# MySql主从复制

## 下载地址：https://dev.mysql.com/downloads/mysql/

## 1、卸载预装mysql

## #查看已安装：

## [root@centos upload]# rpm -qa|grep mysql

## #卸载：

## [root@centos upload]# rpm -e --nodeps mysql-libs-5.1.71-1.el6.x86\_64

## #再次查看：

## [root@centos upload]# rpm -qa|grep mysql

## 2、解压安装包

## [root@centos upload]# tar -zxvf mysql-5.6.31-linux-glibc2.5-x86\_64.tar.gz -C /usr/java

## [root@centos upload]# cd /usr/java

## [root@centos java]# mv mysql-5.6.31-linux-glibc2.5-x86\_64 mysql

## 3、复制mysql的配置文件

## [root@centos java]# cd mysql

## [root@centos java]# cp support-files/my-default.cnf /etc/my.cnf

## [root@centos java]# cp support-files/mysql.server /etc/rc.d/init.d/mysql

## 4、修改my.cnf

## vim /etc/my.cnf

## basedir = /usr/java/mysql

## datadir = /usr/java/mysql/data

## log-error = /usr/java/mysql/data/error.log

## pid-file = /usr/java/mysql/data/mysql.pid

## user = root

## tmpdir = /tmp

## 5、初始化数据库

## [root@centos java]# cd /usr/java/mysql

## [root@centos mysql]# ./scripts/mysql\_install\_db --user=root

## --basedir=/usr/java/mysql --datadir=/usr/java/mysql/data

## --pid-file=/usr/java/mysql/data/mysql.pid --tmpdir=/tmp

## 6、启动和关闭mysql

## [root@centos mysql]# service mysql start

## Starting MySQL.. [确定]

## [root@centos mysql]# service mysql stop

## Shutting down MySQL.. [确定]

## [root@centos mysql]# service mysql restart

## Shutting down MySQL..

## Starting MySQL.. [确定]

## 7、配置mysql命令支持

## 如果提示没有mysql命令，需要添加软连接

## [root@centos mysql]# ln -s /usr/java/mysql/bin/mysql /usr/bin/mysql

## 8、修改MySQL密码

## [root@centos upload]# mysql -u root

## mysql> use mysql;

## mysql> update user set password= password("1111") where user='root';

## mysql> flush privileges;

## 9、开放远程登录权限

## mysql>GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'%' IDENTIFIED BY '1111' WITH GRANT OPTION;

## mysql>FLUSH PRIVILEGES;

## 10、设置开机启动

## [root@centos mysql]# chkconfig mysql on

# MySQL主从复制

## 1、mysql主从简介

## 1. MySQL 默认支持主(master)从(slave)功能.

## 2. 主从复制效果：在主数据库中操作时,从同步进行变化.

## 3. 主从复制本质：主数据的操作写入到日志中,从数据库从日志中读取,进行操作.

## 2、主从备份要素：

## 1. 开启日志功能

## 2. 每个数据库需要有一个 server\_id,主 server\_id 值小于从server\_id(标识从哪server写入的)

## 3. 每个 mysql 都有一个 uuid,由于虚拟机直接进行克隆,需要修改uuid 的值(唯一识别码)

## 4. 必须要在主数据库中有一个用户具有被从数据库操作的权限.

## 3、克隆mysql1的虚拟机

## 作为从数据库

## 4、配置主数据库

## 修改主数据库的my.cnf文件

## log\_bin=master\_log

## server\_id=1

## 重启mysql

## [root@centos upload]# service mysql restart

## 5、配置从数据库

## 1、修改server\_id

## server\_id=2

## 2、data文件夹auto.cnf编写当前mysql的uuid

## server-uuid=b144a074-6ebe-11e9-b501-000c293cc59c

## 3、重启mysql

## [root@centos upload]# service mysql restart

## 4、修改slave

## mysql> stop slave;

## mysql> change master to master\_host='192.168.233.137',master\_user='root',master\_password='1111',master\_log\_file='master\_log.000001';

## mysql> start slave;

## 5、查看slave状态

## mysql>show slave status \G;

## io线程和sql线程已开启：

## Slave\_IO\_Running:Yes

## Slave\_SQL\_Running:Yes

## Last\_IO\_Errno:0

## Last\_IO\_Error:

## Last\_SQL\_Errno:0

## Last\_SQL\_Error:

## 只要没有错误,说明配置成功主从关系：

## 6、验证主从关系

## 在主数据库中新建数据库,新建表,添加数据,观察从数据库的

# MyCat

## 1、MyCat简介

## MyCAT是一个数据库中间件。国产开源项目，前身是cobar项目。

## 2、Mycat对多数据库的支持

## MySQL、甲骨文、SQLServer、PostgreSQL、mongoDB

## 3、MyCAT架构

## 如图所示：MyCAT使用Mysql的通讯协议模拟成了一个Mysql服务器，所有能使用Mysql的客户端

## 以及编程语言都能将MyCAT当成是Mysql Server来使用，不必开发新的客户端协议。

## 4、MyCat分库分表

## 垂直分割（分库）：指按照业务将表进行分类，分布到不同的数据库上面，这样也就将数据或者

## 说压力分担到不同的库上面，就是将不同业务的表分配到不同的库中。

## 水平分割（分表）：一个表格的数据按照行分割到多个节点上，就是将表中的数据平均分到不同库中。

## 典型的分片规则：

## 根据主键编号进行hash、求余，如hash(key)%5

## 5、MyCat安装

## 5.1、下载mycat

## 官方网站：http://www.mycat.org.cn/

## github地址：https://github.com/MyCATApache

## 5.2、安装Mycat

## 1、把MyCat的压缩包上传到linux服务器

## 2、解压缩，得到mycat目录

## [root@centos upload]# tar -zxvf Mycat-server-1.6-RELEASE-20161028204710-linux.tar.gz -C /usr/java

## 3、启动和关闭MyCat

## 进入mycat/bin，启动MyCat

## 启动命令：./mycat start

## 停止命令：./mycat stop

## 重启命令：./mycat restart

## 查看状态：./mycat status

## 4、Mycat分库分表和读写分离

## 需求：把商品表分片存储到两个数据节点上。

## 安装环境：

## mysql节点1环境

## 操作系统版本 : centos6.5

## 数据库版本 :mysql-5.6

## 数据库名 : db1

## ip:192.168.25.134

## mysql节点2环境

## 操作系统版本 :centos6.5

## 数据库版本 :mysql-5.6

## 数据库名 : db2

## ip:192.168.25.135

## mysql节点3环境

## 操作系统版本 :centos6.5

## 数据库版本 :mysql-5.6

## 数据库名 : db3

## ip:192.168.25.136

## mycat节点环境

## 操作系统版本 :centos6.5

## mycat版本：1.6release

## ip:192.168.25.137

## 5、MyCat重要概念

## 1、逻辑库（schema）：一个包含了所有数据库的逻辑上的数据库

## 2、逻辑表（table）：一个包含了所有表的逻辑上的表

## 3、数据主机（dataHost）：数据库软件安装到哪个服务器上

## 4、数据节点（dataNode）：数据库软件中的 database

## 5、分片规则（rule）：拆分规则

## 6、配置schema.xml

## Schema.xml介绍：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。

## schema.xml配置

## <?xml version="1.0"?>

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

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

## <schema name="usian" checkSQLschema="false" sqlMaxLimit="100">

## <table name="tb\_content" dataNode="dn1,dn2,dn3" rule="crc32slot" />

## <table name="tb\_content\_category" dataNode="dn1,dn2,dn3" rule="crc32slot1"/>

## <table name="tb\_item" dataNode="dn1,dn2,dn3" rule="crc32slot2" />

## <table name="tb\_item\_cat" dataNode="dn1,dn2,dn3" rule="crc32slot3" />

## <table name="tb\_item\_desc" dataNode="dn1,dn2,dn3" rule="crc32slot4" />

## <table name="tb\_item\_param" dataNode="dn1,dn2,dn3" rule="crc32slot5" />

## <table name="tb\_item\_param\_item" dataNode="dn1,dn2,dn3" rule="crc32slot6" />

## <table name="tb\_order" dataNode="dn1,dn2,dn3" rule="crc32slot7" />

## <table name="tb\_order\_item" dataNode="dn1,dn2,dn3" rule="crc32slot8" />

## <table name="tb\_order\_shipping" dataNode="dn1,dn2,dn3" rule="crc32slot9" />

## <table name="tb\_user" dataNode="dn1,dn2,dn3" rule="crc32slot10" />

## </schema>

## <dataNode name="dn1" dataHost="localhost1" database="db1" />

## <dataNode name="dn2" dataHost="localhost1" database="db2" />

## <dataNode name="dn3" dataHost="localhost1" database="db3" />

## <dataHost name="localhost1" maxCon="1000" minCon="10" balance="0"

## writeType="0" dbType="mysql" dbDriver="native" switchType="1" slaveThreshold="100">

## <heartbeat>select user()</heartbeat>

## <writeHost host="hostM1" url="192.168.233.137:3306" user="root" password="1111">

## <readHost host="hostS2" url="192.168.233.138:3306" user="root" password="1111" />

## </writeHost>

## </dataHost>

## </mycat:schema>

## 7、配置server.xml

## server.xml介绍

## server.xml几乎保存了所有mycat需要的系统配置信息。最常用的是在此配置用户名、密码及权限。

## server.xml配置

## <user name="root">

## <property name="password">1111</property>

## <property name="schemas">usian</property>

## </user>

## <user name="user">

## <property name="password">1111</property>

## <property name="schemas">usian</property>

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

## </user>

## 8、配置rule.xml

## 分片规则

## auto-sharding-long 规则

## 以 500 万为单位,实现分片规则：

## 1-500 万保存在 db1 中, 500 万零 1 到 1000 万保存在 db2 中,1000 万零 1 到 1500 万保存在 db3 中.

## crc32slot 规则

## 在 CRUD 操作时,根据具体数据的 crc32 算法计算,数据应该保存在哪一个dataNode 中

## rule.xml配置

## 1）<columns>id</columns>中推荐配置主键列

## 2）所有的 tableRule 只能使用一次。如果需要为多个表配置相同的分片规则，那么需要在此重新定义该规则。

## 3) 要分片的数据库节点数量，必须指定，否则没法分片

## <tableRule name="crc32slot1">

## <rule>

## <columns>id</columns>

## <algorithm>crc32slot</algorithm>

## </rule>

## </tableRule>

## <tableRule name="crc32slot2">

## <rule>

## <columns>id</columns>

## <algorithm>crc32slot</algorithm>

## </rule>

## </tableRule>

## <tableRule name="crc32slot3">

## <rule>

## <columns>id</columns>

## <algorithm>crc32slot</algorithm>

## </rule>

## </tableRule>

## <tableRule name="crc32slot4">

## <rule>

## <columns>item\_id</columns>

## <algorithm>crc32slot</algorithm>

## </rule>

## </tableRule>

## <tableRule name="crc32slot5">

## <rule>

## <columns>id</columns>

## <algorithm>crc32slot</algorithm>

## </rule>

## </tableRule>

## <tableRule name="crc32slot6">

## <rule>

## <columns>id</columns>

## <algorithm>crc32slot</algorithm>

## </rule>

## </tableRule>

## <tableRule name="crc32slot7">

## <rule>

## <columns>order\_id</columns>

## <algorithm>crc32slot</algorithm>

## </rule>

## </tableRule>

## <tableRule name="crc32slot8">

## <rule>

## <columns>id</columns>

## <algorithm>crc32slot</algorithm>

## </rule>

## </tableRule>

## <tableRule name="crc32slot9">

## <rule>

## <columns>order\_id</columns>

## <algorithm>crc32slot</algorithm>

## </rule>

## </tableRule>

## <tableRule name="crc32slot10">

## <rule>

## <columns>id</columns>

## <algorithm>crc32slot</algorithm>

## </rule>

## </tableRule>

## <function name="crc32slot" class="io.mycat.route.function.PartitionByCRC32PreSlot">

## <property name="count">3</property><!-- 要分片的数据库数量，必须指定，否则没法分片 -->

## </function>

## 9、测试

## 创建库 在主数据库中分别创建db1、db2、db3

## 创建表并插入数据

## 配置完毕后，重新启动mycat。

## 使用mysql客户端连接mycat，创建表并插入数据。

## 项目测试

## 修改数据库url

## spring:

## application:

## name: usian-item-service

## datasource:

## driver-class-name: com.mysql.jdbc.Driver

## url: jdbc:mysql://192.168.233.139:8066/usian?characterEncoding=UTF-8

## username: root

## password: 1111

## type: com.alibaba.druid.pool.DruidDataSource