

TDSQL-H LibraDB

实践教学



腾讯云

【 版权声明 】

©2013–2024 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【 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。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【 服务声明 】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。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。

【 联系我们 】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或 95716。

文档目录

实践教程

多表归并（数据合并）最佳实践

基于 Merge 表引擎数据合并最佳实践

实践教程

多表归并（数据合并）最佳实践

最近更新时间：2022-05-24 16:36:50

当需要将已有 OLTP 数据库实例中的多个表合并时，可通过开启多表归并开关，将多个表数据实时汇聚到一个 LibraSQL 分析引擎进行分析，实现数据写入读取时均合并。

场景概述

场景	说明
场景一	将多个源 OLTP 实例多个表数据，实时汇集到一个 LibraSQL 分析引擎进行分析。
场景二	将同一个源 OLTP 实例多个表数据，实时汇集到一个 LibraSQL 分析引擎的同一个库表进行分析。

使用限制

请参见 [多表归并（数据合并）](#) 中的使用限制。

注意事项

添加 CDC 任务通用注意事项请参见 [添加 CDC](#)。

场景一

场景示例

多表归并 CDC 任务	源端 OLTP 库表名	合并至目标端库表名
任务一	源端 OLTP 1database1: tableA 和 tableB	databaseA: table1
任务二	源端 OLTP 2database2: tableE 和 tableF	databaseA: table1

任务一和任务二均启动后，源端 OLTP 1和源端 OLTP 2中指定的库表将会实时合并至目标端指定的库表。

操作步骤

以下步骤仅以场景示例为例，其他场景案例请参考该操作步骤执行。

1. 创建 CDC 任务一。

1.1 参考 [添加 CDC 任务](#) 并已进入设置指定对象步骤。

示例：源端为：OLTP 1

1.2 在**设置对象**页面，同步对象选择**指定对象**。

1.3 打开**多表归并**开关。

1.4 在**源库对象**中选择库表。

示例：database1: tableA 和 tableB

1.5 单击 ，在弹出的**将所有源端数据合并到指定库表**对话框中，输入目标实例的库表名，单击**确定**。

示例：databaseA: table1

1.6 单击**下一步**，进入 **添加 CDC 任务** 的高级设置步骤，继续执行其他步骤，启动 CDC 任务后，该配置生效。

任务启动后，源端已指定的表会合并至目标端的指定库表中。示例：database1: tableA 和 tableB 数据会合并至 databaseA: table1 中。

2. 创建 CDC 任务二。

参考创建 CDC 任务一的步骤。

示例：源端为 OLTP 2，在**源库对象**中选择库表 database2: tableE 和 tableF，合并到目标端的指定库表命名为 databaseA: table1。

任务启动后，源端已指定的表会合并至目标端的指定库表中。示例：database2: tableE 和 tableF 数据会合并至 databaseA: table1 中。

操作结果

源 OLTP 1与源 OLTP 2实现多源合一：任务一和任务二中的数据会全部合并至目标端指定的相同库表中。

示例：database1: tableA 和 tableB 与 database2: tableE 和 tableF 的数据会全部合并至 databaseA: table1 中。

场景二

场景示例

多表归并 CDC 任务	源端 OLTP 库表名	合并至目标端库表名
任务一	源端 OLTP 3 database3: tableA 和 tableB database4: tableE 和 tableF	databaseB: table1

任务一启动后，源端 OLTP 3中指定的多个库表将会实时合并至目标端指定的库表。

操作步骤

以下步骤仅以场景示例为例，其他场景案例请参考该操作步骤执行。

1. 参考 **添加 CDC 任务** 并已进入设置指定对象步骤。

示例：源端为：OLTP 3

2. 在设置对象页面，同步对象选择指定对象。

3. 打开多表归并开关。

4. 在源库对象中选择库表。

示例：database3: tableA 和 tableB, database4: tableE 和 tableF

5. 单击 ，在弹出的将所有源端数据合并到指定库表对话框中，输入目标实例的库表名，单击确定。

示例：databaseC: table1

6. 单击下一步，进入 [添加 CDC 任务](#) 的高级设置步骤，继续执行其他步骤，启动 CDC 任务后，该配置生效。

操作结果

源 OLTP 3实现多表合一：任务一中的数据会全部合并至目标端指定的相同库表中。

示例：database3: tableA 和 tableB, database4: tableE 和 tableF 的数据会合并至 databaseB: table1 中。

基于 Merge 表引擎数据合并最佳实践

最近更新时间：2022-05-24 16:36:44

当需要将已有 OLTP 数据库实例中的多个表进行读合并时，可基于 Merge 表引擎合并，将多个表数据实时汇聚到一个 LibraSQL 分析引擎进行分析。

场景概述

场景	说明
场景一	将多个源 OLTP 实例多个表数据，实时汇集到一个 LibraSQL 分析引擎进行分析。
场景二	将同一个源 OLTP 实例多个数据库中的多个表数据，实时汇集到一个 LibraSQL 分析引擎进行分析。

使用限制

请参见 [基于 Merge 表引擎数据合并](#) 中的使用限制。

注意事项

添加 CDC 任务通用注意事项请参见 [添加 CDC](#)。

场景一

场景示例

多表归并 CDC 任务	源端 OLTP 库表名	合并至目标端库表名
任务一	源端 OLTP 1database1: tableA 和 tableB	databaseA: tableA_log 和 tableB_log
任务二	源端 OLTP 2database2: tableC 和 tableD	databaseA: tableC_log 和 tableD_log

任务一和任务二均启动后，源端 OLTP 1和源端 OLTP 2中指定的多个库表将会实时同步至目标端指定的库表，再通过 Merge 表引擎，实现数据读取时合并。

操作步骤

以下步骤仅以场景示例为例，其他场景案例请参考该操作步骤执行。

1. 创建 CDC 任务一。

1.1 参考 [添加 CDC 任务](#) 并已进入设置指定对象步骤。

示例：源端为：OLTP 1对象选择**指定对象**。

1.2 **多表归并**开关默认关闭。

1.3 在**源库对象**中选择库表。

示例：database1: tableA 和 tableB

1.4 单击 ，**已选对象**中将展示已选择的库表。

1.5 将鼠标悬停至目标库，单击  **修改库名称**。

示例：databaseA

1.6 在**已选对象**右侧单击**批量改名**，在弹出的**批量添加已选对象表名后缀**对话框中输入后缀，单击**确定**。

示例： _log

已选对象中将显示修改后的库表名称。

示例：databaseA: tableA_log 和 tableB_log

1.7 单击**下一步**，进入 [添加 CDC 任务](#) 的高级设置步骤，继续执行其他步骤，启动 CDC 任务后，该配置生效。

任务启动后，源端已指定的表会同步至目标端的指定库表中。示例：database1: tableA 和 tableB 数据会分别同步至 databaseA: tableA_log 和 tableB_log 中。

2. 创建 CDC 任务二。

参考创建 CDC 任务一的步骤。

示例：源端为 OLTP 2，在**源库对象**中选择库表 database2: tableC 和 tableD，合并到目标端的指定库表命名为 databaseA: tableC_log 和 tableD_log。

任务启动后，源端已指定的表会同步至目标端的指定库表中。示例：database2: tableC 和 tableD 数据会分别同步至 databaseA: tableC_log 和 tableD_log 中。

3. 利用 Merge 表能力，匹配具有相同命名模式和表结构的数据表，构造虚拟表，完成表数据查询合并。

```
CREATE TABLE ... Engine=Merge(currentDatabase(), tables_regexp)
```

其中，tables_regexp 是一个正则表达式，用于匹配指定数据库中的表名。**示例：**

```
-- 在当前数据库下，按照普通表 tableA_log 的表结构建立 Merge 表 table_all，其中  
table_all 是虚拟表， *_log 匹配当前数据库中表名的正则表达式
```

```
CREATE TABLE table_all on cluster default_cluster as databaseA.tableA_log  
Engine=Merge(currentDatabase(), '.*_log');
```

```
-- 当查询 Merge 表时，LibraSQL 分析引擎会自动查询所有带后缀 _log 的表(即  
tableA_log、tableB_log、tableC_log、tableD_log)，将结果汇聚返回
```

```
select * from table_all;
```

操作结果

源 OLTP 1与源 OLTP 2实现多源合一：在目标端使用 Merge 表引擎查询任务一和任务二中数据时，多个表数据合并返回。

示例：当查询 Merge 表时，LibraSQL 分析引擎会自动查询所有 *_log（即 tableA_log、tableB_log、tableC_log、tableD_log）表，将结果汇聚返回。

场景二

场景示例

多表归并 CDC 任务	源端 OLTP 库表名	合并至目标端库表名
任务一	源端 OLTP 3 A_1: d_1 和 d_2 A_2: c_1 和 c_2	A_1: d_1_log 和 d_2_log A_2: c_1_log 和 c_2_log

任务一启动后，源端 OLTP 3中指定的库表会分别实时同步至目标端指定的库表，再通过 Merge 表引擎，实现数据读取时合并。

操作步骤

以下步骤仅以场景示例为例，其他场景案例请参考该操作步骤执行。

1. 创建 CDC 任务一。

1.1 参考 [添加 CDC 任务](#) 并已进入设置指定对象步骤。

示例：源端为：OLTP 3

1.2 在设置对象页面，同步对象选择指定对象。

1.3 多表归并开关默认关闭。

1.4 在源库对象中选择库表。

示例：A_1: d_1 和 d_2，A_2: c_1 和 c_2

1.5 单击 ，已选对象中将展示已选择的库表。

1.6 在已选对象右侧单击批量改名，在弹出的批量添加已选对象表名后缀对话框中输入后缀，单击确定。

示例：_log

已选对象中将显示修改后的库表名称。

示例：A_1: d_1_log 和 d_2_log，A_2: c_1_log 和 c_2_log

1.7 单击下一步，进入 [添加 CDC 任务](#) 的高级设置步骤，继续执行其他步骤，启动 CDC 任务后，该配置生效。

任务启动后，源端已指定的表会同步至目标端的指定库表中。示例：A_1: d_1和d_2、A_2: c_1和c_2数据会分别同步至A_1: d_1_log和d_2_log、A_2: c_1_log和c_2_log中。

2. 使用 Merge 表引擎，匹配具有相同命名模式的库，且库中有相同命名模式和表结构的数据表，构造虚拟表，完成表数据查询合并。

```
CREATE TABLE ... Engine=Merge(REGEXP(expression), tables_regexp)
```

其中，expression 是用于匹配数据库名称的正则表达式。tables_regexp 是一个正则表达式，用于匹配指定数据库中的表名。示例：

```
-- 按照普通表 d_1_log 的表结构建立 Merge 表 table_all，其中 table_all 是虚拟表，A_* 匹配数据库名称的正则表达式，.*_log 匹配数据库中表名的正则表达式
CREATE TABLE table_all on cluster default_cluster as A_1.d_1_log
ENGINE=Merge(REGEXP('A_*'), '.*_log');

-- 当查询 Merge 表时，LibraSQL 分析引擎会自动查询所有 A_* 数据库(即 A_1、A_2)中的带后缀 _log 的表(即 d_1_log、d_2_log、c_1_log、c_2_log)，将结果汇聚返回
select * from table_all;
```

操作结果

源 OLTP 3实现同源多表合一：在目标端使用 Merge 表引擎查询任务一中数据时，跨库多个表数据合并返回。

示例：当查询 Merge 表时，LibraSQL 分析引擎会自动查询 A_1、A_2 中的带后缀 _log 的表（即 d_1_log、d_2_log、c_1_log、c_2_log），将结果汇聚返回。