30.MySQL主从架构之上的新型复制架构-MSR,MGR×

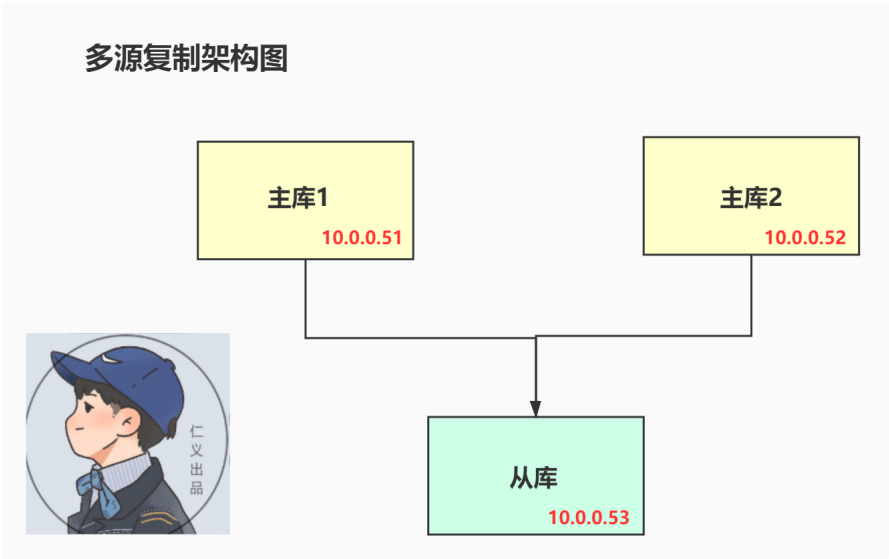
1.多源复制 MSR（Multi-Source-Repliction)-5.7版本出现

比较适合于OLAP(在线分析处理场景)，传统的场景是OLTP(在线事务处理场景)

1.1 介绍

1.2 MSR架构节点规划

主机角色	IP地址	端口
Master1	10.0.0.51	3306
Master2	10.0.0.52	3306
Slave	10.0.0.53	3306



1.3 MSR架构搭建过程

1.3.1 首先准备干净的gtid模式主从

Bash | Copy

```
# 三台虚拟机
10.0.0.51  db01
10.0.0.52  db02
10.0.0.53  db03
防火墙关闭

#清理环境:
pkill mysqld
rm -rf /data/3306/*
mkdir -p /data/3306/data /data/3306/binlog
chown -R mysql.mysql /data/*


# 准备配置文件
主库db01:
mv /etc/my.cnf /tmp
cat > /etc/my.cnf <<EOF
[mysqld]
basedir=/usr/local/mysql
datadir=/data/3306/data
socket=/tmp/mysql.sock
server_id=51
port=3306
secure-file-priv=/tmp
log_bin=/data/3306/binlog/mysql-bin
binlog_format=row
gtid-mode=on
enforce-gtid-consistency=true
log-slave-updates=1
[mysql]
prompt=db01 [\\d]>
EOF

slave1(db02):
mv /etc/my.cnf /tmp
cat > /etc/my.cnf <<EOF
[mysqld]
basedir=/usr/local/mysql
datadir=/data/3306/data
socket=/tmp/mysql.sock
server_id=52
port=3306
secure-file-priv=/tmp
log_bin=/data/3306/binlog/mysql-bin
binlog_format=row
gtid-mode=on
enforce-gtid-consistency=true
log-slave-updates=1
[mysql]
prompt=db02 [\\d]>
```

```
EOF

slave2(db03):
mv /etc/my.cnf /tmp
cat > /etc/my.cnf <<EOF
[mysqld]
basedir=/usr/local/mysql
datadir=/data/3306/data
socket=/tmp/mysql.sock
server_id=53
port=3306
secure-file-priv=/tmp
log_bin=/data/3306/binlog/mysql-bin
binlog_format=row
gtid-mode=on
enforce-gtid-consistency=true

log-slave-updates=1
[mysql]
prompt=db03 [\\d]>
EOF

初始化数据
mysqld --initialize-insecure --user=mysql --basedir=/usr/local/mysql --datadir=/data/3306/data

启动数据库
/etc/init.d/mysqld start

```

1.3.2 开始搭建MSR架构

a.两个主库(51,52)创建复制用户

Bash | Copy

```
为了避免从库不把创建复制用户的操作复制过去，所以开启不记录日志
set sql_log_bin=0;
create user repl@'10.0.0.%' identified with mysql_native_password by '123';
grant replication slave on *.*  to repl@'10.0.0.%' ;
set sql_log_bin=1;
```

b.从库(53)上配置多主库的信息（配置多源复制）

Bash | Copy

```
通过for channel命令 分别设置两个主库对应两个传输通道，为了区分开来。
CHANGE MASTER TO MASTER_HOST='10.0.0.51',MASTER_USER='repl', MASTER_PASSWORD='123', MASTER_AUTO_POSITION=1 FOR

CHANGE MASTER TO MASTER_HOST='10.0.0.52',MASTER_USER='repl', MASTER_PASSWORD='123', MASTER_AUTO_POSITION=1 FOR
```

c.从库（53）开启主从复制线程

```bash
start slave for CHANNEL  'Master_1';
start slave for CHANNEL  'Master_2';
```

1.3.3 MSR架构的监控

Bash | Copy

```
第一种监控方法（粗略）
SHOW SLAVE STATUS FOR CHANNEL 'Master_1'\G;
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Channel_Name: master_1

SHOW SLAVE STATUS FOR CHANNEL 'Master_2'\G;
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Channel_Name: master_2
第二种监控方法（详细）
1.db03 [(none)]>select * from performance_schema.replication_connection_configuration\G
*************************** 1. row ***************************
                 CHANNEL_NAME: master_1
                         HOST: 10.0.0.51
                         PORT: 3306
                         USER: repl
            NETWORK_INTERFACE:
                AUTO_POSITION: 1
                  SSL_ALLOWED: NO
                  SSL_CA_FILE:
                  SSL_CA_PATH:
              SSL_CERTIFICATE:
                   SSL_CIPHER:
                      SSL_KEY:
SSL_VERIFY_SERVER_CERTIFICATE: NO
                  SSL_CRL_FILE:
                  SSL_CRL_PATH:
    CONNECTION_RETRY_INTERVAL: 60
       CONNECTION_RETRY_COUNT: 86400
           HEARTBEAT_INTERVAL: 30.000
                  TLS_VERSION:
              PUBLIC_KEY_PATH:
               GET_PUBLIC_KEY: NO
            NETWORK_NAMESPACE:
        COMPRESSION_ALGORITHM: uncompressed
       ZSTD_COMPRESSION_LEVEL: 3
             TLS_CIPHERSUITES: NULL
*************************** 2. row ***************************
                 CHANNEL_NAME: master_2
                         HOST: 10.0.0.52
                         PORT: 3306
                         USER: repl
            NETWORK_INTERFACE:
                AUTO_POSITION: 1
                  SSL_ALLOWED: NO
                  SSL_CA_FILE:
                  SSL_CA_PATH:
              SSL_CERTIFICATE:
                   SSL_CIPHER:
```

```
                      SSL_KEY:
SSL_VERIFY_SERVER_CERTIFICATE: NO
                  SSL_CRL_FILE:
                  SSL_CRL_PATH:
     CONNECTION_RETRY_INTERVAL: 60
        CONNECTION_RETRY_COUNT: 86400
            HEARTBEAT_INTERVAL: 30.000
                   TLS_VERSION:
               PUBLIC_KEY_PATH:
                GET_PUBLIC_KEY: NO
             NETWORK_NAMESPACE:
         COMPRESSION_ALGORITHM: uncompressed
        ZSTD_COMPRESSION_LEVEL: 3
              TLS_CIPHERSUITES: NULL

2.db03 [(none)]>SELECT * FROM performance_schema.replication_connection_status WHERE CHANNEL_NAME='master
*************************** 1. row ***************************
                                    CHANNEL_NAME: master_1
                                      GROUP_NAME:
                                     SOURCE_UUID: e8267cfb-b62d-11eb-a116-000c29edc386
                                       THREAD_ID: 49
                                   SERVICE_STATE: ON
                           COUNT_RECEIVED_HEARTBEATS: 10
                            LAST_HEARTBEAT_TIMESTAMP: 2021-05-16 18:20:50.992558
                            RECEIVED_TRANSACTION_SET:
                                  LAST_ERROR_NUMBER: 0
                                 LAST_ERROR_MESSAGE:
                               LAST_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
                             LAST_QUEUED_TRANSACTION:
  LAST_QUEUED_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
 LAST_QUEUED_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
      LAST_QUEUED_TRANSACTION_START_QUEUE_TIMESTAMP: 0000-00-00 00:00:00.000000
        LAST_QUEUED_TRANSACTION_END_QUEUE_TIMESTAMP: 0000-00-00 00:00:00.000000
                               QUEUEING_TRANSACTION:
     QUEUEING_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
    QUEUEING_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
         QUEUEING_TRANSACTION_START_QUEUE_TIMESTAMP: 0000-00-00 00:00:00.000000

db03 [(none)]>select * from performance_schema.replication_applier_status_by_worker\G;
*************************** 1. row ***************************
                                       CHANNEL_NAME: master_1
                                          WORKER_ID: 0
                                          THREAD_ID: 50
                                      SERVICE_STATE: ON
                                  LAST_ERROR_NUMBER: 0
                                 LAST_ERROR_MESSAGE:
                               LAST_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
                             LAST_APPLIED_TRANSACTION:
   LAST_APPLIED_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
  LAST_APPLIED_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
       LAST_APPLIED_TRANSACTION_START_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
         LAST_APPLIED_TRANSACTION_END_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
                                   APPLYING_TRANSACTION:
```

```
          APPLYING_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
         APPLYING_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
              APPLYING_TRANSACTION_START_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
                  LAST_APPLIED_TRANSACTION_RETRIES_COUNT: 0
    LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_NUMBER: 0
   LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_MESSAGE:
 LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
                      APPLYING_TRANSACTION_RETRIES_COUNT: 0
        APPLYING_TRANSACTION_LAST_TRANSIENT_ERROR_NUMBER: 0
       APPLYING_TRANSACTION_LAST_TRANSIENT_ERROR_MESSAGE:
     APPLYING_TRANSACTION_LAST_TRANSIENT_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
*************************** 2. row ***************************
                                       CHANNEL_NAME: master_2
                                          WORKER_ID: 0
                                          THREAD_ID: 52
                                      SERVICE_STATE: ON
                                  LAST_ERROR_NUMBER: 0
                                 LAST_ERROR_MESSAGE:
                               LAST_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
                            LAST_APPLIED_TRANSACTION:
      LAST_APPLIED_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
     LAST_APPLIED_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
          LAST_APPLIED_TRANSACTION_START_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
            LAST_APPLIED_TRANSACTION_END_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
                                 APPLYING_TRANSACTION:
          APPLYING_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
         APPLYING_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
              APPLYING_TRANSACTION_START_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
                  LAST_APPLIED_TRANSACTION_RETRIES_COUNT: 0
    LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_NUMBER: 0
   LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_MESSAGE:
 LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
                      APPLYING_TRANSACTION_RETRIES_COUNT: 0
        APPLYING_TRANSACTION_LAST_TRANSIENT_ERROR_NUMBER: 0
```

1.2.4 MSR架构配置过滤复制

Bash | Copy

```
mysql> CHANGE REPLICATION FILTER REPLICATE_WILD_DO_TABLE = ('db1.%') FOR CHANNEL "master_1";
mysql> CHANGE REPLICATION FILTER REPLICATE_WILD_DO_TABLE = ('db2.%') FOR CHANNEL "master_2";
```

<https://service.weibo.com/share/share.php?url=https%3A
Repliction)-5.7%E7%89%88%E6%9C%AC%E5%87%BA%E7%8E%B0%E6%AF%94%E8%BE%83%E9%80%82%E5%90%88%E4%BA%8EOLAP(%E5%9C%A8%E7%BA%BF%E5%88%86%E6%9E%90%E5%A4%