



# 5、深入了解性能优化

T A H N K   Y O U   F O R   W A T C H I N G

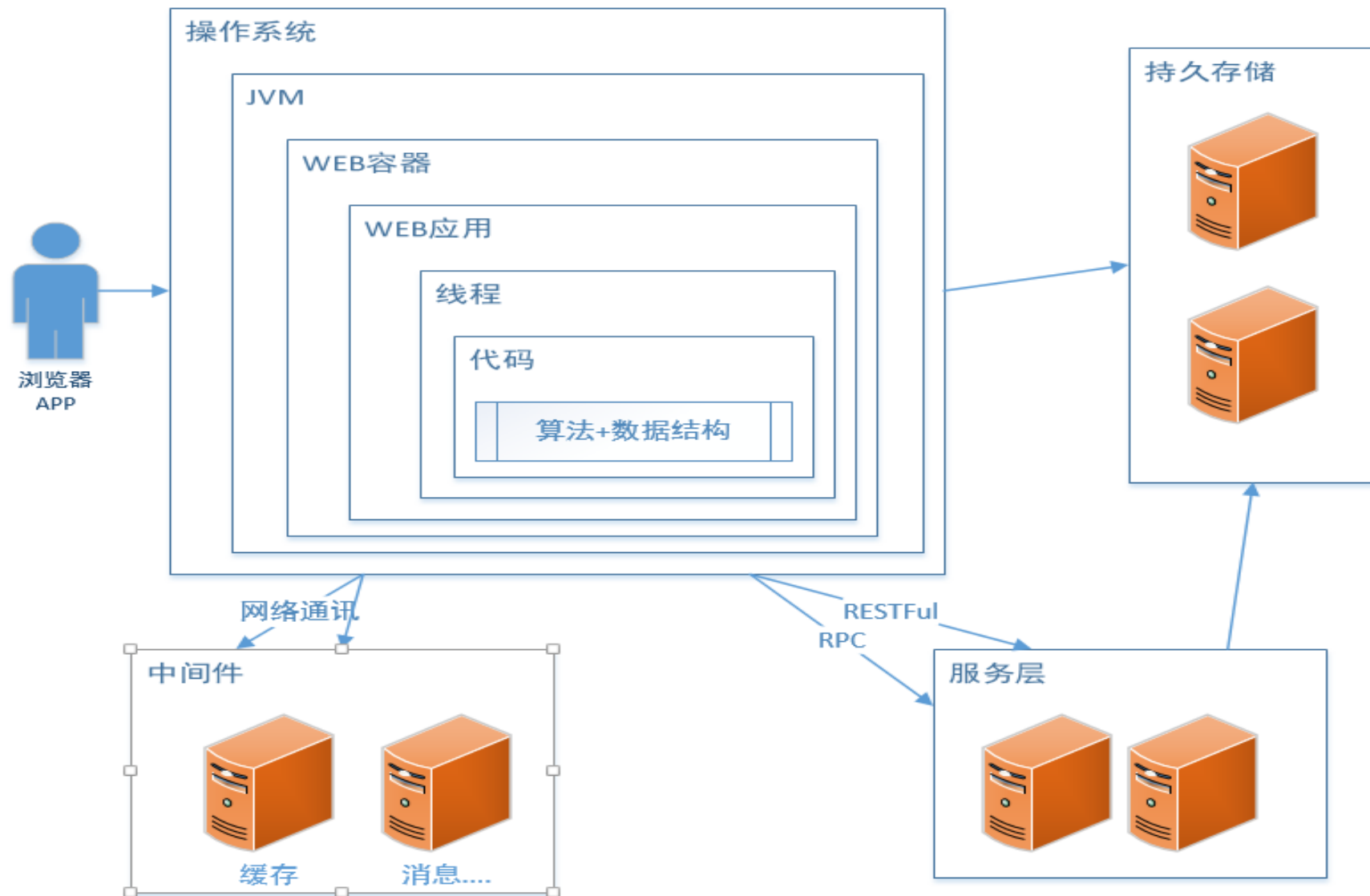


主讲老师Mark : 446106311



课程咨询安生老师 : 669100976

# 影响一个系统性能的方方面面





# 常用的性能评价/测试指标

## ■ 响应时间

- 提交请求和返回该请求的响应之间使用的时间，一般比较关注平均响应时间。
- 如：数据库查询花费的时间，将字符回显到终端上花费的时间，访问 Web 页面花费的时间；

操作	响应时间
打开一个站点	几秒
数据库查询一条记录（有索引）	十几毫秒
机械磁盘一次寻址定位	4毫秒
从机械磁盘顺序读取1M数据	2毫秒
从SSD磁盘顺序读取1M数据	0.3毫秒
从远程分布式换成Redis读取一个数据	0.5毫秒
从内存读取1M数据	十几微妙
Java程序本地方法调用	几微妙
网络传输2Kb数据	1微妙



# 常用的性能评价/测试指标

## ■ 并发数

- 指同一时刻，对服务器有实际交互的请求数。和网站在线用户数的关联。

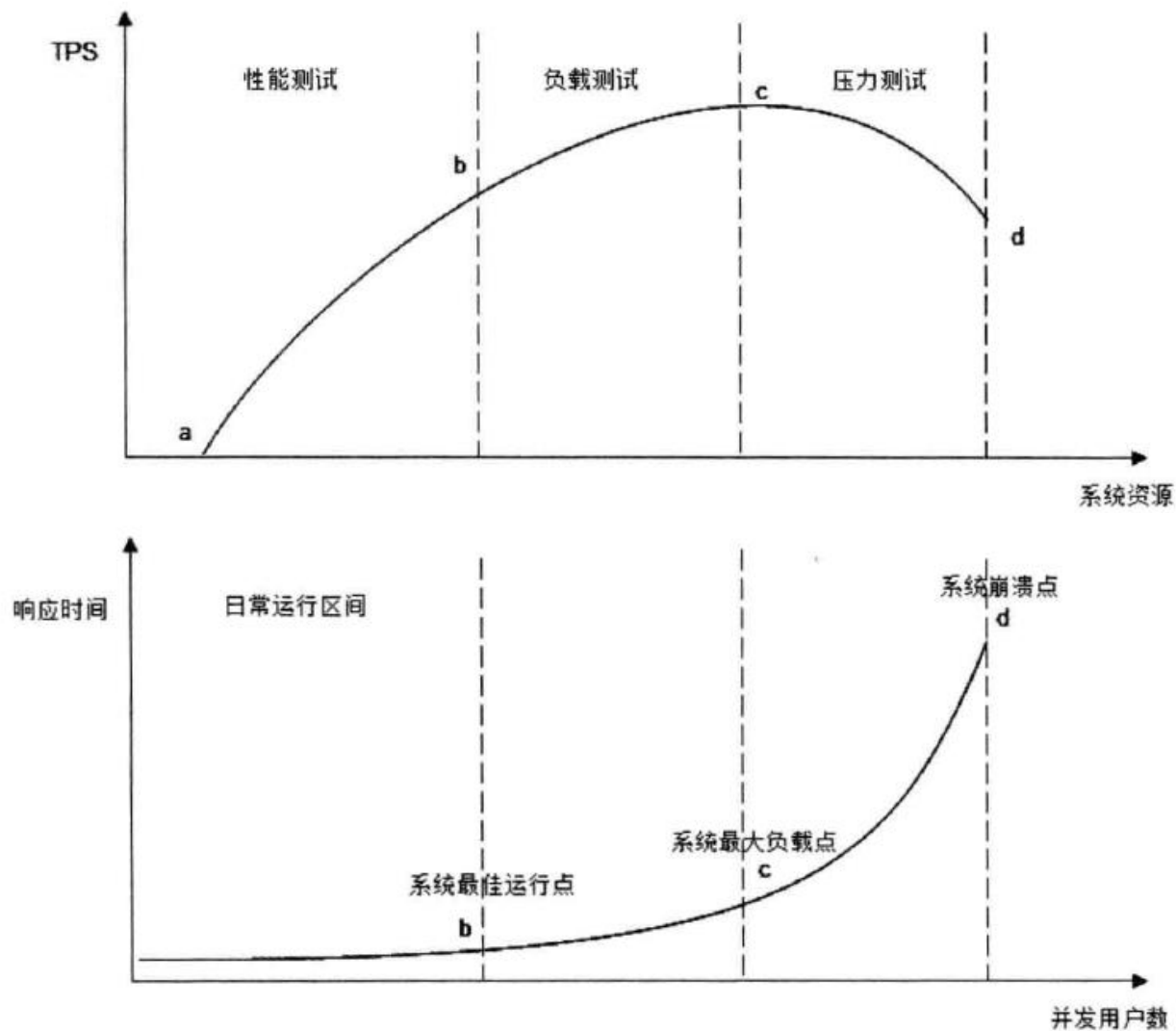
## ■ 吞吐量

- 对单位时间内完成的工作量(请求)的量度。
- 如：每分钟的数据库事务，每秒传送的文件千字节数，每分钟的 Web 服务器命中数

## ■ 关系

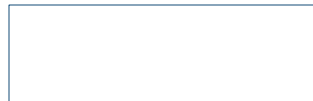
- 通常，平均响应时间越短，系统吞吐量越大；平均响应时间越长，系统吞吐量越小。但是，系统吞吐量越大，未必平均响应时间越短。

# 指标之间的关系曲线



总原则

避免过早优化



进行系统性能测试

寻找系统瓶颈，分而治之，逐步优化

1

前端优化

2

应用服务性能优化

3

存储性能优化



# 前端优化常用手段



## ■ 浏览器/App

- 减少请求数;
- 使用客户端缓冲;
- 启用压缩
- 资源文件加载顺序
- 减少Cookie传输

## ■ CDN加速

## ■ 反向代理缓存

## ■ WEB组件分离

1

缓存

2

集群

3

异步

4

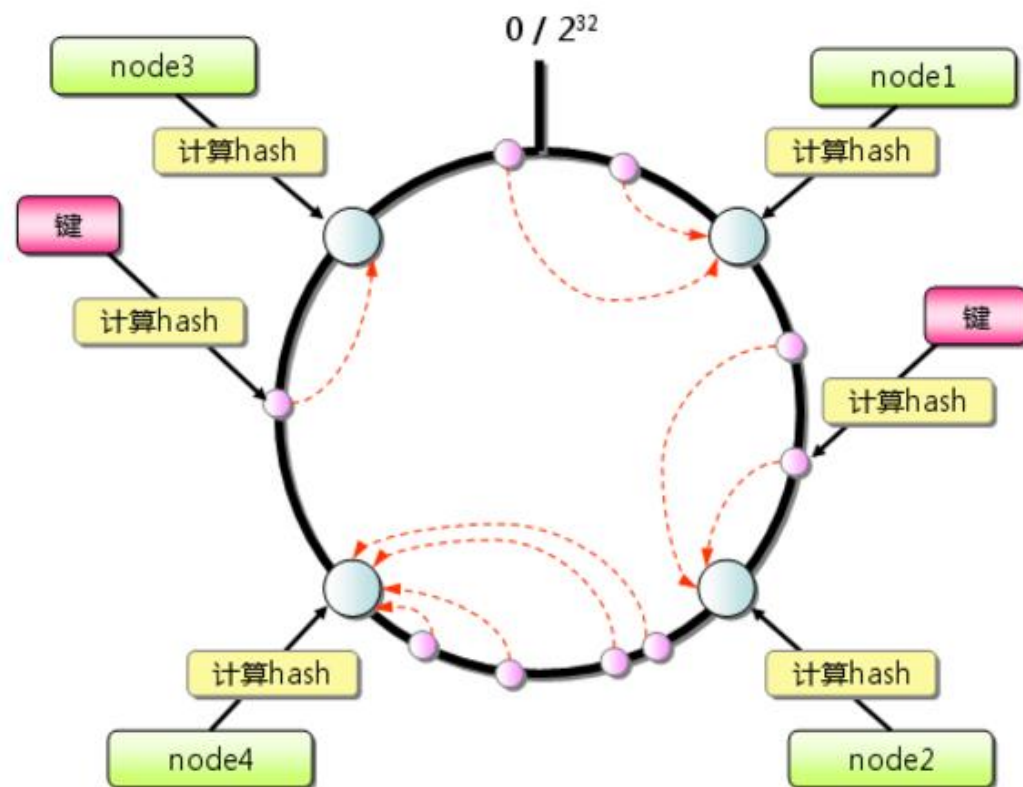
程序



■ 网站性能优化第一定律：优先考虑使用缓存优化性能

■ Mark老师的推论：缓存离用户越近越好

- 缓存的基本原理和本质
- 合理使用缓冲的准则
- 分布式缓存与一致性哈希



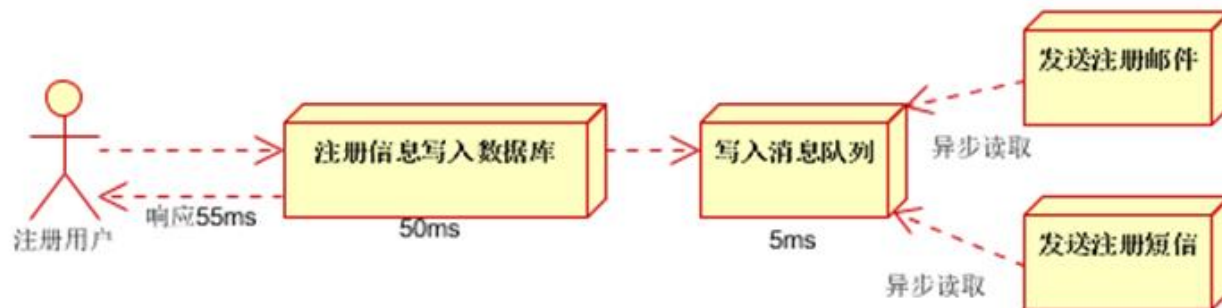
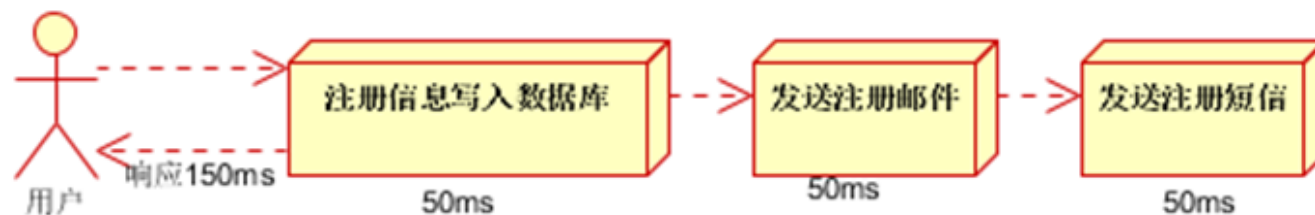
## ■ 同步和异步，阻塞和非阻塞

## ■ 常见异步的手段

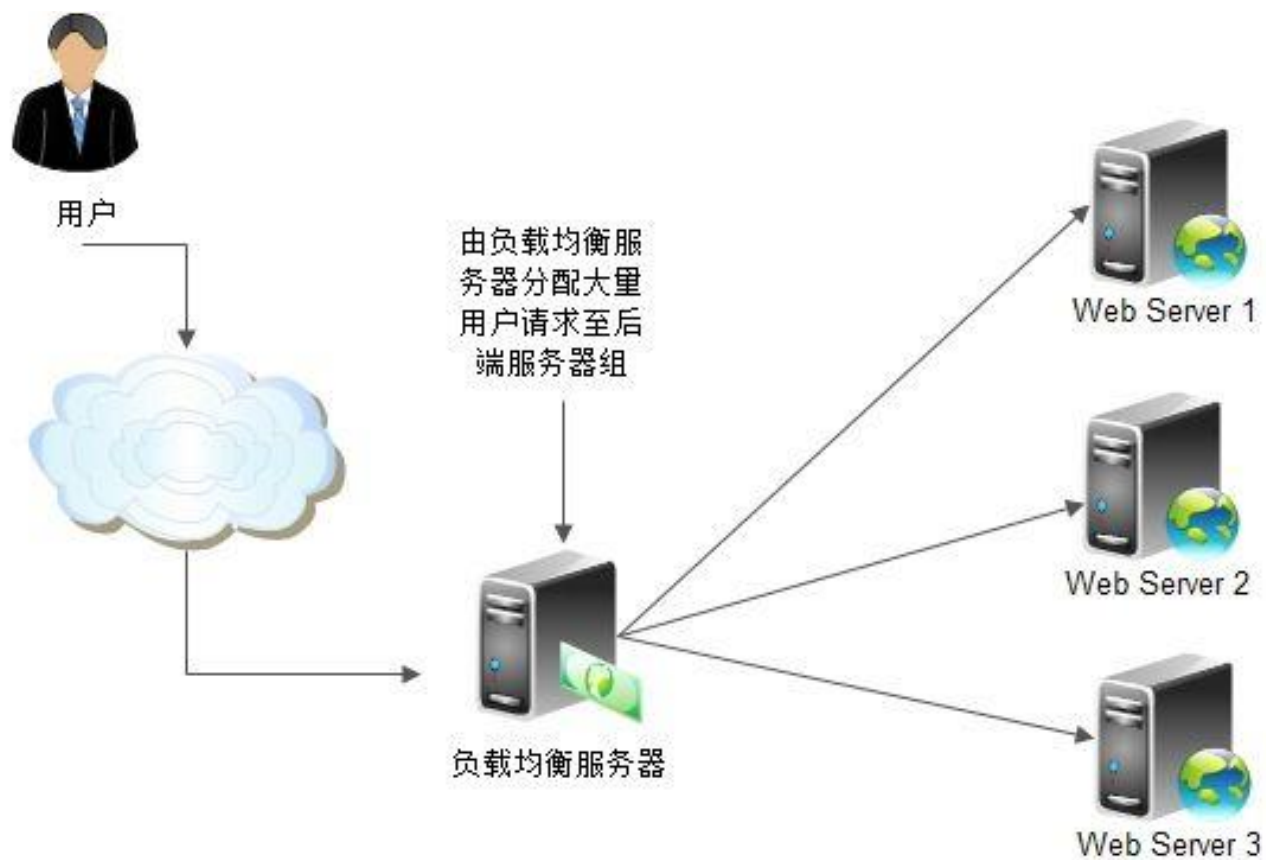
### ➤ Servlet异步

### ➤ 多线程

### ➤ 消息队列



## ■ 集群化



# 程序



- 代码级别
- 并发编程
- 资源的复用
- JVM



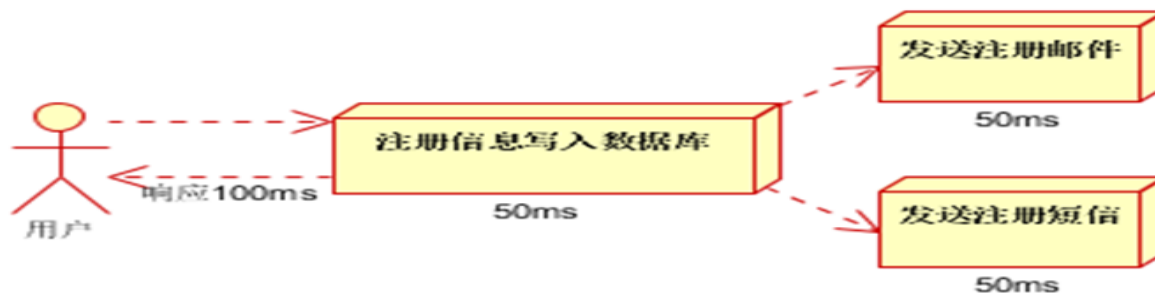


# 代码级别



- 选择合适的数据结构
- 选择更优的算法
- 编写更少的代码
- 参见《4、编写高效优雅Java程序》

- 充分利用**CPU**多核，尽量使用线程池，合理设置线程数量，尽量使用JDK提供的各种并发框架和工具
- 实现线程安全的类，避免线程安全问题
- 同步下减少锁的竞争
  - 缩小锁的范围，减少锁的粒度，锁分段，
  - 替换独占锁，读写锁，CAS代替锁，ThreadLocal等等





# 资源的复用



## ■ 减少开销很大的系统资源的创建和销毁

➤ 单例模式

➤ 池化技术

## ■ 与JIT编译器相关的优化

- 热点编译的概念
- 选择编译器类型 `-server`, `-client`, `-XX:+TieredCompilation`
- 代码缓存相关 `-XX:ReservedCodeCacheSize=N`
- 编译阈值
- 编译线程
- 方法内联
- 逃逸分析



## ■ GC调优

- 目的
- 调优的原则和步骤
- 学会阅读GC日志
- 其他与GC相关的参数
- 推荐策略

## ■ 调优实战

- 不同的内存大小
- 不同的GC回收器



# 存储性能优化



- 尽量使用SSD
- 定时清理数据或者按数据的性质分开存放
- 结果集处理