

移动与嵌入式开发技术

Beginning Smartphone Web Development:
Building JavaScript, CSS, HTML and Ajax-Based Applications for iPhone,
Android, Palm Pre, BlackBerry, Windows Mobile, and Nokia S60



智能手机 Web标准开发实战

(美) Gail Rahn Frederick
Rajesh Lal
杨小冬

著
译

Apress®



清华大学出版社

移动与嵌入式开发技术

智能手机 Web 标准 开发实战

——为 iPhone、Android、Palm Pre、BlackBerry、Windows Mobile 及
Nokia S60 开发通用的基于 JavaScript、CSS、HTML 和 Ajax 的 Web 应用

清华大学出版社

北 京

Gail Rahn Frederick, Rajesh Lal

Beginning Smartphone Web Development: Building JavaScript, CSS, HTML and Ajax-Based Applications for iPhone, Android, Palm Pre, BlackBerry, Windows Mobile, and Nokia S60

EISBN: 978-1-4302-2620-8

Original English language edition published by Apress, 2855 Telegraph Avenue, #600, Berkeley, CA 94705 USA. Copyright © 2009 by Apress L.P. Simplified Chinese-language edition copyright © 2010 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: 01-2010-2257

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

智能手机 Web 标准开发实战——为 iPhone、Android、Palm Pre、BlackBerry、Windows Mobile 及 Nokia S60 开发通用的基于 JavaScript、CSS、HTML 和 Ajax 的 Web 应用/(美)弗雷德里克(Frederick, G. R.), (美)拉尔(Lal, R.) 著; 杨小冬 译. —北京: 清华大学出版社, 2010.12

书名原文: Beginning Smartphone Web Development: Building JavaScript, CSS, HTML and Ajax-Based Applications for iPhone, Android, Palm Pre, BlackBerry, Windows Mobile, and Nokia S60
(移动与嵌入式开发技术)

ISBN 978-7-302-24103-4

I. 智… II. ①弗… ②拉… ③杨… III. 移动通信—携带电话机—应用程序—程序设计 IV. TN929.53

中国版本图书馆 CIP 数据核字(2010)第 221248 号

责任编辑: 王 军 王滋润

装帧设计: 孔祥丰

责任校对: 胡雁翎

责任印制: 王秀菊

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

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

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 北京密云胶印厂

装 订 者: 北京鑫海金澳胶印有限公司

经 销: 全国新华书店

开 本: 185×260 印 张: 19.5 字 数: 475 千字

版 次: 2010 年 12 月第 1 版 印 次: 2010 年 12 月第 1 次印刷

印 数: 1~4000

定 价: 39.00 元

产品编号: 037195-01

作者简介

Gail Rahn Frederick 是一名移动软件架构师、移动 Web 爱好者，也是基于标准的移动 Web 开发的讲师。北美和欧洲等地超过 6 个国家和地区的 10 多家移动运营商都已部署了她开发的移动应用程序和移动 Web 站点。Gail 讲授智能手机和其他移动设备上的移动 Web 开发，包括防御性编程和移动生态系统中的其他生存之道，一切尽在 Learn the Mobile Web (<http://learnthemobileweb.com>)和波特兰社区学院(<http://www.computers.pcc.edu>)。在 Medio Systems 公司(<http://medio.com>)，她领导着一个移动软件小组，主要开发一些个性化的搜索和发现产品，重点关注移动分析领域。

Rajesh Lal 在美国山景城的 Nokia 公司工作，他是一名撰稿人、设计师、开发人员，也是一名技术传授人员。在过去 5 年中，Rajesh 一直参与移动 UI/UX 设计工作，对于各种移动设备(包括 Sony Mylo、Windows Mobile、Apple iPhone、Nokia S60 和 Maemo 设备)都有着丰富的设计开发经验。他喜欢在设计中融入客观实际的理念，并且撰写了多部关于各种小工具和小部件的著作。若要了解他的设计理念和用户体验心得，可以访问他的博客，网址是 <http://abcofdesign.com>。

技术编辑简介

James Pearce 在移动 Web 领域付出了很多的心血，作为移动博客 WAPtastic 的创始人，他从 10 年前就一直在 Argogroup 公司、AnywhereYouGo 公司工作。他最近成为 dotMobi 公司的 CTO。James 开发并运行了许多流行的移动 Web 站点和服务，编写了很多这方面的著作，同时也讲授了很多相关课程。若要了解有关他的其他信息或与他联系，请访问网站 <http://triplecodeon.com>。

Rocco Georgi 是伦敦一家 PavingWays Ltd.咨询公司的联合创始人兼 CTO，这家公司帮助客户将业务拓展到移动 Web 和移动设备领域。从 1999 年起，他就开始从事 Web 开发工作。在 PavingWays Ltd.公司，他主要负责将基于 Web 的应用程序移植到各种移动设备。Rocco 积极创作编写移动 Web 开发和移动小部件等方面的著作，并参与各种相关讨论会议，并与大家分享他的丰富经验。

A. Todd Emerson 从 2000 年就开始在多家移动数据解决方案公司担任领导职务。Todd 在美国和欧洲的多移动运营商中发起并实施了白标(White-Label)移动数据解决方案，他的解决方案设计中包含很多以移动设备为中心的技术，主要包括移动 Web、富客户端、数据同步、消息收发和语音识别。数千万的移动订阅者纷纷采用 Todd 的解决方案。除了移动运营商方面的工作，Todd 还创办了 Kikata 业务解决方案公司，主要为中小型企业提供一些新兴的技术解决方案，包括 RFID、迁移、VoIP 以及基于云的基础结构。

前言

我认同 W3C 提出的一个 Web 的原则，即 Web 中的各种服务和信息在主题上应该是一致的，可以通过各种设备进行访问，而不必考虑在显示功能上的差异。我个人认为，一个 Web 的原则指的就是，假设我在家中通过 Firefox 在线编写了一个商品列表，那么我应该能够在商店使用手机查看该列表并核对是否买齐。但实际上，移动 Web 及其生态系统在很多方面具有与众不同的特性，如访问模式、用户行为、浏览器技术以及客户端功能。

mobiThinking 最近发布的一份报告总结了这样一句箴言：“效用是移动 Web 的动力源泉”。这句话已经成为我在移动 Web 开发工作中恪守的信条，我希望大家也能遵循这样的原则。只有能够为移动用户解决实际的问题，移动 Web 内容才能算是成功的。行车路线、公共交通、企业名单、新闻标题、社交网络以及银行业务都是移动 Web 内容取得成功的范例，因为现实中的人们需要在日常生活中使用手机查询这些信息，他们希望能够随时随地获取所需的信息。

移动 Web 正在经历突飞猛进的发展。大约在 1994 年，当时还是桌面 Web 的天下。大家都认为应该会出现技术上的飞跃，但是几乎所有人都不清楚如何开始。本书主张采用基于标准的方法进行移动 Web 开发。通过对 Web 知识和移动标准的深入了解，以及对不断涌现出来的各种标准化成果持积极的怀疑态度，提供了针对移动设备构建 Web 内容的最佳方法。

本书适合刚刚开始接触移动 Web 的 Web 开发人员和设计人员。我们假设您以前没有移动设备开发和设计经验。本书的前两部分将介绍移动 Web 开发的一些基本概念、标准以及约定。第三部分将介绍移动设计和可用性原则，以及专为智能手机上功能强大的 Web 浏览器而提供的增强功能。本书的最后一部分将介绍如何优化、验证、测试移动 Web 站点，以及如何在公共 Internet 和移动生态系统中部署移动 Web 站点。

我希望本书能对您起到一定的启发指导作用，并在 Web 内容移动化和在移动 Web 上广泛传播 Web 内容的过程中提供正确的参考。祝您在移动开发过程中一帆风顺。

Gail Rahn Frederick

目 录

第 I 部分 移动 Web 开发入门

第 1 章 移动 Web 开发简介	3
1.1 移动 Web 和桌面 Web	4
1.2 移动标记语言	6
1.2.1 HTML 和 XHTML	6
1.2.2 XHTML 移动配置文件	7
1.2.3 WML	7
1.2.4 其他移动标记语言	8
1.3 移动脚本语言	9
1.4 移动样式表	10
1.5 移动行业组织和标准机构	10
1.6 移动生态系统	10
1.7 代码示例	11
1.8 小结	12
第 2 章 设置移动 Web 开发环境	13
2.1 建议的 IDE	14
2.2 移动 MIME 类型	16
2.3 Web 服务器配置	18
2.3.1 Apache	18
2.3.2 Microsoft IIS	19
2.3.3 Nginx	19
2.4 桌面设备上的移动 Web 浏览器	20
2.4.1 Firefox 和移动增件	20
2.4.2 移动浏览器模拟器	29
2.4.3 真实的移动设备	32
2.5 其他开发工具	32
2.5.1 文件比较	32

2.5.2 源代码控制	33
2.6 小结	35

第 II 部分 移动 Web 的语法

第 3 章 移动标记语言	39
3.1 选择移动标记语言	39
3.2 XHTML	40
3.2.1 为什么不选择 HTML	41
3.2.2 HTML 5	41
3.3 XHTML-MP	42
3.3.1 XHTML-MP 文档示例	43
3.3.2 适用于 XHTML-MP 的 DTD	43
3.3.3 XHTML-MP 中不支持的 XHTML 元素	43
3.3.4 XHTML-MP 中更新的功能和移动设备特有的功能	45
3.3.5 常见的实现故障	50
3.4 用于移动设备的 CSS	52
3.4.1 CSS2	52
3.4.2 无线 CSS 和 CSS 移动配置文件	52
3.4.3 确定移动设备上的 CSS 支持	57
3.4.4 移动 CSS 最佳实践	57
3.4.5 外部、内部和内联样式表	58
3.4.6 媒体选择器和依赖媒体的样式表	59
3.5 WML	60
3.5.1 标记层次结构	62

3.5.2	特殊字符	63
3.5.3	头和元数据	64
3.5.4	文本格式	65
3.5.5	链接	67
3.5.6	图像	71
3.5.7	表	72
3.5.8	计时器	74
3.5.9	变量	75
3.5.10	用户输入	78
3.5.11	其他 WML 语言功能	82
3.6	小结	83
第 4 章	设备识别和内容自适应	85
4.1	设备识别	86
4.1.1	使用 HTTP 请求头识别移动设备	86
4.1.2	使用设备数据库获取设备功能	87
4.2	内容自适应	106
4.2.1	创建设备组	108
4.2.2	选择调整点	109
4.2.3	为设备组编写内容自适应规则	109
4.2.4	实现内容自适应	111
4.2.5	移动 Web 中的内容自适应	114
4.3	小结	117
第 5 章	使用 JavaScript 和 AJAX 增加交互性	119
5.1	迭代开发方法	120
5.2	移动浏览器中的 JavaScript	120
5.2.1	ECMAScript 移动配置文件	121
5.2.2	在标记文档中嵌入 JavaScript	121
5.2.3	移动浏览器中的 JavaScript 分段	125
5.2.4	移动 JavaScript 的示例	127
5.3	移动浏览器中的 AJAX	132
5.3.1	移动浏览器的 AJAX 示例	136

5.3.2	测试移动浏览器中的 AJAX 支持	141
5.4	小结	141

第III部分 高级移动 Web 开发技术

第 6 章	移动 Web 可用性	145
6.1	适用于可用移动 Web 站点的最佳实践	146
6.1.1	案例分析 1: Bank of America	146
6.1.2	案例分析 2: CNN	148
6.1.3	案例分析 3: Wikipedia	149
6.1.4	案例分析 4: Flickr	151
6.2	移动浏览器布局比较	152
6.3	设计移动 Web 页面	154
6.3.1	灵活的参考设计	154
6.3.2	标准布局	155
6.3.3	新闻网站	155
6.3.4	搜索网站	156
6.3.5	服务网站	157
6.3.6	门户网站	158
6.3.7	多媒体共享网站	159
6.4	设计指导原则	159
6.4.1	开发移动 Web 页面的提示	160
6.4.2	为最大数量的用户创建 Web 页面	161
6.4.3	创建更好的移动用户体验	162
6.5	小结	163
第 7 章	针对智能手机浏览器增强移动 Web 页面	165
7.1	智能手机浏览器的通用 Web 技术	166
7.1.1	Viewport Meta 标记	166
7.1.2	在 JavaScript 中检测方向变更	167

7.2	移动浏览器中的 WebKit	172	9.3	公共标记检验器	219
7.3	用于 iPhone 的 Safari Mobile	175	9.3.1	W3C 标记验证服务	220
7.4	用于 Android 移动设备的 浏览器	176	9.3.2	W3C CSS 验证服务	221
7.5	用于 Palm Pre 的 webOS 浏览器	177	9.3.3	W3C mobileOK 检查器	223
7.6	BlackBerry 浏览器	178	9.3.4	mobiReady	224
7.7	Nokia Series 60 智能手机中的 Nokia Web 浏览器	180	9.3.5	Validome	226
7.8	用于 Windows Mobile 设备的 Internet Explorer Mobile 浏览器	181	9.4	小结	228
7.9	Opera Mini 和 Opera Mobile 浏览器	182	第 10 章	测试移动 Web 站点	229
7.10	小结	184	10.1	移动 Web 测试方法	230
			10.1.1	移动浏览器测试注意 事项	231
			10.1.2	选择在测试中使用的移动 设备	232
			10.2	在真实的移动设备上进行 测试	232
			10.2.1	获取移动设备	233
			10.2.2	开发人员计划	234
			10.3	在移动模拟器上进行测试	237
			10.4	在桌面浏览器中进行测试	238
			10.5	小结	240
			第 11 章	部署移动 Web 站点	241
			11.1	将移动流量路由到移动 Web 站点	241
			11.1.1	标准移动 Web 域和路径 名称	242
			11.1.2	移动切换算法	243
			11.1.3	移动切换产品	246
			11.2	移动 SEO 和流量获取	247
			11.2.1	移动搜索引擎和爬网 程序	248
			11.2.2	使用链接关系进行移动 站点发现	249
			11.2.3	移动站点地图	250
			11.2.4	适用于移动 Web 的 SEO	251
			11.2.5	需要忘记的一些 SEO 实践	252
			11.3	小结	254

第IV部分 部署到移动生态系统中

第 8 章	优化移动标记	187
8.1	移动标记的后期处理技术	188
8.1.1	最小化外部资源	188
8.1.2	删除空白、注释和不必要的 标记	190
8.1.3	对图像进行调整和代码转换 处理	198
8.1.4	响应文档的 MIME 多节 编码	199
8.2	移动浏览器的 Web 服务器 优化	203
8.2.1	gzip 或 deflate 响应压缩	204
8.2.2	HTTP 响应头中的缓存 指令	206
8.3	小结	211
第 9 章	验证移动标记	213
9.1	有效的标记在移动 Web 中的 重要性	214
9.2	验证过程不会测试的内容	218

第 12 章	如何在移动生态系统中 应对自如	255
12.1	运营商、代码转换器以及 代理	255
12.1.1	公共 Internet 上的代码 转换器	258
12.1.2	标准化代码转换器行为	260
12.2	移动 Web 的防御性编程	261
12.2.1	将标记声明为移动友好	262
12.2.2	标识来自代码转换器的 请求	263
12.3	小结	267

第 13 章	移动 Web 的未来	269
13.1	移动 Web 专家对移动领域 未来发展趋势的看法	270
13.2	小结	279

第 V 部分 附录

附录 A	移动设备中的示例 用户-代理	283
附录 B	移动设备中的示例请求头	287
附录 C	术语表	293
附录 D	案例分析：测试移动 浏览器缓存和性能	299

第 I 部分

移动 Web 开发入门

第 I 部分介绍移动行业、移动用户以及移动 Web。在这部分中，会涉及以下角色：Web 策略制定者、移动 Web 最佳实践的发起者以及移动 Web 标准的起草者。您将学习如何评估一项标准或最佳实践，以及如何判断某项标准或最佳实践是否适合您的移动 Web 项目。

了解了生态系统的概念之后，即开始为移动 Web 开发项目做准备，具体操作包括设置开发环境、选择 IDE(Integrated Development Environment, 集成开发环境)以及配置通用移动 MIME(Multipurpose Internet Mail Extension, 多用途 Internet 邮件扩展)类型的 Web 服务器。然后通过一系列增件扩展 Firefox(火狐)，从而实现以下功能：在桌面系统中查看移动 Web 页面、使用移动模拟器和真实的移动设备实现更精确的移动 Web 视图。

开发环境设置完成后，将检查标记语言和脚本语言，这些语言用于在各种移动设备(尤其是智能手机)上驱动丰富的轻型网络体验。这里将向您介绍 HTML(Hypertext Markup Language, 超文本标记语言)、XHTML-MP(XHTML Mobile Profile, XHTML 移动配置文件)和 WML(Wireless Markup Language, 无线标记语言)，然后是使用 CSS(Cascading Style Sheets, 级联样式表)的样式移动标记。最后，将回顾一下针对移动设备编写 Web 页面代码的最佳实践。

移动 Web 开发简介

移动设备是一种全新的媒介。桌面系统中总结出来的最佳实践在移动系统中完全不适用。移动设备、移动生态系统和移动用户具有独特的特性，要求采用不同于以往的最佳实践进行移动 Web 开发。

本书将讲述移动 Web 的语法、语义和生态系统。您将学习如何构建适应性强、响应迅速并且符合标准的移动 Web 站点，并确保其可以在任意移动浏览器上运行。一些简单的开发提示和技巧将改进小尺寸屏幕中的 Web 可用性。您可以进一步充实移动 Web 站点，使其适用于高级智能手机浏览器(具有电子邮件、桌面功能的 Web 浏览等集成 Internet 功能的高端手机中的浏览器)，能够呈现完整的 HTML 并实现专有扩展。开发工作完成后，您将了解如何在实际移动设备上进行全方位的测试，优化移动 Web 页面以便于网络传输，以及通过避免代码转换保留更多适用于移动生态系统的程序。

移动 Web 好比是西部荒野。一些重量级人物控制着整个生态系统，战争随处可见。意志坚定的人可以在遥远的地方占有财产、构建家园、建立社区以及与世界隔绝的繁华村落。但是，要想生存下来，最好的方式还是全副武装。而对于移动开发人员来说，在这片荒野中生存最有效的武器就是：

- 对移动标准和最佳实践具有专业而深入的了解
- 批判性思维技能和正确对待怀疑
- 千方百计找出正确的语法
- 正确评价移动用户的需求

基于标准的移动 Web 开发方法确保在各种移动浏览器和平台之间的兼容性和可用性。要想在移动 Web 开发领域取得成功，必须了解所有规则，同时还要了解什么时候可以忽略规则。

1.1 移动 Web 和桌面 Web

从根本上说,只有一种 Web。Web 内容是可以使用各种 Web 浏览器查看的标准化标记、样式、脚本和多媒体。在本书中,按照惯例将这种 Web 称为桌面 Web。我们可以在台式计算机、笔记本电脑、上网本上通过 Firefox、Opera 或 Internet Explorer 等浏览器访问桌面 Web,进行网上冲浪。

桌面 Web 由通过 TCP/IP 计算机网络链接在一起的大量服务器构成。这种服务器称为 Web 服务器,很多 Web 服务器实现超文本传输协议(HTTP)共享文档和文件。Web 服务器通过统一资源标识符(Uniform Resource Identifier, URI)提供对文本文件、标记文档和二进制资源的访问。

在 HTTP 请求中,客户端向 Web 服务器发送所需资源的 URI 以及一组请求头,其中一个请求头包含 MIME 类型列表,该列表公布客户端支持的内容类型。

在 HTTP 响应中,Web 服务器除了向客户端发送请求的文档(标记、文本或二进制文件)外,还会附带另外一组头,其中一个头包含 MIME 类型,描述传输到客户端的文档的文件类型。

移动 Web 在桌面 Web 的基础上添加了新的 MIME 类型、标记语言、文档格式和最佳实践,为小尺寸屏幕提供优化的 Web 内容,并可解决移动设备上的资源限制、Web 浏览器可用性差等问题。

移动 Web 在 Web 生态系统中引入了一些新的组件,包括:

- 针对移动设备进行了优化的标记语言和样式
- 可区分移动标记和桌面 HTML 的 MIME 类型
- 具有大量功能的浏览器客户端
- 使内容更适合上述客户端的网络代理

如果将移动 Web 比作西部荒野,桌面 Web 就可称得上是世外桃源。桌面 Web 是一种较好理解的安全开发环境,采用根据已制定的标准建立的客户端技术。截止到本书编写时,桌面 Web 已经走过将近 20 年的风雨历程了。桌面浏览器客户端是公开的、免费的、可轻易获得并且经常更新。目前人们使用的主流 Web 浏览器是由少数软件供应商和开放源代码项目生产的,这样就降低了跨平台 Web 开发中的测试难度。在桌面生态系统中,如果一个 Web 页面到达目标浏览器,则其标记在传输途中几乎一直存在,Internet 中的中介服务器不会更改这些标记。网络所有者和 Internet 服务提供商(Internet Service Provider, ISP)对于通过自动标记适配和内容重新打包优化和改善网络体验没有任何兴趣(请参阅表 1-1,了解定义移动 Web 和桌面 Web 的特征列表)。

表 1-1 移动 Web 和桌面 Web 的特征

	移 动 Web	桌 面 Web
平均会话长度	2~3 分钟	10~15 分钟
最小屏幕尺寸	90×60	800×600
最大屏幕尺寸	对于常用的设备为 240×400	无限制

(续表)

	移 动 Web	桌 面 Web
浏览器供应商	超过 12 家并且还在不断增长	只有两家,市场份额超过 5%
浏览器故障	经常出现故障。除使用可更新操作系统的智能手机外, 其他设备都无法修补	很少出现故障并且可以修补
W3C ¹ 标准	不规范。在移动行业中, 有时会忽略甚至违背这些标准	接受并充分应用
标记语言	WML CHTML XHTML Basic XHTML-MP XHTML HTML	XHTML、HTML
JavaScript 和 AJAX	90%的移动设备都不支持。采用 ECMAScript-MP 和 JavaScript。文档对象模型(Document Object Model, DOM)和支持的事件不同。一般都采用专有的 API	通常情况下支持
可寻址的客户	全球 30 亿移动订阅者	总计 10 亿台笔记本电脑、台式计算机和服务器

桌面 Web 过滤软件可以阻止查看让人讨厌的网页, 但是, Web 过滤器的工作方式是阻止页面访问, 而不是调整页面语法。

移动 Web 开发是一门全新的学科, 这样说的原因如下:

- **移动 Web 生态系统是全新的。**移动 Web 使用桌面 Web 的既有知识, 但它也有一些从移动设备独有的特性衍生出来的新的最佳实践和疑难问题。桌面隐喻并不适用。带宽占用量是一个比较关键的问题, 即使对智能手机也是如此。使用 JavaScript 架构以及异步 JavaScript 和 XML (AJAX)等 Rich Web 2.0 功能时务必要谨慎, 否则有电池电量耗尽的风险。运营商经常会控制和阻止移动 Web 站点的流量。在移动标记传输到移动浏览器的途中, 代码转换代理经常会尝试重置移动标记的格式。最后, 必须编写一些保护性的程序, 降低代码转换器暴露以及出现移动网络问题的概率。
- **移动 Web 用户是全新的。**移动 Web 用户采用独特的使用模式和导航方法。移动用户具有强烈的目标导向性和位置感知能力。在影响移动用户的移动 Web 浏览体验的主要因素中, 即包括在服务区内漫游时的网络访问问题。实际上, 对成本敏感的移动用户宁可取消网络交易, 也不愿冒因错误操作而支付费用的风险。
- **移动 Web 浏览器是全新的。**移动浏览器具有其他浏览器所不具备的优势, 同时也有一些与众不同的问题及相应的解决方法。Web 标准实现不彻底是经常出现的问

1. 万维网联盟(World Wide Web Consortium), 相关信息请访问网站 <http://w3.org>。

题。Web 页面格式错误会对移动设备产生严重的影响,包括浏览器崩溃或设备重置。用户非常需要 JavaScript 和 AJAX 等高级 Web 功能,但这些功能会影响电池的使用寿命。市场上有数十家移动浏览器供应商,确保 Web 标准遵从性的重担就落在了原始设备制造商(Original Equipment Manufacturer, OEM)和运营商肩上。

1.2 移动标记语言

目前,各种移动设备采用的移动浏览器一般是基于一定的标准,但又不一定遵从标准,这样,用户可以查看多种移动标记语言的 Web 内容,这些语言包括:

- XHTML 和 HTML
- XHTML 移动配置文件(XHTML-MP)
- CHTML (iMode)
- 无线标记语言(WML)

1.2.1 HTML 和 XHTML

HTML 是移动标记语言的旗舰产品。作为标准 Web 标记语言,HTML 被 Web 开发人员和设计人员广泛使用。很多移动浏览器都支持完整的 HTML 标记集,但是这些浏览器可能无法满足直接查看桌面 HTML 网站的用户体验。对于移动设备来说,屏幕分辨率、存储容量和带宽都存在限制,有必要开发出更加优化的标记和样式。当然,移动用户还希望针对他们的移动特性开发出一些专用的服务。

XHTML 在严格遵循 XML 语法的基础上,结合运用 HTML 标记集。对于移动浏览器来说,处理和呈现 XML 格式的标记要比处理松散的 HTML 语法规则容易得多。对于支持 HTML 的移动浏览器来说,XHTML 是最佳标记。

Android、iPhone、Nokia Series 60、Windows Mobile 和 BlackBerry 设备中的智能移动浏览器都支持 XHTML、HTML、JavaScript 和 AJAX。此功能集以及可选择添加的大量客户端缓存和 CSS 扩展功能构成了交互式移动 Web 应用程序的基础。本书介绍如何针对智能手机浏览器构建高级的 Web 应用程序。

注意:

仅针对智能手机开发的移动 Web 站点可以使用 HTML 4 的完整功能集,并且在不久的将来,还可以使用 HTML 5 的完整功能集。但是,在移动 Web 站点上使用 HTML 和 XHTML 需要支付一定的费用。使用 HTML 和 XHTML 后,无法与使用旧版浏览器的高容量功能手机(市场上大量销售的低成本、功能很少的手机)兼容(尽管智能手机做了大肆的宣传,但还是有大量的功能手机用户在移动 Web 上冲浪)。使用桌面标记还需要引入代码转换器,也就是通过重置标记格式优化桌面 Web 使之适用于移动设备的网络应用程序。代码转换器可以使系统认为标记是用于桌面浏览器的,同时机器会对标记进行重新调整,使其适合移动浏览器。本书教您应对以下两种情况:如何最大程度地使 HTML 与移动浏览器兼容;如何避免对针对移动设备进行了优化的 HTML 进行双重代码转换。

1.2.2 XHTML 移动配置文件

XHTML 移动配置文件(XHTML-MP)由开放移动联盟(<http://openmobilealliance.org>)指定和维护,实际上,它就是移动 Web 的标准标记。移动配置文件,顾名思义,这种标记语言是 XHTML 的一个子集,专用于移动计算设备,包括手机:

- XHTML-MP 1.0 设定了移动标记语言的基本标记。
- XHTML-MP 1.1 添加了<script>标记并支持移动 JavaScript。
- XHTML-MP 1.2 添加了更多表单标记和文本输入模式。截止到本书编写时,很多移动浏览器还不支持 XHTML-MP 1.2。

实际上,所有新开发的移动 Web 站点都使用 XHTML-MP 为移动用户提供服务。

这种标记语言在移动 Web 中引入了一些常用的概念,如分离标记结构和显示(presentation)。XML 格式的标记定义文档结构,而级联样式表(CSS)控制显示。大多数 XHTML-MP 移动浏览器都支持无线 CSS、CSS 移动配置文件和/或 CSS 2。当然,大多数意味着并不是所有支持 XHTML-MP 的移动浏览器也支持 CSS。

欢迎进入移动 Web 开发的荒凉世界。本书将教您如何使用设备数据库确定移动浏览器的疑难问题,如不能很好地支持 CSS,这样您就可以对标记进行相应的调整。幸运的是,移动浏览器不断发展,Web 标准遵从性和整体质量水平都得到大幅提升。

XHTML-MP 是无线应用协议(Wireless Application Protocol, WAP)第二版中规定的标记语言。尽管从技术上讲不够准确,但业界还是习惯性地将 XHTML-MP 称为 WAP2。

1.2.3 WML

无线标记语言(WML)是一种旧版的简单标记语言,适用于低功耗的移动设备。1998 年,无线应用协议论坛(也就是现在的开放移动联盟)对这种语言进行了标准化处理。WML 是可扩展标记语言(XML)的一种行业说法,主要使用隐喻卡片组和卡片。一个标记文档可以包含多个用户界面(UI)屏幕或卡片。WML 最初设计用于在内存和处理能力极其有限的单色移动设备上显示文本。移动开发人员使用集成开发环境(IDE)或文本编辑器以纯文本的形式编写 WML,或者使用服务器端 Web 脚本语言生成代码。在某些移动网络中,WML 网关服务器会将标记编译为二进制格式,以压缩的形式传输到设备,这样可以提高传输速度。支持 WML 的移动浏览器反编译并显示该二进制 WML,也可以直接显示文本 WML。

WML 包含两个主要的版本:WML 1.1 和 WML 1.3。与前者相比,WML 1.3 引入了对彩色图像的支持。实际上,现在的所有移动浏览器都支持 WML 1.3 以及其他标记语言。目前美国境内所使用的移动设备中,大约 5%的设备其浏览器仅支持 WML,剩下的 95%支持并首选 XHTML-MP、XHTML 和/或 HTML。

WML 是 WAP 规范第一版中规定的标记语言。因此,在移动行业里,也将 WML 称为 WAP1。这在技术上讲并不是很准确,因为 WAP 规范覆盖整个协议栈(包括标记本身),但尽管如此,还是沿用这种叫法。

WML 被认为是旧版移动 Web 语言。这种语言实在是老套,以至于 Apple iPhone 为追赶潮流而在其支持 Web 的移动浏览器中明确表示不再支持 WML。尽管 WML 语言已经存