

移动客户端搜索速度优化

(手机百度性能优化实践)

程逸凡



2015年06月

目录 CONTENTS

- 手机百度中的移动搜索
- 移动客户端常见加载模式
- 移动客户端速度指标构建
- 手机百度的优化方案
- 速度指标测试与监控

手机百度中的移动搜索

- 数以亿计的用户通过手机百度搜索
- 搜索是手机百度最核心业务
- 搜索承载了众多业务，影响用户生活的方方面面

手百1ms的优化，能令数亿用户受益

手机百度中的移动搜索

- 手机百度中经典的搜索流程



目录 CONTENTS

- 手机百度中的移动搜索
- **移动客户端常见加载模式**
- 移动客户端速度指标构建
- 手机百度的优化方案
- 速度指标测试与监控

移动客户端中常见加载模式

- 本地Native界面切换
- Native展现云端数据
- 加载Web Content

本地Native界面切换

- Native → Native
- 优化重点
 - ✓ UI效率
 - ✓ 数据库读取速度
 - ✓ 避免网络连接
 - ✓ 减少非必要运算



Native展现云端数据

- 结构化数据→Native
- 优化重点
 - ✓ UI生成与渲染
 - ✓ 数据解析
 - ✓ 服务器响应
 - ✓ 数据缓存
 - ✓ 网络通信



加载Web Content

- WebView加载
- 优化重点
✓ ??



加载Web Content的优化挑战

- App对加载过程的控制力较弱
- 对于App，WebView加载的过程是一个黑盒
 - 网络连接一般由WebView控制
 - 渲染性能依赖于WebView内核
- 页面优化依赖于前端实现

所以，客户端很难做优化??

目录 CONTENTS

- 手机百度中的移动搜索
- 移动客户端常见加载模式
- **移动客户端速度指标构建**
- 手机百度的优化方案
- 速度指标测试与监控

移动客户端速度指标构建

用户要求就一个字：

快

移动客户端速度指标构建

- 快的要求简单、直接，但是模糊
- 建立有效的评估体系
 - 实际感官的评估复杂且费时
 - 选择合适的技术指标，准确的反映感观速度
 - 尽可能将评估与测试自动化，减少评估成本
- 把握真实的用户场景，让优化效果有所体现
 - 感觉到快才是终极目标
 - 把资源投入到最容易被感受到的地方

移动客户端速度指标构建

- 首屏时间（用户可见速度）
 - 用户看到首屏界面
 - 告知用户加载完成
 - 极大影响用户的感知
- 完成时间（用户可操作速度）
 - 实际加载完成
 - 用户无法实际感知到，一般通过首屏时间推测

目录 CONTENTS

- 手机百度中的移动搜索
- 移动客户端常见加载模式
- 移动客户端速度指标构建
- **手机百度的优化方案**
- 速度指标测试与监控

常规性能优化手段

- 具体参考相关文档及Blog
- 遵守UI开发规范
- 用好SDK提供的工具
 - Trace View
 - Hierarchy Viewer

利用首屏后的时间差

- 人的反应时间，一般在150ms ~ 400ms之间
 - 刘翔的起跑反应时间：139ms
- 首屏之后存在一个时间差：300~600ms
 - 用户在看到界面后才会操作
 - 看到界面后决定如何操作，有考虑的时间
 - 移动手指操作时间

利用首屏后的时间差

让用户更早看到首屏， 感觉到“快”

整体加载顺序优化

- 优先初始首屏可见界面（UI）
- 让首屏尽可能快的显示出来
 - 只做和首屏有关的数据初始化
- 利用时间差，初始化其他必要的数据与逻辑
 - 可以在UI线程上完成必要的初始化工作
 - 用户一般不会操作，对用户打扰小

在手机百度中的应用例子



优化WebView加载速度

- 让WebView占用尽可能多的CPU、网络资源
 - 减少UI刷新和Layout布局
 - 避免后台任务
- 勿让进度条“喧宾夺主”
 - 前台的UI刷新影响后台任务的执行速度
 - 控制动画刷新频率
- 自定义进度条：提高渲染效率
 - 避免复杂运算和图像变换
 - 简化UI渲染

优化WebView加载速度

- 移动网络下，建立TCP连接的时间不可小觑
- 复用连接
- 提前建立连接
 - 避免缓存
 - 维持连接

优化WebView加载速度

- 端与页面配合，减少页面代码
 - 提供Native能力供页面调用
 - ✓ 分享
 - ✓ 登录
 - ✓
 - 页面调用端上的控件



Layout优化

- Activity、Fragment的切换都有一定的耗时
- 在速度敏感场景里，以View来组织界面
- 核心界面切换使用隐藏，而不是移除
- 使用自定义Layout
 - 减少层级
 - 简化measure & layout逻辑

图片资源优化

- 尽可能使用Shape Drawable
- 使用代码绘制简单图形
- 减小.9.png图片中纯色部分的大小
- 注意LevelListDrawable的加载时间

针对移动网络优化

- 在慢速网络下，下发简化的内容和功能
- 利用App能力获得准确的网络状态，传递给服务端
 - 动态调整下发版式
- 云和端协同预测用户连接速度
 - IP判断不精确
 - WIFI也可能很慢
 - 建立速度数据库
 - 监控线上服务质量

目录 CONTENTS

- 手机百度中的移动搜索
- 移动客户端常见加载模式
- 移动客户端速度指标构建
- 手机百度的优化方案
- 速度指标测试与监控

速度测试与监控

- 打点Log

- 数据精准
- 易于自动化
- 需要代码中埋点

- 高速摄像头

- 接近用户实际感知
- 操作复杂，成本高

速度测试

- 效果评估是性能优化工作中极其耗时的一环
- 核心指标测试自动化
- 速度指标可回溯
- 基准测试环境
 - 设备
 - 网络
 - 测试方法
 -

速度监控

- 线上：输出打点数据，抽样收集
 - 真实反映线上用户情况
 - 大数据分析
 - 成本高

- 线下：集成CI，自动化监控
 - 打江山难，守江山更难
 - 粒度细化到每一次代码提交
 - 发现问题立即报警

谢谢