

(/apps/redi  
utm\_sourc  
banner-clic

# MySQL主从配置详解



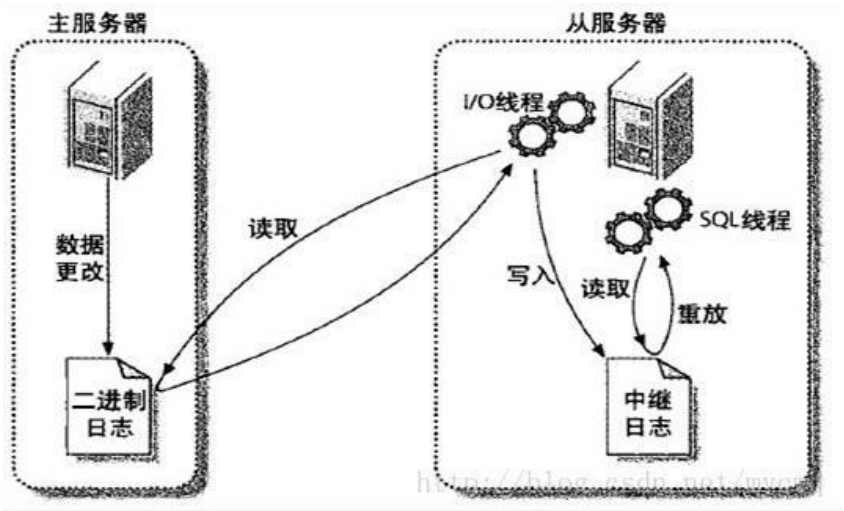
就没一个昵称能用 (/u/6610ec6cda2e) [+ 关注](#)

2017.08.28 11:55\* 字数 618 阅读 546 评论 2 喜欢 12

(/u/6610ec6cda2e)

## MySQL主从复制原理

主服务器数据库的每次操作都会记录在其二进制文件mysql-bin.xxx（该文件可以在mysql目录下的data目录中看到）中，从服务器的I/O线程使用专用账号登录到主服务器中读取该二进制文件，并将文件内容写入到自己本地的中继日志relay-log文件中，然后从服务器的SQL线程会根据中继日志中的内容执行SQL语句



## MySQL主从同步的作用

- 1、可以作为备份机制，相当于热备份
- 2、可以用来做读写分离，均衡数据库负载

## 项目场景

- 1、主服务器10.10.20.111，其中已经有数据库且库中有表、函数以及存储过程
- 2、从服务器10.10.20.116，空的啥也没有

## 准备工作

主从服务器需要有相同的初态

- 1、将主服务器要同步的数据库枷锁，避免同步时数据发生改变

```
mysql>use db;  
mysql>flush tables with read lock;
```



- 2、将主服务器数据库中数据导出

```
mysql>mysqldump -uroot -pxxxx db > db.sql;
```

这个命令是导出数据库中所有表结构和数据，如果要导出函数和存储过程的话使用

```
mysql>mysqldump -R -ndt db -uroot -pxxxx > db.sql
```

其他关于mysql导入导出命令的戳这里 (<http://www.cnblogs.com/chevin/p/5683281.html>)

3、备份完成后，解锁主服务器数据库

```
mysql>unlock tables;
```

4、将初始数据导入从服务器数据库

```
mysql>create database db;
mysql>use db;
mysql>source db.sql;
```

好了，现在主从服务器拥有一样的初态了

## 主服务器配置

1、修改MySQL配置

```
vi /etc/my.cnf
```

在[mysqld]中添加

```
#主数据库端ID号
server_id = 1
#开启二进制日志
log-bin = mysql-bin
#需要复制的数据库名，如果复制多个数据库，重复设置这个选项即可
binlog-do-db = db
#将从服务器从主服务器收到的更新记入到从服务器自己的二进制日志文件中
log-slave-updates
#控制binlog的写入频率。每执行多少次事务写入一次(这个参数性能消耗很大，但可减小MySQL崩溃造成的损失)
sync_binlog = 1
#这个参数一般用在主主同步中，用来错开自增值，防止键值冲突
auto_increment_offset = 1
#这个参数一般用在主主同步中，用来错开自增值，防止键值冲突
auto_increment_increment = 1
#二进制日志自动删除的天数，默认值为0，表示“没有自动删除”，启动时和二进制日志循环时可能删除
expire_logs_days = 7
#将函数复制到slave
log_bin_trust_function_creators = 1
```

2、重启MySQL，创建允许从服务器同步数据的账户

```
#创建slave账号account，密码123456
mysql>grant replication slave on *.* to 'account'@'10.10.20.116' identified by '123456';
#更新数据库权限
mysql>flush privileges;
```

(/apps/redi  
utm\_sourc  
banner-clic



### 3、查看主服务器状态

```
mysql>show master status\G;
***** 1. row *****
                File: mysql-bin.000033 #当前记录的日志
            Position: 337523 #日志中记录的位置
        Binlog_Do_DB:
        Binlog_Ignore_DB:
```

(/apps/redi  
utm\_sourc  
banner-cl

执行完这个步骤后不要再操作主服务器数据库了，防止其状态值发生变化

## 从服务器配置

### 1、修改MySQL配置

```
vi /etc/my.cnf
```

在[mysqld]中添加

```
server_id = 2
log-bin = mysql-bin
log-slave-updates
sync_binlog = 0
#log buffer将每秒一次地写入log file中，并且log file的flush(刷到磁盘)操作同时进行。该模式下在事
innodb_flush_log_at_trx_commit = 0
#指定slave要复制哪个库
replicate-do-db = db
#MySQL主从复制的时候，当Master和Slave之间的网络中断，但是Master和Slave无法察觉的情况下（比如防
slave-net-timeout = 60
log_bin_trust_function_creators = 1
```

### 2、执行同步命令

```
#执行同步命令，设置主服务器ip，同步账号密码，同步位置
mysql>change master to master_host='10.10.20.111',master_user='account',master_passw
#开启同步功能
mysql>start slave;
```

### 3、查看从服务器状态

```
mysql>show slave status\G;
***** 1. row *****
                Slave_IO_State: Waiting for master to send event
            Master_Host: 10.10.20.111
            Master_User: account
            Master_Port: 3306
            Connect_Retry: 60
            Master_Log_File: mysql-bin.000033
        Read_Master_Log_Pos: 337523
            Relay_Log_File: db2-relay-bin.000002
            Relay_Log_Pos: 337686
        Relay_Master_Log_File: mysql-bin.000033
            Slave_IO_Running: Yes
            Slave_SQL_Running: Yes
            Replicate_Do_DB:
            Replicate_Ignore_DB:
            ...
```

