

HTTP 服务性能测试

HTTP 服务性能测试

项目开发

性能优化

项目上线

HTTP 服务性能测试

- 想要优化性能，首先要做性能检查。

HTTP 服务性能测试

- 压力测试工具
 - ab
 - webbench

HTTP 服务性能测试

- 服务器性能指标
 - QPS
 - 吞吐量

HTTP 服务性能测试

- 找到性能瓶颈
 - top
 - iostat

Node.js 性能分析工具

Node.js 性能分析工具

- 工具
 - Node.js 自帶 profile

Node.js 性能分析工具

- 工具
 - Node.js 自帶 profile
 - Chrome devtool

Node.js 性能分析工具

- 工具
 - Node.js 自帶 profile
 - Chrome devtool
 - npm - clinic

JavaScript 代码性能优化

JavaScript 代码性能优化

- 根据性能分析结果，优化下载页性能。

JavaScript 代码性能优化

- 性能优化的准则
 - 减少不必要的计算

JavaScript 代码性能优化

- 性能优化的准则
 - 减少不必要的计算
 - 空间换时间

JavaScript 代码性能优化

- 性能优化的准则
 - 减少不必要的计算
 - 空间换时间
- 思考：在用户能感知到的时间里，这个计算是不是必要的？

JavaScript 代码性能优化

- Node.js HTTP 服务性能优化准则：
 - 提前计算



内存优化管理

内存优化管理

- 垃圾回收

内存优化管理

- 新生代
- 老生代

内存优化管理

- 新生代
 - 容量小，垃圾回收更快
- 老生代
 - 容量大，垃圾回收更慢

内存优化管理

- 新生代
 - 容量小，垃圾回收更快。
- 老生代
 - 容量大，垃圾回收更慢。
- 减少内存使用，也是提高服务性能的手段。
- 如果有内存泄漏，会导致服务性能大大降低。

内存优化管理

- Node.js Buffer 的内存分配策略

内存优化管理

- 节省内存的最好方式就是
 - 使用池

Node.js C++ 插件

Node.js C++ 插件

- 编译环境安装
 - node-gyp
 - node-gyp 本身
 - 对应的 node 版本编译环境
 - python2

Node.js C++ 插件

- 将计算量转移到 C++ 进行
 - 收益：C++ 运算比 JavaScript 更快的部分
 - 成本：C++ 变量和 v8 变量的转换

Node.js C++ 插件

- 将计算量转移到 C++ 进行
 - 收益：C++ 运算比 JavaScript 更快的部分
 - 成本：C++ 变量和 v8 变量的转换
- 收益是否抵得过成本？

Node.js 子进程与线程

Node.js 子进程与线程

- 进程
 - 操作系统挂载运行程序的单元
 - 拥有一些独立的资源，如内存等

Node.js 子进程与线程

- 线程
 - 进行运算调度的单元
 - 进程内的线程共享进程内的资源

Node.js 子进程与线程

- 线程
 - 进行运算调度的单元
 - 进程内的线程共享进程内的资源
- 进程类似“公司”
- 线程类似公司的“职员”

Node.js 子进程与线程

- Node.js 的事件循环
 - 主线程运行 v8 与 JavaScript
 - 多个子线程通过事件循环被调度

Node.js 子进程与线程

- 使用子进程或线程利用更多 CPU 资源

Node.js cluster 模块

Node.js cluster 模块

- 主进程启动多个子进程，由主进程轮流分发请求，子进程代为处理

Node.js 进程守护与管理

Node.js 进程守护

- Node.js 的稳定性

Node.js 进程守护

- 防止僵尸进程
 - 心跳

Node.js 进程守护

- 死亡重启

Node.js 进程守护

- 数据监控

动静分离

动静分离

- 静态内容
 - 基本不会变动，也不会因为请求参数不同而变化
- 动态内容
 - 各种因为请求参数不同而变动，且变种的数量几乎不可枚举

动静分离

- 静态内容
 - 基本不会变动，也不会因为请求参数不同而变化
 - -> CDN 分发, HTTP 缓存 等
- 动态内容
 - 各种因为请求参数不同而变动，且变种的数量几乎不可枚举
 - -> 用大量的源站机器承载，结合反向代理进行负载均衡

动静分离

- 使用腾讯云云服务器的动静分离例子

动静分离

- Nginx 的静态服务能力和 Node.js 对比

动静分离

- Nginx 配置反向代理的方法

反向代理与缓存

反向代理与缓存

