

<<Web2.0高级编程>>

图书基本信息

书名：<<Web2.0高级编程>>

13位ISBN编号：9787115168917

10位ISBN编号：7115168911

出版时间：2008-1

出版时间：人民邮电出版社

作者：威利斯

页数：465

译者：赵子鹏

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## 内容概要

Web 2.0旨在让Web重归发明它的初衷，即让用户创建内容。

从技术角度说，Web 2.0就是以全新的方式使用大量已有的技术，这意味着专业开发人员必须了解众多Web 2.0技术，如HTML、CSS、HTTP、JavaScript、XML、服务器端编程等。

市面上的大部分图书都过于专业化，对于Web 2.0应用使用的每项技术，都可找到大量介绍它们的优秀图书，但从专业角度阐述如何综合使用这些技术的图书凤毛麟角。

本书旨在填补这一空白，让读者对Web 2.0技术及其如何协同工作有初步认识，从而真正理解Web 2.0应用的工作原理以及如何开发它们。

本书首先通过一个Web 2.0应用演示了服务器和浏览器之间的交互过程，让读者对本书中介绍的技术在Web 2.0应用中扮演的角色有更深入的认识；然后介绍了客户端技术，包括(X)HTML与CSS、JavaScript与Ajax、客户端设计准则、有望成为主流的客户端技术以及胖客户端替代品；接下来讨论了用于在客户端和服务端之间通信的协议和格式，包括HTTP与URI、XML及其替代品、联合、微格式和Web服务；最后，讨论了服务器端编程技术，包括通过HTTP提供XML、如何使用非XML数据源、如何创建联合频道、创建mashup、如何实现和维护URI空间、提供多媒体内容以及安全。

本书适合参与开发Web 2.0项目的专业开发人员阅读，也可供对Web 2.0开发感兴趣的人员参考。

## 书籍目录

第1章 Web 2.0简介1.1 BuzzWatch简介1.2 概况1.3 探究后台实现1.4 让BuzzWatch成为更好的Web公民1.5 让BuzzWatch更易于维护1.6 完成最后一步1.7 总结第2章 页面表示2.1 创建简洁的页面2.1.1 生成有效的HTML2.1.2 使用级联样式表2.1.3 选择合适的元素2.2 从HTML到XHTML2.2.1 XHTML 1.02.2.2 XHTML 1.12.2.3 为何使用XHTML2.2.4 与HTML的差别2.3 文档对象模型2.3.1 DOM级别2.3.2 DOM API概述2.3.3 DOM API参考2.4 级联样式表2.4.1 圆角2.4.2 选项卡2.5 工具2.5.1 Firefox的DOM查看器2.5.2 Firefox的Web开发工具栏2.5.3 Internet Explorer开发工具栏2.5.4 用于Firefox的DevBoi2.6 总结第3章 JavaScript和Ajax3.1 JavaScript:了解鲜为人知但至关重要的特性3.1.1 undefined的值和类型3.1.2 ===运算符3.1.3 使用for-in进行迭代3.1.4 函数式编程3.1.5 函数参数3.2 JavaScript优化3.2.1 缩短JavaScript的下载时间3.2.2 最大限度地减少DOM更新3.3 Ajax3.3.1 一切为了用户体验3.3.2 XMLHttpRequest的历史3.3.3 XMLHttpRequest示例3.3.4 Yahoo! UI Library3.3.5 Google Web Toolkit3.3.6 处理内存泄露3.4 总结第4章 设计准则4.1 常见的设计问题4.1.1 书签4.1.2 导航4.1.3 最大限度地减少传输量4.1.4 支持低级设备4.1.5 易用性需求及指导原则4.2 总结第5章 Web 2.0的发展趋势5.1 XSLT和XPath5.1.1 XSLT Hello world示例5.1.2 更多XSLT示例5.1.3 XSLT样式5.1.4 更多XSLT示例5.1.5 在浏览器中使用XSLT5.2 SVG5.2.1 SVG Hello world示例5.2.2 在SVG中指定样式5.2.3 更多有关SVG的知识5.3 XForms5.3.1 XForms Hello World示例5.3.2 给XForms指定样式5.3.3 更多XForms示例5.4 HTML的发展趋势5.4.1 W3C的提案5.4.2 WHATWG的提案5.4.3 比较XHTML 2.0和HTML 55.5 总结第6章 胖客户端6.1 从浏览器到胖客户端6.1.1 浏览器的缺点6.1.2 一种解决方案——胖客户端6.1.3 当前的胖客户端6.2 比较胖客户端框架6.2.1 OpenLaszlo6.2.2 XUL6.2.3 XAML6.3 总结第7章 HTTP和URI7.1 Web是如何获得成功的7.2 Web 1.0: HTML、URL和HTTP7.2.1 表象之下7.2.2 URL和URI7.2.3 隐藏代码的危险7.2.4 Web服务器究竟做什么7.2.5 Web是双向的7.3 Web模型和REST7.3.1 资源7.3.2 表示7.3.3 状态7.3.4 使用HTTP方法进行传输7.4 创建HTTP服务时需要考虑的因素7.4.1 媒体类型7.4.2 感兴趣的资源7.4.3 所需的表示7.4.4 服务器响应7.5 电缆上发生的情况7.5.1 参考工具包7.5.2 HTTP示例7.6 其他表示7.6.1 内容协商7.6.2 使用HTTPTracer查看消息7.7 总结第8章 XML及其替代品8.1 XML8.1.1 XML基础知识8.1.2 XML命名空间8.1.3 Unicode和字符编码8.1.4 处理二进制数据8.1.5 通常不需要的特性8.1.6 数据类型8.1.7 模式语言8.1.8 XSLT和XPath8.1.9 其他XML技术8.1.10 现有的数据格式和协议8.1.11 API8.2 XML的替代品8.2.1 JSON8.2.2 JavaScript8.2.3 HTML8.2.4 其他格式8.3 总结第9章 联合9.1 有关联合的基础知识9.1.1 微格式9.1.2 提要9.2 联合过程9.3 联合格式9.3.1 RSS 2.0: 简单内容9.3.2 RSS 1.0: 资源描述9.3.3 RSS 1.0规范和模块9.3.4 Atom: 在两方面都最好9.3.5 Atom规范和扩展9.3.6 比较不同格式的元素9.3.7 enclosure和播客9.4 总结第10章 微格式10.1 微格式基础10.1.1 CSS和微格式10.1.2 Semantic HTML10.1.3 内容、表示和数据10.1.4 消除模糊性10.1.5 覆盖范围10.1.6 2.0标准组织10.2 创建微格式文档10.3 总结第11章 结合使用多种协议创建Web服务11.1 澄清Web服务11.2 REST服务11.2.1 一个应用程序示例11.2.2 Atom发布协议11.2.3 当前的REST服务11.2.4 更多有关REST的资源11.3 WS-\*服务11.3.1 WS-\*服务和SOAP11.3.2 WS-\*服务和WSDL11.3.3 WS-\*栈11.4 REST Vs. WS-\*11.4.1 SOAP11.4.2 WSDL11.4.3 工具支持11.5 总结第12章 通过HTTP提供XML12.1 与提供HTML有何不同12.1.1 媒体类型12.1.2 字符编码12.1.3 提供定义良好的XML12.2 提供静态内容12.3 提供动态内容12.3.1 原则12.3.2 使用Java提供XML12.3.3 使用ASP.NET和C#提供XML12.3.4 使用Ruby on Rails提供XML12.3.5 使用PHP、Perl、Python等提供XML12.3.6 使用服务器端XSLT12.3.7 使用XML平台12.4 XQuery和XML数据库12.4.1 使用eXist提供XML12.4.2 WebDAV和Subversion12.5 提供JSON12.6 总结第13章 数据库和非XML数据源13.1 处理非XML数据源13.2 将关系数据转换为XML13.2.1 使用SQL Server进行服务器端转换13.2.2 使用ADO.NET进行客户端转换13.2.3 使用Oracle XSQL13.3 将二进制数据转换为XML13.3.1 手工转换13.3.2 自动转换13.4 总结第14章 创建联合频道14.1 简单的Atom服务14.1.1 用户界面14.1.2 应用程序的架构14.1.3 Abdera Atom工具包14.1.4 代码概览14.2 运行应用程序14.2.1 编译和部署应用程序14.2.2 部署方面的考虑因素14.3 添加电子邮件支

持14.3.1 理解邮件支持架构14.3.2 建立本地邮件服务器14.3.3 使用MailClient类14.3.4  
MessageReader接口14.3.5 MailEntryCollector14.4 总结第15章 Mashup、HTML抓取和Web服务15.1  
流行的示例：地图mashup15.2 为什么使用mashup15.3 mashup的商业模式15.4 屏幕抓取15.4.1  
优点和缺点15.4.2 公平竞争15.4.3 下载页面15.4.4 分析页面15.4.5 简单的屏幕抓取示例15.5 创建  
提要15.5.1 在服务器上部署15.5.2 通过外部服务部署15.6 地图服务和Badge15.6.1 badge15.6.2  
Google Maps15.7 总结第16章 实现和维护URI空间16.1 确保URI经受住时间的考验16.1.1 独立于  
技术16.1.2 层次结构和集合16.1.3 末尾的斜杠和位置无关性16.2 管理URI空间变更16.2.1 HTTP重  
定向基础16.2.2 永久性重定向资源16.2.3 暂时重定向资源16.2.4 不应使用的重定向方法16.2.5 服务器  
端重定向16.3 URI映射工具箱16.3.1 Java Servlet16.3.2 Apache mod\_alias和mod\_rewrite16.3.3  
ASP.NET和IIS16.4 总结第17章 播客和提供多媒体17.1 格式迷宫17.1.1 描述格式17.1.2 容器和  
编码格式17.2 协议17.2.1 组播17.2.2 单播17.2.3 选择协议17.3 总结第18章 安全18.1 什么是安  
全18.1.1 资源安全18.1.2 消息安全18.2 历史教训18.3 分层方法18.4 身份验证和授权18.4.1 IIS  
和ASP.NET中的身份验证和授权18.4.2 IIS和ASP.NET中各种身份验证方法18.4.3 Apache中的身份验  
证和授权18.5 消息加密18.5.1 对称加密18.5.2 公钥设施18.6 消息摘要18.7 数字证书18.8 安全套  
接字层18.8.1 SSL通信的步骤18.8.2 在IIS中配置SSL18.8.3 在Apache中设置SSL18.9 代码安全18.10  
Web服务安全18.10.1 使用Web服务安全18.10.2 该Web服务的工作原理18.10.3 添加策略18.10.4  
添加自定义身份验证18.10.5 创建客户端18.10.6 改善安全性18.11 总结

## <<Web2.0高级编程>>

### 编辑推荐

Web 2.0架构在灵活的Web设计、创造性复用和方便更新方面提供了无限的可能性。除介绍Web 2.0涉及的重要技巧和语言外，这本独特的图书还从专业角度阐述了Web 2.0涉及的各种技术。

每章都提供了多个示例应用程序，它们是使用流行的框架创建的。

本书首先介绍了用于创建Web 2.0应用程序的技术。

这包括深入探讨XHTML、级联样式表（CSS）、JavaScript和Ajax；然后，引导读者更深入地认识用于在Web客户端和服务端之间交换信息的协议和格式；最后介绍了实现新想法和开发健壮的应用程序所需的服务器端编程知识。

通过阅读本书。

读者将学到：  
· 如何开发Web 2.0应用程序；  
· 结合使用主流客户端技术的新方式；  
· 有关HTTP与URI、XML、联合、微格式和Web服务的知识；  
· 实现和维护URI空间的技巧；  
· 如何通过HTTP提供XML文档；  
· 创建mashup以聚合多个数据源中信息的步骤；  
· 改善应用程序安全性的方法。

本书针对的读者：  
本书是为对HTML、CSS、JavaScript和XML有基本认识的专业开发人员编写的。

#### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>