

移动Hybrid App开发与架构设计

西涛

offbye@gmail.com

<http://blog.csdn.net/offbye>

2014年3月

Agenda

- ▶ 概念
- ▶ 与桌面开发的异同
- ▶ 争议
- ▶ 常见踩坑点
- ▶ 常用工具介绍
 - 调试工具
 - Cordova/PhoneGap框架
 - Ionic、AngularJS框架
- ▶ Hybrid App架构实践
- ▶ 项目选用的js框架和项目总结

概念

- ▶ Web Application
- ▶ App：应用程序
 - 完成某项或者几项任务
 - 满足某些需求
- ▶ Web：构建于Web技术之上
 - HTML/CSS/JS
 - 服务端

概念（续）

- ▶ 形态
 - Web
 - 纯WebApp
 - Hybrid App

	Device Access	Speed	Development Cost	App Store	Approval Process
Native	Full	Very Fast	Expensive	Available	Mandatory
Hybrid	Full	Native Speed as Necessary	Reasonable	Available	Low Overhead
Web	Partial	Fast	Reasonable	Not Available	None

与桌面开发的异同

▶ 同

- 技术范畴基本相同
- 大部分经验可以迁移

▶ 异

- 不同载体上的特殊问题
- 性能
- 库和框架

技术范畴

- ▶ HTML/CSS/JS
 - DOM/BOM/...
- ▶ 服务端
 - WebServer/PHP/...
- ▶ 主要针对浏览器
 - 浏览器兼容问题
 - Android碎片化，不同的android版本，分辨率。。。。

开发经验

- ▶ 架构经验
 - 展现与逻辑分离
 - 模板引擎
 - 各种优化
- ▶ 代码层面的经验
 - 模块化
 - 模板编译
 - 代码混淆

开发经验（续）

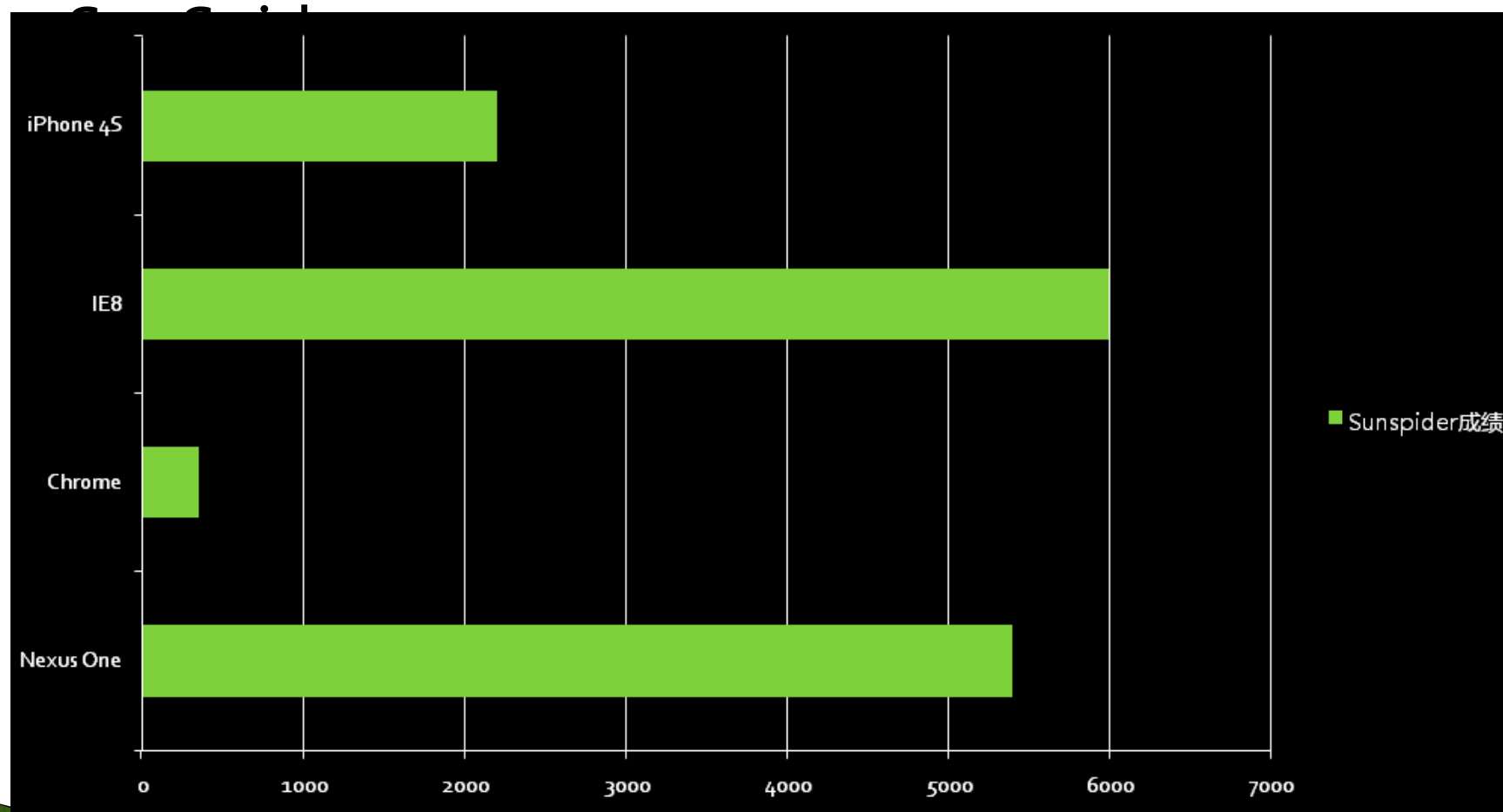
▶ 与服务端交互数据

- 使用JSON格式
- Ajax/JSONP
- JSONP即JSON with Padding。由于同源策略的限制，XmlHttpRequest只允许请求当前源（域名、协议、端口）的资源。如果要进行跨域请求，我们可以通过使用html的script标记来进行跨域请求，并在响应中返回要执行的script代码，其中可以直接使用JSON传递javascript对象。这种跨域的通讯方式称为JSONP。
具体参考
<http://www.cnblogs.com/chopper/archive/2012/03/24/2403945.html>

载体不同

- ▶ 移动设备屏幕
 - 屏幕偏小，ViewPort概念受到空前关注
 - 布局pixel与物理pixel不一定相等
- ▶ 操作方式
 - 不能依赖鼠标悬浮
 - 触摸控制精度低于鼠标控制
 - 触屏设备（通常）没有物理键盘

性能



性能（续）

- ▶ 顶级手机性能不输桌面平均水平
- ▶ 高端和低端性能差异巨大
 - 手机更新换代速度也非常快
- ▶ 手机上不适合用setInterval做动画

性能

▶ 幸哉

- iOS和Android平台都以webkit为主
- CSS 3 Transition/Animation可用
- 半透明、阴影、圆角、渐变都可以用

▶ 悲哉

- CSS 3动画在低端手机上的丢帧现象
- CSS 3效果对性能要求不低
- 同期设备iOS浏览器性能明显高于Android
 - 但国内（搜索）市场上Android流量高于iOS

性能优化Tips

1. 用HTML5的离线存储和本地存储进行缓存，或者将页面直接打包到客户端中，减少在网络下载中的耗时。
2. 减少DOM数量，尽可能少的使用`position:relative`，减少对DOM的操作
3. 用CSS动画代替JS动画，在Android平台上可以平稳退化，放弃动画效果（包括CSS3动画）
4. 避免GIF图片的使用（消耗内存）
5. 如果只是在移动端使用的话，请使用iScroll-lite来代替iScroll（iScroll里面增加了很多额外的功能，比如在PC上模拟滚动），在允许的情况下，可以关闭滚动条（滚动条也是创建的DOM元素）

库和框架

- ▶ 我们不再有伟大的jQuery
 - 虽然有不再那么伟大的jQuery Mobile
- ▶ 但是有伟大的webkit
 - jQuery的浏览器兼容大多数都不再需要
- ▶ 需要精简的库
 - 同时最好有jQuery like API
 - [Zepto](#)

库和框架

▶ 其他库

- dojo
- Sencha
- 还有一大堆

▶ 问题所在

- 资料——学习成本
- 贪大求全（是否符合我们的需要？）

库和框架

- ▶ UI框架
 - jQuery Mobile
 - dojo Toolkit
 - Sencha Touch
- ▶ MVC框架
 - Backbone.js
- ▶ 一切跟着需求走

争议

- ▶ 有页面间跳转还算不算WebApp
 - 正方：有页面间跳转还是可以算WebApp
 - 基于Web技术
 - 能满足特定需求，是一个应用程序
 - 反方：有页面间跳转就不算WebApp
 - 破坏“闭环”
 - 没有做到类似原生App的体验

争议（续）

- ▶ WebApp是否应效仿原生App
 - 正方：应该效仿
 - 原生App交互效果华丽，用户体验水平高
 - 形成App内“闭环”，锁住用户
 - 反方：不应效仿
 - WebApp先是Web，不应失去Web理念
 - 使用传统Web理念仍有改善用户体验的空间
 - “闭环”不是任何时候都是好事

常见踩坑点

► Viewport

- 小屏幕显示大网页
- 缩放
- 480px屏幕显示320px网页
- 布局中的pixel \neq 屏幕上的pixel
- 高分辨率图片的使用
 - 为什么iPhone4上看是模糊的？不是说iPhone4屏幕效果更好吗？

常见踩坑点

- ▶ touch 不等于 mouse
 - 移动设备上通常用touch事件而非mouse事件
 - WP没有touch，还是用mouse
 - touchstart -> touchend的过程通常比click事件更灵敏
 - 可以用这个特点来自己制作快速点击的事件
 - iOS较新版本支持很多点，而Android直到2.3还是只支持一个点
 - 双指缩放图片

常见踩坑点

- ▶ 抓狂的onscroll
 - 大多数手机和浏览器都有惯性滑动
 - 惯性滚动停止以后才触发onscroll事件
 - 有的手机和有的UCWeb下面是以抽筋一样的频率触发
- ▶ scroll周边问题
 - 有的浏览器下取窗口滚动位置不准确

常见踩坑点

- ▶ position:fixed
 - 不支持
 - iOS 5支持
 - 诡异的1px来回抖动
 - Android 2.3支持
 - 版本和第三方定制差异，可用性较低
- ▶ window.innerHeight
 - iOS下OK
 - Android下弹出虚拟键盘时常会造成悲剧
 - 使用一个height:100%的绝对定位div来取高度

常见踩坑点

- ▶ translate3d
 - 硬件加速？忽悠？
 - Android有时候会和rotate冲突
 - 这种时候即使用TransformMatrix都不给力
 - 性能！性能！性能！
 - 卡：少用特效
 - 虚：参考“卡”，碰运气
 - 闪：`-webkit-backface-visibility:hidden;`
`-webkit-perspective:1000;`

工具

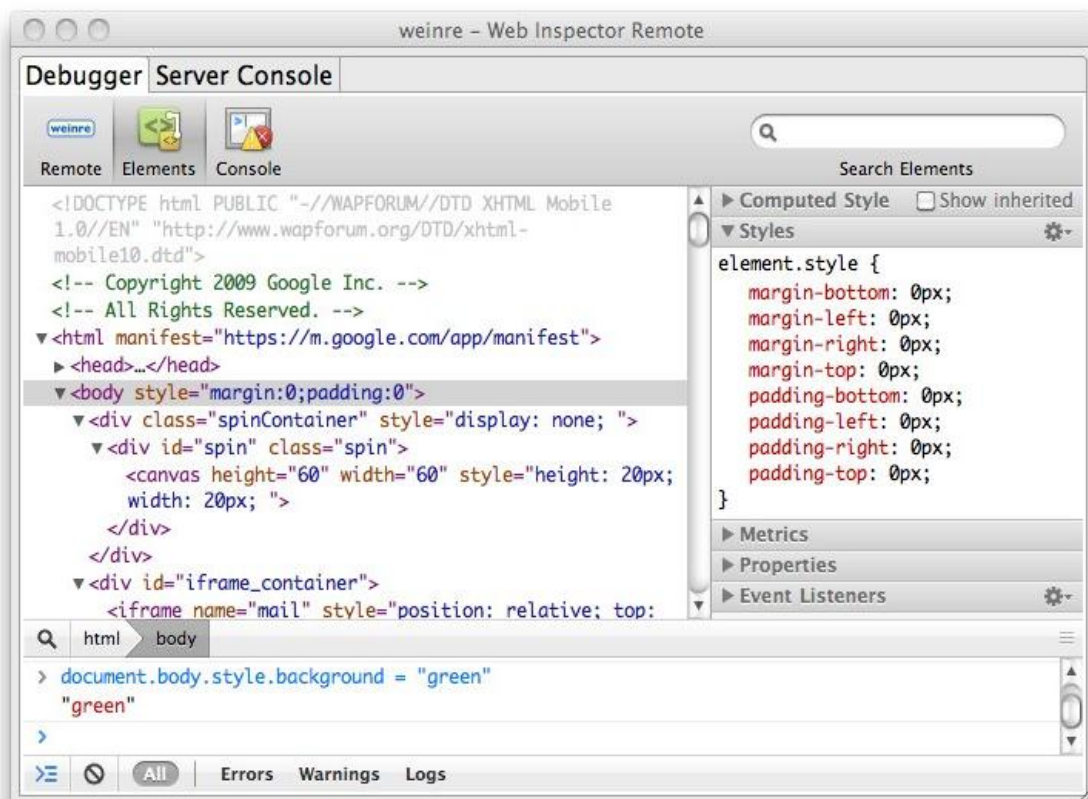
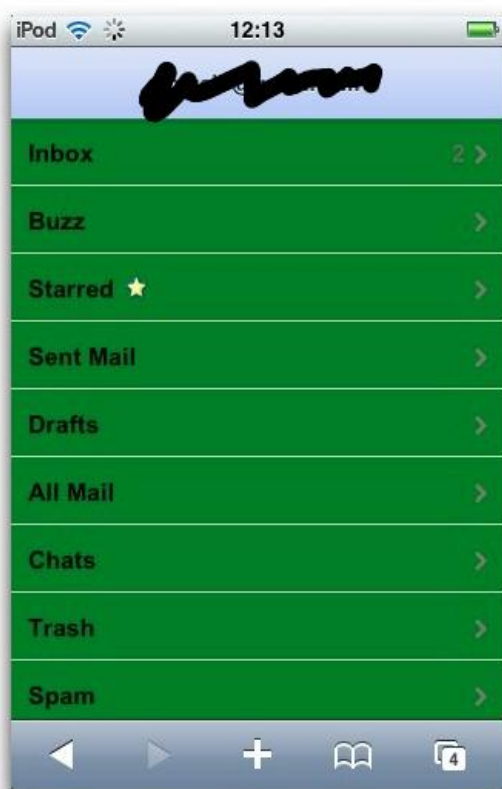
- ▶ Chrome
 - webkit引擎，强大的开发者工具
 - 点击变touch
- ▶ Weinre
 - 远程修改DOM和CSS
- ▶ 真机、真机.....
- ▶ Wiki
 - 建设经验库，避免走回头路

常用工具

Framework & Tools	Current	Maturity
Debug	Chrome for Android Weinre	Available for Android Proxy based (All Device)
IDE	Sencha Designer	Beta
JavaScript Library	Jquery Jquery Mobile JQ.Mobi	Industry Standard Performance issue Growing(5k)
JavaScript Framework	Sencha Touch Yahoo! Cocktails - YUI Sproutcore Backbone JS LungoJS Joshfire Jo	Mature, Heavy MobileFramework Unknown Code Generation Lightweight,Key-Value binding Event Semantic structure in project Cross platform, Adapt Event Design for Web App
UI Framework	Sencha Touch Chart Jquery UI	Mature UI Widgets, Theme
Data/Logic Framework	YQL	Unknown
Cross Platform Framework	PhoneGap Titanium	Dominate N/A

Web移动应用调试工具——Weinre

- ▶ Weinre代表Web Inspector Remote，是一种远程调试工具。举个例子，在电脑上可以即时 的更改手机上对应网页的页面元素、样式表，或是查看Javascript变量，同时还可以看到手机上页面的错误和警告信息.



Cordova/PhoneGap框架

- ▶ 开源手机开发框架，使用HTML5, CSS3, and JavaScript开发跨平台应用，针对每个平台提供应用包装，提供一致性的多平台硬件设备能力API

	amazon- fireos	android	blackberry10	ios	Ubuntu	wp7 (Windows Phone 7)	wp8 (Windows Phone 8)	win8 (Windows 8)	tizen
cordova CLI	✓ Mac, Windows, Linux	✓ Mac, Windows, Linux	✓ Mac, Windows	✓ Mac	✓ Ubuntu	✓ Windows	✓ Windows	✓	✗
Embedded WebView	✓ (see details)	✓ (see details)	✗	✓ (see details)	✓	✗	✗	✗	✗
Plug-in Interface	✓ (see details)	✓ (see details)	✓ (see details)	✓ (see details)	✓	✓ (see details)	✓ (see details)	✓	✗

Cordova/PhoneGap框架

	amazon- fireos	android	blackberry10	ios	Ubuntu Platform	wp7 (Windows Phone 7) APIs	wp8 (Windows Phone 8)	win8 (Windows 8)	tizen
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓	✓
Capture	✓	✓	✓	✓	✓	✓	✓	✗	✗
Compass	✓	✓	✓	✓ (3GS+)	✓	✓	✓	✓	✓
Connection	✓	✓	✓	✓	✓	✓	✓	✓	✓
Contacts	✓	✓	✓	✓	✓	✓	✓	✗	✗
Device	✓	✓	✓	✓	✓	✓	✓	✓	✓
Events	✓	✓	✓	✓	✓	✓	✓	✓	✓
File	✓	✓	✓	✓	✓	✓	✓	✓	✗
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓	✓
Globalization	✓	✓	✗	✓	✓	✓	✓	✗	✗
InAppBrowser	✓	✓	✓	✓	✓	✓	✓	(partial)uses iframe	✗
Media	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification	✓	✓	✓	✓	✓	✓	✓	✓	✓
Splashscreen	✓	✓	✓	✓	✓	✓	✓	✓	✗
Storage	✓	✓	✓	✓	✓	✓ localStorage & indexedDB	✓ localStorage & indexedDB	✓ localStorage & indexedDB	✓

Cordova/PhoneGap框架

▶ Cordova CLI命令行工具

- 提供创建项目，添加目标平台，编译运行项目等功能，自动化

▶ 设备API插件化

- From version 3.0 onward, Cordova implements all device APIs as plugins, and leaves them disabled by default
- Cordova plugin add 和 pluginman命令

▶ 配置文件config.xml

- `<content src="index.html" />`
`<access origin="*" />`
`<preference name="Fullscreen" value="true" />`
`<preference name="WebViewBounce" value="true" />`

▶ 跨域问题

- Cordova adheres to the [W3C Widget Access](#) specification, which relies on the `<access>` element within the app's config.xml
- 各平台行为有差异
- 默认`<access origin="*" />`

Ionic框架

- ▶ Ionic是类似Bootstrap概念的WebApp框架，提供了Hybrid App的基础Css，UI控件，工程架构和最佳实践
- ▶ Ionic框架主要关注移动应用的UI展示和UI交互，不是phoneGap框架的替代品
- ▶ Ionic框架需要依赖AngularJS
- ▶ powerful UI interactions, gestures, animations

Ionic框架提供了什么

- ▶ 基于Ionic命令的项目模板
- ▶ 整合了phonegap和angularJS，并压缩
- ▶ 提供常用的UI控件
- ▶ View Controller

Hybrid App架构实践



Hybrid App架构设计原则

- ▶ 使用phonegap提供跨平台的设备资源访问能力
- ▶ 服务器端基于REST + jsonp提供数据
- ▶ 大部分WebApp文件（html，css，js）打包在App内，安装到设备上
- ▶ Html和js完全分离，html页面模板化，通过js模板引擎渲染服务器端json数据到html模板页面
- ▶ Html页面优先采用响应式设计适配不同的屏幕分辨率，特殊页面分别实现多个html页面
- ▶ 使用单html页面，html页面模板对应不同的功能模块，降低单个文件的复杂度，通过基于#锚点的Url路由跳转页面

服务器接口设计

- ▶ 基于HTTP+json的REST接口，扩展支持jsonp（只需要进行简单改造）
- ▶ 隔离APP端和接口，各自独立开发，前期客户端可以使用 mock接口测试数据开发
- ▶ 接口可以兼容ios和android，并兼容native版本的客户端实现
 - 当某些WebApp页面出现难以解决的性能问题时，可以切换到原生界面实现，后面也可以在接口不变的前提下开发 native app版本

js框架选择原则

- ▶ 需要使用大量UI控件，表单密集型的应用
 - jQuery Mobile
 - Sencha Touch
 - Ionic
- ▶ UI较单一，注重性能，注重页面特效的应用（淘宝的）
 - 使用多个小型js库
 - zepto.js作为底层库，兼容jquery语法
 - 使用sea.js进行模块的管理和发布
 - 使用backbone.js为基础的MVC架构，用来剥离应用的数据部分
 - 使用underscore.js做为前端模板引擎（或使用juicer）

项目选用的js框架

- ▶ 基于Ionic UI组件的界面实现，DOM结构简单，不容易出现性能问题，默认IOS7风格
- ▶ AngularJS， MVC， 模板引擎， Url Route， 模块化等
- ▶ PhoneGap， 提供设备能力API， App封装
- ▶ Ionic， 类似Bootstrap概念的WebApp框架， 提供了Hybrid App的基础Css， 工程架构和最佳实践
- ▶ Ionic/cordova的命令行工具， 生成android和ios项目工程， 并且可以直接安装到模拟器或设备

项目面临的风险

- ▶ AngularJS框架很新很先进，有很多新的概念，和传统js框架差别较大，学习曲线比较陡
- ▶ 项目中没有人有AngularJS的经验，需要边学边做
- ▶ 项目需要实现的功能点很多，时间紧，项目进入开发阶段后需要增加人力
- ▶ AppStore审核可能会出现问題，延误上线时间
 - 可以先提交一个包含主要业务流程的功能较少的版本审核
- ▶ WebApp在iPad上必然会出现一些性能和用户体验问题，解决会比较困难

项目阶段回顾

- ▶ 经过2周的一个迭代
- ▶ 项目客户端结构基本稳定下来，ionic表现让人满意
- ▶ 项目人员基本适应了AngularJS，但也开始踩到坑
- ▶ 项目人员都很有经验
- ▶ 总体来看比用jquery的框架好太多了！

Q&A