NSD DBA2 DAY05

1. 案例1：搭建mycat 分片服务器

1 案例1：搭建mycat 分片服务器

1.1 问题

- 数据库主机 192.168.4.55 使用db1库存储数据
- 数据库主机 192.168.4.56 使用db2库存储数据
- 主机 192.168.4.54 运行mycat服务，逻辑库名称为test，连接用户名为admin，密码123456
- 在主机 192.168.4.254 访问测试配置

1.

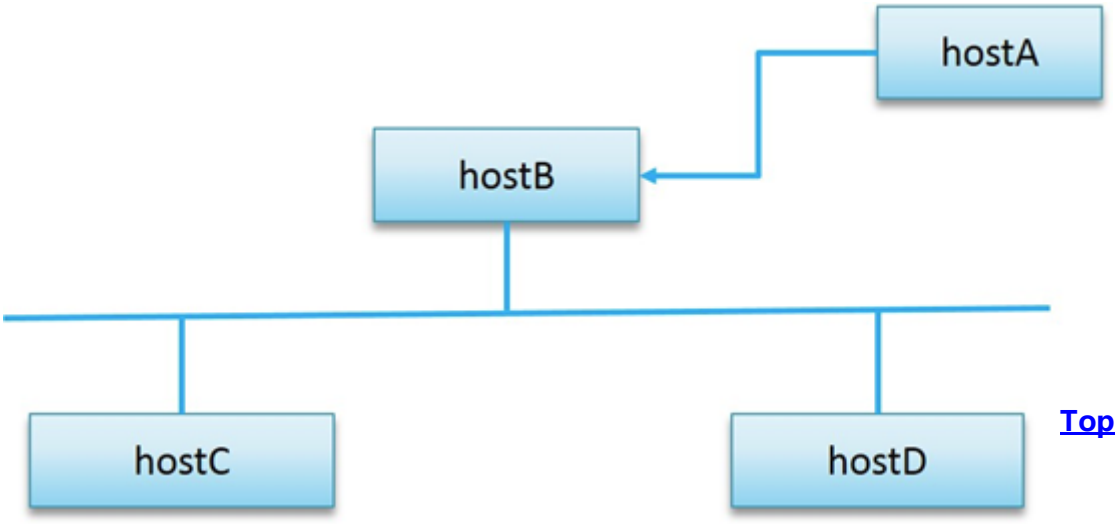
1.2 方案

准备四台主机，搭建mycat分片服务器，通过某种特定条件，将存放在一个数据库(主机)中的数据，分散存放到多个数据库(主机)中，已达到分散单台设备负载的效果。其中192.168.4.56作为mycat服务器，192.168.4.54和192.168.4.55作为数据库服务器，192.168.4.254作为客户端。如图-1所示：

拓扑名称	主机名	角色	数据库名	IP地址
hostA	client	客户端	无	192.168.4.254/24
hostB	mycat	mycat服务器	无	192.168.4.56/24
hostC	c1	数据库服务器	db1	192.168.4.55/24
hostD	c2	数据库服务器	db2	192.168.4.54/24

图-1

数据分片的拓扑如图-2所示：



Top

图-2

1.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：公共配置

1）关闭防火墙和selinux，配置yum源（系统源），这里不再操作

2）把54，55还原成独立数据库服务器，且只保留默认4个库其他都删除，停止56主机的mha管理服务，下载mycat 软件到56主机

3）在54（主机c2）和55（主机c1）上面创建db1和db2库

```
[root@c1 ~]# mysql -u root -p123456
mysql> create database db1;        //c1上面创建db1库
Query OK, 1 row affected (0.00 sec)

[root@c2 ~]# mysql -u root -p123456
mysql> create database db2;        //c2上面创建db2库
Query OK, 1 row affected (0.00 sec)
```

在54上面授权一个用户

```
mysql> grant all on *.* to admin@"%" identified by  "123456";
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

在55上面授权一个用户

```
mysql> grant all on *.* to admin@"%" identified by  "123456";
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

4）修改数据库的配置文件

注意：1代表不区分表名的大小写，0为区分大小写

主机c1上面：

```
[root@c1 ~]# vim /etc/my.cnf
[mysqld]
lower_case_table_names=1       //表名忽略大小写
[root@c1 ~]# systemctl restart mysqld
```

Top

主机c2上面：

```
[root@c2 ~]#vim /etc/my.cnf
[mysqld]
lower_case_table_names=1
[root@c2 ~]#systemctl restart mysqld

```

5）在56主机上面安装JDK

```
[root@mycat ~]#rpm -qa | grep -i jdk      //安装自带的即可
java-1.8.0-openjdk-1.8.0.131-11.b12.el7.x86_64
java-1.8.0-openjdk-headless-1.8.0.131-11.b12.el7.x86_64
copy-jdk-configs-2.2-3.el7.noarch
[root@mycat ~]#yum -y install java-1.8.0-openjdk
```

6）在56主机上面安装mycat

```
[root@mycat ~]#cd mysql/
[root@mycat mysql]#tar -xf Mycat-server-1.4-beta-20150604171601-linux.tar.gz   //免
[root@mycat mysql]#  mv mycat/ /usr/local/
[root@mycat mysql]#ls /usr/local/mycat/
bin  catlet  conf  lib  logs  version.txt
[root@mycat mysql]#cd /usr/local/mycat/
[root@mycat mycat]#./bin/mycat --help
Usage: ./bin/mycat { console | start | stop | restart | status | dump }
```

7）修改配置文件

目录结构说明：

bin mycat命令，如 启动 停止 等

catlet 扩展功能

conf 配置文件

lib mycat使用的jar

log mycat启动日志和运行日志

wrapper.log mycat服务启动日志

Top

mycat.log 记录SQL脚本执行后的报错内容

重要配置文件说明：

server.xml 设置连mycat的账号信息

schema.xml 配置mycat的真实库表

rule.xml 定义mycat分片规则

配置标签说明

<user>.. ..</user> 定义连mycat用户信息

<datanode>.. ..</datanode> 指定数据节点

<datahost>.. ..</datahost> 指定数据库地址及用户信息

查看server.xml配置文件

```
[root@mycat mycat]# cd conf/
[root@mycat conf]# vim server.xml
        </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,服务
        </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,不写这一行
        </user>
```

修改schema.xml配置文件

```
[root@mycat conf]# vim schema.xml
<table name="travelrecord" dataNode="dn1,dn2" rule="auto-sharding-long" />
//travelrecord（逻辑上的，名字不能随便写，一般不动）表分片到数据节点dn1和dn2，
<table name="company" primaryKey="ID" type="global" dataNode="dn1,dn2" />
<table name="hotnews" primaryKey="ID" dataNode="dn1,dn2" rule="mod-long" />
<dataNode name="dn1" dataHost="c1" database="db1" />
//数据节点对应的服务器 name="dn1"名称要与上面的对应 dataHost="c1"写本机主机名
```

Top

```
<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db1库在物理主机mysql55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" switchType="1"  slaveThresh
                <heartbeat>select user()</heartbeat>
            <!-- can have multi write hosts -->
            <writeHost host="c1" url="192.168.4.55:3306" user="admin"
                password="123456">
//访问数据库时，mycat服务连接数据库服务器时使用的用户名和密码
                <!-- can have multi read hosts -->

            </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" switchType="1"  slaveThresh
                <heartbeat>select user()</heartbeat>
            <!-- can have multi write hosts -->
            <writeHost host="c2" url="192.168.4.54:3306" user="admin"
                 password="123456">
//访问数据库时，mycat服务连接数据库服务器时使用的用户名和密码
                <!-- can have multi read hosts -->

            </writeHost>
        </dataHost>
```

8）添加PATH路径

```
[root@mycat conf]#  export PATH=/usr/local/mycat/bin:$PATH
[root@mycat conf]# echo "PATH=/usr/local/mycat/bin:$PATH" >> /etc/profile
[root@mycat conf]# source /etc/profile
[root@mycat conf]# echo $PATH
/usr/local/mycat/bin:/usr/local/mycat/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr
```

Top

```
[root@mycat conf]# which mycat
/usr/local/mycat/bin/mycat
```

9）启动服务并查看端口

```
[root@mycat conf]#  mycat start
Starting Mycat-server...
[root@mycat conf]# netstat -antup | grep :8066
tcp6       0      0 :::8066                 :::*                    LISTEN      6421/java
[root@mycat conf]# ps -C java
  PID TTY          TIME CMD
 6421 ?        00:00:04 java
```

用admin用户登录

```
[root@client ~]#  mysql -h192.168.4.54 -uadmin -p123456
mysql> show processlist;
+----+-------+--------------------+------+---------+------+----------+-----------------------+
| Id | User  | Host               | db   | Command | Time | State    | Info                  |
+----+-------+--------------------+------+---------+------+----------+-----------------------+
|  3 | admin | 192.168.4.56:34580 | db2  | Sleep   |   80 |          | NULL                  |
|  4 | admin | 192.168.4.56:34570 | db2  | Sleep   |  100 |          | NULL                  |
|  5 | admin | 192.168.4.56:34572 | db2  | Sleep   |   40 |          | NULL                  |
|  6 | admin | 192.168.4.56:34562 | db2  | Sleep   |   30 |          | NULL                  |
|  7 | admin | 192.168.4.56:34564 | db2  | Sleep   |   90 |          | NULL                  |
|  8 | admin | 192.168.4.56:34566 | db2  | Sleep   |   60 |          | NULL                  |
|  9 | admin | 192.168.4.56:34574 | db2  | Sleep   |   70 |          | NULL                  |
| 10 | admin | 192.168.4.56:34576 | db2  | Sleep   |   10 |          | NULL                  |
| 11 | admin | 192.168.4.56:34578 | db2  | Sleep   |   20 |          | NULL                  |
| 12 | admin | 192.168.4.56:34568 | db2  | Sleep   |   50 |          | NULL                  |
| 14 | admin | 192.168.4.51:58354 | NULL | Query   |    0 | starting | show processlist      |
+----+-------+--------------------+------+---------+------+----------+-----------------------+
11 rows in set (0.00 sec)
```

Top

```
[root@client ~]#  mysql -h192.168.4.55 -uadmin -p123456
mysql> show processlist;
```

```
+----+------+------------------+------+---------+------+----------+-----
| Id | User | Host             | db   | Command | Time | State    | Info        |
+----+------+------------------+------+---------+------+----------+-----
|  3 | root | localhost        | NULL | Sleep   | 2352 |          | NULL        |
|  4 | admin | 192.168.4.56:45148 | db1 | Sleep   |    2 |          | NULL        |
|  5 | admin | 192.168.4.56:45150 | db1 | Sleep   |   62 |          | NULL        |
|  6 | admin | 192.168.4.56:45160 | db1 | Sleep   |   12 |          | NULL        |
|  7 | admin | 192.168.4.56:45162 | db1 | Sleep   |   92 |          | NULL        |
|  8 | admin | 192.168.4.56:45152 | db1 | Sleep   |   32 |          | NULL        |
|  9 | admin | 192.168.4.56:45154 | db1 | Sleep   |   42 |          | NULL        |
| 10 | admin | 192.168.4.56:45156 | db1 | Sleep   |   22 |          | NULL        |
| 11 | admin | 192.168.4.56:45158 | db1 | Sleep   |   82 |          | NULL        |
| 12 | admin | 192.168.4.56:45164 | db1 | Sleep   |   52 |          | NULL        |
| 13 | admin | 192.168.4.56:45166 | db1 | Sleep   |   72 |          | NULL        |
| 14 | admin | 192.168.4.51:32796 | NULL | Query   |    0 | starting | show processlist |
+----+------+------------------+------+---------+------+----------+-----
12 rows in set (0.00 sec)

[root@mycat conf]# ls /usr/local/mycat/logs/
mycat.log  mycat.pid  wrapper.log       //wrapper.log为错误日志
[root@mycat conf]# ldconfig -v       //更新加载的模块
```

10）客户端访问

命令： mysql -hmycat主机的IP -P端口号 -u用户 -p密码

```
[root@clent ~]# mysql -h192.168.4.56 -P8066 -utest -ptest
mysql> show databases;
+----------+
| DATABASE |
+----------+
| TESTDB   |
+----------+
1 row in set (0.00 sec)

mysql> USE TESTDB;
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
```

Top

```
mysql>
mysql> show tables;
+-----------------+
| Tables in TESTDB |
+-----------------+
| company          |
| customer         |
| customer_addr    |
| employee         |
| goods            |
| hotnews          |
| orders           |
| order_items      |
| travelrecord     |
+-----------------+
9 rows in set (0.00 sec)
```

11）客户端测试：

```
mysql> create table employee(id int not null primary key,name varchar(100),sharding_id i
Query OK, 0 rows affected (0.10 sec)

mysql> insert into employee(id,name,sharding_id) values(1,"bob",10000);
Query OK, 1 row affected (0.03 sec)

mysql> insert into employee(id,name,sharding_id) values(1,"lucy",10010);
Query OK, 1 row affected (0.02 sec)

mysql> select * from employee;
+----+------+-------------+
| id | name | sharding_id |
+----+------+-------------+
|  1 | bob  |       10000 |
|  1 | lucy |       10010 |
+----+------+-------------+
2 rows in set (0.06 sec)
```

Top

12）在c1上面查看结果

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| db1                |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> use db1;
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_db1 |
+---------------+
| employee      |
+---------------+
1 row in set (0.00 sec)

mysql> select * from employee;        //查看结果
+----+------+-------------+
| id | name | sharding_id |
+----+------+-------------+
|  1 | bob  |       10000 |
+----+------+-------------+
1 row in set (0.00 sec)
```

13）在c2上面查看结果

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
```

Top

```
| db2                |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> use db2;
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_db2 |
+---------------+
| employee      |
+---------------+
1 row in set (0.00 sec)

mysql> select * from employee;
+----+------+-------------+
| id | name | sharding_id |
+----+------+-------------+
|  1 | lucy |       10010 |
+----+------+-------------+
1 row in set (0.00 sec)
```

Top