Ckafka 中消息大小设置的上限需要大于源库表中单行数据的最大值，以便数据可以正常同步到目标端。
- 在同步指定库/表对象（非源实例全部），并且采用 Kafka 单分区的场景中，DTS 解析增量数据后，仅将同步对象的数据写入 Kafka Topic 中，其他非同步对象的数据会转成空事务写入 Kafka Topic，所以在消费数据时会出现空事务。空事务的 Begin/Commit 消息中保留了事务的 GTID 信息，可以保证 GTID 的连续性和完整性。同时，在 MySQL/TDSQL-C MySQL 的消费 Demo 中，多个空事务也做了压缩处理以减少消息数量。
- 为了保证数据可重入，DTS 同步到 Kafka 链路引入 Checkpoint 机制。消息写入 Kafka Topic 时，一般每10秒会插入一个 Checkpoint，用来标识数据同步的位点，在任务中断后再重启识别断点位置，实现断点续传。另外，消费端遇到 Checkpoint 消息会做一次 Kafka 消费位点提交，以便及时更新消费位点。
- 数据格式选择 JSON 时，如果您使用过或者熟悉开源订阅工具 Canal，可以选择将这里消费出来的 JSON 格式数据转换成 Canal 工具兼容的数据格式，再进行后续处理，我们的 Demo 中已经提供了相关支持，在启动 Demo 的参数中添加参数 trans2canal 即可实现。目前该功能仅限 Java 语言支持。

消费 Demo 下载

在配置同步任务中，您可以选择不同的数据格式，Avro 和 JSON。Avro 采用二进制格式，消费效率更高，JSON 采用轻量级的文本格式，更加简单易用。选择的数据格式不同，参考的 Demo 示例也不同。

如下提供的 Demo 示例，均已包含对应的 Avro/JSON 协议文件，无需另外下载。

Demo 中的逻辑讲解及关键参数说明，请参考 [Demo 说明](#)。

Demo 语言	Avro 格式	JSON 格式
Go	地址	地址
Java	地址	地址

Python	地址	地址
--------	----	----

Java Demo 使用说明

编译环境：Maven 或者 Gradle 包管理工具，JDK8。用户可自行选择打包工具，如下以 Maven 为例进行介绍。

操作步骤：

1. 下载 Java Demo，然后解压该文件。
2. 进入解压后的目录，为方便使用，目录下分别放置了 Maven 模型文件、pom.xml 文件，用户根据需要选用。使用 Maven 进行打包：mvn clean package。

3. 运行 Demo。

使用 Maven 打包后，进入目标文件夹 target，运行如下代码：`java -jar consumerDemo-avro-1.0-SNAPSHOT.jar --brokers xxx --topic xxx --group xxx --trans2sql`

`broker` 为 CKafka 的访问地址，`topic` 为同步任务中设置的 topic 名称，如果是多 topic 需要分别消费。这两个可通过[数据同步 > 操作 > 查看](#)获取。

`group` 为消费组名称，用户可提前在 Ckafka 中创建消费组，也可以在此处自定义输入。

`trans2sql` 表示是否转换为 SQL 语句，java 代码中，携带该参数表示转换为 SQL 语句，不携带则不转换。

`trans2canal` 表示是否转换为 Canal 格式打印出来，携带该参数表示转换为 Canal 格式，不携带则不转换。

说明：

携带 `trans2sql` 时，将使用 `javax.xml.bind.DatatypeConverter.printHexBinary()` 将 byte 值转成16进制，请使用 JDK1.8 版本及以上避免不兼容。如果不需要转 SQL，可以注释此处代码。

4. 观察消费情况。

```
BEGIN
-->[partition: 0, offset: 87008, partitionSeq: 87009] [mysql-bin.000004:24574], happenedAt: 2021-03-01T
17:49:14
INSERT INTO `kafka-subscribe`.`table1` VALUES (_binary'subscribe-kafka', 61)
-->[partition: 0, offset: 87008, partitionSeq: 87009] [mysql-bin.000004:24605], happenedAt: 2021-03-01T
17:49:14
COMMIT
```

Golang Demo 使用说明

编译环境：Golang 1.12 及以上版本，配置好 Go Module 环境。

操作步骤：

1. 下载 Golang Demo，然后解压该文件。
2. 进入解压后的目录，运行 `go build -o subscribe ./main/main.go`，生成可执行文件 `subscribe`。
3. 运行 `./subscribe --brokers=xxx --topic=xxx --group=xxx --trans2sql=true`。

`broker` 为 CKafka 的访问地址，`topic` 为同步任务中设置的 topic 名称，如果是多 topic 需要分别消费。这两个可通过[数据同步 > 操作 > 查看](#)获取。

`group` 为消费组名称，用户可提前在 `Ckafka` 中创建消费组，也可以在此处自定义输入。

`trans2sql` 表示是否转换为 SQL 语句。

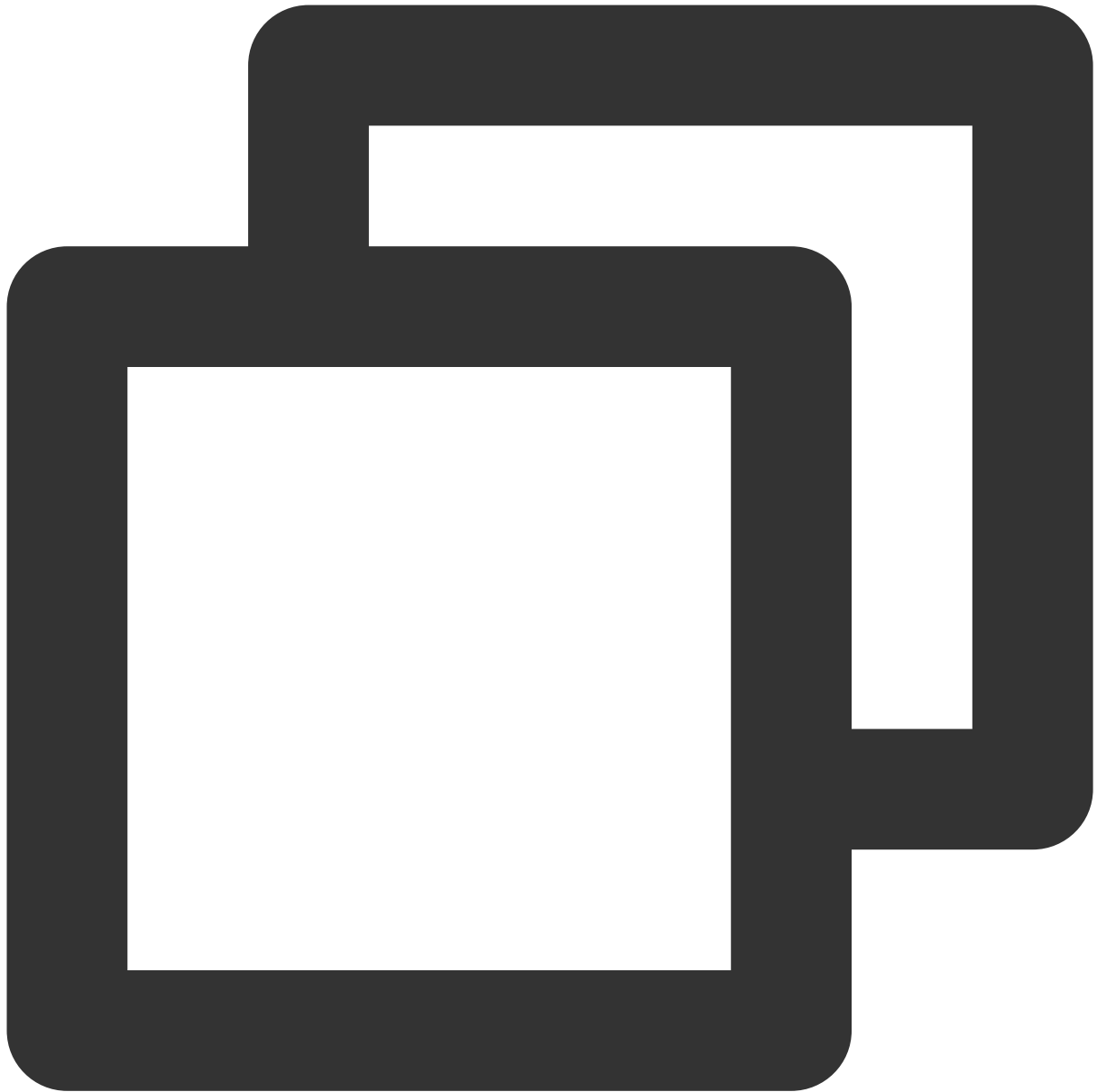
4. 观察消费情况。

```
BEGIN
-->[partition: 0, offset: 86991, partitionSeq: 86992] [mysql-bin.000004:24272], happenedAt: 2021-03-01
17:47:49 +0800 CST
INSERT INTO `kafka-subscribe`.`table1` VALUES (_binary'subscribe-kafka', 60)
-->[partition: 0, offset: 86991, partitionSeq: 86992] [mysql-bin.000004:24303], happenedAt: 2021-03-01
17:47:49 +0800 CST
COMMIT
```

Python3 Demo 使用说明

编译运行环境：安装 Python3, pip3（用于依赖包安装）。

使用 pip3 安装依赖包：



```
pip install flag
pip install kafka-python
pip install avro
```

操作步骤：

1. 下载 Python3 Demo ，然后解压该文件。
 2. 运行 `python main.py --brokers=xxx --topic=xxx --group=xxx --trans2sql=1` 。
- `broker` 为 CKafka 的访问地址， `topic` 为同步任务中设置的 topic 名称，如果是多 topic 需要分别消费。这两个可通过[数据同步 > 操作 > 查看](#)获取。

`group` 为消费组名称，用户可提前在 **Ckafka** 中创建消费组，也可以在此处自定义输入。

`trans2sql` 表示是否转换为 SQL 语句。

3. 观察消费情况。

```
BEGIN
-->[partition: 0, offset: 89083, partitionSeq: 89084] [mysql-bin.000004:24876], happenedAt: 2021-03-01
20:43:31
INSERT INTO `kafka-subscribe`.`table1` VALUES (_binary'subscribe-kafka', 62)
-->[partition: 0, offset: 89083, partitionSeq: 89084] [mysql-bin.000004:24907], happenedAt: 2021-03-01
20:43:31
COMMIT
```

任务管理

任务状态说明

最近更新时间：2024-07-08 19:11:11

计费任务状态说明

说明

包年包月购买后即扣费。按量计费任务配置期间不计费，等任务正式启动才开始计费，且仅在任务正常运行期间计费，任务中断期间不计费。

仅使用包年包月计费模式的任务实例显示**续费**的操作。

仅使用按量计费模式的任务实例显示**按量计费转包年包月**的操作。

状态	说明	支持的操作类型
未初始化	完成购买资源操作，未配置同步任务。	查看、配置、变配、按量计费转包年包月、续费、销毁/退货
已初始化	已完成同步任务配置。	查看、配置、校验、变配、按量计费转包年包月、续费、销毁/退货
校验中	同步任务校验进行中。	查看、变配、按量计费转包年包月、续费、销毁/退货
校验通过	同步任务校验完成。	查看、配置、启动、校验、变配、按量计费转包年包月、续费、销毁/退货
校验不通过	同步任务校验不通过。	查看、配置、校验、变配、按量计费转包年包月、续费、销毁/退货
准备运行	校验通过，准备启动同步任务的过渡状态。	查看、变配、按量计费转包年包月、续费、销毁/退货
运行中	同步任务运行中。	查看、暂停、结束、修改同步配置、变配、按量计费转包年包月、续费、销毁/退货
任务出错	同步过程中，出现异常导致任务中断，可通过界面操作 重试 ，继续同步任务的阶段。	查看、重试、结束、变配、按量计费转包年包月、续费、销毁/退货
重试中	单击 重试 后的过渡状态。	查看、结束、变配、按量计费转包年包月、续费、销毁/退货

暂停中	同步任务运行中，用户需要调整业务，临时暂停任务。用户操作 暂停 后的过渡状态为暂停中	查看、销毁/退货
已暂停	任务已暂停，暂停时间如果超过7天任务会失败。	查看、恢复、结束、销毁/退货
恢复中	暂停后，再次启动任务，恢复的过渡状态，恢复后任务状态将转为“运行中”。	查看、销毁/退货
结束中	用户不再需要同步任务，手动单击 结束 ，任务进入结束中，结束操作不可恢复。	查看、销毁/退货
已结束	同步任务已结束。	查看、销毁/退货、重置
重置中	对于已结束的任务，重置后将清除旧的任务配置数据，任务变为“未初始化”的状态，用户可重配置新的同步任务。	查看、销毁/退货
任务失败	同步任务失败。	查看、销毁/退货
变配中	用户单击变配后的中间状态。	查看、续费
隔离中	用户不再需要任务实例，手动单击 销毁/退货 。实例到期或者欠费1天后，自动进入隔离状态。	查看、续费、立即下线
已隔离	用户单击 销毁/退费 ，或者到期/欠费后实例进入回收站，进入隔离状态。 因实例到期/欠费导致进入隔离状态，补缴费用后可恢复正常。	查看、解隔离（恢复）、续费、立即下线
恢复中	采用按量计费的任务，用户因为误操作 销毁 ，导致实例进入隔离状态，重新单击 解隔离（恢复） ，任务将恢复到销毁前的状态。	查看
下线中	隔离7天后，实例自动下线，数据将被清除不可恢复。	查看
冲正中	按量计费实例因到期或欠费进入回收站，补缴费后，实例冲正中，后续可恢复正常。	查看、续费
续费中	对包年包月实例进行续费操作后的状态。	查看

未计费任务状态说明

状态	说明	支持的操作类型
未初始化	完成购买资源操作，未配置同步任务。	查看、配置、删除
已初始化	已完成同步任务配置。	查看、配置、校验、删除

校验中	同步任务校验进行中。	查看、删除
校验通过	同步任务校验完成。	查看、配置、启动、校验、删除
校验不通过	同步任务校验不通过。	查看、配置、校验、删除
准备运行	校验通过，准备启动同步任务的过渡状态。	查看、删除
运行中	同步任务运行中。	查看、暂停、结束、修改同步配置、删除
任务错误	同步过程中，出现异常导致任务中断，通过界面操作 重试 ，继续同步任务的阶段。	查看、重试、结束、删除
重试中	单击 重试 后的状态。	查看、结束、删除
结束中	用户手动单击 结束 ，任务进行到结束中。	查看、删除
已结束	结束任务完成。	查看、删除
任务失败	同步任务失败。	查看、删除
删除中	对于不需要的任务，用户手动单击 删除 任务，任务删除后不可恢复，资源会被清除。	查看

查看任务

最近更新时间：2024-07-08 19:11:11

操作场景

通过任务详情信息，用户可查看同步任务、同步配置、结构初始化、数据初始化、监控数据等信息。

前提条件

已成功创建数据同步任务，并已登录 [数据同步控制台](#)。

操作步骤

方式1：在 [数据同步](#) 列表，选择指定的同步任务，单击任务名称。

方式2：在 [数据同步](#) 列表，选择指定的同步任务，在**操作**列单击**查看**。

同步任务页面

展示任务基本信息、源库信息以及目标库信息。

Sync Task

Sync Configuration

Structure Initialization

Data Initialization

Monitor

Basic Info

Task ID	replicat[REDACTED]0g
Task Name	replicate[REDACTED]
Running Mode	Immediate execution
Billing Mode	Pay as you go
Sync Task Specification	Standard Edition
Task Type	MySQL -> MySQL
Creation Time	2021-10-11 20:51:46

Source Database Info

Instance ID	cdb-[REDACTED] Instance Monitoring
Source Database Type	MySQL
Service Provider	Others
Access Type	Database
Region	South China (Guangzhou)
Account	root

Target Database Info

Instance ID	cdl-[REDACTED] Instance Monitoring
Target Database Type	MySQL
Access Type	Database
Region	South China (Guangzhou)
Account	root

同步配置页面

展示同步任务相关配置。

Sync Task

Sync Configuration

Structure Initialization

Data Initialization

Monitorir

Initialization

Structure initialization/Full data initialization

If Target Already Exists

Precheck and report error

SQL Type

DML

Insert、Update、Delete

DDL

Select All

Conflict Resolution Method

Ignore

Sync Object

Sync Object ⓘ

Unfold all Fold all

db-dst (Entire database selected)

结构初始化页面

在同步配置中选择了**结构初始化**后，在该页面展示相关信息。

参数	描述
对象名	目标实例中表的名称
源库	源库对象的名称，包括数据库和数据表
目标库	目标库中对象的名称，包括数据库和数据表
状态	展示当前状态，若失败会展示出失败原因
操作	单击【查看创建语法】查看在目标库中执行的创建语法

Sync Task

Sync Configuration

Structure Initialization

Data Initialization

Monitoring Data

Task Log

Object Name	Source Database	Target Database	Status
	db-dst.tb	db-dst.tb	

数据初始化页面

在同步配置中选择了**数据初始化**后，在该页面展示相关信息。

说明：

若处理失败，可单击**查看失败原因**排查失败原因。

监控数据信息展示

在还未监控到数据的情况下，默认展示-。



配置任务

最近更新时间：2024-07-08 19:11:11

操作场景

用户完成购买同步资源操作后，需要配置同步任务详细参数，也可以对已完成的配置任务，进行参数修改，如修改同步对象，冲突处理策略等。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的同步任务，在**操作**列单击**配置**。
2. 在同步任务页，配置或修改相关参数，并进行保存。

不同链路配置的操作要求和指导，请参考 [数据同步](#) 下的对应文档。

校验任务

最近更新时间：2024-07-08 19:11:11

操作场景

当初次校验任务失败，用户根据界面提示修复完成后，需要重新发起校验任务。

前提条件

已完成校验任务报错处理。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的迁同步任务，在**操作**列选择**更多 > 校验**。
2. 在弹出的对话框中，可创建校验任务和查询校验结果。

启动任务

最近更新时间：2024-07-08 19:11:11

操作场景

任务开启前，用户在校验阶段出现异常，处理异常后重新进行校验，待校验完成后需要启动同步任务。

前提条件

数据校验通过。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的同步任务，然后在**操作**列选择**更多 > 启动**。
2. 在弹出的对话框中，确认无误后，单击**确定**。

重试任务（MySQL 系列）

最近更新时间：2024-07-08 19:11:11

操作场景

同步任务启动后，如果遇到异常情况导致任务中断，用户可以进行手动重试。

全量阶段：全量阶段基于分块（chunk）导出/导入机制，每个分块会进行标记，任务重新启动后，DTS 识别中断的分块位置，然后从未同步的分块位置继续同步数据。

增量阶段：增量阶段基于 Checkpoint 机制，DTS 解析源库的 Binlog 再写入目标端时，一般每10秒会插入一个 Checkpoint 消息，用来标识数据同步的位点，任务重新启动后，DTS 从中断的位点继续同步数据。

数据同步场景支持重试的情况如下：

全量导出：全量导出仅对无锁导出并且源库待同步表有主键的场景支持重试，有锁导出的场景不支持重试。

全量导入：支持重试。

同步增量：支持重试。

适用范围

MySQL/TDSQL-C MySQL/MariaDB/Percona/Kafka 之间的数据同步链路中支持重试。

操作步骤

登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的同步任务，在**操作**列选择**更多 > 重试**。

限速

最近更新时间：2024-07-08 19:11:11

操作场景

在 DTS 数据同步任务配置中，或者任务运行中，可根据源库的数据写入量，源库和目标库的资源配置，来灵活调整任务并发数和 RPS。

说明：

本文仅提供了 DTS 性能参数的操作指导，如果您需要全面提升 DTS 传输性能，需要结合源/目标端的负载，网络情况等进行分析，具体请参考 [DTS 性能调优最佳实践](#)。

适用范围

数据同步：MySQL/MariaDB/Percona/TDSQL-C MySQL/TDSQL TDSQL Store 相互之间的链路。

注意事项

1. 在任务配置阶段，或者当任务状态为“运行中”时，可进行限速操作。
2. 降低任务速率时，请注意观察延迟数据量和延迟时间，DTS 默认只缓存7天的中间数据，长期延迟量过大，可能导致中间缓存数据过期，任务彻底失败。
3. 全量导出阶段调整速率会导致任务重启，部分重启场景任务需要从头开始同步，具体如下。增量阶段调整速率也会发生任务重启，但对数据传输无影响。
主键冲突策略选择“冲突报错”的任务，默认走有锁同步。在有锁同步场景中，重启任务后需要从头开始同步。
主键冲突策略未选择“冲突报错”的其他任务，默认走无锁同步。在无锁同步场景中，在结构导出阶段、无主键表导出阶段，重启任务需要从头开始同步；在有主键表导出阶段，重启任务会接续同步，无需从头开始。
4. 如果用户初次修改了限速中的全量速率参数（全量导出线程数据和 RPS，全量导入线程数和 RPS），后续进行修改同步对象操作时，会先跳转到限速参数调整界面，建议用户将全量速率参数尽量调大一些（根据源和目标库负载情况综合判断），避免新追加对象后，对新对象的全量同步耗时较久

调整原则

DTS 任务全量导出并发线程数默认为8，一般情况下不需要修改。如遇需要调整，请结合监控指标，源库/目标库配置，在不影响源库/目标库性能的前提下，最大程度提升 DTS 传输速率。

DTS 传输性能与源/目标库的配置、DTS 任务规格、网络延迟、网络带宽等都有关系，所以提升 DTS 传输性能时，单独的调增大整线程数，或增大 RPS，不一定有效，需要分析性能瓶颈在哪个环节，针对具体问题具体解决。

任务阶段	调整场景	调整方案	调整结果
任务配置阶段	源库配置太低，无法承担 DTS 增加的负载，在 DTS 任务启动后源库出现崩溃。	减小源库导出线程数，降低源库导出 RPS。	有效。
	目标库配置太低，无法承担 DTS 增加的负载，在 DTS 任务启动后目标库出现崩溃。	减小目标库导入线程数，降低目标库导入 RPS。	有效。
全量阶段	源库/目标库配置高，DTS 全量传输速率慢	如果源库配置高，增大源库导出线程数 如果目标库配置高，增大目标库导入线程数。 全量阶段默认 RPS 非常大，提升传输速率时一般不需要调整 RPS。	不一定有效。 如果调整后未生效，请排查网络延迟，带宽等因素，因为这些也会影响传输性能。
	目标库配置低，DTS 写入目标库速度太快，对目标库负载产生影响（如 CPU 占用率高）	减小目标库导入线程数，降低目标库导入 RPS。	有效。
增量阶段	源库配置高，源库写入数据量大，网络延迟大	查看 RPS 是否达到上限，DTS 任务不同规格，对应的 RPS 上限 不同。 如果 RPS 已达上限，则先升级 DTS 任务规格（通过 变配 操作实现）。	如果升级任务后未生效，请排查网络延迟，带宽等因素，因为这些也会影响传输性能。
	目标库配置低，DTS 写入目标库速度太快，对目标库负载产生影响（如 CPU 占用率高）	1. 降低 DTS 任务规格。 2. 减小目标库导入并发线程数。	有效。
	目标 Kafka 配置低，DTS 写入目标库速度太快，对目标库负载产生影响	在同步到 Ckafka 的链路中，遇到写入速率太快，降低目标端写入 RPS 和线程数效果可能不显著，请先调整 CKafka 带宽和分区数。 1. 提高 CKafka 网络带宽配置。 2. 调整 Ckafka 的分区数，多个分区可以并发写入，提升写入 Ckafka 的速率。	有效。

	目标库配置低，DTS 连接数过多，目标库线程达上限	降低目标库导入线程数。	有效。
	DTS 写入到目标库，与目标库本身的写入产生锁争抢（例如 Lock wait timeout 错误）	降低目标库导入线程数，减少锁竞争。	有效。

任务配置中限速

同步任务初始配置阶段，在**设置同步选项**和**同步对象**页面的高级选项中，可设置传输速率参数。

高级选项

限制传输速率

源库全量导出并发线程数 ⓘ *

-

8

+

个

源库全量导出 RPS ⓘ

-

0

+

千 ▾

Counts/s

5 千万 Counts/s 表示不限制

目标库全量导入并发线程数 ⓘ *

-

8

+

个

目标库全量导入 RPS ⓘ

-

0

+

千 ▾

Counts/s

5 千万 Counts/s 表示不限制

目标库增量导入并发线程数 *

-

32

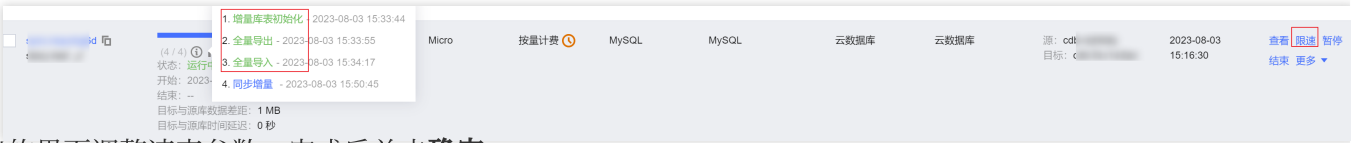
+

个

任务运行中限速

任务在全量数据同步阶段进行限速操作，可以修改全量+增量的速率参数（全量导出线程数和 RPS，全量导入线程数和 RPS，增量导入线程数）；任务在增量数据同步阶段进行限速操作，仅可修改增量的速率参数（增量导入线程数），具体操作参考如下。

- 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的同步任务。
- 在全量数据同步阶段进行限速操作。
在**操作**列选择**更多 > 限速**。



在弹出的界面调整速率参数，完成后单击**确定**。

限制任务速率

当前阶段调整任务速率将导致任务从头开始全量数据导出，请谨慎调整速率。

当源库或者目标库因数据传输导致实例负载过高时，减少导出、导入线程数，可有效减少任务运行对实例和带宽资源的占用；当源库或者目标库实例规格、网络带宽较高时，增大线程数可加快数据传输速度。[详情参考调整任务速率](#)

注意：在数据量大时，降低全量导入或者导出速率，可能会导致全量阶段变慢，增量数据接管过期；推荐根据业务需求，提升源库和目标库规格

您已选1个任务，[收起详情](#)

任务 ID	任务名	计费模式	现有规格
		按量计费	Micro

源库全量导出并发线程数

-

8

+

个

默认值：8个

源库全量导出 RPS

-

0

+

Count/s

全量阶段无系统默认值，可根据历史监控数据调整

目标库全量导入并发线程数

-

8

+

个

默认值：8个

目标库全量导入 RPS

-

0

+

Count/s

全量阶段无系统默认值，可根据历史监控数据调整

目标库增量导入并发线程数

-

32

+

个

确定

取消

3. 在增量数据同步阶段进行限速操作。

在**操作列**选择**更多 > 限速**。

版权所有：腾讯云计算（北京）有限责任公司

第277 共300页

限制任务速率

×

⚠ 大部分情况下，无需调整增量传输速率，[详情参考调整任务速率](#) 

您已选1个任务，[收起详情](#) ▼

任务 ID	任务名	计费模式	现有规格
syr 		按量计费	Micro

目标库增量导入并发线程数 *

−

32

+

个

确定

取消

结束任务

最近更新时间：2024-07-08 19:11:11

操作场景

数据同步是一个持续进行的任务，不会自动结束，需要用户选择合适的时机，手动结束数据同步任务。

说明

对于按量计费任务，结束任务后则停止计费。

对于包年包月任务，结束任务后不会自动退费，用户若需要退费，需自行操作**销毁/退费**。设置了自动续费的任务，仅停止任务未销毁，在每个计费周期开始时仍会继续扣费。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的同步任务，然后在**操作**列选择**更多 > 结束**。

说明

建议在结束任务前，查看任务的状态，满足如下条件，避免因数据差距大或者延迟大导致同步结果不一致。

目标与源库数据差距为**0MB**。

目标与源库时间延迟为**0秒**。

☐	replicate-replicate	(3 / 3)   Status: Running Start: 2021-12-16 11:02:07 End: -- Data lag between target and source databases: 0 MB Time lag between target and source databases: 0s	Immediate execution	Pay as you go	MySQL -> MySQL	MySQL	MySQL	Database
☐	replicate-vm		Immediate execution	Pay as you go	MySQL -> MySQL	MySQL	MySQL	Database

2. 在弹出的对话框中，确认无误后，单击**确定**。

Stop Sync Task



Are you sure you want to stop the sync task `replic-f0g (repl-f0g)`?

Once confirmed:

1. The task will be stopped.
2. The task will be in the "Stopped" status.

Confirm

Cancel

修改同步配置

最近更新时间：2024-07-08 19:11:11

操作场景

用户初次配置完同步任务，同步任务已经开始运行，后续如果需要增加同步对象、删除同步对象、修改同步参数配置，都可以通过本操作进行。

修改同步配置时，已有的同步任务不会暂停，也不会受影响。修改的参数，除数据初始化选项不支持修改，其他的主键冲突策略，SQL 同步策略，库、表、高级对象等，都可以进行修改。

增加对象：同步任务类型为 全量 + 增量同步时，新增加一个同步对象，DTS 会将新增对象的存量数据和增量数据都同步到目标库；同步任务类型为仅增量同步时，新增加一个同步对象，DTS 只将新增对象的增量数据同步到目标库。

删除对象：修改配置后，删除对象新增的数据不会再同步到目标库，修改操作前已经同步到目标库的数据仍保留，不会删除。

修改主键冲突策略：仅对修改配置后新同步的数据应用新的主键冲突策略。

适用范围

仅如下链路支持修改同步配置操作。

MySQL/MariaDB/Percona/TDSQL-C MySQL -> MySQL

MySQL/MariaDB/Percona -> MariaDB

MySQL/MariaDB/TDSQL-C MySQL -> TDSQL-C MySQL

MySQL/MariaDB/Percona/TDSQL-C MySQL -> Kafka

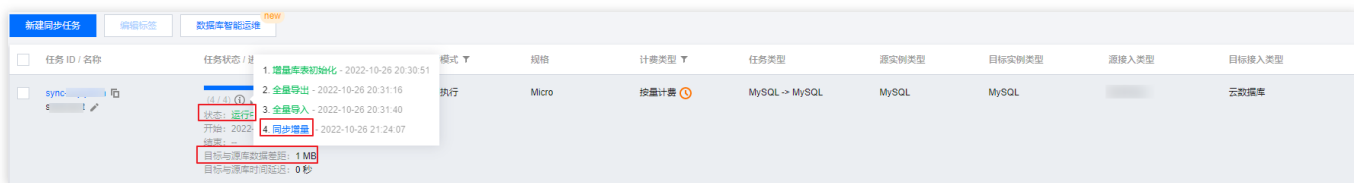
注意事项

1. 修改同步配置功能于2022年12月上线，仅支持对上线后新发起的任务进行修改，上线前已有的存量任务暂不支持修改。存量任务如需修改请 [提交工单](#) 对任务升级。
2. 同步任务需要满足在“同步增量”阶段，任务状态为“运行中”，“目标库与源库时间延迟”小于1GB，这三个条件，才支持修改同步配置。
3. 新增或者删除同步对象时，需要保证相互关联的对象一起选择，否则会导致同步任务失败。
4. 修改同步配置时，仅支持对新增加的库表对象进行重命名操作，历史选择的库表对象，此时不能再进行重命名。如果用户在修改同步配置时，同时又对历史对象进行重命名操作，系统会在校验阶段报错，需要用户自行改回之前的名称。

5. 如果用户初次修改了限速中的全量速率参数（全量导出线程数数据和 RPS，全量导入线程数和 RPS），后续进行修改同步对象操作时，会先跳转到限速参数调整界面，建议用户将全量速率参数尽量调大一些（根据源和目标库负载情况综合判断），避免新追加对象后，对新对象的全量同步耗时较久。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的同步任务，在**操作**列单击**修改同步配置**。
任务需要在“同步增量”阶段，任务状态为“运行中”，“目标库与源库数据差距”小于1GB，才支持修改配置。其他的“任务失败”、“任务出错”等都不支持修改配置。



2. （可选）确认是否需要调整速率参数，确认后单击**确定**。

说明：

如果用户初次修改了限速中的全量速率参数（全量导出线程数数据和 RPS，全量导入线程数和 RPS），后续进行修改同步对象操作时，会先跳转到限速参数调整界面，建议用户将全量速率参数尽量调大一些（根据源和目标库负载情况综合判断），避免新追加对象后，对新对象的全量同步耗时较久。

确定修改同步对象?

为了尽快完成修改同步配置过程，请先调整当前任务速率限制。

建议:

观察源库及目标库负载，如果负载较低，尽可能提升源库导出和目标库导入的并发线程数及速率，促使修改同步配置尽快完成。

注意:

在修改同步配置过程中，无法调整线程数，请先调整限速后，再启动修改同步配置。

如果修改同步配置失败或者中断修改配置，本次限速调整将在任务重启后生效，或在任务列表内点击【限速】重新配置后生效。

源库全量导出并发线程数 *

-

8

+

个

源库全量导出 RPS

-

0

+

千

Counts/s

5 千万 Counts/s 表示不限制

目标库全量导入并发线程数 *

-

8

+

个

目标库全量导入 RPS

-

0

+

千

Counts/s

5 千万 Counts/s 表示不限制

目标库增量导入并发线程数 *

-

48

+

个

确定

取消

3. 进入配置同步任务页，在**设置同步选项和同步对象**中，修改相关参数，如增加对象、删除对象、修改主键冲突策略等，然后单击**保存并下一步**。除数据初始化选项不支持修改，其他的选择都支持修改。

1 设置同步源和目标数据库

2 设置同步选项和同步对象

3 校验任务

数据初始化选项

初始化类型

结构初始化

全量数据初始化

已存在同名表

前置校验并报错

忽略并继续执行

数据同步选项

主键冲突处理

冲突报错

冲突忽略

冲突覆盖

同步操作类型

DML

Insert

Update

Delete

DDL

DDL

DDL 自定义

同步对象选项

高级迁移对象

Procedure

Function

高级对象的复制是一次性动作，在任务启动后，不会再复制新建的高级对象。

源实例对象搜索结果最多展示 200 条记录，如需查看更多对象，请指定对象名称进行针对性搜索。

源库对象

搜索库名，支持模糊匹配

共有 14 个数据库，当前展示全部 14 个

db1

db2

db2_10w

dfs

migrate

obj

obj1

obj2

obj3

刷新 全选 清空

已选对象

全局搜索源对象名，支持模糊匹配

db1_928_2126(原名: db1) (选中)

db2_928_2143(原名: db2) (选中)

展开所有 折叠所有 全选 清空 恢复原名称

上一步

保存并下一步

4. 任务重新校验，校验通过后，单击**启动任务**。

创建校验任务

查询校验结果

校验项说明

连接DB检查

通过

成本检查

通过

源实例权限检查

警告 查看详情

部分实例参数检查

通过

目标实例权限检查

通过

目标实例内容冲突检查

通过

目标实例空间检查

通过

binlog参数检查

通过

外键依赖检查

通过

外键部分库表依赖检查

通过

视图检查

通过

警告项检查

警告 查看详情

高级对象检查

通过

配置修改检查

通过

周边检查

上一步

重新校验

启动任务

5. 返回任务列表，任务状态显示为“运行中（动态修改配置）”，表示 DTS 正在对修改配置后的数据进行处理。

说明：

修改同步配置后，当任务出于“运行中（动态修改配置）”，请勿在源数据库执行 DDL 操作，否则可能会导致任务失败。

用户新增同步对象时，DTS 会先将新增对象的存量数据同步到目标库，如果新增对象的存量数据较多，则“运行中（动态修改配置）”这个状态的持续时间也会较长。

任务 ID / 名称	任务状态 / 进度	运行模式	规格	计费类型	任务类型	源实例类型	目标实例类型	源接入类型	目标接入类型	操作
sync- s	运行中 (动态修改配置) 状态: 运行中 (动态修改配置) 开始: 2022-10-26 20:30:43 结束: -- 目标与源库数据差距: 0 MB 目标与源库时间延迟: 0 秒	立即执行	Micro	按量计费	MySQL -> MySQL	MySQL	MySQL		云数据库	查看 创建类似任务 编辑标签 启动 更多

当任务状态转换为“运行中”，表示本次修改配置的操作已完成，后续如有需要可以再次发起修改配置的操作。

任务 ID / 名称	任务状态 / 进度	运行模式	规格	计费类型	任务类型	源实例类型	目标实例类型	源接入类型	目标接入类型	操作
sync- s	运行中 状态: 运行中 开始: 2022-10-26 20:30:43 结束: -- 目标与源库数据差距: 1 MB 目标与源库时间延迟: 0 秒	立即执行	Micro	按量计费	MySQL -> MySQL	MySQL	MySQL		云数据库	查看 结束 修改同步配置 创建类似任务 更多

暂停任务

最近更新时间：2024-07-08 19:11:11

操作场景

在任务启动后，如果源数据库连接数过多或者业务需要调整，用户可以选择暂停任务，待业务调整完成后继续同步任务。

适用范围

MySQL/TDSQL-C MySQL/MariaDB/Percona 之间的数据同步链路。

注意事项

同步任务在“运行中”的状态下，才可进行暂停操作。

任务暂停期间，DTS 会持续拉取源库的数据，但不会向目标库中写入数据，待任务恢复后继续任务。

全量导出阶段：任务暂停再恢复后，不支持断点续传，DTS 需要重新导出源库的全量数据。

全量导入阶段、增量数据同步阶段：任务暂停再恢复后，DTS 将基于暂停前同步的 Binlog 位点继续数据传输。

同步任务期间 Binlog 记录最多保留7天，若暂停时长超过7天则会导致任务失败，无法重新启动。建议任务暂停时间不超过3天。

因为任务暂停期间，DTS 还会持续拉取源库的数据，所以对于计费链路，同步任务暂停期间仍会保持计费。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的同步任务，在**操作**列单击**暂停**。
2. 在弹出的对话框中，确认无误后，单击**确定**。

暂停同步任务



您已选1个任务，[收起详情](#)

任务 ID	任务名	计费模式	现有规格
sync-i-3	S-1...	包年包月	large

- 服务暂停期间，DTS 服务会持续拉取源端的数据和 binlog，但不会向目标库中写入数据，恢复任务后，将基于任务暂停前的 binlog 数据点继续执行该任务
- 同步期间 binlog 记录最多保留 7 天，若暂停时长超过 7 天则会导致任务失败，无法重新启动；建议任务暂停时间不超过 3 天
- 对于计费链路，同步任务暂停期间仍会保持计费

确定

取消

恢复任务

最近更新时间：2024-07-08 19:11:11

操作场景

用户因为业务需要暂停了同步任务，后续需要继续任务，可通过本操作进行。

适用范围

MySQL/TDSQL-C MySQL/MariaDB/Percona 之间的数据同步链路。

注意事项

任务暂停期间，DTS 会持续拉取源库的数据，但不会向目标库中写入数据，待任务恢复后继续任务。

全量导出阶段：任务暂停再恢复后，不支持断点续传，DTS 需要重新导出源库的全量数据。

全量导入阶段、增量数据同步阶段：任务暂停再恢复后，DTS 将基于暂停前同步的 Binlog 位点继续数据传输。

同步任务期间 Binlog 记录最多保留7天，若暂停时长超过7天则会导致任务失败，无法重新启动。建议任务暂停时间不超过3天。

对于计费链路，同步任务暂停期间仍会保持计费。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的同步任务，在**操作**列单击**恢复**。
2. 在弹出的对话框中，确认无误后，单击**确定**。

重置任务

最近更新时间：2024-07-08 19:11:11

操作场景

使用包年包月的同步实例，当一个同步任务实例结束，需要重新发起新的任务实例时，可通过重置操作，清除旧的任务配置数据，重新配置新的任务数据。这样可以使用同一个实例 ID 来计费，避免用户先退费旧实例，再购买新实例，频繁进行购买操作。

重置操作后，实例 ID 保持不变，实例名称会变更为与 ID 一致，旧任务数据将被释放不可恢复，任务重置到“未初始化”状态。

注意事项

任务仅在“已结束”状态下才可以进行重置操作。

重置操作前请确保旧任务已结束或不再需要。

适用范围

MySQL/MariaDB/Percona/TDSQL-C MySQL 之间的同步任务，且采用包年包月计费模式支持重置。

重置任务功能于2022年12月上线，上线后新建的同步任务支持重置，上线前已有存量的任务不支持。存量任务如需重置请 [提交工单](#) 对任务升级。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定任务实例，在**操作**列选择**重置**。
2. 在弹出的对话框中，确认无误后勾选，然后单击**重置**。

删除任务

最近更新时间：2024-07-08 19:11:11

操作场景

对于已结束或者失败等不再需要的实例任务，用户可删除任务，任务删除后就不会存在，资源也会被清除。

适用范围

未计费的数据同步任务。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定任务实例，在**操作**列选择**更多 > 删除**。
2. 在弹出的对话框，确认无误后，单击**删除**。

变更配置

最近更新时间：2024-07-08 19:11:11

操作场景

用户业务诉求变更，需要对已选择的同步链路规格进行修改时，可通过本操作进行修改。

规格变更后会立即生效，如果使用包年包月，当月剩余的链路规格费用，会按照差价进行多退少补，之前的计费周期不变。

说明

用户自助变更配置后，需要3-5分钟才能生效，配置生效期间任务会暂停。

适用范围

已计费的数据同步任务。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定任务实例，在**操作**列选择**更多 > 变配**。
2. 在弹出的对话框，选择需要修改的规格，然后单击**确定**。

按量计费转包年包月

最近更新时间：2024-07-08 19:11:11

操作场景

DTS 支持将按量计费任务转为包年包月。用户转换计费模式后，DTS 会生成包年包月的续费订单，需要用户及时支付续费订单才能转换成功。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的同步任务，然后在**操作**列选择**更多 > 按量计费转包年包月**。
2. 在弹出的对话框中，选择续费时长和自动续费设置，确认无误后，单击**确定**。

销毁/退货实例

最近更新时间：2024-07-08 19:11:11

操作场景

销毁：对于已结束或者失败等不再需要的实例任务，用户可销毁任务。

退货：使用包年包月购买的实例任务，用户不需要使用，可以进行退货处理。相关的退费请参考[退费说明](#)。

销毁或退货后实例进入隔离状态，隔离7天后自动下线，请谨慎处理。

按量计费实例因误操作销毁后，可通过[解隔离任务](#)来进行恢复。

包年包月实例销毁或退货后，可通过续费任务 来进行恢复。

如果采用按量计费模式，在创建任务时冻结一个小时的费用，后续用户主动销毁实例或由于欠费被动隔离销毁后会解冻，所以在同步任务使用按量计费购买后未启动等，不会产生计费的场景，计费系统都会预先冻结1小时的费用。
建议用户对已完成的任务或者不需要的任务，及时进行销毁，以便及时解冻费用。

适用范围

已计费的数据同步任务。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定任务实例，在**操作**列选择**更多 > 销毁 / 退货**。
2. 在弹出的对话框中，确认无误后勾选**已阅读并同意销毁规则**，然后单击**立即隔离**。

续费任务

最近更新时间：2024-07-08 19:11:11

操作场景

计费模式采用包年包月的数据同步实例，可以对实例进行续费操作，续费以月为单位。当实例因为欠费进入隔离状态时，依然可以通过续费恢复实例。

对同步任务的续费，可以在 **DTS 控制台**，也可以在费用中心。当需要对多个同步任务进行批量续费时，请在费用中心进行操作。

适用范围

已计费的数据同步任务。

DTS 控制台续费

1. 登录 **DTS 控制台**，在左侧导航选择**数据同步**页，选择指定任务实例，在**操作**列选择**更多 > 续费**。
2. 在弹出的对话框，选择续费时长，然后单击**确定**。

费用中心续费

1. 在 **DTS 控制台** 右上角选择**费用 > 计费中心**。
2. 选择**续费管理**，筛选时间范围和对应的服务后进行续费。

对单个任务进行续费，直接单击对应任务操作列的**续费**。

对多个任务进行续费，请先勾选需要批量操作的任务，然后单击**批量操作**。

解隔离任务

最近更新时间：2024-07-08 19:11:11

操作场景

采用按量计费的同步任务，用户因为误操作销毁任务进入隔离状态，可通过本操作指导对任务进行恢复。恢复后任务将回到销毁前的状态。

适用范围

采用按量计费的同步任务。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定任务实例，在**操作**列选择**更多 > 解隔离（恢复）**。
2. 在弹出的对话框中，确认无误后单击**恢复**。

下线实例

最近更新时间：2024-07-08 19:11:11

操作场景

用户自助销毁或者因为欠费导致实例进入回收站，状态显示为隔离中时，才可以进行下线操作。
实例进入回收站后，默认保留7天后会自动下线，如果用户需要立即下线，可通过本操作进行处理。

适用范围

已计费的数据同步任务。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定任务实例，在**操作**列选择**更多 > 立即下线**。
2. 在弹出的对话框中，确认无误后勾选**已阅读并同意下线规则**，然后单击**立即下线**。

创建类似任务

最近更新时间：2024-07-08 19:11:11

操作场景

创建类似任务，可以让用户快速创建一个和当前任务配置相同的任务，新的任务中数据库类型、接入方式、计费模式、冲突策略、DDL 同步策略等默认填充的选项都和历史任务保持一致，用户如果需要也可根据情况进行修改。在任务失败、多对一同步、或者一对多同步等场景中，可以通过创建类似任务，快速配置新的任务。

操作步骤

1. 登录 DTS 控制台，在左侧导航选择**数据同步**页，选择指定任务，在**操作**列选择**更多 > 创建类似任务**。
2. 在弹出的对话框中，确认无误后，单击**创建类似任务**。

查看日志

最近更新时间：2024-07-08 19:11:11

操作场景

同步任务进行中，用户可以查看同步任务日志，了解任务进行的阶段。

操作步骤

1. 登录 [DTS 控制台](#)，通过以下两个方式查看任务。
- 方式一：在数据同步页面，选择指定的同步任务，单击任务名称。

• 方式二：在数据同步页面，选择指定的同步任务，在操作列选择更多 > 查看。
2. 查看同步日志。
- 切换页签查看任务日志。

Sync Task	Sync Configuration	Structure Initialization	Data Initialization	Monitoring Data	Task Log
All	Today	Yesterday	Last 7 days	Last 30 days	Select date 
Operation Time	Status 				
2021-12-16 11:02:09	Task running				
2021-12-16 11:02:05	Preparing				
2021-12-16 10:59:05	Verification passed				
2021-12-16 10:58:08	Checking				
2021-12-16 10:58:07	Initialized				

任务状态说明

状态	说明
----	----

状态	说明
未初始化	完成购买操作，未配置同步任务。
已初始化	已完成同步任务配置。
校验中	同步任务校验进行中。
校验通过	同步任务校验完成。
校验不通过	同步任务校验不通过。
准备运行	准备启动同步任务。
任务运行	同步任务运行中。
可重试错误	同步过程中，出现异常导致任务中断，通过界面操作重试，继续同步任务的阶段。
结束中	同步任务进行中，用户手动结束了迁移任务，结束任务进行中。
已结束	结束任务完成。
任务失败	同步任务失败。
释放中	对于已结束、失败、或者不需要的任务，用户手动选择释放任务，释放任务进行中。
已释放	释放任务完成，任务释放后就不会存在，也不会占用资源。

释放任务

最近更新时间：2024-07-08 19:11:11

操作场景

对于已结束、失败、或者不需要的任务，用户可选择释放任务。任务释放后就不会存在，也不会占用资源。请谨慎执行释放任务。

前提条件

同步任务已结束、失败或者任务不再需要。

操作步骤

1. 登录 [DTS 控制台](#)，在左侧导航选择**数据同步**页，选择指定的同步任务，在**操作**列选择**更多 > 释放**。
2. 在弹出的对话框中，确认无误后，单击**确定**。