[**Mycat入门教程**](http://blog.csdn.net/u013467442/article/details/56955846)

标签： [数据库](http://www.csdn.net/tag/%e6%95%b0%e6%8d%ae%e5%ba%93)[mysql](http://www.csdn.net/tag/mysql)

2017-02-24 22:50 1061人阅读 [评论](http://blog.csdn.net/u013467442/article/details/56955846#comments)(0) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/u013467442/article/details/56955846#report)

http://static.blog.csdn.net/images/category_icon.jpg分类：

MySQL*（9）* http://static.blog.csdn.net/images/arrow_triangle%20_down.jpg

版权声明：本文为原创文章，转载请标明出处。

目录[(?)[+]](http://blog.csdn.net/u013467442/article/details/56955846)

**mycat介绍**

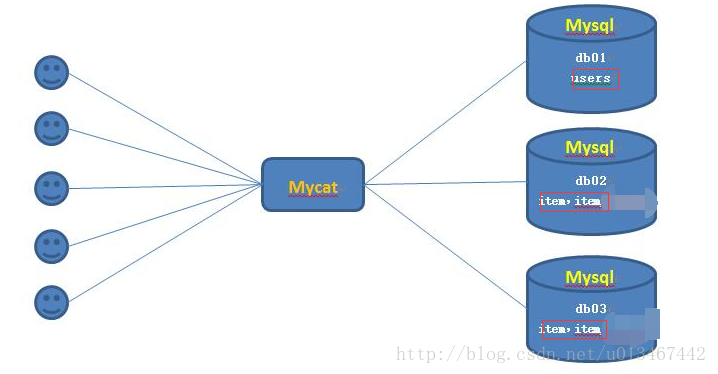
* 介绍在官方网站上有比较详细的介绍，在这里复制粘贴没什么意思，大家到官网上看
* [官网链接](http://www.mycat.org.cn/)

**前置条件**

本教程是在window环境下运行的，实际生产推荐在Linux上运行。   
必备条件(自行安装,如果不会装请先打好基础在来学习)：

* JDK：建议是1.7及其以上。
* MySQL：必须是5.5及其以上。

**拓扑结构**



* 两张表users和item，三个数据库db01,db02,db03（三个库在一个数据库实例上）
* users只在db01中存储。
* item表被分割到db02和db03中存储。

create database db01;

CREATE TABLE users (

id INT NOT NULL AUTO\_INCREMENT,

name varchar(50) NOT NULL default '',

indate DATETIME NOT NULL default '0000-00-00 00:00:00',

PRIMARY KEY (id)

)AUTO\_INCREMENT= 1 ENGINE=InnoDB DEFAULT CHARSET=utf8;

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8

在db02和db03中分别创建item表，SQL脚本如下

create database db02;

CREATE TABLE item (

id INT NOT NULL AUTO\_INCREMENT,

value INT NOT NULL default 0,

indate DATETIME NOT NULL default '0000-00-00 00:00:00',

PRIMARY KEY (id)

)AUTO\_INCREMENT= 1 ENGINE=InnoDB DEFAULT CHARSET=utf8;

create database db03;

CREATE TABLE item (

id INT NOT NULL AUTO\_INCREMENT,

value INT NOT NULL default 0,

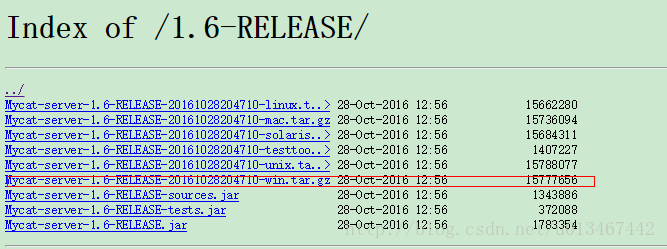
indate DATETIME NOT NULL default '0000-00-00 00:00:00',

PRIMARY KEY (id)

)AUTO\_INCREMENT= 1 ENGINE=InnoDB DEFAULT CHARSET=utf8;

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15

**开始使用**

* 先打mycat官方网站上下载安装包，网址为[mycat](http://dl.mycat.io/1.6-RELEASE/) 下载图如下所示：   
  
* 然后在conf目录下编辑service.xml，rule.xml，schema.xml三个文件。
* service.xml主要配置mycat服务的参数，比如端口号，myact用户名和密码使用的逻辑数据库等
* role.xml主要配置路由策略，主要有分片的片键，拆分的策略（取模还是按区间划分等）
* schema.xml文件主要配置数据库的信息，例如逻辑数据库名称，物理上真实的数据源以及表和数据源之间的对应关系和路由策略等。
* 配置如下所示：
* server.xml

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

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

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

<system>

<!--

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

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

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

<property name="frontWriteQueueSize">4096</property>

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

<property name="mutiNodePatchSize">100</property>

-->

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

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

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

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

</system>

<!-- 任意设置登陆 mycat 的用户名,密码,数据库 -->

<user name="test">

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

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

</user>

<user name="user">

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

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

<property name="readOnly">true</property>

</user>

<!--

<quarantine>

<whitehost>

<host host="127.0.0.1" user="mycat"/>

<host host="127.0.0.2" user="mycat"/>

</whitehost>

<blacklist check="false"></blacklist>

</quarantine>

-->

</mycat:server>

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22
* 23
* 24
* 25
* 26
* 27
* 28
* 29
* 30
* 31
* 32
* 33
* 34
* 35
* 36
* 37
* 38
* 39
* role.xml   
  该路由表表明采用mod2的方式进行路由。通过下面可知是对item表的id键进行取模分表的，取模分到db02和db03，而users表直接路由到db01中。

<?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:rule SYSTEM "rule.dtd">

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

<tableRule name="role1">

<rule>

<columns>id</columns>

<algorithm>mod-long</algorithm>

</rule>

</tableRule>

<function name="mod-long" class="io.mycat.route.function.PartitionByMod">

<!-- how many data nodes -->

<property name="count">2</property>

</function>

</mycat:rule>

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22
* 23
* 24
* 25
* schema.xml 数据库的url,usename,password按实际情况填写。

<?xml version="1.0"?>

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

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

<!-- 设置表的存储方式.schema name="TESTDB" 与 server.xml中的 TESTDB 设置一致 -->

<schema name="TESTDB" checkSQLschema="false" sqlMaxLimit="100">

<table name="users" primaryKey="id" dataNode="node\_db01" />

<table name="item" primaryKey="id" dataNode="node\_db02,node\_db03" rule="role1" />

</schema>

<!-- 设置dataNode 对应的数据库,及 mycat 连接的地址dataHost -->

<dataNode name="node\_db01" dataHost="dataHost01" database="db01" />

<dataNode name="node\_db02" dataHost="dataHost01" database="db02" />

<dataNode name="node\_db03" dataHost="dataHost01" database="db03" />

<!-- mycat 逻辑主机dataHost对应的物理主机.其中也设置对应的mysql登陆信息 -->

<dataHost name="dataHost01" maxCon="1000" minCon="10" balance="0" writeType="0" dbType="mysql" dbDriver="native">

<heartbeat>select user()</heartbeat>

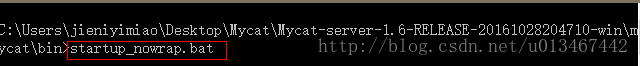
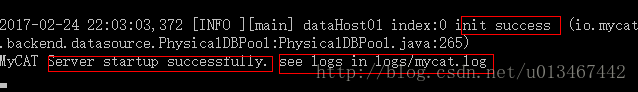
<writeHost host="server1" url="127.0.0.1:3306" user="root" password="123456"/>

</dataHost>

</mycat:schema>

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22

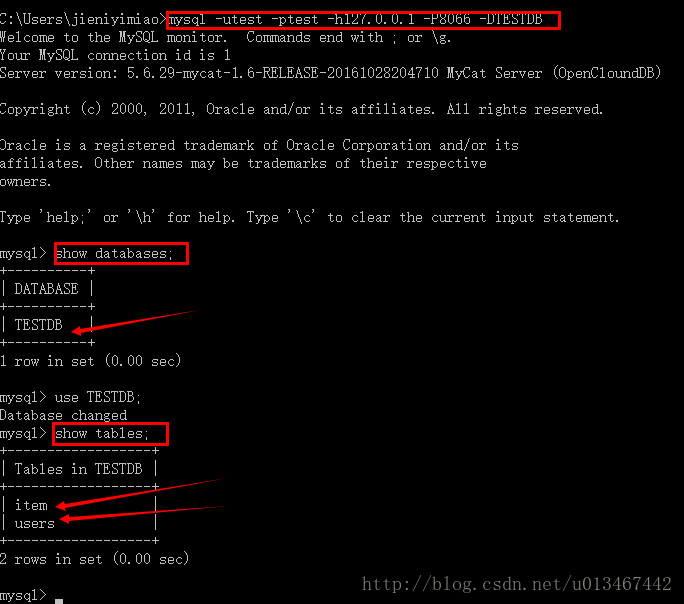
**启动测试**

* 在命令行切换到bin目录下，然后执行如下命令：   
    
  正确启动后会显示如下命令：   
    
  然后表示我们启动服务成功。

**验证结果**

* 在命令行访问mycat逻辑数据库，采用如下的命令：

mysql -utest -ptest -h127.0.0.1 -P8066 -DTESTDB

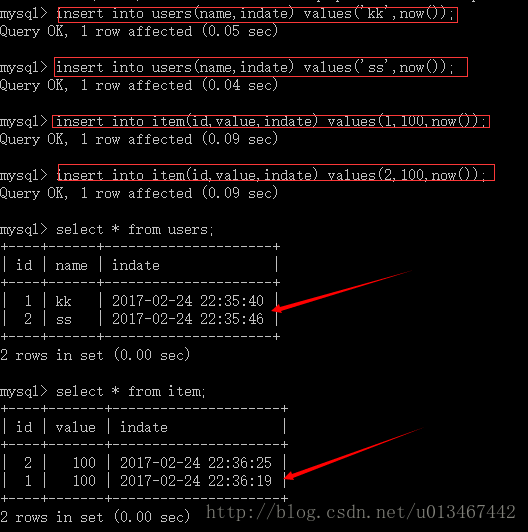
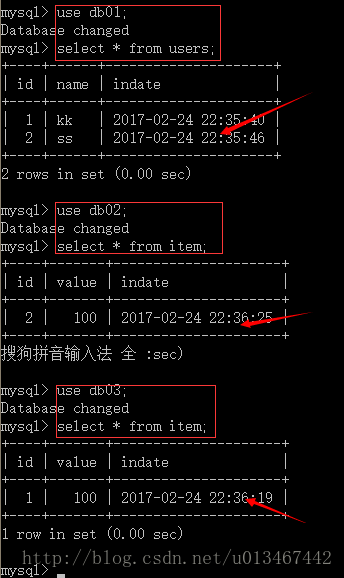
* 现在通过数据库查询数据库和表，发现只有逻辑数据库TESTDB而不是db01, db02, db03；而且表也是统一显示的，而不是分布在不同的实际数据库中。参考图片如下：   
  
  + 现在在访问Mycat在数据库中插入数据，看数据能否按照前面配置的路由规则进行分表。
  + 现在执行如下的SQL语句进行数据的插入。

insert into users(name,indate) values('kk',now());

insert into users(name,indate) values('ss',now());

insert into item(id,value,indate) values(1,100,now());

insert into item(id,value,indate) values(2,100,now());

* 1
* 2
* 3
* 4
* 然后在mycat上查看是否插入成功，下图显示插入成功。   
  
* 然后登录实际的数据库，看分表是否成功。下图显示分表成功。   
    
  图上显示，插入的users表中的数据全部在db01中，而item表中的数据通过Id取模后均匀的分布在db02和db03中。这样就根据实际的路由策略进行了分表。
* **测试完成！！~数据进行了分库分表！**

**参考文章**

[1. MySQL 高可用：mysql+mycat实现数据库分片（分库分表）](http://blog.csdn.net/kk185800961/article/details/51147029)