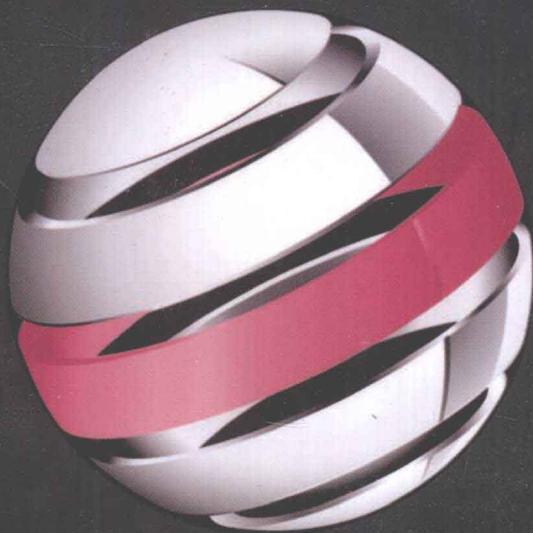


移动App开发的“一次构建”方法



# Android Web应用 高级编程

Pro Android Web Apps: Develop for Android Using  
HTML5, CSS3 & JavaScript

(美) Damon Oehlman 著  
Sébastien Blanc 译  
张 龙



发技术

# Android Web 应用

## 高级编程

(美) Damon Oehlman      著  
Sébastien Blanc      著  
张 龙      译

清华大学出版社

北 京

Damon Oehlman, Sébastien Blanc

Pro Android Web Apps: Develop for Android Using HTML5, CSS3 & JavaScript

EISBN: 978-1-4302-3276-6

Original English language edition published by Apress, 2855 Telegraph Avenue, #600, Berkeley, CA 94705 USA. Copyright © 2011 by Damon Oehlman, Sébastien Blanc. Simplified Chinese-language edition copyright © 2012 by Tsinghua University Press. All rights reserved.

本书中文简体字版由 Apress 出版公司授权清华大学出版社出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字：01-2011-2257

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

### 图书在版编目(CIP)数据

Android Web 应用高级编程/(美)奥尔蒙(Oehlman, D.), (美)布兰克(Blanc, S.) 著; 张龙 译.

—北京: 清华大学出版社, 2012.3

(移动与嵌入式开发技术)

书名原文: Pro Android Web Apps: Develop for Android Using HTML5, CSS3 & JavaScript

ISBN 978-7-302-28050-7

I. ①A… II. ①奥… ②布…③张… III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字(2012)第 024160 号

责任编辑: 王 军 于 平

装帧设计: 牛艳敏

责任校对: 成凤进

责任印制: 王静怡

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈: 010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 刷 者: 清华大学印刷厂

装 订 者: 三河市溧源装订厂

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 19.75 字 数: 481 千字

版 次: 2012 年 3 月第 1 版 印 次: 2012 年 3 月第 1 次印刷

印 数: 1~3500

定 价: 48.00 元

---

产品编号: 040481-01

# 译者序

随着科技的日益进步，现在我们已经逐渐步入了移动世界。移动开发浪潮席卷全球，智能手机与平板电脑的销量日益增加，这不仅为最终用户的的生活和工作带来了极大的便利，同时也为众多的开发者创造了展示自己的舞台。

自 Google 推出 Android 系统以来，Android 家族一直呈现出蓬勃发展的态势。世界上已经有多家厂商在开发、研制基于 Android 系统的手机与平板电脑，如三星、摩托罗拉、联想、HTC、华为等，Android 生态圈也表现出了欣欣向荣的发展势头。无数的开发者涌入 Android 开发的世界中，本书也在这个时机下应运而生。

本书与其他讲述 Android 开发的书籍最大的区别在于：目前绝大多数 Android 开发书籍依然在介绍如何构建原生应用，而本书的重点则在于如何通过最先进的 HTML5、CSS3 与 JavaScript 等标准和技术构建基于 Android 的 Web 应用。Web 应用是应用未来的发展趋势，而本书正是契合了这种趋势，讲述了 Android Web 应用开发所涉及的方方面面的知识与技术。

本书通过深入浅出的讲解帮助读者迅速进入到 Android Web 应用开发的世界中。即便读者不具备 Web 应用开发相关的技能也能通过本书快速掌握移动 Web 开发所涉及的各方面技能与技巧。本书通过大量示例帮助读者深刻理解原生应用与 Web 应用之间的差异，如何构建基于 Android 的 Web 应用，同时还详细讲解了如何通过桥接框架将 Android Web 应用转换为原生应用以在 Google 市场上发布；此外，本书还对目前移动世界中的流行 UI 框架 Jo、jQTouch、jQuery Mobile 与 Sencha Touch 进行了分析与比较，这样读者就可以根据书中的介绍为自己的应用选择合适的框架。

除了 Android Web 应用开发本身所涉及的内容外，本书还通过大量篇幅详细介绍了如何实现基于位置的服务与移动地图、如何集成 Social API 并探讨了移动计算的未来，这些都是移动 Web 开发者的兴趣所在。相信通过本书的学习，读者会对 Android Web 应用开发拥有更加深刻和清晰的认识。

翻译技术书籍是一项艰苦的劳动，在这里我要将我最真挚的谢意送给我的妻子张明辉。本书翻译期间，我们在经历了 8 年的相识后终于步入了婚姻的殿堂，本书的出版也是赐予我们最好的结婚礼物。将最真诚的感谢送给我的父母，没有你们的哺育和教诲就不会有今天的我。

感谢清华大学出版社的李阳老师，认识你是我的荣幸，你的专业与认真都给我留下了深刻的印象。

本人从事 Android 相关开发工作已经有两年多，在这期间积累了不少经验，深谙移动

开发的精髓。曾翻译过《Spring 高级程序设计》、《iPhone 游戏开发》、《iPhone SDK 编程入门经典：使用 Objective-C》、《设计原本：计算机科学巨匠 Frederick P. Brooks 的思考》等著作。本人还是国内高端技术站点 InfoQ 中文站编辑、满江红开放技术研究组织成员，参与了 Spring 2.5 官方文档的翻译工作，同时拥有 5 年以上的 Java EE 培训讲师经历。

对于译者来说，能将英文转换为中文并给读者带来切实的帮助是我最大的荣幸。因此，您在阅读过程中所发现的问题都将是我的责任，也真诚希望各位读者不吝赐教。由于译者水平有限，失误和遗漏之处在所难免，恳请读者批评指正。敬请广大读者提供反馈意见，读者可以将意见反馈到 [wkservice@vip.163.com](mailto:wkservice@vip.163.com)，我会仔细阅读读者发来的每一封邮件，以求进一步提高今后译著的质量。

译者

# 作者简介

**Damon Oehlman** 是一位经验丰富的软件开发人员与技术经理，目前居住在澳大利亚布里斯班。他曾在多种平台上进行过开发，从 Windows 到 Web 开发，现在则转向了移动开发，Damon 对移动 Web 应用开发“一次编写，到处运行”的理念有着独到的见解，这也促使他将精力转向了这里。

由于看到了移动开发日益增长的势头，Damon 放弃了稳定的工作，与人联合创立了移动开发公司 Sidelab([www.sidelab.com](http://www.sidelab.com))。Sidelab 为移动 Web 应用开发提供专业的开发服务，特别是在地图、基于位置的服务以及数据可视化等领域。Damon 还维护着一个技术博客 Distractable([www.distractable.net](http://www.distractable.net))并创建了 HTML5 移动地图 JavaScript 库 Tile5([www.tile5.org](http://www.tile5.org))。

在编码或写作的闲暇时间，Damon 喜欢与妻子和孩子们共度时光，他们让 Damon 知道除了编写软件外，生活也是五彩缤纷的。

**Sébastien Blanc** 是一位资深 JEE 软件工程师。他就职于荷兰公司 E-id([www.e-id.nl](http://www.e-id.nl))。此外，Sébastien 还将很多精力放在了移动 Web 应用开发上。他是各种会议演讲的常客，深信基于 Web 的移动开发拥有美好的明天。与 Damon 一样，在编码的闲暇之余，他喜欢与家人共度。

# 技术审校者简介

**Kunal Mittal** 是 Sony Pictures Entertainment 的技术执行理事，负责 SOA、身份管理和内容管理程序。他为不同的业务线提供了集中的工程服务，并且为 Sony Pictures Enterprise IT 环境引入了新的平台与技术。

**Kunal** 是一位企业家，帮助初创公司制定技术战略、产品路线图和开发计划。由于他与全球几家开发伙伴保持着紧密的关系，因此可以帮助初创公司甚至是大型公司构建适当的开发伙伴关系。通常他的工作职责是顾问或咨询 CTO，但也在项目管理与技术架构领域表现活跃。他曾创作和编辑过几本关于 J2EE、云计算和移动技术方面的书籍和文章。

# 致 谢

首先感谢我的妻子与孩子。2010 是忙碌的一年，充满了各种机遇，你们不仅支持我的工作，还不忘提醒我，花些时间陪陪家人也是非常重要的事情。我爱你们。其次，我想感谢 Apress 团队能给我写作本书的机会以及在漫长的写作过程中所给予我的支持和建议。我在这个过程中获益颇多，感谢你们自始至终的耐心与专业。

Damon

将本书献给 Mathilde、我的孩子、Damon、Douglas、Mary、Kunal 与 Steve。

Sébastien

# 前 言

我们现在正步入新的世界，移动设备逐渐成为人们与 Internet 互联的主要方式，开发移动设备应用正成为人们新的追求。供应商之间的竞争也日趋白热化，这导致市场中充满了各种各样的设备。

我们看到各个供应商都在为自己的设备提供开发工具与市场，并希望围绕着自己的产品创建软件生态圈。在很大程度上，一些供应商的战略也是这样的。开发者使用这些工具并为特定的设备创建“原生”应用，然后需要重新构建应用的大部分内容以面向各种不同的设备。

对于构建移动应用的公司来说，这是可接受的方式。然而，从长远来看，这种方式是行不通的。考虑到拥有 Web 产品的每个公司在未来几年间都需要为多种设备提供桌面版的 Web 应用与适合的移动客户端。再考虑软件开发者的数量——就像你我，还有其他开发者。我们拥有必要的资源满足这种需求吗？我没有。一定会有更好的方式。确实是有的。

构建移动 Web 应用是更好的方式。这是一种移动应用开发的方式，你只需要重写很少的代码就可以满足市场上存在的众多设备的需求。本书关注于为 Android 编写移动 Web 应用，但实际上，很多概念也可以轻松移植到其他移动设备上(这是重点)。

## 何谓移动 Web 应用

移动 Web 应用就是使用核心客户端 Web 技术 HTML、CSS 与 JavaScript 构建，并且专门针对移动设备的应用。吸引人们关注移动 Web 应用是 HTML5 与 CSS3 的趋势——这两个技术的“最新版”。本书会详细介绍 HTML5、CSS3 和 JavaScript。

JavaScript 是令很多开发者又爱又恨的语言。有些人甚至都不把它看做是一门编程语言。然而，JavaScript 很可能会成为未来 5 年内最需要的技能之一。

## 本书使用了哪些技术

本书将会使用大量 JavaScript 代码。显然还会使用很多 HTML 与 CSS，但 JavaScript 确实是适合于移动 Web 应用开发的语言。

如果不熟悉 JavaScript，那么我们并不会将你置于纷繁复杂的 JavaScript 代码中，我们会推荐一些学习资料，因为这并非一本 JavaScript 基础书。我们还大量使用了优秀的 jQuery JavaScript 库以简化开发。如果不熟悉 jQuery，那么我们建议你看看 jQuery 教程。如果熟悉 Prototype、MooTools 或是其他的 jQuery “竞争者”，那么你可以很轻松地修改书中的示例代码。

对于移动 Web 应用来说(以及其他的富 JavaScript Web 应用)，掌握如何组织应用以保证可读性与可维护性是很重要的。这正是本书中使用了很多小应用项目而非代码片段来展



示特定功能的原因所在。这样你就可以熟悉移动 Web 应用的方方面面，还能理解如何高效地整合真实世界中的移动 Web 应用。

如果熟悉 Web 应用开发，那么通过本书的学习，你就可以轻松转换到移动 Web 应用开发上来。但如果你熟悉移动应用开发，想要学习 Web 应用开发方式，那么学习额外的资料将是非常重要的。

## 本书内容

本书围绕着两个应用示例展开，你会从中学习到移动 Web 应用的方方面面。第 2 章~第 6 章介绍了首个迷你应用，它是一个简单的“任务清单列表”应用，第 8 章~第 12 章则从头开始介绍如何构建一个简单的位感知游戏。

在这两顿大餐之间还有 3 章作为“点心”。第 1 章主要介绍了编写 Android Web 应用的基本概念。第 7 章简要介绍了交互性与 HTML5 canvas。第 13 章介绍了移动应用世界的未来。

# 目 录

<b>第 1 章 入门</b> ..... 1	<b>第 3 章 HTML5 Storage API</b> ..... 39
1.1 了解 Android 平台功能 ..... 1	3.1 Web Storage API ..... 39
1.1.1 设备连接 ..... 2	3.1.1 使用 JSON 将对象保存到 Web Storage 中 ..... 41
1.1.2 触摸 ..... 2	3.1.2 Local Storage 与 Session Storage ..... 45
1.1.3 地理位置 ..... 3	3.2 Web SQL Database ..... 45
1.1.4 硬件传感器 ..... 3	3.2.1 使用客户端数据库保存任务清 单条目 ..... 46
1.1.5 本地数据库与存储 ..... 4	3.2.2 数据库的版本化与升级 ..... 51
1.1.6 相机支持 ..... 4	3.3 小结 ..... 53
1.1.7 消息与推送通知 ..... 4	<b>第 4 章 构建多页面应用</b> ..... 55
1.1.8 WebKit Web 浏览器 ..... 5	4.1 单个 HTML 文件、多个应用 页面 ..... 55
1.1.9 进程管理 ..... 5	4.1.1 创建视图管理器 ..... 58
1.1.10 Android OS 特性小结 ..... 6	4.1.2 实现视图动作 ..... 60
1.2 准备开发环境 ..... 6	4.2 构建应用的主界面 ..... 62
1.2.1 文本编辑器与工作目录 ..... 7	4.2.1 修改 ViewManager 功能 ..... 65
1.2.2 Web 服务器 ..... 7	4.2.2 主界面的存储需求 ..... 66
1.2.3 模拟器 ..... 9	4.2.3 装配主界面 ..... 70
1.3 Hello World ..... 13	4.3 构建所有任务界面 ..... 72
1.4 小结 ..... 15	4.4 实现视图栈 ..... 78
<b>第 2 章 构建移动 HTML 输入表单</b> ..... 17	4.5 小结 ..... 81
2.1 面向移动 Web 的 HTML ..... 17	<b>第 5 章 与云进行同步</b> ..... 83
2.1.1 面向移动的网页 ..... 17	5.1 探索在线存储的选择 ..... 83
2.1.2 添加表单元元素 ..... 21	5.1.1 在线同步存储的需求 ..... 83
2.2 添加样式 ..... 22	5.1.2 避免 3 层架构 ..... 84
2.2.1 使用 CSS3 的表单样式 ..... 24	5.1.3 用户认证 ..... 84
2.2.2 改进页面标题的外观 ..... 26	5.1.4 JavaScript 同步库 ..... 84
2.2.3 针对不同的屏幕尺寸编写代码 ..... 27	
2.2.4 处理设备方向的变化 ..... 28	
2.3 添加表单验证 ..... 32	
2.4 小结 ..... 38	

5.1.5	可能的同步解决方案	85
5.2	Google App Engine 入门	85
5.2.1	在本地部署 jsonengine	86
5.2.2	选择合适的同步模式	87
5.2.3	向 jsonengine 发送离线数据	87
5.3	针对在线同步更新用户界面	89
5.4	开发桌面界面	92
5.4.1	查询 jsonengine 实例	92
5.4.2	将应用部署到云中	95
5.5	小结	95
<b>第 6 章</b>	<b>与原生应用竞争</b>	<b>97</b>
6.1	添加轻量级动画与类似于原生应用的布局	97
6.1.1	添加简单的旋转加载器	97
6.1.2	添加可滚动的内容	100
6.1.3	修改动作栏	102
6.2	为应用添加位置感知特性	103
6.3	离线运行应用	108
6.3.1	离线缓存清单文件	108
6.3.2	探索隐蔽的离线缓存特性	109
6.3.3	检测连接状态	111
6.4	小结	112
<b>第 7 章</b>	<b>探索交互性</b>	<b>113</b>
7.1	HTML5 Canvas 简介	113
7.2	为 Canvas 绘制增加交互性	115
7.2.1	交互: 使用鼠标	115
7.2.2	交互: 使用触摸	117
7.3	实现 Canvas 动画	119
7.3.1	创建动画循环	120
7.3.2	绘制一帧动画	121
7.3.3	绘制图像: 考虑设备 DPI	124
7.4	高级动画技术	128
7.4.1	在动画中实现逼真的移动	128
7.4.2	Canvas 变换与动画	132
7.4.3	小汽车动画的变换	134
7.5	小结	137

<b>第 8 章</b>	<b>基于位置的服务与移动地图</b>	<b>139</b>
8.1	基于位置的服务	139
8.2	地理社交网络	140
8.3	移动地图	141
8.3.1	使用 Google Maps 显示地图	141
8.3.2	Tile5: 另一个 HTML5 Mapping API	143
8.3.3	向 Google Map 添加标记	146
8.3.4	显示标记的详细信息	147
8.4	针对移动优化的地图 UI	149
8.4.1	地图 UI 模型	149
8.4.2	编写移动地图 UI 样板	150
8.4.3	在样板代码中实现 UI 导航	154
8.4.4	使用导航栏选择标记	159
8.5	小结	166
<b>第 9 章</b>	<b>使用 PhoneGap 桥接原生应用</b>	<b>167</b>
9.1	桥接框架简介	167
9.1.1	何时使用 PhoneGap	168
9.1.2	下载 PhoneGap	168
9.2	示例 PhoneGap 应用	168
9.2.1	构建示例应用	170
9.2.2	探索示例应用	175
9.3	一个简单的 PhoneGap 地图应用	179
9.3.1	修改示例 PhoneGap 项目	179
9.3.2	将现有代码转换为 PhoneGap 应用	184
9.4	小结	187
<b>第 10 章</b>	<b>集成 Social API</b>	<b>189</b>
10.1	连接到 Web API	189
10.1.1	何谓 JSONP	189
10.1.2	处理缺少 JSONP 支持的 API	194
10.1.3	Geominer API 简介	196
10.2	在 Moundz 中定位资源	198
10.2.1	使用 Geominer API 找到附近资源	200
10.2.2	使用 Geolocation 追踪位置	203

10.3	实现用户登录.....	206	12.2.1	构建资源详细信息界面...	261
10.3.1	构建欢迎与登录界面.....	206	12.2.2	使用 Geominer 实现资源 追踪.....	267
10.3.2	Twitter Anywhere 与登录 过程.....	208	12.3	将 Moundz 打包为原生应用.....	269
10.3.3	另一种通过 Geominer 实现的 Twitter 认证.....	212	12.3.1	针对 PhoneGap 打包.....	269
10.4	小结.....	215	12.3.2	处理应用权限.....	272
<b>第 11 章</b>	<b>移动 UI 框架比较.....</b>	<b>217</b>	12.4	PhoneGap、认证与 Intent.....	273
11.1	移动 UI 框架概览.....	217	12.4.1	之前的 Web 认证流程.....	273
11.1.1	框架之间的异同点.....	217	12.4.2	Android Intent 概览.....	274
11.1.2	为框架比较做好准备.....	219	12.4.3	使用 PhoneGap 插件处理 Intent.....	276
11.2	Jo.....	222	12.5	打包应用以进行发布.....	281
11.2.1	Jo 入门.....	222	12.6	小结.....	284
11.2.2	当 Moundz 遇到 Jo.....	224	<b>第 13 章</b>	<b>移动计算的未来.....</b>	<b>285</b>
11.3	jQuery Touch.....	229	13.1	移动计算时代.....	285
11.3.1	jQuery Touch 入门.....	229	13.1.1	全球范围的现象.....	285
11.3.2	将 jQuery Touch 应用到 Moundz 上.....	231	13.1.2	桌面已死.....	286
11.4	jQuery Mobile.....	235	13.1.3	拥抱渐进增强.....	287
11.4.1	jQuery Mobile 入门.....	236	13.2	移动技术预测.....	288
11.4.2	Moundz 与 jQuery Mobile.....	238	13.2.1	工具与库的改进.....	289
11.5	Sencha Touch.....	243	13.2.2	设备架构的变化.....	290
11.5.1	Sencha Touch 入门.....	243	13.2.3	面向未来架构的编码.....	291
11.5.2	Moundz 与 Sencha Touch.....	245	13.3	物联网.....	291
11.6	小结.....	252	13.3.1	硬件传感器网络.....	292
<b>第 12 章</b>	<b>完善与打包应用以进行发布.....</b>	<b>255</b>	13.3.2	人体传感器.....	293
12.1	继续使用 jQuery Mobile.....	255	13.4	小结.....	294
12.1.1	恢复登录界面.....	255	<b>附录 A</b>	<b>调试 Android Web 应用.....</b>	<b>295</b>
12.1.2	改进导航布局.....	259			
12.2	搜集资源.....	261			

# 入 门

欢迎来到 Android Web 应用开发的精彩世界。本书将带你领略移动 Web 应用构建的过程。虽说本书内容主要面向 Android，但大部分代码都可以用在 Chrome OS 上。事实上，应用代码的重用并不局限于 Chrome OS——本书的代码可以运行在提供了基于 WebKit 浏览器的任何设备上。如果你现在不熟悉 WebKit 或 Chrome OS，那也没关系——学完本书后就会熟悉了。

本章将会介绍一些高层次主题，这样你就能尽快开始构建应用了：

- Android 平台功能概览
- 通过 Web 浏览器可以使用哪些功能(默认情况下以及使用诸如 PhoneGap 之类的桥接框架时)
- 为编写本书示例及创建自己的应用配置开发环境
- Android 开发包自带的工具以及能够帮助你构建 Web 应用的一些支持工具概览

## 1.1 了解 Android 平台功能

Android 操作系统(OS)是一种面向移动设备(包括智能手机与平板电脑)的通用 OS。Android 的计划是成为多个设备厂商自己的设备 OS，接下来各个厂商就可以定制它并在其上构建应用。这个愿景在很大程度上已经实现了，众多厂商已经开始发布预装 Android 的设备并成为开放手机联盟(<http://openhandsalliance.com>)的成员。

但 Android 并非唯一的移动 OS，这意味着必须重写 Android 原生应用来支持其他(非 Android)移动设备。这样就需要针对希望支持的每个平台管理移动应用的开发过程。虽然大型公司能够承担得起这些成本，但小型组织或创业公司就很难。开发移动 Web 应用的吸引力正在于此——编写一次应用代码就可以运行在多种设备上。

本节将会简要介绍 Android OS 的现有特性，以及在构建 Web 应用时是否能够使用这些功能。

若更想了解系统功能概要以及通过浏览器或桥接框架可以访问哪些功能，则可以直接翻阅本节最后的表 1-2。

### 桥接框架

桥接框架向开发者提供了这样一种技术：可以构建部署到移动设备上的 Web 应用。你还可以通过框架对原生 API 的包装(通常使用 JavaScript)访问部分原生设备功能(比如加速计与照相机)。

本书将会介绍一些使用 PhoneGap(<http://phonegap.com>)来桥接部分原生功能的示例。PhoneGap 是出现较早的桥接框架，此外还有不少桥接框架也可以使用。但本书将专注于 PhoneGap，因为它提供了一种简单、轻量级的方式来包装移动 Web 应用而实现原生部署。

要想了解关于各种移动 Web 应用框架的更多信息，请参考我撰写的关于该主题的几篇博文。其中，下面这篇文章有来自于项目贡献者的很好的评论，介绍了这些框架擅长的领域，地址是 <http://distractable.net/coding/iphone-android-web-application-frameworks>。

虽然我很乐意在本书中对每个框架都进行详尽的介绍，但本书的关注点是构建移动 Web 应用。依我看来，它们是可以部署到 Web 并通过设备浏览器访问的应用。桥接框架应该是一种可选的手段而非必要条件。在这个特定的情况下，PhoneGap 毫无疑问是胜者。

## 1.1.1 设备连接

作为消费者，我们可能认为将自己的移动设备连接起来是理所当然的事情，但对于移动开发者(Web 应用或原生应用)来说，最好不要这么想。如果在构建移动应用时就假设应用时刻与 Web 相连，那么当连接受限时就会限制应用的可用性——这种情况的发生概率远远超出你的想象。

理解应用在不同时间具有不同的连接级别对于创建拥有良好用户体验的应用来说至关重要。

简单来说，从 Web 角度来看移动设备具有以下 3 种连接级别：

- 高带宽连接(比如 WiFi)
- 低带宽连接(比如 3G)
- 限制或无连接(离线)

目前，在构建纯 Web 应用时，只能测试是否具备连接(无法通过下载等手段测试连接速度)。这与构建原生 Android 应用不同，因为这些应用可以访问原生 API，而原生 API 提供了关于设备当前的连接类型与质量等信息。

第 5 章将会介绍 HTML5 API 中的一些特性，应用可以通过这些特性实现脱机使用；第 9 章将会介绍一些示例，这些示例使用桥接框架来访问某些原生的连接检测功能。

## 1.1.2 触摸

令目前的移动设备有别于老式设备的一个特性就是触摸界面。根据 Android 版本的不同，在原生层次上可以使用多点触摸事件与单点触摸事件。但目前，Web 应用只能支持单点触摸事件。

**说明:**

由于 Web 应用不支持多点触摸事件,因此原生应用应用 UI 的实现上具有一定的优势。未来这种状况极有可能得到改观,但现在我们还是要面对这样一种境况:某些 Android 设备支持 Web 应用的多点触摸,而某些设备则不支持。

至少在未来几年中,我们编写的代码主要还是针对单点触摸,同时还会为那些在 Web 浏览器中支持多点触摸的设备提供改进的功能(如果时间允许)。

第 7 章将会深入介绍触摸事件。

### 1.1.3 地理位置

Android OS 可以通过各种不同的实现支持地理位置检测,包括 GPS(全球定位系统)与手机信号塔,此外还有使用了 IP 嗅探等技术检测位置的 Internet 服务。在原生 API 层次中,地理位置是在 android.location 包中实现的(请参阅 <http://developer.android.com/reference/android/location/package-summary.html>),大多数桥接框架都是通过原生 API 公开此功能的。

由于 HTML5 逐渐为人所接受并且已经部分实现了(完整实现需要等到规范完成时,这还需要几年时间),因此我们无需桥接框架就可以直接在浏览器中访问位置信息了。这是通过 HTML5 Geolocation API 实现的([www.w3.org/TR/geolocation-API](http://www.w3.org/TR/geolocation-API))。第 6 章详细介绍了 HTML5 Geolocation API。

### 1.1.4 硬件传感器

现代智能手机最酷的特性之一就是配备了一系列的硬件传感器,随着技术的不断普及,硬件传感器的使用率也将会不断攀升。目前使用最为广泛的传感器之一就是三轴加速计,开发者编写的软件能以创新的方式使用它来追踪用户的交互。目前除了加速计以外,Android OS 还可以使用很多其他的硬件传感器,如果查看针对原生开发的硬件传感器 API 参考,就会发现原生 API 已经可以支持很多传感器了(参见 <http://developer.android.com/reference/android/hardware/Sensor.html>)。表 1-1 列出了各种传感器,并且说明了桥接框架 PhoneGap 是否可以访问每个传感器。如果不熟悉表中所列的传感器,可参考维基百科,上面有详细的说明信息——只需要搜索传感器名字。注意,虽然 Android SDK(软件开发包)支持很多硬件传感器,但大多数都无法通过移动 Web 应用访问。

表 1-1 Android SDK 支持的传感器

传 感 器	PhoneGap 支持
加速计	支持
陀螺仪	不支持
光	不支持
磁场	不支持

(续表)

传 感 器	PhoneGap 支持
方向	支持
按压	不支持
接近度	不支持
温度	支持

原生开发优于 Web 开发最明显的佐证之一就是前者可以访问各种传感器,随着技术的不断发展,向移动设备添加的传感器也会越来越多。虽然这是个不争的事实,但如果使用桥接框架来构建 Web 应用,那么也可以访问一些常用的传感器。

此外,PhoneGap 是开源框架,支持用户编写插件(虽然不太容易找到有价值的信息),因此完全可以使用它访问其他传感器。

### 1.1.5 本地数据库与存储

长久以来,移动设备都以各种形式支持本地存储,但直到最近才出现实现存储的标准化技术(以及技术选择)。当然了,在原生 API 层次上,Android 通过 `android.database.sqlite` 包(参见 <http://developer.android.com/reference/android/database/sqlite/package-summary.html>)实现了对 SQLite(<http://sqlite.org>)的支持。

SQLite 很快成为了嵌入式数据库事实上的标准,在实现面向 Web 技术的本地存储与数据库时更是如此。如果客户端能够访问诸如 SQLite 这样的轻量级数据库,那么我们创建的应用就可以同时存储和缓存“信息的位置拷贝”,而这些信息通常是保存在远程服务器上的。

有两个新的处于发展中的 HTML5 标准提供了除 JavaScript 以外无需任何外部服务就可以持久化数据的机制。这些新的 API,HTML5 Web Storage(<http://dev.w3.org/html5/webstorage>)与 Web SQL Database(<http://dev.w3.org/html5/webdatabase>)提供了一些优秀的工具使应用在离线情况下依然可以运行。我们将在第 3 章详细介绍这些 API。

### 1.1.6 相机支持

在触摸成为移动设备最受欢迎的一个特性之前,合适的相机当然会成为影响用户购买的一个因素了。这一点通过应用了相机的各种原生应用就能反映出来。在原生层次,访问相机是通过 `android.hardware.Camera` 类(参见 <http://developer.android.com/reference/android/hardware/Camera.html>)实现的;但在浏览器中就无法访问了——而 HTML Media Capture 规范则正在制订过程中(参见 [www.w3.org/TR/capture-api](http://www.w3.org/TR/capture-api))。

在规范制订完毕前,Web 应用可以通过桥接框架访问设备上的相机与照片库。

### 1.1.7 消息与推送通知

在 Android 2.2 中的原生层次上实现了一个叫做 Cloud to Device Messaging(C2DM,参见 <http://code.google.com/android/c2dm/index.html>)的服务。原生应用开发者可以凭借该服务



将应用注册为推送通知，这样当新的内容出现或是某些内容发生变化时移动用户就会收到通知。

浏览器中推送通知的实现还尚需时日，因为前不久才开始有工作组就这个特定领域展开讨论并给出提案(参见 [www.w3.org/2010/06/notification-charter](http://www.w3.org/2010/06/notification-charter))。

但是，由于 C2DM 是面世不久的服务，因此桥接框架尚需要一些时间才能为 Android 实现该服务。

### 1.1.8 WebKit Web 浏览器

Android OS 实现了一个基于 WebKit 的浏览器。WebKit(<http://webkit.org>)是开源的浏览器引擎，很多桌面与移动浏览器都在使用它。WebKit 引擎是众多流行的浏览器背后的基石，例如桌面上的 Chrome 与 Safari 及移动设备上的移动版 Safari 和原生 Android 浏览器(仅举几例)。单单这一点就可以说明为移动设备构建 Web 应用要优于原生应用。因为 Android 与 iPhone 都实现了原生的 WebKit 浏览器(Mobile Safari 的核心是 WebKit)，因此只需要面向 WebKit 进行开发，就可以轻松在这两种设备上实现了。

为何说拥有共同的 WebKit 很重要呢？考虑到 HTML5 与 CSS3 都是新规范，因此 Web 标准还需要经过多年的发展才会完善，移动浏览器也需要在多年之后才会统一。目前，在两个占据统治地位的消费智能手机平台间有 WebKit 这个共同元素是一个巨大的优势。作为开发者，我们构建的应用可以利用日趋稳定的 HTML5 组件(这些组件正被不断发展着的浏览器引擎所实现，如 WebKit)，并且可以在 Android 与 iPhone 上运行这些应用。接下来不妨用原生的 Android Java 代码或 iPhone Objective-C 代码试试看。

说明：

将 WebKit 作为“移动浏览器的选择”这种势头正在不断增大。BlackBerry 背后的公司 Research In Motion(RIM)已经在其新的 BlackBerry Torch 中采用了 WebKit 与 HTML5。这对于移动 Web 应用开发者来说是好消息，我相信未来的发展趋势将是跨平台的 Web 开发而非原生开发。

### 1.1.9 进程管理

自从 Apple 的 iOS 4 发布后，Android 与 iOS 设备上的进程处理方式就很相似了；但在这之前，当用户“退出”应用时，Android 与 iPhone 应用的行为有着很大差别。在 iPhone 上，一旦退出了应用，应用就会停止运行——这意味着应用无法在后台做任何事情。但在 Android 上，如果用户在没有停止的前提下退出了应用(包括 Web 应用)，应用还会继续在后台运行。

为了证实这一点，我们在 Android 手机上运行如下代码来确保当应用(在该示例中是浏览器)处于不活动状态时请求依然可以得到处理。

```
<html>
<body>
<script type="text/javascript">
```