

30.MySQL主从架构之上的新型复制架构-MSR,MGRx

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

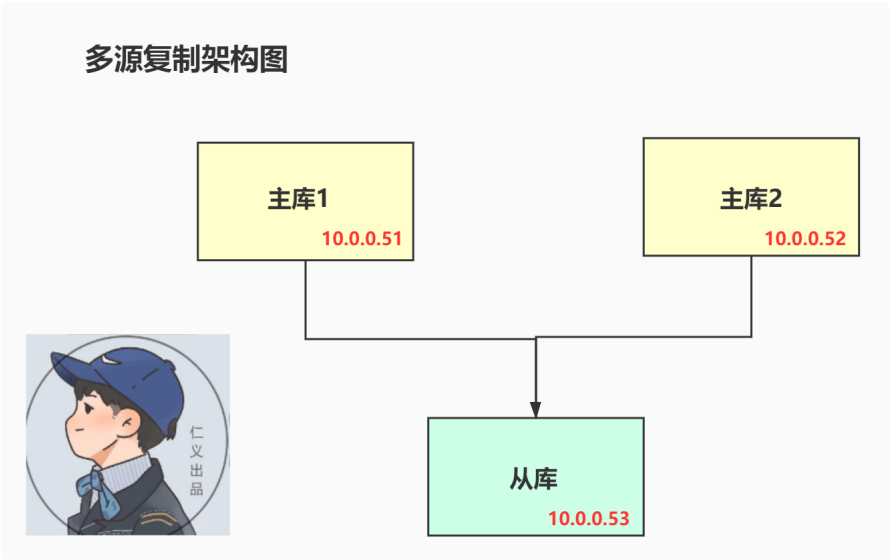
比较适合于OLAP(在线分析处理场景)，传统的场景是OLTP(在线事务处理场景)

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

```
1 # 三台虚拟机
2 10.0.0.51 db01
3 10.0.0.52 db02
4 10.0.0.53 db03
5 防火墙关闭
6
7 #清理环境:
8 pkill mysqld
9 rm -rf /data/3306/*
10 mkdir -p /data/3306/data /data/3306/binlog
11 chown -R mysql:mysql /data/*
12
13
14 # 准备配置文件
15 主库db01:
16 mv /etc/my.cnf /tmp
17 cat > /etc/my.cnf <<EOF
18 [mysqld]
19 basedir=/usr/local/mysql
20 datadir=/data/3306/data
21 socket=/tmp/mysql.sock
22 server_id=51
23 port=3306
24 secure-file-priv=/tmp
25 log_bin=/data/3306/binlog/mysql-bin
26 binlog_format=row
27 gtid-mode=on
28 enforce-gtid-consistency=true
29 log-slave-updates=1
30 [mysql]
31 prompt=db01 [\\d]>
32 EOF
33
34 slave1(db02):
35 mv /etc/my.cnf /tmp
36 cat > /etc/my.cnf <<EOF
37 [mysqld]
38 basedir=/usr/local/mysql
39 datadir=/data/3306/data
40 socket=/tmp/mysql.sock
41 server_id=52
42 port=3306
43 secure-file-priv=/tmp
44 log_bin=/data/3306/binlog/mysql-bin
45 binlog_format=row
46 gtid-mode=on
47 enforce-gtid-consistency=true
48 log-slave-updates=1
49 [mysql]
50 prompt=db02 [\\d]>
```

```
51 EOF
52
53 slave2(db03):
54 mv /etc/my.cnf /tmp
55 cat > /etc/my.cnf <<EOF
56 [mysqld]
57 basedir=/usr/local/mysql
58 datadir=/data/3306/data
59 socket=/tmp/mysql.sock
60 server_id=53
61 port=3306
62 secure-file-priv=/tmp
63 log_bin=/data/3306/binlog/mysql-bin
64 binlog_format=row
65 gtid-mode=on
66 enforce-gtid-consistency=true
67
68 log-slave-updates=1
69 [mysql]
70 prompt=db03 [\\d]>
71 EOF
72
73 初始化数据
74 mysqld --initialize-insecure --user=mysql --basedir=/usr/local/mysql --datadir=/data/3306/data
75
76 启动数据库
77 /etc/init.d/mysqld start
78
```

1.3.2 开始搭建MSR架构

a.两个主库(51,52)创建复制用户

```
1 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
2 set sql_log_bin=0;
3 create user repl@'10.0.0.%' identified with mysql_native_password by '123';
4 grant replication slave on *.* to repl@'10.0.0.%' ;
5 set sql_log_bin=1;
```

b.从库(53)上配置多主库的信息（配置多源复制）

```
1 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。
2 CHANGE MASTER TO MASTER_HOST='10.0.0.51',MASTER_USER='repl', MASTER_PASSWORD='123', MASTER_AUTO_POSITION=1 FOR
3
4 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 |  Copy

```
1  start slave for CHANNEL 'Master_1';  
2  start slave for CHANNEL 'Master_2';
```

1.3.3 MSR架构的监控

Bash | Copy

```

1  第一种监控方法（粗略）
2  SHOW SLAVE STATUS FOR CHANNEL 'Master_1'\G;
3  Slave_IO_Running: Yes
4  Slave_SQL_Running: Yes
5  Channel_Name: master_1
6
7  SHOW SLAVE STATUS FOR CHANNEL 'Master_2'\G;
8  Slave_IO_Running: Yes
9  Slave_SQL_Running: Yes
10 Channel_Name: master_2
11 第二种监控方法（详细）
12 1.db03 [(none)]>select * from performance_schema.replication_connection_configuration\G
13 ***** 1. row *****
14         CHANNEL_NAME: master_1
15         HOST: 10.0.0.51
16         PORT: 3306
17         USER: repl
18         NETWORK_INTERFACE:
19         AUTO_POSITION: 1
20         SSL_ALLOWED: NO
21         SSL_CA_FILE:
22         SSL_CA_PATH:
23         SSL_CERTIFICATE:
24         SSL_CIPHER:
25         SSL_KEY:
26 SSL_VERIFY_SERVER_CERTIFICATE: NO
27         SSL_CRL_FILE:
28         SSL_CRL_PATH:
29 CONNECTION_RETRY_INTERVAL: 60
30 CONNECTION_RETRY_COUNT: 86400
31 HEARTBEAT_INTERVAL: 30.000
32 TLS_VERSION:
33 PUBLIC_KEY_PATH:
34 GET_PUBLIC_KEY: NO
35 NETWORK_NAMESPACE:
36 COMPRESSION_ALGORITHM: uncompressed
37 ZSTD_COMPRESSION_LEVEL: 3
38 TLS_CIPHERSUITES: NULL
39 ***** 2. row *****
40         CHANNEL_NAME: master_2
41         HOST: 10.0.0.52
42         PORT: 3306
43         USER: repl
44         NETWORK_INTERFACE:
45         AUTO_POSITION: 1
46         SSL_ALLOWED: NO
47         SSL_CA_FILE:
48         SSL_CA_PATH:
49         SSL_CERTIFICATE:
50         SSL_CIPHER:

```

```

51         SSL_KEY:
52     SSL_VERIFY_SERVER_CERTIFICATE: NO
53         SSL_CRL_FILE:
54         SSL_CRL_PATH:
55     CONNECTION_RETRY_INTERVAL: 60
56     CONNECTION_RETRY_COUNT: 86400
57     HEARTBEAT_INTERVAL: 30.000
58     TLS_VERSION:
59     PUBLIC_KEY_PATH:
60     GET_PUBLIC_KEY: NO
61     NETWORK_NAMESPACE:
62     COMPRESSION_ALGORITHM: uncompressed
63     ZSTD_COMPRESSION_LEVEL: 3
64     TLS_CIPHERSUITES: NULL
65
66 2.db03 [(none)]>SELECT * FROM performance_schema.replication_connection_status WHERE CHANNEL_NAME='master
67 ***** 1. row *****
68             CHANNEL_NAME: master_1
69             GROUP_NAME:
70             SOURCE_UUID: e8267cfb-b62d-11eb-a116-000c29edc386
71             THREAD_ID: 49
72             SERVICE_STATE: ON
73             COUNT_RECEIVED_HEARTBEATS: 10
74             LAST_HEARTBEAT_TIMESTAMP: 2021-05-16 18:20:50.992558
75             RECEIVED_TRANSACTION_SET:
76             LAST_ERROR_NUMBER: 0
77             LAST_ERROR_MESSAGE:
78             LAST_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
79             LAST_QUEUED_TRANSACTION:
80     LAST_QUEUED_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
81     LAST_QUEUED_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
82     LAST_QUEUED_TRANSACTION_START_QUEUE_TIMESTAMP: 0000-00-00 00:00:00.000000
83     LAST_QUEUED_TRANSACTION_END_QUEUE_TIMESTAMP: 0000-00-00 00:00:00.000000
84     QUEUEING_TRANSACTION:
85     QUEUEING_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
86     QUEUEING_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
87     QUEUEING_TRANSACTION_START_QUEUE_TIMESTAMP: 0000-00-00 00:00:00.000000
88
89 db03 [(none)]>select * from performance_schema.replication_applier_status_by_worker\G;
90 ***** 1. row *****
91             CHANNEL_NAME: master_1
92             WORKER_ID: 0
93             THREAD_ID: 50
94             SERVICE_STATE: ON
95             LAST_ERROR_NUMBER: 0
96             LAST_ERROR_MESSAGE:
97             LAST_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
98             LAST_APPLIED_TRANSACTION:
99     LAST_APPLIED_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
100    LAST_APPLIED_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
101    LAST_APPLIED_TRANSACTION_START_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
102    LAST_APPLIED_TRANSACTION_END_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
103    APPLYING_TRANSACTION:

```

```

104     APPLYING_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
105     APPLYING_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
106     APPLYING_TRANSACTION_START_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
107     LAST_APPLIED_TRANSACTION_RETRIES_COUNT: 0
108     LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_NUMBER: 0
109     LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_MESSAGE:
110     LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
111     APPLYING_TRANSACTION_RETRIES_COUNT: 0
112     APPLYING_TRANSACTION_LAST_TRANSIENT_ERROR_NUMBER: 0
113     APPLYING_TRANSACTION_LAST_TRANSIENT_ERROR_MESSAGE:
114     APPLYING_TRANSACTION_LAST_TRANSIENT_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
115     ***** 2. row *****
116     CHANNEL_NAME: master_2
117     WORKER_ID: 0
118     THREAD_ID: 52
119     SERVICE_STATE: ON
120     LAST_ERROR_NUMBER: 0
121     LAST_ERROR_MESSAGE:
122     LAST_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
123     LAST_APPLIED_TRANSACTION:
124     LAST_APPLIED_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
125     LAST_APPLIED_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
126     LAST_APPLIED_TRANSACTION_START_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
127     LAST_APPLIED_TRANSACTION_END_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
128     APPLYING_TRANSACTION:
129     APPLYING_TRANSACTION_ORIGINAL_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
130     APPLYING_TRANSACTION_IMMEDIATE_COMMIT_TIMESTAMP: 0000-00-00 00:00:00.000000
131     APPLYING_TRANSACTION_START_APPLY_TIMESTAMP: 0000-00-00 00:00:00.000000
132     LAST_APPLIED_TRANSACTION_RETRIES_COUNT: 0
133     LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_NUMBER: 0
134     LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_MESSAGE:
135     LAST_APPLIED_TRANSACTION_LAST_TRANSIENT_ERROR_TIMESTAMP: 0000-00-00 00:00:00.000000
136     APPLYING_TRANSACTION_RETRIES_COUNT: 0
137     APPLYING_TRANSACTION_LAST_TRANSIENT_ERROR_NUMBER: 0

```

4.2.4 MGR架构配置过滤复制

Bash | Copy

```

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

```

<<https://service.weibo.com/share/share.php?url=https%3A>

Replication)-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%

