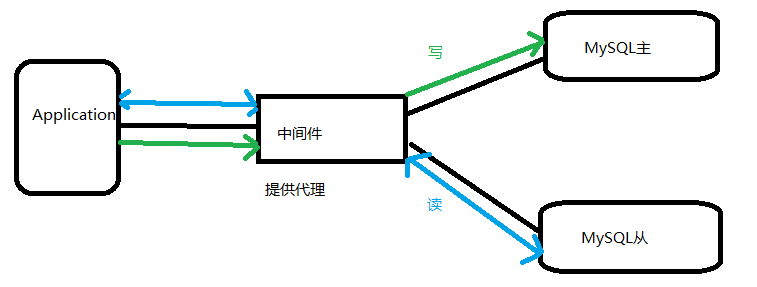
# **MySQL读写分离的概述**

**Mysql作为目前世界上使用最广泛的免费数据库，相信所有从事系统运维的工程师都一定接触过。但在实际的生产环境中，由单台Mysql作为独立的数据库是完全不能满足实际需求的，无论是在安全性，高可用性以及高并发等各个方面。**

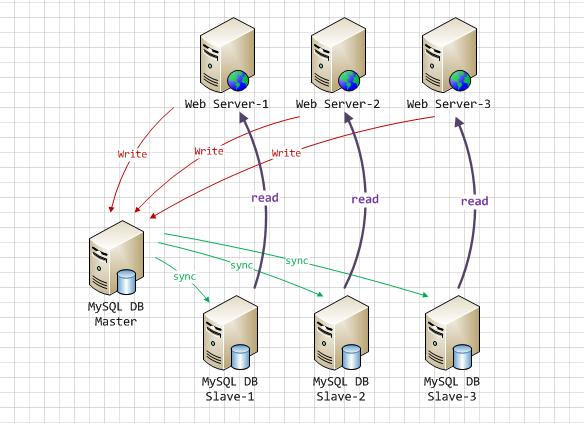
**因此，****一般来说都是通过 主从复制（Master-Slave）的方式来同步数据，再通过读写分离（MySQL-Proxy/Amoeba）来提升数据库的并发负载能力 这样的方案来进行部署与实施的。**

## **读写分离工作原理：**

**基本的原理是让主数据库处理事务性增、改、删操作（INSERT、UPDATE、DELETE），而从数据库处理SELECT查询操作。数据库复制被用来把事务性操作导致的变更同步到集群中的从数据库。**

****

**数据内部交换过程：**

****

## **为什么要读写分离：**

**面对越来越大的访问压力，单台的服务器的性能成为瓶颈，需要分担负载**

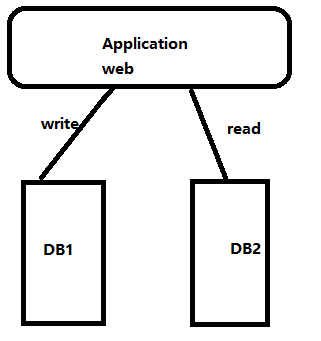
1. **主从只负责各自的写和读，极大程度的缓解X锁和S锁争用**
2. **从库可配置myisam引擎，提升查询性能以及节约系统开销**
3. **增加冗余，提高可用性**

## **实现读写分离的方式：**

**一般有两种方式实现**

### **应用程序层实现，网站的程序实现**

**应用程序层实现指的是在应用程序内部及连接器中实现读写分离**

****

**优点：**

**A：应用程序内部实现读写分离，安装既可以使用**

**B：减少一定部署难度**

**C：访问压力在一定级别以下，性能很好**

**缺点：**

**A：架构一旦调整，代码要跟着变**

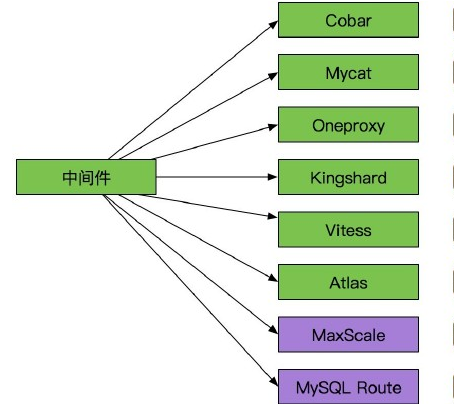
**B：难以实现高级应用，如自动分库，分表**

**C：无法适用大型应用场景**

### **中间件层实现**

**中间件层实现是指在外部中间件程序实现读写分离**

**常见的中间件程序：**



**Cobar:**

**阿里巴巴B2B开发的关系型分布式系统，管理将近3000个MySQL实例。 在阿里经受住了考验，后面由于作者的走开的原因cobar没有人维护 了，阿里也开发了tddl替代cobar。**

**MyCAT:**

**社区爱好者在阿里cobar基础上进行二次开发，解决了cobar当时存 在的一些问题，并且加入了许多新的功能在其中。目前MyCAT社区活 跃度很高，目前已经有一些公司在使用MyCAT。总体来说支持度比 较高，也会一直维护下去，**

**OneProxy:**

**数据库界大牛，前支付宝数据库团队领导楼总开发，基于mysql官方 的proxy思想利用c进行开发的，OneProxy是一款商业收费的中间件， 楼总舍去了一些功能点，专注在性能和稳定性上。有朋友测试过说在 高并发下很稳定。**

**Vitess:**

**这个中间件是Youtube生产在使用的，但是架构很复杂。 与以往中间件不同，使用Vitess应用改动比较大要 使用他提供语言的API接口，我们可以借鉴他其中的一些设计思想。**

**Kingshard:**

**Kingshard是前360Atlas中间件开发团队的陈菲利用业务时间 用go语言开发的，目前参与开发的人员有3个左右， 目前来看还不是成熟可以使用的产品，需要在不断完善。**

**Atlas:**

**360团队基于mysql proxy 把lua用C改写。原有版本是支持分表， 目前已经放出了分库分表版本。在网上看到一些朋友经常说在高并 发下会经常挂掉，如果大家要使用需要提前做好测试。**

**MaxScale与MySQL Route:**

**这两个中间件都算是官方的吧，MaxScale是mariadb (MySQL原作者维护的一个版本)研发的，目前版本不支持分库分表。MySQL Route是现在MySQL官方Oracle公司发布出来的一个中间件。**

**优点：**

**A：架构设计更灵活**

**B：可以在程序上实现一些高级控制，如：透明化水平拆分，failover,监控**

**C：可以依靠些技术手段提高mysql性能，**

**D：对业务代码的影响小，同时也安全**

**缺点：**

**需要一定的开发运维团队的支持**

# **什么是MYCAT**

 一个彻底开源的，面向企业应用开发的大数据库集群

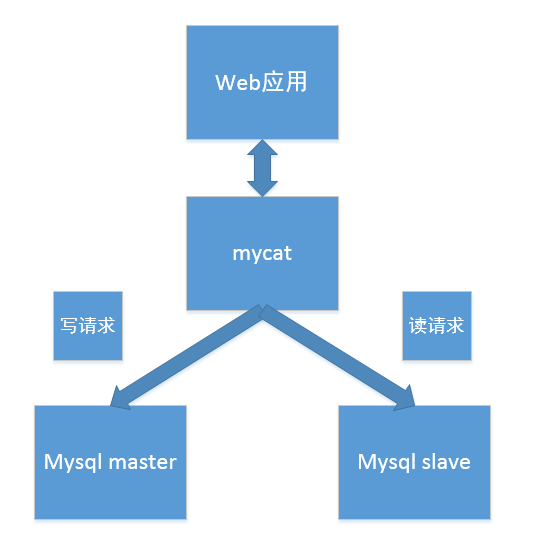
 支持事务、ACID、可以替代MySQL的加强版数据库

 一个可以视为MySQL集群的企业级数据库，用来替代昂贵的Oracle集群

 一个融合内存缓存技术、NoSQL技术、HDFS大数据的新型SQL Server

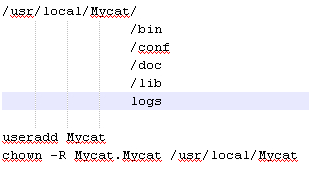
 结合传统数据库和新型分布式数据仓库的新一代企业级数据库产品

 一个新颖的数据库中间件产品



**2.1 服务安装与配置**

MyCAT有提供编译好的安装包，支持windows、Linux、Mac、Solaris等系统上安装与运行。 官方下载主页<http://www.mycat.io>

linux下可以下载Mycat-server-xxxxx.linux.tar.gz 解压在某个目录下，注意目录不能有空格，在Linux(Unix)下，建议放在usr/local/Mycat目录下，如下： 

下面是修改MyCAT用户密码的方式(仅供参考)：



目录解释如下：

bin 程序目录，存放了window版本和linux版本，除了提供封装成服务的版本之外，也提供了nowrap的shell脚本命令，方便大家选择和修改，进入到bin目录：

Linux下运行：./mycat console,首先要chmod +x \*

注：mycat支持的命令{ console | start | stop | restart | status | dump }

conf目录下存放配置文件，server.xml是Mycat服务器参数调整和用户授权的配置文件，schema.xml是逻辑库定义和表以及分片定义的配置文件，rule.xml是分片规则的配置文件，分片规则的具体一些参数信息单独存放为文件，也在这个目录下，配置文件修改，需要重启Mycat或者通过9066端口reload.

lib目录下主要存放mycat依赖的一些jar文件.

日志存放在logs/mycat.log中，每天一个文件，日志的配置是在conf/log4j.xml中，根据自己的需要，可以调整输出级别为debug，debug级别下，会输出更多的信息，方便排查问题.

注意：Linux下部署安装MySQL，默认不忽略表名大小写，需要手动到/etc/my.cnf 下配置 lower\_case\_table\_names=1 使Linux环境下MySQL忽略表名大小写，否则使用MyCAT的时候会提示找不到表的错误！

Mycat 需要安装JDK 1.7 或者以上版本

第一步：下载jdk-8u131-linux-x64.tar.gz文件 http://haixi.sfkcn.com:8080/201704/tools/jdk-linux-x64.tar.gz

第二步：新建/usr/java文件夹，将jdk-8u131-linux-x64.tar.gz放到该文件夹中，并解压到当前目录下

第三步：配置环境变量 在/etc/profile底部加入如下内容：

JAVA\_HOME=/usr/java/jdk1.8.0\_131 PATH=$JAVA\_HOME/bin:$PATH CLASSPATH=$JAVA\_HOME/jre/lib/ext:$JAVA\_HOME/lib/tools.jar export PATH JAVA\_HOME CLASSPATH[root@xuegod68 java]# source /etc/profile #使环境变量生效

查看java –version

**2.2 服务启动与启动设置**

MyCAT在Linux中部署启动时，首先需要在Linux系统的环境变量中配置MYCAT\_HOME,操作方式如下：

1. vi /etc/profile,在系统环境变量文件中增加

**MYCAT\_HOME=/usr/local/mycat PATH=$MYCAT\_HOME/bin:$PATH**

2) 执行 source /etc/profile 命令，使环境变量生效。

如果是在多台Linux系统中组建的MyCAT集群，那需要在MyCAT Server所在的服务器上配置对其他ip和主机名的映射，配置方式如下：

vi /etc/hosts

例如：我有3台机器，配置如下：

IP 主机名：

10.10.10.67 xuegod67.cn

10.10.10.68 xuegod68.cn

10.10.10.70 xuegod70.cn

编辑完后，保存文件。

(mycat的用户账号和授权信息是在conf/server.xml文件中配置

**Vim /usr/local/mycat/conf/server.xml**

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

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

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

<system>

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

</system>

<user name="root">

<property name="password">123456</property>

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

</user>

<user name="user">

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

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

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

</user>

</mycat:server>

**编辑MyCAT的配置文件schema.xml，关于dataHost的配置信息如下：**

**Vim /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="ha" checkSQLschema="false" sqlMaxLimit="100" dataNode='dn1'>

</schema>

<dataNode name="dn1" dataHost="dthost" database="ha"/>

<dataHost name="dthost" maxCon="500" minCon="10" balance="1" writeType="0" dbType="mysql" dbDriver="native" switchType="-1" slaveThreshold="100">

<heartbeat>select user()</heartbeat>

<writeHost host="xuegod68.cn" url="localhost:3306" user="root" password="123456">

</writeHost>

<writeHost host="xuegod68.cn" url="10.10.10.68:3306" user="root" password="123456" />

</dataHost>

</mycat:schema>

有两个参数需要注意，balance和 switchType。

其中，balance指的负载均衡类型，目前的取值有4种：

1. 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。

2. balance="1"，全部的readHost与stand by writeHost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。

3. balance="2"，所有读操作都随机的在writeHost、readhost上分发。

4. balance="3"，所有读请求随机的分发到wiriterHost对应的readhost执行，writerHost不负担读压力

switchType指的是切换的模式，目前的取值也有4种：

1. switchType='-1' 表示不自动切换

2. switchType='1' 默认值，表示自动切换

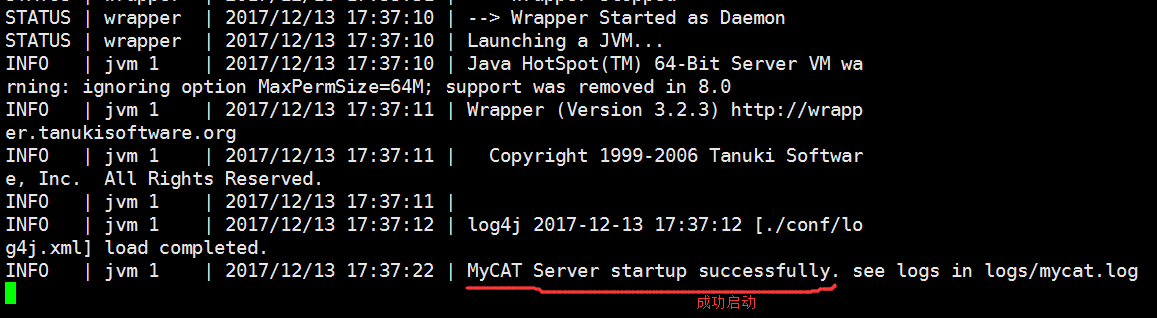
3. switchType='2' 基于MySQL主从同步的状态决定是否切换,心跳语句为 show slave status

4. switchType='3'基于MySQL galary cluster的切换机制（适合集群）（1.4.1），心跳语句为 show status like 'wsrep%'。

经过以上两个步骤的配置，就可以到/usr/local/mycat/bin 目录下执行：

./mycat start

即可启动mycat服务！



# **配置Mysql主从**

## **xuegod68上创建数据库和表，用于实现写操作：**

**[root@xuegod68 ~]# mysql -uroot -p123456**

**mysql> create database ha;**

**mysql> use ha;**

**mysql> create table test(id int,name varchar(20));**

**mysql> insert into test values(1,’man’);**

**授权给mycat登陆数据库使用的帐号**

**mysql> GRANT ALL PRIVILEGES ON \*.\* TO 'mycat'@"%" IDENTIFIED BY "123456";**

**mysql> grant replication slave on \*.\* to** [**slave@"10.10.10.%"**](mailto:slave@%2210.10.10.%25%22) **identified by "123456";**

**mysql>flush privileges;**

**mysqldump –uroot –p123456 -B ha>ha.sql #导出给从库**

**关闭密码强度审计，在my.cnf添加validate-password=off**

## **xuegod70上导入数据库和表，用于实现读操作：**

**[root@xuegod70 ~]# mysql -uroot -p123456<ha.sql**

**授权给mycat登陆数据库使用的帐号**

**mysql> grant all on db.\* to mycat@'%' identified by '123456';**

**mysql>flush privileges;**

**mysql>stop slave; #停止slave**

**mysql> change master to master\_host='10.10.10.68',master\_user='slave',master\_password='123456';**

**mysql> start slave; #启动slave**

**mysql> show slave status\G 查看状态 ，有两个yes主从同步成功！**

## **启动服务mycat服务**

**[root@xuegod67 ~]# /usr/local/mycat/bin/mycat start**

**测试读写分离**

# **当我们slave挂掉后咋样呢**

**模拟故障：从服务器挂掉了**

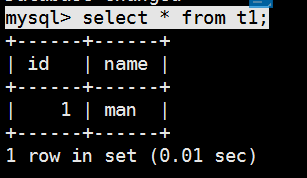
**[root@xuegod70 ~]# systemctl stop mysqld**

**在客户端上测试读写**

**[root@xuegod67 ~]# mysql -uroot -p123456 -h 10.10.10.67 -P8066**

**mysql> use ha;**

**mysql> select \* from t1;**



**说明读操作的路由切换到master上了，对外没有任何影响！**

# **当我们master挂掉后咋样呢**

**模拟故障：主服务器挂掉了**

**[root@xuegod68 ~]# systemctl stop mysqld**

**在客户端上测试读写**

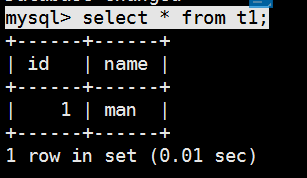
**[root@xuegod67 ~]# mysql -uroot -p123456 -h 10.10.10.67 -P8066**

**mysql> create table t3(id int);**

**ERROR 1184 (HY000): 拒绝连接 #主数据库停止了，是无法写操作的，但不影响读.**

**mysql> use ha;**

**mysql> select \* from t1;**



**同学们，好好看一下Mycat官方中文档。会有很大的收获！**