

The
Pragmatic
Programmers

TURING 图灵程序设计丛书

34个实例，消灭亟待解决的Web开发问题，实现轻松进阶

HTML5与CSS3 实例教程（第2版）

HTML5 and CSS3, Second Edition

Level Up with Today's Web Technologies

【美】Brian P. Hogan 著
卢俊祥 译



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

HTML5与CSS3 实例教程（第2版）

HTML5 and CSS3, Second Edition

Level Up with Today's Web Technologies

【美】Brian P. Hogan 著

卢俊祥 译



人民邮电出版社

北京

图书在版编目 (C I P) 数据

HTML5与CSS3实例教程 / (美) 霍根 (Hogan, B. P.)
著 ; 卢俊祥译. -- 2版. -- 北京 : 人民邮电出版社,
2014.8

(图灵程序设计丛书)

ISBN 978-7-115-36340-4

I. ①H… II. ①霍… ②卢… III. ①超文本标记语言
—程序设计—教材②网页制作工具—教材 IV. ①TP312
②TP393.092

中国版本图书馆CIP数据核字(2014)第154195号

内 容 提 要

HTML5 和 CSS3 技术是目前整个网页的基础。本书共分 3 部分,集中讨论了 HTML5 和 CSS3 规范及其技术的使用方法。这一版全面讲解了最新的 HTML5 和 CSS3 技术,所有实例均使用最新特性实现,针对的是最新版本的浏览器。

本书适合所有使用 HTML 和 CSS 的 Web 开发人员学习参考。



- ◆ 著 [美] Brian P. Hogan
- 译 卢俊祥
- 责任编辑 朱 巍
- 执行编辑 陈婷婷
- 责任印制 焦志炜
- ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市海波印务有限公司印刷
- ◆ 开本: 800×1000 1/16
印张: 15.5
字数: 367千字 2014年8月第2版
印数: 9 501 - 13 000册 2014年8月河北第1次印刷
著作权合同登记号 图字: 01-2014-0767号

定价: 49.00元

读者服务热线: (010)51095186转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京崇工商广字第 0021 号

前言

对Web开发者来说，Web世界中的三个月犹如真实世界的一年。也就是说，自本书上一版出版至今已经有12个“Web年”了。

作为Web开发者，总能耳闻目睹一些新的技术趋势。在几年以前，HTML5和CSS3看起来还很遥远，但由于当前各种浏览器，如Chrome、Safari、Firefox、Opera以及IE等，都在努力实现HTML5和CSS3规范中的各项内容，因而现今企业和组织也都在业务系统的建设过程中纷纷采用这些新技术。

在Web应用交互方面，HTML5和CSS3帮助开发者打下了坚实的基础。新技术让我们能够创建更易开发和维护以及更具交互友好性的网站。HTML5提供了能够定义站点结构并嵌入内容的元素，这就意味着我们不必采用额外的属性、标记或插件。CSS3提供了高级选择器、图像增强功能以及更好的字体支持，让我们的网站更有吸引力，同时将我们从以往使用图片替代技术、复杂的JavaScript代码或者图像工具等旧有开发模式中解放出来。更好的可访问性能够提升动态JavaScript客户端应用对残障用户的支持，离线应用支持特性使得我们创建不需要网络连接即可运行的应用成为可能。

在本书中，我们将实践HTML5和CSS3技术，即使你的用户所使用的浏览器目前还不支持这些新特性，本书也能够帮助你弄清楚如何在项目中应用它们。在开始之前，我们先花点时间来讨论一下HTML5及相关的流行观点。

HTML5：平台还是规范

HTML5是一个规范，它描述了一些新的标签和标记，同时还包括了一些非常棒的JavaScript API，但HTML5陷入了炒作与浮夸的漩涡。于是，本是标准的HTML5最终演变成了平台性质的HTML5，给开发者和用户造成了很多困惑。在某些情况下，CSS3规范中的一些模块，诸如阴影、渐变和转换，也被当成了HTML。浏览器厂商正试图通过他们支持了多少“HTML5”新特性的方式，来达到其领先于其他浏览器厂商的目的。于是，人们开始提出诸如“用HTML5来创建网站”这类的奇怪要求。

本书的大部分内容都将聚焦于HTML5和CSS3规范本身，以及你如何根据所有常用Web浏览器对这些技术的描述方式来使用这些技术。在本书最后，我们将探索一系列跟HTML5紧密相关

的规范，它们可以即刻应用到多个平台上，如Geolocation（地理定位）和Web Sockets。虽然这些技术从严格意义上讲并非HTML5，但它们在跟HTML5和CSS3协同工作时却可以帮助你创建出令人难以置信的丰富应用。

本书内容

本书的每个章节分别聚焦于某一类特定问题，我们可以使用HTML5和CSS3来解决这些问题。每章都有一个概述以及一个覆盖全章标签、特性或概念的汇总列表。每章的主要内容将拆分为多个实例部分，这些实例将介绍特定概念，并引导你用这些概念创建一个简单的示例。各章内容分门别类并据此展开讨论。相较于将内容分成HTML5和CSS3两部分来讲解，基于亟待解决的实际问题进行分类讨论的方式更具实际意义。你会发现其中几章重点聚焦于CSS3，同时CSS3技术又很合时宜地遍及其他章节。

许多实例都包含了一个“回退方案”的章节，为某些用户的浏览器无法直接支持我们基于新特性的实现提供了解决方案。我们将通过各种技术实现来确保回退方案的正确运行，从第三方JavaScript库到我们自己的JavaScript代码实现，以及jQuery解决方案，等等。

每章最后还包括了一个“未来展望”章节，其中讨论了当新特性在将来得以更广泛采纳时，如何更好地应用这些技术。

首先，我们会快速了解一下HTML5和CSS3的概貌，并认识一些新的结构化标签，它们可以用来描述你的页面内容。之后，我们构造一个表单，你将有机会了解表单字段及其特性，诸如自动聚焦和就地编辑等。从那开始，你将使用一些CSS3的新选择器来给元素应用样式，从此告别通过给内容添加额外标记来实现的旧有做法。

接下来，我们将探究HTML5的音频和视频支持，并且你将掌握如何通过画布绘制形状。你也将了解到如何使用CSS3的阴影、渐变与转换等特性，以及字体、过渡和动画特性。

然后，我们将使用HTML5客户端的一些新特性，如Web Storage、IndexedDB以及离线支持，来创建客户端应用程序。同时，我们会通过Web Sockets实现一个简单的聊天服务通信程序，并讨论HTML5是如何实现跨域消息发送的。你还将有机会接触到Geolocation API的使用，以及管理浏览器历史记录的知识。

本书聚焦于在现代浏览器中你可以实现怎样有趣的应用。尽管一些附加介绍的HTML5和CSS3新特性当下还不具备广泛应用的成熟度，但仍然是有价值的。你将在第11章进一步了解到这些新特性。

附录A是全书所涉及新特性的一个完整列表，并提供了具体特性所对应描述章节的快速参考。本书会大量应用jQuery，因此，附录B会为你提供一个简短的jQuery快速入门教程。附录C将带你一览如何对音频和视频进行编码，以便在HTML5中使用它们。

浏览器兼容性

每章开头都会给出本章所涉及的HTML5特性，每个特性的浏览器支持情况会在括号内以浏览器名称简写码以及所支持最低版本号来表示。简写码的含义为：C代表Chrome、F代表Firefox、S代表Safari、IE代表Internet Explorer、O代表Opera、iOS指带Safari的iOS设备、A指Android浏览器。

本书未涉及的内容

我们在本书里不会讨论IE 8之前的IE版本。微软已经在积极推动用户放弃这些老旧浏览器了。

我们也不会涉及HTML5和CSS3的方方面面。有些内容并没有太大的讨论意义，因为其实现已经改变，或者还不具备实际应用能力。比如，CSS网格布局确实不错^①，但不值得我们花太多时间投入研究，除非所有浏览器都支持该特性。^②本书重点为你演示如何使用HTML5和CSS3技术来为最大用户群体增强应用功能。

本书没有涉及HTML和CSS的基础内容，因此，它并非针对完全初学者。本书主要针对已经具备了较多的HTML和CSS知识的Web开发者。如果你才刚刚接触HTML和CSS，可以阅读本书参考文献中列出的Jon Duckett的著作*HTML and CSS: Design and Build Websites*[Duc11]，这本书很好地覆盖了HTML和CSS基础内容。也可以参考Jeffrey Zeldman的著作*Designing with Web Standards*[Zel09]。

本书假设你至少对JavaScript和jQuery^③有着基本的了解，我们将使用它们来实现很多回退解决方案。附录B提供了一个关于jQuery的简单介绍，内容覆盖了本书使用的jQuery基本方法，你可以考虑阅读Christophe Porteneuve的*Pragmatic Guide to JavaScript*[Por10]一书，将其作为对JavaScript的深入参考，这本书的最后部分描述了非常精彩的JavaScript高级用法，而我相信你会做得更好。

第2版的更新内容

本书第2版结合新的技术发展趋势对内容作了全面更新，并删除了特别针对IE 7及之前IE版本的内容。增加了更多关于HTML5可访问性的内容、更稳定且切实可行的回退方案以及9个新实例：

- 实例2：用<meter>元素实现进度条；
- 实例4：常见问题描述列表；
- 实例8：不借助JavaScript验证用户输入；

① <http://www.w3.org/TR/css3-grid-layout/>

② 目前只有IE 10、IE 11支持，<http://caniuse.com/#feat=css-grid>。——译者注

③ <http://www.jquery.com>

- 实例19: 使用SVG绘制矢量图形;
- 实例22: 视频播放的可访问性;
- 实例16: 提升表格的可访问性;
- 实例26: 通过过渡和动画特性移动物体;
- 实例28: 使用IndexedDB将数据存储到客户端数据库中;
- 实例34: 通过拖放来整理内容。

另外,你将在第11章探索CSS3的弹性盒子模型、跨域资源共享、Web Workers、服务器发送事件以及CSS滤镜效果等内容。

除了上述新内容,其他实例内容也已根据需要更新了相应的回退解决方案。同时,你会发现在本书源代码中新增加了一个便利的基于Node.js实现的Web服务器,使用它可以很方便地测试所有跨浏览器的应用功能。

如何阅读本书

没有必要从头至尾阅读本书。本书已经将内容分解成了便于单独理解与学习的各个实例,每个实例都聚焦于一到两个核心概念。每一章都会涉及一些相关项目。如果你下载了本书源代码^①,将看到一个template/文件夹,其中包含了将要用到的通用模板文件,这将会是一个不错的着手点。

如下示例代码的第一行标注了该示例代码文件在本书源代码中的位置:

```
html5_new_tags/index.html
```

```
<link rel="stylesheet" href="style.css">
```

如果你阅读的是英文电子版,可以点击该标注来打开整个示例代码文件,以便在上下文中查看代码。这个标注指出了示例代码文件在本书源代码中的位置,但它也许并不总是与你实际操作的文件相匹配。

最后,请跟随本书代码,放心大胆地练习和调整本书提供的完整示例代码。接下来,让我们来详细了解一下运行本书示例代码之前应做何种准备。

准备工作

你需要Firefox 20或更高版本、Chrome 20或更高版本、Opera 10.6或者Safari 6等浏览器来测试本书代码。由于每款浏览器在功能实现上都会有一些差别,因此,你可能还希望在所有浏览器上都测试我们编写的代码。有一台Android或iOS设备在手边会很有帮助,但不是必需的。

^① <http://pragprog.com/titles/bhh52e/>

在 IE 浏览器上进行测试工作

你同时也应该考虑到使用IE 8及更高版本来测试网站功能的现实需要，这样才能确保我们创建的回退解决方案运行无误。最简单的方式就是在VirtualBox上安装微软Windows来进行测试。^①微软在Modern.IE网站上提供了免费的虚拟机，可用于测试Web应用程序，你可以从上面下载现成可用的VirtualBox、Parallels或VMware镜像文件。^②这些虚拟机软件有30天的免费试用期，30天之后需要重新下载。

Node.js 与示例服务器

测试书中的某些特性时，需要通过一个Web应用服务器来运行HTML和CSS文件。另外，测试其他的一些特性时也需要一个相对复杂的后台应用服务。本书源代码中提供了一个应用服务器，可以满足你的需要。要运行这个服务器，你需要事先按照Node.js网站^③上的相关说明安装好Node.js，并需要0.10.0或以上版本的Node.js，以避免服务器崩溃。

你还需要用到npm，这是一个命令行工具，用来安装Node打包模块，因此，你可以用它来安装依赖模块。这个工具是Node.js安装文件中的一部分。

一旦安装好了Node.js，就可以访问本书站点并下载示例代码。对压缩文件进行解压，然后通过终端窗口（在Windows平台则是命令行窗口）进入解压后的文件所在位置，并运行以下命令，下载所有的依赖模块（注意不包括\$符号）：

```
$ npm install
```

之后输入以上命令，同样不要输入\$符号：

```
$ node server
```

我们需要在端口8000上运行应用服务器。在浏览器中加载<http://localhost:8000>，就可以浏览应用示例了。如果你在虚拟机上进行测试，虚拟机应该使用运行示例服务器的计算机的真实IP地址来连接。最棒的是，与server文件放在同一文件夹里的文件与文件夹将会通过示例服务器来运行，因此，你可以直接使用本书源代码文件夹的既有组织方式，来跟随本书内容按部就班地进行学习。

使用 JavaScript 和 jQuery 的注意事项

在本书中，我们将大量使用JavaScript。在过去，在页面<head>标签里加载JavaScript文件是

① <http://virtualbox.org>

② <http://modern.ie>

③ <http://nodejs.org>

一种很常用的实践方式，之后使用诸如jQuery的`document.ready()`方法来等待DOM（Document Object Model，文档对象模型）准备就绪，以对DOM进行操作。但是现在的推荐做法是在页面底部加载所有的脚本，这样可以获得更好的性能。因此，我们也将这么做，所有的脚本包括jQuery，都将放在页面底部，但是需要在元素加载之前对DOM进行操作的场景除外。

此外，我们将在合适的地方使用jQuery，如果只是打算简单地通过元素ID来查找某个元素，我们将直接使用`document.getElementById()`。但如果需要进行事件处理或者更复杂的DOM操作以在IE8中实现回退方案时，我们会考虑使用jQuery。

换言之，我们要“根据任务的不同选择合适的工具”。这可能会带来一些矛盾，但当我们开始介绍回退方案以使应用在老式浏览器上正常运行时，这将是一个不错的折中办法。在后面，我会解释为什么要这样做。

在线资源

本书网站提供了连到讨论论坛以及勘误表的链接，并提供了本书所有示例源代码的下载链接。^①

如果你在阅读过程中发现了错误，请在勘误页面提交勘误项，以便后期修改。本书英文电子版中，在每页的页脚处都提供了一个勘误页面链接，方便你提交勘误信息。

最后，务必访问本书博客Beyond HTML5 and CSS3^②。我将发布本书相关的内容、更新以及应用示例。

准备好了吗？让我们开始精彩纷呈的HTML5和CSS3之旅吧。

^① <http://www.pragprog.com/titles/bhh52e/>

^② <http://www.beyondhtml5andcss3.com>

版 权 声 明

Copyright © 2013 The Pragmatic Programmers, LLC. Original English language edition, entitled *HTML5 and CSS3: Level Up with Today's Web Technologies, 2nd Edition*.

Simplified Chinese-language edition copyright © 2014 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由 The Pragmatic Programmers, LLC.授权人民邮电出版社独家出版。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。

版权所有，侵权必究。

致 谢

对于一本书的第2版来说，面世速度理应很快，因为它不过是针对第1版的错误进行更正，对内容进行完善和更新。但是，你可知道，本书第2版却差不多是另写了一本新书！在写作过程中，有很多人给予了我莫大的帮助，借此机会在这里对他们表示感谢！

首先，感谢读者选择了这本书。希望你读完它后，其中的这些知识能够切实帮助你实现一些很酷且有趣的项目。

还有 Pragmatic Bookshelf 的优秀团队，我不仅要致以谢意，还会记住他们在本书出版过程中的功劳。Susannah Pfalzer 再一次保证了我又一本书的品质。她是本书的策划编辑，感谢她花费时间指导我以及对书中各种细节内容的关注，尤其是与 HTML5 和 CSS3 相关的书，有无数细节需要注意。Dave Thomas 和 Andy Hunt 给了我很多的意见反馈，感谢他们长久以来的支持！感谢大家！

我很幸运地遇到了本书的技术评审组，他们能力超强！点评和意见反馈都非常精彩、详实，各种改进建议内容丰富。感谢你们给予我的所有帮助：Cheyenne Clark、Joel Clermont、Jon Cooley、Chad Dumler-Montplaisir、Jeff Holland、Michael Hunter、Karoline Klever、Stephen Orr、Dan Reedy、Loren Sands-Ramshaw、Brian Schau、Matthew John Sias、Tibor Simic、Charley Stran 和 Colin Yates，感谢你们的深入审核！你们提供的大量建议和真知灼见在很大程度上决定了本书的最终内容。

感谢 Jessica Janiuk 提供了本书用到的 Android 设备上的截屏图片。

感谢我的业务伙伴 Chris Warren、Chris Johnson、Mike Weber、Nick LaMuro、Austen Ott、Erich Tesky、Kevin Gisi 以及 Jon Kinney，感谢你们一直以来的支持！

最后，我的妻子 Carissa 非常努力地工作，以确保我也可以非常努力（安心）地写作。她是我写作期间在身后默默支持我的“合作伙伴”，我永远感激她对我的爱以及所给予我的支持！谢谢你，亲爱的 Carissa，感谢你为我做的一切！

目 录

第 1 章 HTML5 和 CSS3 概述.....	1	2.1.3 页脚标签.....	14
1.1 强大的 Web 开发平台.....	1	2.1.4 导航标签.....	14
1.1.1 向后兼容.....	1	2.1.5 区块和文章.....	15
1.1.2 更具描述性的标记.....	2	2.1.6 旁白与侧边栏.....	16
1.1.3 更加简化.....	2	2.1.7 为博客设置样式.....	18
1.1.4 用户界面增强.....	2	2.1.8 回退方案.....	20
1.1.5 更好的表单控件.....	3	2.2 实例 2: 用<meter>元素实现进度条.....	21
1.1.6 可访问性增强.....	3	2.2.1 回退方案.....	22
1.1.7 更高级的选择器.....	4	2.2.2 进度条标签.....	24
1.1.8 视觉效果.....	4	2.3 实例 3: 用自定义数据属性构造弹出窗体.....	25
1.1.9 更少依赖插件的多媒体功能实现.....	4	2.3.1 行为与内容分离, 以及 onclick() 的问题.....	25
1.1.10 应用能力增强.....	4	2.3.2 增强可访问性.....	25
1.1.11 跨文档通信.....	4	2.3.3 不用 onclick().....	26
1.1.12 Web Sockets.....	4	2.3.4 引入自定义数据属性.....	27
1.1.13 客户端存储.....	5	2.3.5 回退方案.....	27
1.2 注定充满挑战的未来之路.....	5	2.4 实例 4: 常见问题描述列表.....	28
1.2.1 处理老版本的 IE.....	5	2.5 未来展望.....	29
1.2.2 可访问性.....	5	第 3 章 创建友好的 Web 表单.....	30
1.2.3 废弃的标签.....	6	3.1 实例 5: 使用新的输入字段描述数据.....	31
1.2.4 市场利益的激烈竞争.....	7	3.1.1 表单描述.....	32
1.2.5 HTML5 和 CSS3 的标准仍在不断发展中.....	7	3.1.2 使用范围字段创建滑动条.....	33
3.1.3 使用选值框处理数值.....	33	3.1.3 使用选值框处理数值.....	33
3.1.4 日期选择字段.....	34	3.1.4 日期选择字段.....	34
3.1.5 电子邮件字段.....	34	3.1.5 电子邮件字段.....	34
3.1.6 URL 字段.....	35	3.1.6 URL 字段.....	35
3.1.7 颜色选择字段.....	35	3.1.7 颜色选择字段.....	35
3.1.8 给表单设置样式.....	36	3.1.8 给表单设置样式.....	36
3.1.9 回退方案.....	36	3.1.9 回退方案.....	36
第一部分 用户界面增强			
第 2 章 新的结构化标签和属性.....	10		
2.1 实例 1: 用语义标记重新定义博客.....	11		
2.1.1 一切皆始于正确的文档类型.....	13		
2.1.2 页眉标签.....	13		

3.2	实例 6: 借助 autofocus 跳到表单第一个字段	41
3.3	实例 7: 通过占位文本提供提示信息	41
3.4	实例 8: 不借助 JavaScript 验证用户输入	45
3.4.1	正则表达式验证	46
3.4.2	为字段添加样式	47
3.4.3	回退方案	48
3.5	实例 9: 通过 contenteditable 属性实现就地编辑功能	49
3.5.1	用户资料表单	50
3.5.2	数据持久化	51
3.5.3	回退方案	52
3.6	未来展望	55
第 4 章	设置内容及界面的样式	57
4.1	实例 10: 使用伪类设置表格样式	58
4.1.1	使用 :nth-of-type 给表格行设置条纹	60
4.1.2	使用 :nth-child 对齐表格列的内容	61
4.1.3	使用 :last-child 加粗表格最后一行	62
4.1.4	使用 :nth-last-child 反向遍历元素	63
4.1.5	回退方案	64
4.2	实例 11: 使用 :after 和 content 生成打印友好的链接	65
4.2.1	创建只适用于打印的样式表	66
4.2.2	双冒号语法	67
4.3	实例 12: 使用媒体查询构建移动页面	67
4.4	实例 13: 创建多栏布局	70
4.4.1	拆分栏	70
4.4.2	特定厂商的前缀	73
4.4.3	回退方案	74
4.5	未来展望	75
第 5 章	构建可访问性界面	76
5.1	实例 14: 使用 ARIA role 属性提供导航提示	77

5.1.1	地标角色	78
5.1.2	文档结构角色	80
5.1.3	回退方案	80
5.2	实例 15: 创建访问性良好的可更新区域	81
5.2.1	创建页面	82
5.2.2	隐藏内容区域	85
5.2.3	回退方案	86
5.3	实例 16: 提升表格的可访问性	86
5.3.1	把标题与列结合起来	88
5.3.2	用题注和描述解释表格	89
5.4	未来展望	90

第二部分 新视角、新声音

第 6 章	在浏览器中画图	92
6.1	实例 17: 在画布上绘制商标	92
6.1.1	绘制线条	95
6.1.2	添加 “AwesomeCo” 文本信息	96
6.1.3	移动原点坐标	97
6.1.4	为对象设置渐变效果	98
6.1.5	回退方案	98
6.2	实例 18: 使用 RGraph 实现图表统计	99
6.2.1	使用 HTML 描述数据	100
6.2.2	将 HTML 描述内容转换成柱状图	102
6.2.3	显示替代内容	103
6.2.4	回退方案	103
6.3	实例 19: 使用 SVG 绘制矢量图形	104
6.3.1	绘制线条	105
6.3.2	添加文本信息	105
6.3.3	绘制形状	106
6.3.4	通过路径进行手工绘制	106
6.3.5	回退方案	107
6.4	未来展望	108
第 7 章	嵌入音频和视频	109
7.1	历史回顾	109
7.2	容器与编解码器	110

7.2.1	视频编解码器	110
7.2.2	音频编解码器	112
7.2.3	容器与编解码器协同工作	113
7.3	实例 20: 音频特性	113
7.3.1	创建基本的音频列表	113
7.3.2	回退方案	115
7.4	实例 21: 嵌入视频	117
7.5	实例 22: 视频播放的可访问性	121
7.5.1	添加字幕	122
7.5.2	HTML5 视频技术的局限性	123
7.6	未来展望	124
第 8 章	视觉特效	125
8.1	实例 23: 设置圆角	126
8.1.1	柔化登录表单	126
8.1.2	回退方案	128
8.2	实例 24: 阴影、渐变及转换	129
8.2.1	基本结构	130
8.2.2	添加渐进效果	131
8.2.3	添加阴影效果	132
8.2.4	旋转姓名牌	133
8.2.5	使用矩阵精确转换	134
8.2.6	透明背景	134
8.2.7	回退方案	135
8.3	实例 25: 设置字体	136
8.3.1	@font-face	137
8.3.2	字体格式	137
8.3.3	改变字体	137
8.3.4	回退方案	139
8.4	实例 26: 通过过渡和动画特性移动 物体	140
8.4.1	使用 CSS 过渡特性实现淡入 淡出效果	140
8.4.2	理解调速函数	141
8.4.3	创建过渡特效	143
8.4.4	利用 CSS3 动画特性实现表单 晃动效果	144
8.4.5	回退方案	147
8.5	未来展望	149

第三部分 标记之外

第 9 章	客户端数据储存	152
9.1	实例 27: 用 Web Storage 存储偏好 设置	153
9.1.1	创建偏好设置表单	154
9.1.2	保存及恢复偏好设置数据	155
9.1.3	为页面应用偏好设置值	156
9.1.4	回退方案	157
9.2	实例 28: 使用 IndexedDB 将数据存 储到客户端数据库中	158
9.2.1	注意事项界面	158
9.2.2	创建并连接数据库	160
9.2.3	创建注意事项表	162
9.2.4	加载注意事项	162
9.2.5	读取特定记录	164
9.2.6	创建、更新以及删除记录	165
9.2.7	回退方案	169
9.3	实例 29: 离线应用	170
9.3.1	通过 manifest 文件定义应用 程序缓存	170
9.3.2	manifest 文件与服务器端缓存 设置	172
9.3.3	检测网络连通性	172
9.4	未来展望	173
第 10 章	创建交互式 Web 应用	174
10.1	实例 30: 保存历史记录	174
10.1.1	存储当前状态	175
10.1.2	恢复上一状态	176
10.1.3	设置默认状态	176
10.1.4	回退方案	177
10.2	实例 31: 跨域通信	178
10.2.1	联系人列表	179
10.2.2	发送消息	181
10.2.3	支持网站	181
10.2.4	接收消息	183
10.2.5	IE 8 及 IE 9 中的限制	183
10.3	实例 32: WebSocket 聊天	184

10.3.1 聊天界面.....	184	11.2 跨域资源共享	205
10.3.2 与聊天服务器的通信	186	11.3 Web Workers	205
10.3.3 回退方案	188	11.4 服务器发送事件	210
10.3.4 Flash Socket 策略文件	189	11.4.1 监听事件	210
10.4 实例 33: 你在哪儿: Geolocation	190	11.4.2 实现你自己的服务器	212
10.4.1 为 AwesomeCo 公司开发 定位功能	191	11.5 滤镜效果	212
10.4.2 如何被找到	192	11.6 WebGL	213
10.4.3 回退方案	192	11.7 前进!	214
10.5 实例 34: 通过拖放来整理内容	194	附录 A 快速参考	215
10.5.1 创建基本用户界面	194	附录 B jQuery 快速入门	220
10.5.2 添加卡片到界面	195	附录 C 针对 Web 的音频和视频编码	229
10.5.3 整理卡片	196	附录 D 相关资源	231
10.5.4 回退方案	199	参考文献	233
10.6 未来展望	200		
第 11 章 未来之路	202		
11.1 使用弹性盒子模型定义布局	202		

第 1 章

HTML5和CSS3概述

HTML5和CSS3是万维网联盟（W3C）及其工作组提出的两个新标准，但它们的意义远超出了新标准本身，它们将是你每天都会用到的下一代Web新技术，可以帮助你更好地构建现代Web应用。在深入探究HTML5和CSS3之前，我们先来讨论一下这两个标准带来的好处，以及面临的一些挑战。

1.1 强大的 Web 开发平台

HTML5包含了众多新特性，力图打造更好的Web应用开发平台。从更丰富的描述性标签、更好的跨站点以及跨窗口通信到动画和完善的多媒体支持，借助于HTML5，开发者就可以拥有大量新工具，创造出更出色的用户体验。

1.1.1 向后兼容

我们之所以学习HTML5，最主要的原因之一是现今的绝大多数浏览器都支持它。即使在IE 6上，你也可以使用HTML5并慢慢转换旧的标记。你甚至可以通过W3C验证服务来验证HTML5代码的标准化程度（当然，这也是有条件的，因为标准仍在不断演进）。

如果你用过HTML或XML，肯定会知道文档类型（doctype）声明。其用途在于告知验证器和编辑器可以使用哪些标签和属性，以及文档将如何组织。此外，众多Web浏览器会通过它来决定如何渲染页面。一个有效的文档类型常常通知浏览器用“标准模式”来渲染页面。

以下是许多网站使用的相当冗长的 XHTML 1.0 Transitional 文档类型：

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

相对于这一长串，HTML5的文档类型声明出乎意料地简单。

html5_why/index.html

```
<!DOCTYPE html>
```

把上述代码放在文档开头,就表明在使用HTML5标准。当然,你不能使用目标浏览器尚不支持的新HTML5元素,但你的文档会验证为HTML5文档。

1.1.2 更具描述性的标记

每个版本的HTML都会引入新标记,但从未像HTML5这样增加了这么多与描述内容直接相关的标记。第2章将介绍更多的元素,如定义各级标题、页脚、导航区块、侧边栏以及文章等,同时还将介绍数值计量元素(meter)、进度条以及如何自定义数据属性,了解如何标识数据。

1.1.3 更加简化

在HTML5中,大量的元素得以改进,并有了更明确的默认值。我们已经见识了文档类型的声明是多么简单,除此之外还有许多其他输入方面的简化。例如,以往我们一直这样定义JavaScript的标签:

```
<script language="javascript" type="text/javascript">
```

但在HTML5中,我们希望所有的<script>标签定义的都是JavaScript,因此,你可以放心地省略多余的属性(指language和type)。

如果想要指定文档的字符编码为UTF-8方式,只需按下面的方式使用<meta>标签即可:

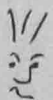
```
<meta charset="utf-8">
```

上述代码取代了以往笨拙的、通常靠复制粘贴方式来完成处理的方式:

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

1.1.4 用户界面增强

用户界面对于Web应用而言十分重要,以至于我们每时每刻都在想方设法让浏览器呈现出我们期望的效果。过去,为了设计出表或圆角效果,我们不得不借助JavaScript库,或者添加一大堆的额外标记来实现目标样式。HTML5和CSS3让这种处理方式成为过去时。



小乔爱问:

但是我喜欢XHTML的自动闭合标签,在HTML5里还能这么用吗?

当然可以!看看 Polyglot 标记¹。与HTML相比,许多开发者因XHTML对标记的严格要求而更喜欢用它。XHTML文档强制要求属性带引号、内容标签自动闭合、使用小写属性名称,给互联网带来了一种组织良好的标记表达方式。迁移到HTML5并不意味着方式的改

1. <http://www.w3.org/TR/html-polyglot/>