一　mycat

1，前言

－－分库分表

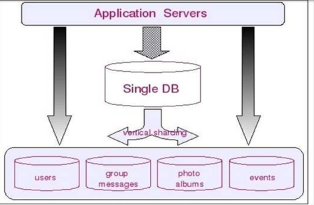
将存放在一个数据库（主机）中的数据，按照特定方式进行拆分，分散存放到多个数据库（主机）中，以达到分散单台设备负载的效果

－－垂直分割

纵向切分

将单个表，拆分成多个表，分散到不同的数据库

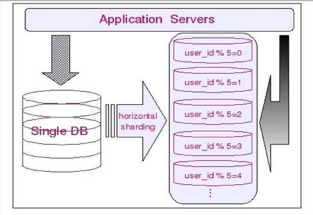
将单个数据库的多个表进行分类，按业务类别分散到不同的数据库上



－－水平分割

横向切分

按照表中某个字段的某种规则，把表中的许多记录按行切分，分散到多个数据库中



２，mycat

默认端口号：8066

－－软件介绍

Mycat是基于java的分布式数据库系统中间层，为高并发环境的分布式访问提供解决方案

支持jdbc形式连接

支持mysql,oracle,sqlserver,mongodb等

提供数据读写分离服务

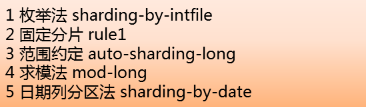
可以实现数据服务器的高可用

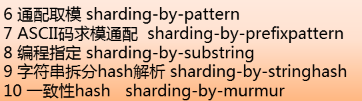
提供数据分片服务

基于阿里巴巴cobar进行研发的开源软件

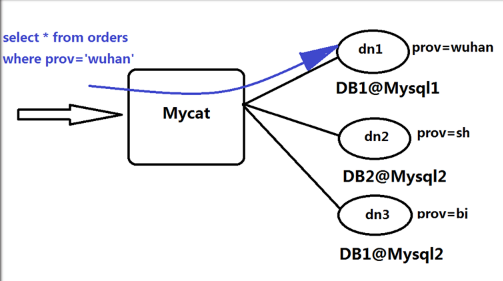
适合数据大量写入的存储需求

－－分片规则





－－工作过程



当mycat收到一个sql查询时

先解析这个sql查询涉及到的表

然后看此表的定义，如果有分片规则，则获取sql里分片字段的值，并匹配分片函数，获得分片列表

然后将sql发往这些分片取执行

最后收集和处理所有分片结果数据，并返回到客户端

－－配置mycat

１）目录说明

bin #mycat命令

Catlet #扩展功能

Conf #配置文件

Lib　　　 #mycat使用的jar

Logs #mycat启动日志和运行日志

Wrapper.log #mycat服务启动日志

Mycat.log　　　#记录sql脚本执行后的报错内容

２）修改mycat配置文件

Server.xml #设置连mycat的帐号信息

Schema.xml　　　　　#配置mycat的真实库表

Rule.xml　　　　　　　#定义mycat分片规则

[root@mycat-57 mycat]# vim conf/server.xml

<user name="admin">　　　　#定义连mycat用户信息

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

<property name="schemas">TESTDB</property>　#虚拟库/逻辑库

</user>

<user name="user">

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

<property name="schemas">TESTDB</property>　#虚拟库/逻辑库

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

</user>

[root@mycat-57 mycat]# vim conf/schema.xml

dataNode="dn1,dn2"　必须dn开头

54，55主机操作

[root@mysql-54 mysql]# vim /etc/my.cnf

[mysqld]

lower\_case\_table\_names=1 #表名忽略大小写

[root@mysql-54 ~]# systemctl restart mysqld

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

－－启动mycat

[root@mycat-57 mycat]# ./bin/mycat --help

Usage: ./bin/mycat { console | start | stop | restart | status | dump }

[root@mycat-57 mycat]# ./bin/mycat start

Starting Mycat-server...

[root@mycat-57 mycat]# ss -tunlp | grep :8066　#查看端口号，mycat是否起来

tcp LISTEN 0 100 :::8066 :::\* users:(("java",pid=6846,fd=49))

－－测试配置

格式：

Mysql -h服务器地址　-p端口　-u用户名　-p密码

例子：

[root@mysql-50 ~]# mysql -uadmin -p123456 -h192.168.4.57 -P8066

mysql> create table employee (

-> id int primary key auto\_increment ,

-> name char(20),

-> age int,

-> sharding\_id int

-> );