Mycat 部署文档 1.0

1. mysql 同步环境配置

在使用mycat管理一个mysql集群的时候经常需要使用主从数据复制的功能，以下是基于mysql5.5以上版本最新的主从配置：

开启mysql主从数据复制，主要在mysql的my.ini文件中设置：

windows系统一般在c:\ProgrameData\MySQL中；

Linux系统一般在 /etc/my.cnf中；

**1、M1数据库服务器配置：**

server-id=1 ##主库和从库的server-id一定要不同

lower\_case\_table\_names=1 ##让mysql不区分大小写

log-bin=mysql-bin

binlog-do-db=test       ##只同步test数据库

binlog-ignore-db=mysql   ##避免同步mysql用户配置，以免不必要的麻烦

**2、M2数据库服务器配置：**

server-id=101

lower\_case\_table\_names=1 ##让mysql不区分大小写

log-bin=mysql-bin

replicate-do-db= test     ##指定只同步的数据库

replicate-ignore-db=mysql   ##屏蔽对mysql用户配置的同步，以免不必要的麻烦

**3、M1 同步到 M2**

1）M1上创建一个用户  
mysql> grant replication slave,file on \*.\* to testuser@' 60.205.182.32' identified by 'm01123456';

## 说明：“m01”为给备服务创建的一个用户,“m01123456”为密码  
mysql> flush privileges;  
mysql> show master status\G;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
File: master-bin.000031  
Position: 909  
Binlog\_Do\_DB:  
Binlog\_Ignore\_DB:  
Executed\_Gtid\_Set:  
1 row in set (0.00 sec)  
ERROR:  
No query specified

2）在M2数据库服务器上给M1服务器授权：

mysql>

change master to master\_host=' 60.205.182.32',master\_port=33061,master\_user=testuser,

master\_password='m01123456', master\_log\_file='master-bin.000009', master\_log\_pos=106;

mysql> slave start;

show slave status\G;

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

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

Master\_Host: 60.205.182.32

Master\_User: testuser

Master\_Port: 3306

Connect\_Retry: 60

Master\_Log\_File: mysql-bin.000009

Read\_Master\_Log\_Pos: 106

Relay\_Log\_File: hansc-pc-relay-bin.000002

Relay\_Log\_Pos: 251

Relay\_Master\_Log\_File: mysql-bin.000009

Slave\_IO\_Running: Yes ## Slave\_IO\_Running，Slave\_SQL\_Running 为Yes表示

Slave\_SQL\_Running: Yes ##设置成功

Replicate\_Do\_DB:

Replicate\_Ignore\_DB:

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: 409

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)

**4、M2 同步到 M1**

1）M1上创建一个用户  
2）在M2数据库服务器上给M1服务器授权：

详细步骤同第三步。

1. mycat 配置简介及配置

1、安装(linux)

下载：http://pan.baidu.com/s/1jHZTIR0

解压：tar xvf Mycat-server-1.6-RELEASE-20161028204710-linux.tar.gz

启动：./bin/startup\_nowrap.sh

默认数据库的端口为8066，管理端口为9066

2、配置及说明

1) server.xml（需要配置）

配置数据库名称、用户名、密码,G2系统JDBC要使用这个数据库名称；详细参考mycat使用指南。

例如：配置一个用户名为root，密码为101202，可访问的数据库为TESTDB

<user name="root">

<property name="password">101202</property>

<property name="schemas">TESTDB</property>

</user>

G2系统JDBC路径配置：

gbpDataSource.url=jdbc\:mysql\://localhost\:8066/TESTDB?characterEncoding\=UTF-8&use OldAliasMetadataBehavior\=true

gbpDataSource.user=root

gbpDataSource.password=101202

2) rule.xml （需要配置）

分片规则，列出了常用的分片策略。

例如：

<tableRule name="sharding-by-murmur">

<rule>

<columns>config\_id</columns>

<algorithm>murmur</algorithm>

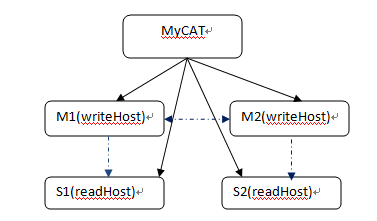
</rule>

</tableRule>

3) server.xml（需要配置）

Schema.xml 作为 MyCat 中重要的配置文件之一，管理着 MyCat 的逻辑库、表、分片规则、 DataNode 以及 DataSource。弄懂这些配置，是正确使用 MyCat 的前提。

1. 双主双从模式



上图仅为一个分片的关系图。

1. M1和M2 之间是双向同步，M1和S1是单向同步，M2和S2 是单向同步。
2. M1，M2，S1，S2读写分离
3. M1挂掉，主数据库会切换到M2

4、schema文件配置:

<?xml version="1.0"?>

<!DOCTYPE mycat:schema SYSTEM "schema.dtd">

<mycat:schema xmlns:mycat="http://io.mycat/">

<!-- 双主双从模式-->

<schema name="TESTDB" dataNode="dn2">

<table name="g2\_t\_attach\_config" dataNode="dn1,dn2" rule="sharding-by-murmur" primaryKey="config\_id"/>

</schema>

<dataNode name="dn1" dataHost="60.205.182.31" database="test" />

<dataNode name="dn2" dataHost="192.168.1.91" database="test" />

<!-- 分片 1 -->

<dataHost name="60.205.182.31" maxCon="1000" minCon="10" balance="2"

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

<heartbeat>show slave status</heartbeat>

<!-- 读写 分离-->

<writeHost host="hostM1" url="60.205.182.31:3306" user="root" password="root">

<readHost host="hostS1" url="60.205.182.32:3306" user="root"

password="root" />

</writeHost>

<writeHost host="hostM2" url="60.205.182.33:3306" user="root" password="root">

<readHost host="hostS2" url="60.205.182.34:3306" user="root"

password="root" />

</writeHost>

</dataHost>

<!-- 分片 2 -->

<dataHost name="192.168.1.91" maxCon="1000" minCon="10" balance="2"

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

<heartbeat>show slave status</heartbeat>

<writeHost host="hostM11" url="192.168.1.90:3306" user="root" password="root">

<readHost host="hostS11" url="192.168.1.91:3306" user="root"

password="root" />

</writeHost>

<writeHost host="hostM22" url="192.168.1.92:3306" user="root" password="root">

<readHost host="hostS22" url="192.168.1.93:3306" user="root"

password="root" />

</writeHost>

</dataHost>

</mycat:schema>

1、设置balance="1"与writeType="0"

1) Balance参数设置：

负载均衡类型，目前的取值有3种：

<1>balance="0", 不开启读写分离机制，所有读操作都发送到当前可用的writeHost上。

<2> balance="1"，全部的readHost与stand by writeHost参与select语句的负载均衡，简单的说，当双主双从模式(M1->S1，M2->S2，并且M1与 M2互为主备)，正常情况下，M2,S1,S2都参与select语句的负载均衡。

<3>balance="2"，所有读操作都随机的在writeHost、readhost上分发。

2) WriteType参数设置：

<1> writeType="0",所有写操作发送到配置的第一个writeHost，第一个挂了切到还生存的第二个writeHost。如果都挂了，那就悲剧了，起来哪个，写哪个，很有可能数据不一致；所以不要数据库恢复之后不要让mycat切回。

<2>writeType="1"，所有写操作都随机的发送到配置的writeHost。

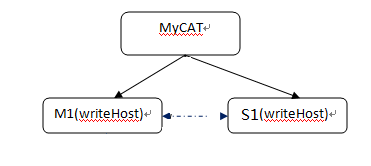
<3>writeType="2"，没实现。

 “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。”

2、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。“

1. 主从复制模式



上图仅为一个分片的关系图。

1、主从复制模式就是只保留双主双从模式的<wirteHost />去掉<readHost /> 。

2、M1 和 S1之间是双向同步，M1挂掉，S1变成主。

3、M1，S1读写分离

4、schema文件配置:

<?xml version="1.0"?>

<!DOCTYPE mycat:schema SYSTEM "schema.dtd">

<mycat:schema xmlns:mycat="http://io.mycat/">

<!-- 主从复制模式-->

<schema name="TESTDB" dataNode="dn2">

<table name="g2\_t\_attach\_config" dataNode="dn1,dn2" rule="sharding-by-murmur" primaryKey="config\_id"/>

</schema>

<dataNode name="dn1" dataHost="127.0.0.1" database="g2" />

<dataNode name="dn2" dataHost="60.205.182.32" database="test" />

<!-- 分片 1 -->

<dataHost name="60.205.182.32" maxCon="1000" minCon="10" balance="2"

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

<heartbeat>show slave status</heartbeat>

<!-- 读写 分离-->

<writeHost host="hostM1" url="60.205.182.32:3306" user="root" password="101202">

</writeHost>

<!-- 从（备）-->

<writeHost host="hostS1" url="127.0.0.1:3306" user="root" password="101202">

</writeHost>

</dataHost

<!-- 分片 2 -->

<dataHost name="127.0.0.1" maxCon="1000" minCon="10" balance="2"

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

<heartbeat>show slave status</heartbeat>

<writeHost host="hostM2" url="127.0.0.1:3306" user="root" password="101202">

</writeHost>

<writeHost host="hostS2" url="60.205.182.32:3306" user="root" password="101202">

</writeHost>

</dataHost>

</mycat:schema>

1. 资料下载

包括一些mycat的官方文档，地址：<http://pan.baidu.com/s/1jHZTIR0>

1. 常见问题

1、Could not find first log file name in binary log index file'的解决办法

参考 <http://blog.csdn.net/billfanggs/article/details/8905991>

2、[Got fatal error 1236 from master when reading data from binary log](http://blog.csdn.net/edwzhang/article/details/17226975)

参考<http://blog.csdn.net/edwzhang/article/details/17226975>

[3、error code 1146 > Error 'Table 'tts.t2' doesn't exist' on query. Default database: 'test'. Query: 'i](http://blog.csdn.net/snowpay/article/details/52313025)

参考<http://blog.csdn.net/snowpay/article/details/52313025>

1. 本文介绍了双主双从模式和主从复制模式配置，前者使用的多于后者，推荐双主双从模式。