

京峰教育 MYSQL 主从实战

3.7.1 MySQL 数据库集群拓展

MySQL 是一个开放源码的小型关联式数据库管理系统，开发者为瑞典 MySQL AB 公司，目前属于 Oracle 公司，MySQL 被广泛地应用在 Internet 上的中小型网站中。由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，许多中小型网站为了降低网站总体拥有成本而选择了 MySQL 作为网站数据库。

对应目前主流的 LAMP 架构来说，Mysql 更是得到各位 IT 运维、DBA 的青睐，目前 mysql 已被 Oracle 收购，不过好消息是原来 mysql 创始人已独立出来自己重新开发了一个 MariaDB，而且使用的人数越来越多。而且 MariaDB 兼容 mysql 所有的功能和相关参数。

Mysql 常用的两大引擎有 MyISAM 和 InnoDB，那他们有什么明显的区别呢，什么场合使用什么引擎呢？

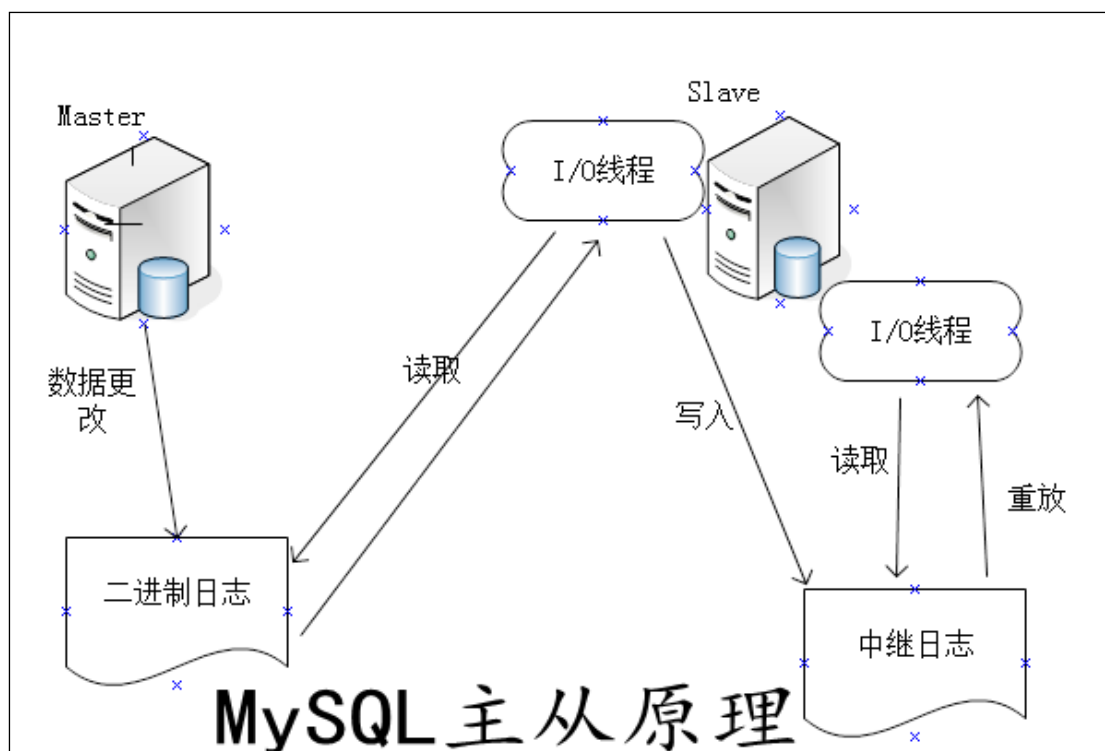
MyISAM 类型的表强调的是性能，其执行速度比 InnoDB 类型更快，但不提供事务支持，如果执行大量的 SELECT 操作，MyISAM 是更好的选择，支持表锁。

InnoDB 提供事务支持事务，外部键等高级数据库功能，执行大量的 INSERT 或 UPDATE，出于性能方面的考虑，应该使用 InnoDB 表，支持行锁。

随着访问量的不断增加，Mysql 数据库压力不断增加，需要对 mysql 进行优化和架构改造，可以使用高可用、主从复制、读写分离来、拆分库、拆分表进行优化。下面我们来学

MySQL 主从复制

高可用如何来实现。



3.7.2 MySQL 主从复制原理剖析

Mysql 主从同步其实是一个异步复制的过程，要实现复制首先需要在 master 上开启 bin-log 日志功能，整个过程需要开启 3 个线程，分别是 Master 开启 IO 线程，slave 开启 IO 线程和 SQL 线程。

- 在从服务器执行 slave start，从服务器上 IO 线程会通过授权的用户连接上 master，并请求 master 从指定的文件和位置之后发送 bin-log 日志内容。**
- Master 服务器接收到来自 slave 服务器的 IO 线程的请求后，master 服务器上的 IO 线程根据 slave 服务器发送的指定 bin-log 日志之后的内容 然后返回给 slave 端的 IO 线程。（返回的信息中除了 bin-log 日志内容外，还有本次返回日志内容后在 master

服务器端的新的 binlog 文件名以及在 binlog 中的下一个指定更新位置。)

- c) Slave 的 IO 线程接收到信息后，将接收到的日志内容依次添加到 Slave 端的

relay-log 文件的最末端，并将读取到的 Master 端的 bin-log 的文件

名和位置记录到 master.info 文件中，以便在下一次读取的时候能够清楚的告诉 Master

“我需要从某个 bin-log 的哪个位置开始往后的日志内容，请发给我”；

- d) Slave 的 Sql 线程检测到 **relay-log 中新增加了内容后**，会马上解析 relay-log 的内容

成为在 Master 端真实执行时候的那些可执行的内容，并在自身执行。

3.7.3 MySQL 主从复制架构实战

环境准备：192.168.1.103 为 master 主服务器，192.168.33.11 为 slave 从服务器。

在主和从服务器都安装 mysql 相关软件，命令如下：

```
yum install -y mysql mysql-devel mysql-server mysql-libs
```

安装完毕后，在 Master 修改 vi /etc/my.cnf 内容为如下：

```
[mysqld]
```

```
datadir=/data/mysql
```

```
socket=/var/lib/mysql/mysql.sock
```

```
user=mysql
```

```
# Disabling symbolic-links is recommended to prevent assorted security risks
```

```
symbolic-links=0
```

```
log-bin=mysql-bin
```

```
server-id = 1
```

```
auto_increment_offset=1
```

auto_increment_increment=2

[mysqld_safe]

log-error=/var/log/mysql.log

pid-file=/var/run/mysql/mysql.pid

replicate-do-db =jfedu

创建/data/mysql 数据目录，mkdir -p /data/mysql ;chown -R mysql:mysql

/data/mysql

启动 mysql 即可，service mysql restart

然后修改 slave Mysql 数据库 my.cnf 配置文件内容如下：

[mysqld]

datadir=/data/mysql

socket=/var/lib/mysql/mysql.sock

user=mysql

Disabling symbolic-links is recommended to prevent assorted security risks

symbolic-links=0

server-id = 2

auto_increment_offset=2

auto_increment_increment=2

[mysqld_safe]

log-error=/var/log/mysql.log

pid-file=/var/run/mysql/mysql.pid

在 Master 数据库服务器上设置权限，执行如下命令：

```
grant replication slave on *.* to 'tongbu'@'%' identified by
'123456';
```

在 Master 数据库执行如下命令：

```
mysql> show master status;
+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000006 |      98 |              |                  |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

然后在 slave 服务器指定 master IP 和同步的 pos 点：

change master to

```
master_host='192.168.1.103',master_user='tongbu',master_password='123456
```

```
',master_log_file='mysql-bin.000006',master_log_pos=98;
```

在 slave 启动 slave start，并执行 show slave status\G 查看 Mysql 主从状态：

```
[root@node2 ~]# mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.0.95-log Source distribution

Copyright (c) 2000, 2011, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> slave start;
Query OK, 0 rows affected (0.00 sec)

mysql> show slave status\G
***** 1. row *****
      Slave_IO_State: Waiting for master to send event
        Master_Host: 192.168.149.128
        Master_User: tongbu
        Master_Port: 3306
        Connect_Retry: 60
        Master_Log_File: mysql-bin.000006
        Read_Master_Log_Pos: 98
        Relay_Log_File: mysqld-relay-bin.000004
        Relay_Log_Pos: 235
        Relay_Master_Log_File: mysql-bin.000006
        Slave_IO_Running: Yes
        Slave_SQL_Running: Yes
        Replicate_Do_DB:
```

Slave_IO_Running: Yes

Slave_SQL_Running: Yes 两个状态为 YES，代表 slave 已经启动两个线程，一个为 IO

线程，一个为 SQL 线程。

然后在 Master 服务器创建一个数据库和表，命令如下：

```
192.168.149.128 x 192.168.149.129
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database mysql_ab_test charset=utf8;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| mysql_ab_test |
| test |
+-----+
4 rows in set (0.00 sec)

mysql> use mysql_ab_test;
Database changed
mysql>
mysql> create table t0 (id varchar(20),name varchar(30));
Query OK, 0 rows affected (0.00 sec)

mysql> show tables;
+-----+
| Tables_in_mysql_ab_test |
+-----+
| t0 |
+-----+
1 row in set (0.00 sec)
```

然后去 slave 服务器查看是否有 mysql_ab_test 数据库和相应 t0 的表 如果存在则代表

Mysql 主从同步搭建成功：

```
192.168.149.128 192.168.149.129 x
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| mysql_ab_test |
| test |
+-----+
4 rows in set (0.00 sec)

mysql> use mysql_ab_test;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_mysql_ab_test |
+-----+
| t0 |
+-----+
1 row in set (0.00 sec)
```

同样还可以测试在 master 服务器插入两条数据,在 slave 查看 insert 数据是否已同步:

128 master 上执行如下图:

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use mysql_ab_test;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> insert into t0 values ("001","wugk1");
Query OK, 1 row affected (0.00 sec)

mysql> insert into t0 values ("002","wugk2");
Query OK, 1 row affected (0.00 sec)

mysql> select * from t0;
+-----+-----+
| id    | name  |
+-----+-----+
| 001   | wugk1 |
| 002   | wugk2 |
+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

129 slave 上执行如下图,在 master 插入的数据已经同步到 slave 上:

```
192.168.149.128 192.168.149.129 x
[root@node2 ~]# mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 5.0.95-log Source distribution

Copyright (c) 2000, 2011, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use mysql_ab_test;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from t0;
+-----+-----+
| id    | name  |
+-----+-----+
| 001   | wugk1 |
| 002   | wugk2 |
+-----+-----+
2 rows in set (0.00 sec)
```

自此 Mysql 主从搭建完毕,现在有一个问题,如果 master 服务器 down 机了,如何快速恢复服务呢?

可以通过两种方法:

第一种方法，如果程序连接的是 master 的 IP，直接在 slave 服务器上添加 master 的 IP 即可。这个手动去操作，而且需要花费时间比较长，可能还会出现误操作的情况，不推荐。

第二种方法，可以使用 keepalived、heartbeat 作为 HA 检测软件，检查 MySQL 服务是否正常，不正常则自动切换到 slave 上，推荐使用。

3.7.4 MySQL 主从注意事项

mysql 主从同步的原理：

- 1、在 master 上开启 bin-log 日志功能，记录更新、插入、删除的语句。
- 2、必须开启三个线程，主上开启 io 线程，从上开启 io 线程和 sql 线程。
- 3、从上 io 线程去连接 master，master 通过 io 线程检查有 slave 过来的请求，请求日志、position 位置。
- 4、master 将这些相应的日志返回给 slave，slave 自己去下载到本地的 relay_log 里面，写入一个 master-info
- 日志记录同步的点。
- 5、slave 的 sql 线程检查到 relay-log 日志有更新，然后在本地去 exec 执行。
- 6、主从同步是属于异步方式。

20133

```
insert into table values ('wgkgood','wgkgood@163.com');
```

20134

```
insert into t1 values ('Linux 系统安装方法', 'dsfjsdklsdfjsdlk');
```

20135

change master to

```
master_host='192.168.1.12',master_user='tongbu',master_password='123456',master_log_file='mysql-bin.000001',master_log_pos=272;
```

二、Mysql 主从同步，master 突然 down 机，如何恢复：

1、slave 数据库必须启动，在 slave 上授权网站 IP 对数据库的访问权限。

2、修改网站服务器 config 目录下:

config_global.php config_ucenter.php 把原先 master ip 改成 slave ip 地址。

3、重启 httpd 服务，切换成功。

作业：

1、在 master 上已经创建好了 discuz 数据库，同时有论坛，新建从，怎么同步数据？

mysqldump > mysql.sql ; 导入进去，再同步。

3.7.5 MySQL 主从同步故障解决方案

方法一：忽略错误后，继续同步

该方法适用于主从库数据相差不大，或者要求数据可以不完全统一的情况，数据要求不严格的情况

解决：

Master 上执行：flush tables with read lock;

Slave 上执行：

stop slave;

#表示跳过一步错误，后面的数字可变

set global sql_slave_skip_counter =1;

start slave;

之后再用 `mysql> show slave status\G` 查看：

Slave_IO_Running: Yes

Slave_SQL_Running: Yes

ok，现在主从同步状态正常了。。。

方式二：重新做主从，完全同步

该方法适用于主从库数据相差较大，或者要求数据完全统一的情况

解决步骤如下：

1)先进入主库，进行锁表，防止数据写入

使用命令：

`mysql> flush tables with read lock;`

注意：该处是锁定为只读状态，语句不区分大小写

2)进行数据备份

#把数据备份到 `mysql.bak.sql` 文件

`[root@server01 mysql]#mysqldump -uroot -p -hlocalhost > mysql.sql`

这里注意一点：数据库备份一定要定期进行，可以用 shell 脚本或者 python 脚本，都

比较方便，确保数据万无一失

3)查看 master 状态

`mysql> show master status;`

```
+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
```

```
| mysql-bin.000001 |      272 |
```

```
+-----+-----+-----+-----+
```

1 row in set (0.00 sec)

4)把 mysql 备份文件传到从库机器，进行数据恢复

#使用 scp 命令

```
[root@server01mysql]# scp mysql.sql root@192.168.1.13:/tmp/
```

5)停止从库的状态

```
mysql> stop slave;
```

6)然后到从库执行 mysql 命令，导入数据备份

```
mysql> source /tmp/mysql.sql
```

7)设置从库同步，注意该处的同步点，就是主库 show master status 信息里的| File|
Position 两项

```
change master to master_host = '192.168.1.12', master_user =  
'tongbu',master_password='123456', master_log_file = 'mysql-bin.000001',  
master_log_pos=272;
```

8)重新开启从同步

```
mysql> start slave;
```

9)查看同步状态

```
mysql> show slave status\G 查看：
```

Slave_IO_Running: Yes

Slave_SQL_Running: Yes

10)在 master 上解锁：

unlock tables;

京峰教育