



分布式融合数据库 HTAP

数据迁移手册

天翼云科技有限公司

数据迁移工具 DM 简介

DM（数据迁移）是一个高效的数据迁移工具，支持从兼容 MySQL 协议的数据库（如 MySQL、MariaDB 等）向分布式融合数据库 HTAP 进行全量数据迁移和增量数据同步。使用 DM 工具能够简化数据迁移流程，降低运维成本。

DM 具备以下特性：

- **兼容 MySQL**。高度兼容 MySQL 5.7 协议、MySQL 5.7 常用的功能及语法。
- **支持 DML 和 DDL 事件同步**。能够解析并同步 binlog 中的 DML 和 DDL 事件。
- **合库合表的同步模式**。可将多个上游 MySQL 实例中的分表数据合并同步至下游的分布式融合数据库 HTAP 数据库表中，支持自定义同步规则，自动识别和处理上游 DDL 变更，从而显著降低运维复杂性。
- **灵活的过滤器**。提供多种过滤选项，包括预定义事件类型、正则表达式和 SQL 表达式，以适应不同的场景需求。
- **集中管理**。DM 支持上千个节点的集群规模，可同时运行并集中管理大量数据迁移同步任务。
- **高可用**。支持迁移任务在不同节点自由调度，少量工作节点宕机并不会影响进行中的任务。

DM 使用说明

第 1 步：创建数据源

首先，新建数据源配置文件 source1.yaml：

唯一标识符，确保不重复使用。

```
source_id: "mysql-02"
```

DM-worker 是否启用全局事务标识符 (GTID) 以拉取 binlog。需确保上游 MySQL 已开启 GTID 模式，且在主从切换时使用。

```
use_gtid: true
```

```
source:
```

```
address: "${host}"      # 示例: 192.168.1.100
username: "admin"
passwd: "${password}"  # 不推荐使用明文密码, 建议通过 dmctl encrypt 加密后使用
connection_port: 3306
```

其次, dmctl 执行 `operate-source create source1.yaml` 将数据源配置加载到 DM 集群中。

第 2 步: 创建迁移任务

新建迁移任务配置文件 `sample_task.yaml`:

任务名称, 确保同时运行的任务不重名。

```
task_name: "sample_task"
```

任务类型, 可选值:

full: 仅执行全量数据迁移

incremental: 实时同步 binlog

all: 综合全量与 binlog 迁移

```
mode: "all"
```

下游数据库配置。

```
destination_database:
```

```
ip: "${host}"          # 示例: 192.168.1.101
```

```
port: 4000
```

```
username: "admin"
```

```
passwd: "${password}"  # 不推荐使用明文密码, 建议通过 dmctl encrypt 加密后使用
```

当前数据迁移任务所需的所有上游 MySQL 实例配置。

```
mysql_instances:
```

```
-
```

上游实例或复制组的标识。

```
instance_id: "mysql-02"
```

迁移库名或表名的黑白名单配置项, 用于引用全局黑白名单设置, 详见下方的 `block_allow_list` 配置。

```
block_allow_list: "listB"
```

黑白名单全局配置，各实例通过配置项名进行引用。

block_allow_list:

listB: *# 名称*

allow_tables: *# 需要迁移的上游表的白名单。*

- database_name: "example_db" *# 需要迁移的表的数据库名。*

table_name: "example_table" *# 需要迁移的表的名称。*

第 3 步：启动任务

在你启动数据迁移任务之前，建议使用 dmctl 执行 `check-task sample_task.yaml` 命令检查配置是否符合 DM 的配置要求，以避免后期报错。

然后使用 dmctl 执行 `start-task sample_task.yaml` 执行以下命令启动数据迁移任务。

第 4 步：查看任务状态

使用 dmctl 执行 `query-status` 命令，了解正在运行的迁移任务及状态等信息。