



链滴

# MyCat 基本配置

作者: [momoda](#)

原文链接: <https://ld246.com/article/1507214005416>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

**MyCat开发那么一定需要MySQL数据库。本次做一个MyCat基本配置，需要的使用一个MySQL服务。**

如果要想取得MyCat开发包，下载地址为：

<https://github.com/MyCATApache/Mycat-download>

然后选择相应的版本即可

本次下载的的版本为：

<http://dl.mycat.io/1.6-RELEASE/Mycat-server-1.6-RELEASE-20161028204710-linux.tar.gz>

## 本次主要使用两台主机实现服务的搭建

No.	主机名称	IP地址	描述
1 MySQL服务	mysql-server-single	192.168.1.165	
2 MyCat服务	mycat-server-01	192.168.1.166	

### 1. 【mysql-server-single】登录mysql数据库：

```
mysql -u root -phahaha
```

### 2. 【mysql-server-single】创建一个数据库脚本

```
DROP DATABASE IF EXISTS zsl;  
CREATE DATABASE zsl CHARACTER SET UTF8 ;  
use zsl ;  
CREATE TABLE member(  
    mid VARCHAR(50) ,  
    name VARCHAR(50) ,  
    age INT ,  
    salary DOUBLE ,  
    birthday DATE ,  
    note TEXT ,  
    CONSTRAINT pk_mid PRIMARY KEY(mid)  
);  
INSERT INTO member(mid,name,age,salary,birthday,note) VALUES ('zsl','hello',10,2000.0,'2017-11-11','very good') ;  
INSERT INTO member(mid,name,age,salary,birthday,note) VALUES ('admin','admin',10,2000.0,'2017-11-11','very good') ;  
INSERT INTO member(mid,name,age,salary,birthday,note) VALUES ('guest','admin',10,2000.0,'2017-11-11','very good') ;
```

### 3. 【mycat-server-01】将mycat的开发包下载到系统之中，而后对其进行解压缩

```
wget -c http://dl.mycat.io/1.6-RELEASE/Mycat-server-1.6-RELEASE-20161028204710-linux.tar.  
z
```

```
tar xzvf Mycat-server-1.6-RELEASE-20161028204710-linux.tar. -C /usr/local/
```

解压缩之后将得到 /usr/local/mycat 目录,所有的配置项都在此目录之中

#### 4. 【mycat-server-01】编辑mycat的核心配置文件:schema.xml(所有的库分离配置)。

打开编辑器:

```
vim /usr/local/mycat/conf/schema.xml
```

```
<?xml version="1.0"?>  
<!DOCTYPE mycat:schema SYSTEM "schema.dtd">  
<mycat:schema xmlns:mycat="http://io.mycat/">  
  <!-- 定义一个MyCat的模式, 此处定义了一个路基数据库名称TestDB -->  
  <!-- "checkSQLschema" :描述的是当前的连接是否需要检测数据的模式 -->  
  <!-- "sqlMaxLimit" :表示返回的最大的数据量的行数 -->  
  <!-- "dataNode="dn1"" :该操作使用的数据节点是dn1的逻辑名称 -->  
  <schema name="TESTDB" checkSQLschema="false" sqlMaxLimit="100" dataNode="dn1"/>  
  
  <!-- 定义个数据的操作节点, 以后这个节点会进行一些苦逼哦分离使用 -->  
  <!-- dataHost="localhost1" :定义数据节点的逻辑名称 -->  
  <!-- database="zsl":定义数据界定啊要使用的数据库名称 -->  
  <dataNode name="dn1" dataHost="localhost1" database="zsl" />  
  <!-- 定义数据节点, 包括了各种逻辑项的配置 -->  
  <dataHost name="localhost1" maxCon="1000" minCon="10" balance="0"  
    writeType="0" dbType="mysql" dbDriver="native" switchType="1" slaveThreshold=  
100">  
    <!-- 配置真实 MySQL与MyCat的心跳 -->  
    <heartbeat>select user()</heartbeat>  
    <!-- can have multi write hosts -->  
    <!-- 配置真实的MySQL的连接路径 -->  
    <writeHost host="hostM1" url="192.168.1.165:3306" user="root"  
      password="hahaha">  
    </writeHost>  
  </dataHost>  
</mycat:schema>  
-->
```

该文件由如下几个部分组成:

##### A.定义MyCat整体的配置项:

```
<!-- 定义一个MyCat的模式, 此处定义了一个路基数据库名称TestDB -->  
<!-- "checkSQLschema" :描述的是当前的连接是否需要检测数据的模式 -->  
<!-- "sqlMaxLimit" :表示返回的最大的数据量的行数 -->  
<!-- "dataNode="dn1"" :该操作使用的数据节点是dn1的逻辑名称 -->  
<schema name="TESTDB" checkSQLschema="false" sqlMaxLimit="100" dataNode="dn1"/>
```

## B.定义数据节点项:

```
<!-- dataHost="localhost1" :定义数据节点的逻辑名称 -->
<!-- database="zsl":定义数据界定啊要使用的数据库名称 -->
<dataNode name="dn1" dataHost="localhost1" database="zsl" />
```

## C.定义数据主机的信息:

```
<!-- 定义数据节点，包括了各种逻辑项的配置 -->
<dataHost name="localhost1" maxCon="1000" minCon="10" balance="0"
    writeType="0" dbType="mysql" dbDriver="native" switchType="1" slaveThreshold="1
0">
    <!-- 配置真实 MySQL与MyCat的心跳 -->
    <heartbeat>select user()</heartbeat>
    <!-- can have multi write hosts -->
    <!-- 配置真实的MySQL的连接路径 -->
    <writeHost host="hostM1" url="192.168.1.165:3306" user="root"
        password="hahaha">
    </writeHost>
</dataHost>
```

### 定义数据主机信息

```
<dataHost name="localhost1" maxCon="1000" minCon="10" balance="0"
    writeType="0" dbType="mysql" dbDriver="native" switchType="1" slaveThreshold="1
0">
```

balance属性负载均衡类型;

负载均衡类型，目前的取值有4种:

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;

balance="1",全部的readHost与stand bywriteH魔术贴参与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;

balance="2"，所有读操作都随机的再writeHost,readhost上分发;

balance="3"，所有读请求随机的分发到writeHost对应的readhost执行，writerHost不负读压力，注意balance=3只在1.4级其月以后版本有1.3没有;

writeType 属性，负载均衡类型;

负载均衡类型，目前的取值有三种:

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,重新启动后以切换后的为准，切换记录在配置文件中: dnindex.properties

writeType="1",所有写操作都随机的发送到配置writeHost;

writeType="2",没实现

switchType 属性，有三种取值

"-1"：表示不自动切换;

"1"：默认值，自动切换;

"2"：基于MySQL主从同步的状态决定是否切换;

## 5.【mycat-server-01】 启动mycat服务:

/usr/local/mycat/bin/mycat console;

### MyCat操作命令

启动MyCat : mycat start;  
停止Mycat : mycat stop;  
前台运行 : mycat console;  
重启服务 : mycat restart;  
暂停 : mycat pause;  
查看启动状态:mycat sttus;

## 6.【mysql-server-single】 如果要想操作Mycat必须使用mysql的命令。

对于连接的时候需要考虑一个端口问题，在Mycat里面提供有两类的操作端口:

- |- 管理端口 : 9066 (查看 Mycat的运行状态):
- |- 数据端口 : 8066 (进行数据的CRUD操作)

如果要知道连接的用户名则可以查看配置文件:

more /usr/local/mycat/conf/server.xml

```
<user name="root">  
  <property name="password">123456</property>  
  <property name="schemas">TESTDB</property>  
</user>
```

### 连接MyCat管理界面

mysql -u root -p123456 -h 192.168.1.166 -P9066 -DTESTDB;

查看所有的可用命令: show @@help  
查看所有的逻辑数据 show @@database;  
查看所有的数据节点: show @@datanode;  
查看所有的数据源: show @@datasource;

### 连接MyCat数据操作端口:

mysql -u root -p123456 -h 192.168.1.166 -P8066 -DTESTDB;

可以看见的都是逻辑数据库: show databases;  
使用指定的数据库: use TESTDB;  
查看所有的数据表: show tables;  
查看所有的数据: select \* from member;