++++++++++++++++++++++++编辑配置文件

[root@localhost ~]# cat /usr/local/mycat/conf/server.xml #定义连接客户端连接mycat服务使用的用户名和密码及逻辑库的名字

<?xml version="1.0" encoding="UTF-8"?>

<!-- - - Licensed under the Apache License, Version 2.0 (the "License");

- you may not use this file except in compliance with the License. - You

may obtain a copy of the License at - - http://www.apache.org/licenses/LICENSE-2.0

- - Unless required by applicable law or agreed to in writing, software -

distributed under the License is distributed on an "AS IS" BASIS, - WITHOUT

WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. - See the

License for the specific language governing permissions and - limitations

under the License. -->

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

<mycat:server xmlns:mycat="http://org.opencloudb/">

<system>

<property name="defaultSqlParser">druidparser</property>

<!-- <property name="useCompression">1</property>--> <!--1为开启mysql压缩协议-->

<!-- <property name="processorBufferChunk">40960</property> -->

<!--

<property name="processors">1</property>

<property name="processorExecutor">32</property>

-->

<!--默认是65535 64K 用于sql解析时最大文本长度 -->

<!--<property name="maxStringLiteralLength">65535</property>-->

<!--<property name="sequnceHandlerType">0</property>-->

<!--<property name="backSocketNoDelay">1</property>-->

<!--<property name="frontSocketNoDelay">1</property>-->

<!--<property name="processorExecutor">16</property>-->

<!--

<property name="mutiNodeLimitType">1</property> 0：开启小数量级（默认） ；1：开启亿级数据排序

<property name="mutiNodePatchSize">100</property> 亿级数量排序批量

<property name="processors">32</property> <property name="processorExecutor">32</property>

<property name="serverPort">8066</property> <property name="managerPort">9066</property>

<property name="idleTimeout">300000</property> <property name="bindIp">0.0.0.0</property>

<property name="frontWriteQueueSize">4096</property> <property name="processors">32</property> -->

</system>

<user name="test"> #连接mycat服务时使用的用户名 test

<property name="password">test</property> #使用test用户连接mycat用户时使用的密码

<property name="schemas">TESTDB</property> #连接上mycat服务后，可以看到的库名多个时，使用逗号分隔 （是逻辑上的库名）

</user>

<user name="user">

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

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

<property name="readOnly">true</property> #定义只读权限，使用定义的user用户连接mycat服务后只有读记录的权限

</user>

<!-- <cluster> <node name="cobar1"> <property name="host">127.0.0.1</property>

<property name="weight">1</property> </node> </cluster> -->

<!-- <quarantine> <host name="1.2.3.4"> <property name="user">test</property>

</host> </quarantine> -->

</mycat:server>

[root@localhost ~]#

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@localhost ~]# cat /usr/local/mycat/conf/schema.xml

<?xml version="1.0"?>

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

<mycat:schema xmlns:mycat="http://org.opencloudb/">

<schema name="TESTDB" checkSQLschema="false" sqlMaxLimit="100"> #逻辑库名 要与server.xml定义的一样

<table name="travelrecord" dataNode="dn1,dn2" rule="auto-sharding-long" /> #定义分片的表

<table name="company" primaryKey="ID" type="global" dataNode="dn1,dn2" /> #定义分片的表

<table name="goods" primaryKey="ID" type="global" dataNode="dn1,dn2" /> #定义分片的表

<table name="hotnews" primaryKey="ID" dataNode="dn1,dn2" rule="mod-long" /> #定义分片的表

<table name="employee" primaryKey="ID" dataNode="dn1,dn2" rule="sharding-by-intfile" /> #定义分片的表

<table name="customer" primaryKey="ID" dataNode="dn1,dn2" rule="sharding-by-intfile" /> #定义分片的表

</schema>

<dataNode name="dn1" dataHost="c1" database="db1" />

#定义分片使用的库，所在的物理主机 ,真正存储数据的db1库在物理主机c1上

<dataNode name="dn2" dataHost="c2" database="db2" />

#定义分片使用的库，所在的物理主机 ,真正存储数据的db2库在物理主机c2上

#指定c1名称主机对应的ip地址

<dataHost name="c1" maxCon="1000" minCon="10" balance="0"

writeType="0" dbType="mysql" dbDriver="native" >

<heartbeat>select user()</heartbeat>

<writeHost host="hostM1" url="192.168.4.55:3306" user="admin"

password="123456"> #访问数据时 mycat服务连接数据库服务器时使用的用户名和密码

</writeHost>

</dataHost>

#指定c2名称主机对应的ip地址

<dataHost name="c2" maxCon="1000" minCon="10" balance="0"

writeType="0" dbType="mysql" dbDriver="native" >

<heartbeat>select user()</heartbeat>

<writeHost host="hostM2" url="192.168.4.54:3306" user="admin"

password="123456"> #访问数据时 mycat服务连接数据库服务器时使用的用户名和密码

</writeHost>

</dataHost>

</mycat:schema>

[root@localhost ~]#

++++++++++++++++++++启动服务

[root@localhost ~]# sed -n '4,5p' /usr/local/mycat/conf/wrapper.conf

# Java Application

wrapper.java.command=java

[root@localhost ~]#

[root@localhost ~]# which java

/usr/bin/java

[root@localhost ~]#echo "export PATH=/usr/local/mycat/bin" >> /etc/profile

[root@localhost ~]#source /etc/profile

[root@localhost ~]# mycat --help

Usage: /usr/local/mycat/bin/mycat { console | start | stop | restart | status | dump }

[root@localhost ~]#

[root@localhost ~]# mycat start

Starting Mycat-server...

[root@localhost ~]#

[root@localhost ~]# netstat -utnalp | grep :8066

tcp6 0 0 :::8066 :::\* LISTEN 4524/java

[root@localhost ~]#

++++++++++++++++++++++++测试MyCAT

[root@room9pc17 ~]# mysql -h192.168.4.56 -P8066 -utest -ptest //客户端连接mycat服务器，存储数据

MySQL [(none)]> show databases;

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

| DATABASE |

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

| TESTDB |

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

1 row in set (0.00 sec)

MySQL [(none)]> use TESTDB;

MySQL [TESTDB]> show tables;

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

| Tables in TESTDB |

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

| company |

| customer |

| employee |

| goods |

| hotnews |

| travelrecord |

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

6 rows in set (0.00 sec)

MySQL [TESTDB]> MySQL [TESTDB]> create table employee( id int not null primary key, name varchar(100), sharding\_id int not null );

MySQL [TESTDB]> MySQL [TESTDB]> insert into employee(id,name,sharding\_id)values(1,"bob",10000),(2,"lucy",10010),(3,"alice",10000),(4,"jerry",10010);

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++

mysql> select @@hostname;

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

| @@hostname |

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

| c1 |

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

1 row in set (0.00 sec)

mysql> select \* from employee;

Empty set (0.00 sec)

mysql> select \* from employee;

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

| id | name | sharding\_id |

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

| 1 | bob | 10000 |

| 3 | alice | 10000 |

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

2 rows in set (0.00 sec)

mysql>

++++++++++++++++++++++++++++++++++++++++++++++

mysql> select @@hostname;

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

| @@hostname |

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

| c2 |

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

1 row in set (0.00 sec)

mysql> select \* from employee;

Empty set (0.00 sec)

mysql> select \* from employee;

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

| id | name | sharding\_id |

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

| 2 | lucy | 10010 |

| 4 | jerry | 10010 |

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

2 rows in set (0.00 sec)

mysql>