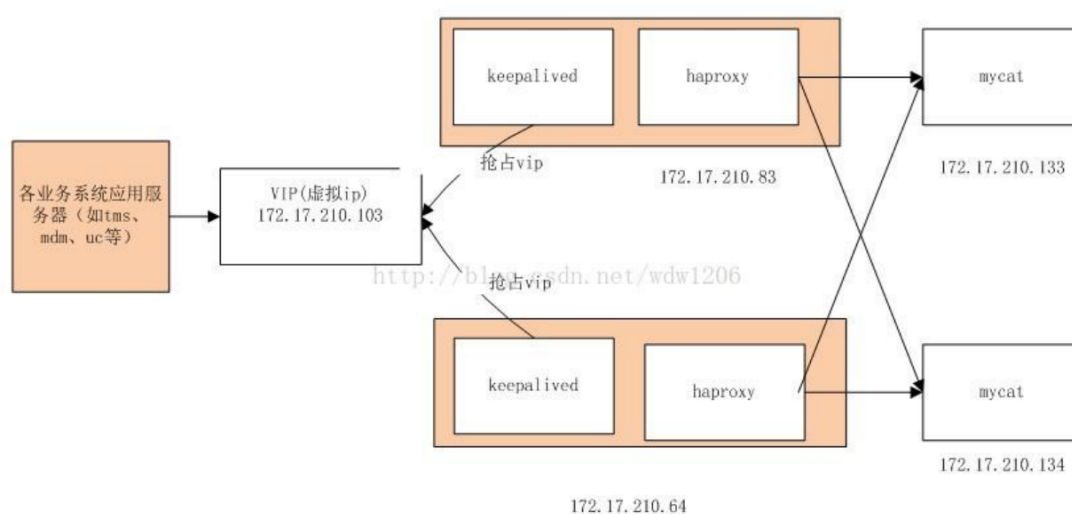


## 读写分离

一些对数据实时性要求不高的业务场景，可以考虑使用读写分离。但是对数据实时性要求比较高的场景，比如订单支付状态，还是不建议采用读写分离的，或者你在写程序时，老老实实的从写库去读取数据。我也咨询过专门做数据同步的机构，他们给出的建议是，如果你做数据的同步，你的网络延迟应该在5ms以内，这个对网络环境要求是非常高的，大家可以ping一下你网络中的其他机器，看看能不能达到这个标准。如果你的网络环境很好，达到了要求，那么使用读写分离是没有问题的，数据几乎是实时同步到读库，根本感觉不到延迟。

## haproxy + keepalived + mycat 高可用与负载均衡集群配置



## 数据库实现方案

数据库实现方案：（按数据量从小到大）单体、mycat中间件、oracle数据库、hadoop、云存储数据

## 单表数据达到多少的时候会影响数据库的查询性能？为什么？

- 一般mysql达到100w，就影响数据库的查询性能，

## 主从复制机制的原理概述是怎样的？常见的存在形式有哪些？

- mysql主从复制是master将所有的事务操作写入到binlog,slave获取binlog读入自己的中继区，然后再进行执行。
- MySQL的复制默认是异步的
- 多线程复制
- 半同步复制

## 分库分表中解释一下垂直和水平2种不同的拆分？

- 垂直拆分：是将单表，或者是有关联的表放在一个数据库，把原有的一个数据库拆分成若干个数据库。
- 水平拆分：是将一个很大的表，通过取模，按照日期范围等等拆分成若干个小表

## 分布式数据存储中间件如mycat的核心流程是什么？

sql解析 -> 数据源分配 -> 请求响应 -> 结果整合

## 解释一下全局表，ER表，分片表？

- 全局表：一个字典类数据的表，每个表都有可能用到，在各个数据节点上都会冗余。
- 分片表：按照一定的规则后，表按照设置的primaryKey来分配到不同的数据节点上。
- ER表：和分片表有外键关系的表，也是通过设置的primaryKey即与分片表的外键，和分片表按照一样的规则分配到不同的数据节点上。

## Mycat的在分库分表之后，它是怎么支持联表查询的？

- 使用好ER表
- 善用全局表
- 在sql上添加注解 `/*!mycat:catlet=io.mycat.catlets.ShareJoin */`

## 分布式ID都有哪些生成方式？

- UUID  
不适用于实际的业务需求
- 数据库自增ID
- 数据库多主模式
- 号段模式
- Redis
- 雪花算法 (Snowflake)  
Snowflake ID组成结构：正数位（占1比特）+ 时间戳（占41比特）+ 机器ID（占5比特）+ 数据中心（占5比特）+ 自增值（占12比特），总共64比特组成的一个Long类型。
- 滴滴出品 (TinyID)
- 百度 (Uidgenerator)
- 美团 (Leaf)

## mycat是怎样实现分库分表的？

mycat里面通过定义路由规则来（路由规则里面会定义分片字段，以及分片算法）。分片算法有多种，你所说的hash是其中一种，还有取模、按范围分片等等。在mycat里面，会对所有传递的sql语句做路由处理（路由处理的依据就是表是否分片，如果分片，那么需要依据分片字段和对应的分片算法来判断sql应该传递到哪一个、或者哪几个、又或者全部节点去执行）。

## mycat适用于哪些场景？相对于海量存储的Nosql的适用场景又如何？

数据量大到单机hold不住，而又不希望调整架构切换为NoSQL数据库，这个场景下可以考虑适用mycat。当然，使用前也应该做规划，哪些表需要分片等等。另外mycat对跨库join的支持不是很好，在使用mycat的时候要注意规避这种场景。

## Mycat 中，旧系统数据如何迁移到 Mycat 中？

旧数据迁移目前可以手工导入，在 mycat 中提取配置好分配规则及后端分片数据库，然后通过 dump或 loaddata 方式导入，后续 Mycat 就做旧数据自动数据迁移工具。

## Mycat 如何对旧分片数据迁移或扩容，支持自动扩容么？

目前除了一致性 hash 规则分片外其他数据迁移比较困难，目前暂时可以手工迁移，未提供自动迁移方案。

### 数据库中间件

- Mycat是服务端
- sharding-jdbc是客户端