基于Mycat开源数据库中间件

MySQL数据库主从读写分离配置详解与示例

参考文档：《Mycat权威指南》

王金剑

本文说明：

根据【心跳问题heartbeat bug #393】以及【bug407：修复主从状态监控和读写分离】，做了修正，请使用附件中的Mycat安装包：

Mycat-Server-1.4-RC-Linux-RW-20150724.tar.gz

或者将附件中的Mycat-server-1.4-RC.jar 更新到Mycat安装目录下的lib目录，替换原文件

**一、MySQL主服务器Master 配置**

1、修改主服务器配置:

#vi /etc/my.cnf

binlog-do-db=db1

binlog-do-db=db2

binlog-do-db=db3

binlog-ignore-db = mysql

log-bin=mysql-bin #启用二进制日志

server-id=158 #服务器唯一ID，一般取IP最后一段

2、重启MySQL

/etc/init.d/mysql restart

3、建立帐户并授权slave:

#/usr/local/mysql/bin/mysql -uroot -p123456

mysql>GRANT FILE ON \*.\* TO 'backup'@'%' IDENTIFIED BY '123456';

mysql>GRANT REPLICATION SLAVE, REPLICATION CLIENT ON \*.\* to 'backup'@'%' identified by '123456';

#一般不用root帐号，“%”表示所有客户端都可能连，只要帐号，密码正确，此处可用具体客户端IP代替，如192.168.145.226，加强安全。

刷新权限

mysql> FLUSH PRIVILEGES;

查看mysql现在有哪些用户

mysql>select user,host from mysql.user;

4、登录主服务器的mysql，查询master的状态

mysql> show master status;

+------------------+----------+--------------+------------------+

| File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB |

+------------------+----------+--------------+------------------+

| mysql-bin.000010 | 106 | db1,db2,db3 | mysql |

+------------------+----------+--------------+------------------+

Master 重启后会修改mysql-bin（序号加1）

**二、MySQL从服务器Slave 配置**

1、修改从服务器配置:

#vi /etc/my.cnf

replicate-do-db=db1

replicate-do-db=db2

replicate-do-db=db3

replicate-ignore-db=mysql

master-connect-retry=60

server-id=78

master-host=10.1.176.158

master-user=backup

master-password=123456

master-port=3306

master-connect-retry=60

#log-slave-updates

skip-slave-start #防止复制随着mysql启动而自动启动。即slave端的mysql服务重启后需手动来启动主从复制（slave start），最好加上，slave端数据库服务重启后手动启动slave比较安全

2、重启MySQL

/etc/init.d/mysql restart

3、登录mysql并停止slave服务

# cd /usr/local/mysql/bin/

# ./mysql -uroot –p123456 或

# /usr/local/mysql/bin/mysql -uroot –p123456

mysql>slave stop;

设置与master服务器相关的配置参数

mysql>change master to master\_host='10.1.176.158', master\_user='backup', master\_password='123456',MASTER\_LOG\_FILE='mysql-bin.000015',MASTER\_LOG\_POS=106;

注意：Master重启后slave 要修改MASTER\_LOG\_FILE，106无单引号。

启动从服务器复制功能

Mysql>start slave;

ERROR 1201 (HY000):Could not initialize master info structure的问题

解决方案是：运行命令 stop slave;

成功执行后继续运行 reset slave;

4、检查从服务器复制功能状态

mysql> show slave status\G

以下两个参数必须为YES：

Slave\_IO\_Running: Yes

Slave\_SQL\_Running: Yes

返回如下：

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Slave\_IO\_State: Waiting for master to send event

Master\_Host: 10.1.176.158

Master\_User: backup

Master\_Port: 3306

Connect\_Retry: 60

Master\_Log\_File: mysql-bin.000010

Read\_Master\_Log\_Pos: 106

Relay\_Log\_File: bogon-relay-bin.000002

Relay\_Log\_Pos: 251

Relay\_Master\_Log\_File: mysql-bin.000010

**Slave\_IO\_Running: Yes**

**Slave\_SQL\_Running: Yes**

Replicate\_Do\_DB: db1,db2,db3

Replicate\_Ignore\_DB: mysql

Replicate\_Do\_Table:

Replicate\_Ignore\_Table:

Replicate\_Wild\_Do\_Table:

Replicate\_Wild\_Ignore\_Table:

Last\_Errno: 0

Last\_Error:

Skip\_Counter: 0

Exec\_Master\_Log\_Pos: 106

Relay\_Log\_Space: 406

Until\_Condition: None

Until\_Log\_File:

Until\_Log\_Pos: 0

Master\_SSL\_Allowed: No

Master\_SSL\_CA\_File:

Master\_SSL\_CA\_Path:

Master\_SSL\_Cert:

Master\_SSL\_Cipher:

Master\_SSL\_Key:

Seconds\_Behind\_Master: 0

Master\_SSL\_Verify\_Server\_Cert: No

Last\_IO\_Errno: 0

Last\_IO\_Error:

Last\_SQL\_Errno: 0

Last\_SQL\_Error:

1 row in set (0.00 sec)

**三、数据库中间件Mycat配置**

1、不使用Mycat托管MySQL主从服务器，简单使用如下配置

<dataNode name=*"dn1"* dataHost=*"localhost1"* database=*"db1"* />

<dataNode name=*"dn2"* dataHost=*"localhost1"* database=*"db2"* />

<dataNode name=*"dn3"* dataHost=*"localhost1"* database=*"db3"* />

<dataHost name=*"localhost1"* maxCon=*"1000"* minCon=*"10"* balance=*"0"*

writeType=*"0"* dbType=*"mysql"* dbDriver=*"native"* switchType=*"1"* slaveThreshold=*"100"*>

<heartbeat>select user()</heartbeat>

<writeHost host=*"hostM"* url=*"10.1.176.158:3306"* user=*"root"*

password=*"123456"*>

</writeHost>

</dataHost>

2、使用Mycat托管主从切换，使用如下配置：

<dataHost name=*"localhost1"* maxCon=*"1000"* minCon=*"500"* balance=*"2"*

writeType=*"0"* dbType=*"mysql"* dbDriver=*"native"* switchType=*"2"* slaveThreshold=*"100"*>

<heartbeat>show slave status</heartbeat>

<!-- can have multi write hosts -->

<writeHost host=*"hostM1"* url=*"10.1.176.158:3306"* user=*"root"*

password=*"123456"*>

</writeHost>

<writeHost host=*"hostS1"* url=*"10.1.176.78:3306"* user=*"root"*

password=*"123456"* />

</dataHost>

2、Mycat 1.4 支持MySQL主从复制状态绑定的读写分离机制，让读更加安全可靠，配置如下：

<dataNode name=*"dn1"* dataHost=*"localhost1"* database=*"db1"* />

<dataNode name=*"dn2"* dataHost=*"localhost1"* database=*"db2"* />

<dataNode name=*"dn3"* dataHost=*"localhost1"* database=*"db3"* />

<dataHost name=*"localhost1"* maxCon=*"1000"* minCon=*"10"* **balance=*"1"***

**writeType=*"0"*** dbType=*"mysql"* dbDriver=*"native"* **switchType=*"2"* slaveThreshold=*"100"***>

<heartbeat>**show slave status**</heartbeat>

<writeHost host=*"hostM"* url=*"10.1.176.158:3306"* user=*"root"*

password=*"123456"*>

<readHost host=*"hostS"* url=*"10.1.176.78:3306"* user=*"root"*

password=*"123456"* />

</writeHost>

</dataHost>

1. **设置 balance="1"与writeType="0"**

Balance参数设置：

1. balance=“0”, 所有读操作都发送到当前可用的writeHost上。

2. balance=“1”，所有读操作都随机的发送到readHost。

3. balance=“2”，所有读操作都随机的在writeHost、readhost上分发

WriteType参数设置：

1. writeType=“0”, 所有写操作都发送到可用的writeHost上。

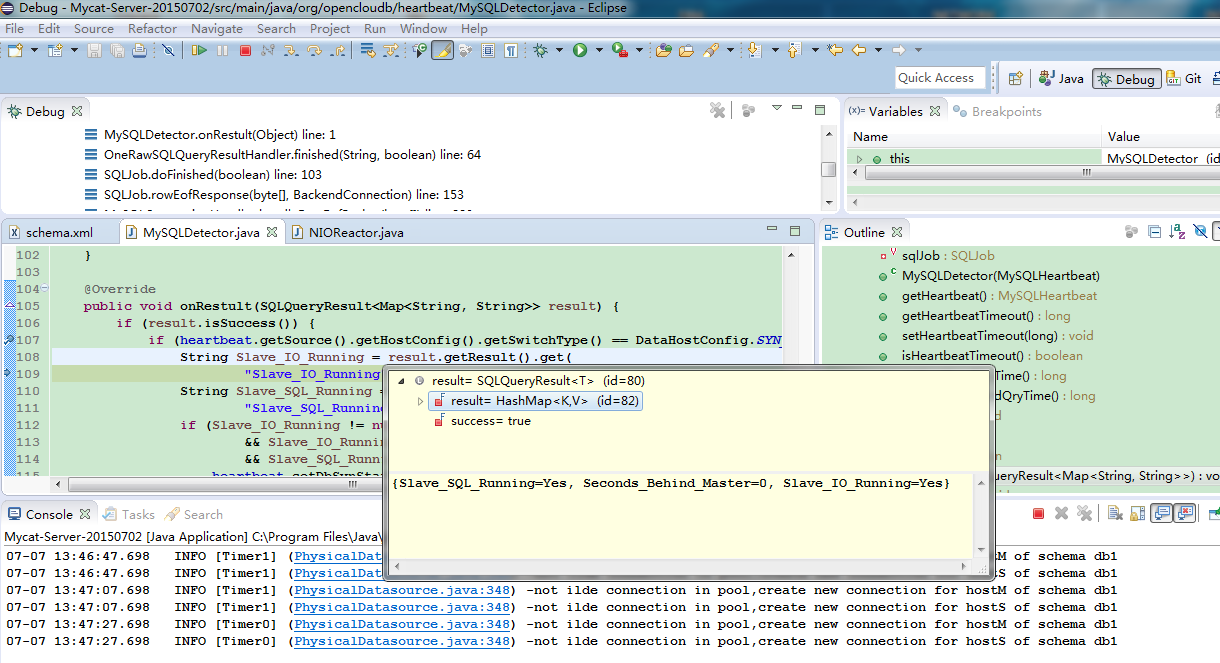
2. writeType=“1”，所有写操作都随机的发送到readHost。

3. writeType=“2”，所有写操作都随机的在writeHost、readhost分上发。

“readHost是从属于writeHost的，即意味着它从那个writeHost获取同步数据，因此，当它所属的writeHost宕机了，则它也不会再参与到读写分离中来，即“不工作了”，这是因为此时，它的数据已经“不可靠”了。基于这个考虑，目前mycat 1.3和1.4版本中，若想支持MySQL一主一从的标准配置，并且在主节点宕机的情况下，从节点还能读取数据，则需要在Mycat里配置为两个writeHost并设置banlance=1。”

1. **设置 switchType="2" 与slaveThreshold="100"**

“Mycat心跳检查语句配置为 show slave status ，dataHost 上定义两个新属性： switchType="2" 与slaveThreshold="100"，此时意味着开启MySQL主从复制状态绑定的读写分离与切换机制。Mycat心跳机制通过检测 show slave status 中的 "Seconds\_Behind\_Master", "Slave\_IO\_Running", "Slave\_SQL\_Running" 三个字段来确定当前主从同步的状态以及Seconds\_Behind\_Master主从复制时延。“



**四、主从测试，执行以下命令**

mysql> explain create table company(id int not null primary key,name varchar(100));

+-----------+---------------------------------------------------------------------+

| DATA\_NODE | SQL |

+-----------+---------------------------------------------------------------------+

| dn1 | create table company(id int not null primary key,name varchar(100)) |

| dn2 | create table company(id int not null primary key,name varchar(100)) |

| dn3 | create table company(id int not null primary key,name varchar(100)) |

+-----------+---------------------------------------------------------------------+

3 rows in set (0.32 sec)

mysql> create table company(id int not null primary key,name varchar(100));

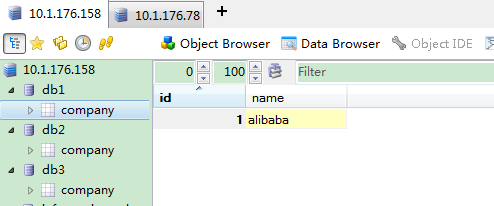
Query OK, 0 rows affected (0.26 sec)

mysql> insert into company(id,name) values(1,'alibaba');

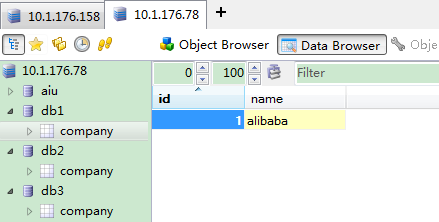
Query OK, 3 rows affected (0.15 sec)

查看执行结果：

Master 端：



Slave 端：



**读写分离测试请将日志的debug（log4j.xml文件）模式打开，通过select查询观察日志中是否通过已配置的读节点执行。**

有任何问题请与我联系，欢迎与大家交流

王金剑 微信：life3721com QQ：1526330277

博客：<http://blog.csdn.net/dreamcode>