**使用Percona XtraBackup对MySQL进行在线热备份**

物理备份缺点：跨平台性差 备份时间长 冗余备份 浪费储存空间

Mysqldump备份缺点：效率较低 备份和还原速度慢 锁表 备份过程中 数据插入和更新操作被阻塞

Xtrabackup工具：一款强大的在线热备份工具 适合生产环境 有专业组织percona 提供

语法：

完全备份 innobackupex --user 用户名 --password 密码 备份目录名 --no-timestamp

完全恢复：inbobackupex --apply-log 目录名 //准备恢复数据

Inbobackupex --copy-back 目录名 //恢复数据

增量备份：innobackupex --user 用户名 --password 密码 \

--incrememtal=增量目录 --incremental-basedir=上次备份目录名 --no-timestamp

增量恢复：innobackupex --appley-log --redo-only 目录名 --incremental-dir=目录名 //准备恢复数据

Innobackupex --copy-back 目录名 //恢复数据



**Mysql主从复制模式**

主从同步介绍： 实现数据的自动同步的服务结构 主服务器：介绍客户端访问连接 从服务器：自动同步主服务器的数据

主从同步原理：主服务器启用binlog日志 从服务器：slave\_io复制master主机binlog日志文件里的sql命令到本机的relay-log文件里

slave\_sql：执行本机relay-log文件里的sql语句实现与master数据一致

构建思路：

配置主服务器：启用binlog日志 授权用户 查看binlog日志信息

配置从服务器：设置server\_id（多台服务器的值不可重复） 确保与主服务器数据一致 指定主库信息 启用slave程序 查看状态信息

主从从服务器：中间的服务器即做从服务器 也做主服务器 需要允许级联复制：（log\_slave\_updates）

复制模式介绍：

异步复制：主服务器执行完一次事务后 立即将结果返给客户端 不关心从服务器是否已经同步数据

半同步复制：介于以部复制和同步复制之间 主服务器在执行完一次事务后 等待至少一台从服务器同步数据完成 才将结果返回客户端

查看是否允许动态加载模块（默认是允许）：show variables like “have\_dynamic\_loading”;

**数据读写分离**

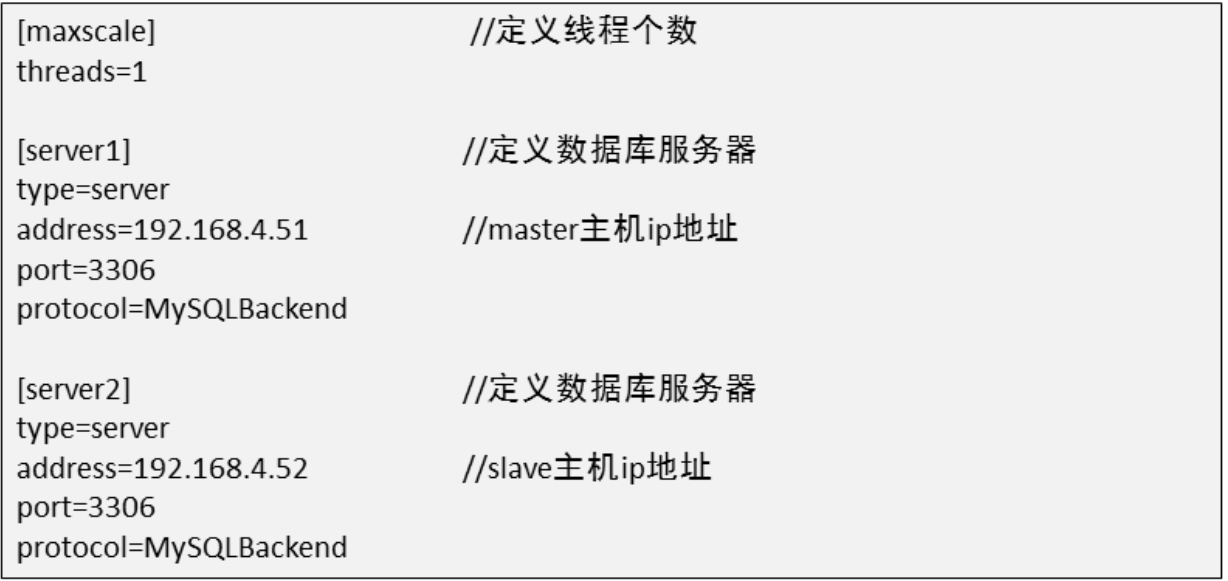
读写分离原理：有mysql代理面向客户端提供服务 收到sql写请求时 交有master服务器处理

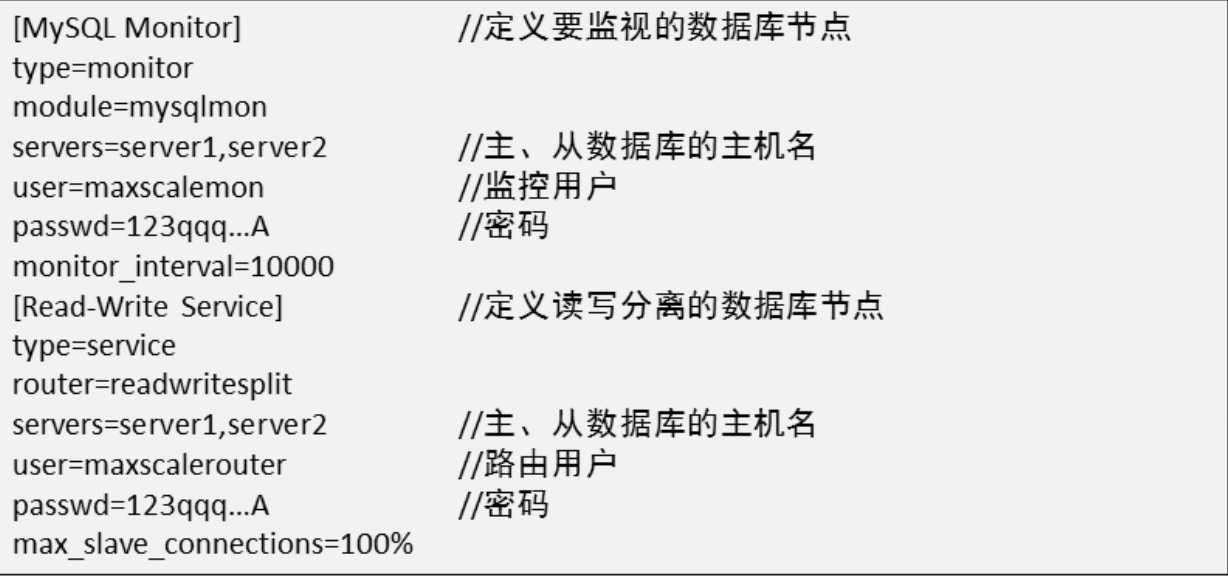
有sql命令读请求时 交给salve服务器处理

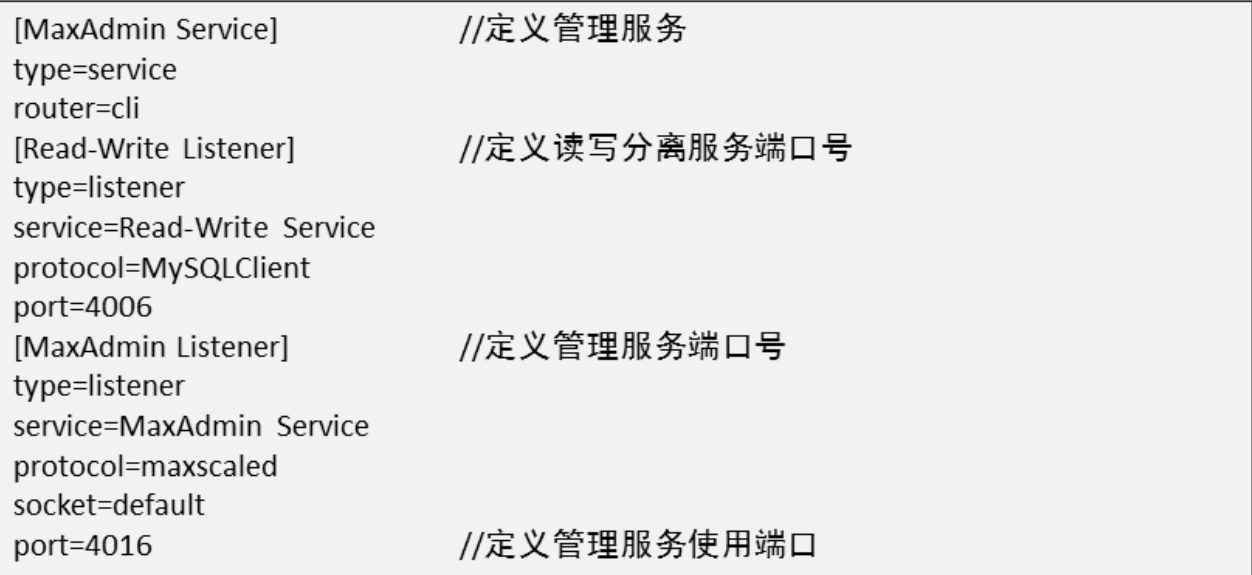
构建思路：

部署mysql一主一从结构 部署mysql代理服务器 装包 修改配置文件 启动服务 测试配置 客户端连接访问

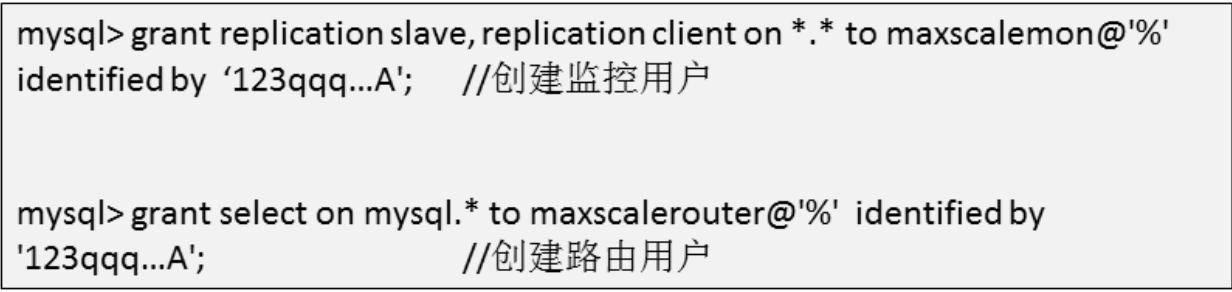
Maxscale代理软件：有mariadb开发 修改配置文件 /etc/maxscale.cnf







在主从服务器上创建授权用户



Maxscale -f /etc/maxscale.cnf //启动服务

Ss -ntulp | grep maxscale //查看服务

Kill -9 进程号 //杀死进程

Mysql -h主机IP -P端口号 -u用户名 -p密码

**数据分片 mycat服务**

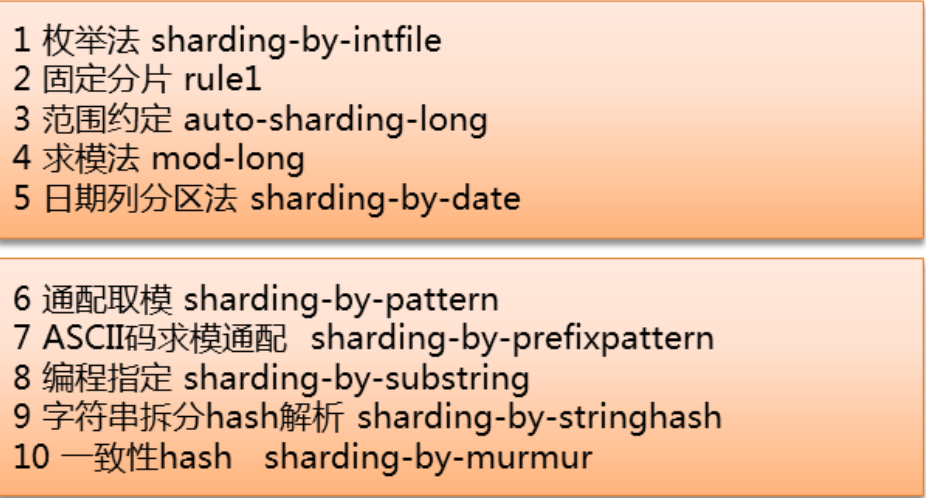
什么是分库分表：将存放在一改数据库服务器中的数据 按照特定的方式进行拆分 分散存放到多台数据库服务器中 以达到分散单台服务器的效果

横向切分：按照表中指定字段的分片规则 将表记录按行切分 分散存储到多个数据库中

纵向切分：将单个数据的多个表按照业务类型分类 分散存储到不同的数据库

Mycat是基于java大分布式存储数据库系统的中间件 为高并发环境的分布存储提供解决方案 适合数据大量写入的存储需求 支持mysql oracle sqlserver mongodb等 提供数据的读写分离服务 提供数据的分片服务 基于阿里巴巴cobar 进行研发的开源软件

Mycat支持提供10中分片规则

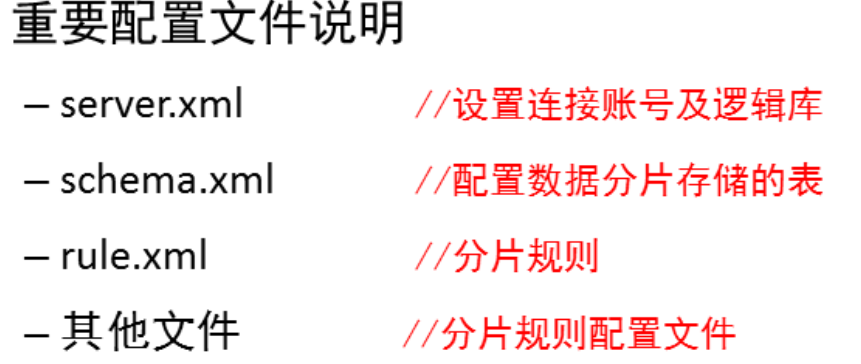
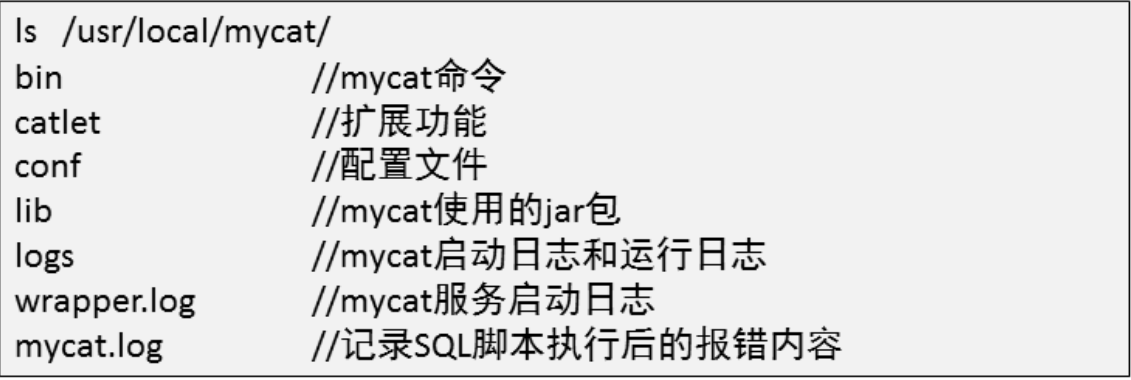


Mycat 的工作过程

当mycat收到一个sql命令时1解析sql命令涉及到的表2然后看对表的配置 如果有分片规则 则获取sql命令里分片字段的值 并匹配分片函数 获得分片规则列表3然后将sql命令发往对应的数据库服务器去执行4最后收集和处理所有分片结果数据 并返回到客户端



Mycat 的目录结构



Mha集群

是由日日本dena公司开发 是一套优秀的实现mysql高可用的解决方案 数据库的自动故障切换操作能做到在0~30秒之内完成 mha能确保在故障切换过程中最大限度的保证数据的一致性 以达到真正意义上的高可用

Mha组成

Mha manager(管理节点)： 管理所有数据库服务器 可以单独部署在一台独立的机器上 也可以部署在某台数据服务器上

Mha nod（数据节点）：存数据的mysql服务器 运行在每台mysql服务器上

Mha工作过程

具体：由manager定时探测群中的maste节点

当master故障时 manager自动将拥有最新数据的slave提升我新的master

**pxc高可用集群**

Pxc介绍：基于galera的mysql高可用集群解决方案

Galera cluster时codership公司开发的一套免费开源的高可用方案

Pxc集群主要由两部分组成：percona server with xtradb和write set replication patches (同步 多出复制插件)

pxc具体如下：数据强一性 无同步延迟 没有主从切换操作 无需使用虚拟ip 支持innodb存储引擎 多线程复制 部署使用简单 支持节点自动加入 无需手动拷贝数据

