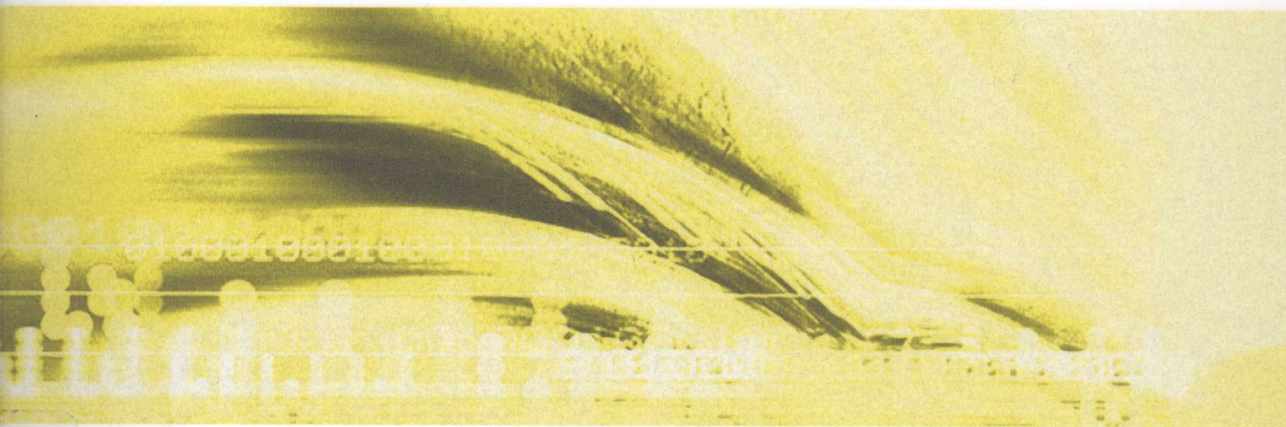


# HTML 5

## 和样式应用指南



NLIC 2970656254

(美) Larry Aronson 著  
刘红伟 等译

# HTML

## MANUAL OF STYLE

A CLEAR, CONCISE REFERENCE FOR  
HYPERTEXT MARKUP LANGUAGE

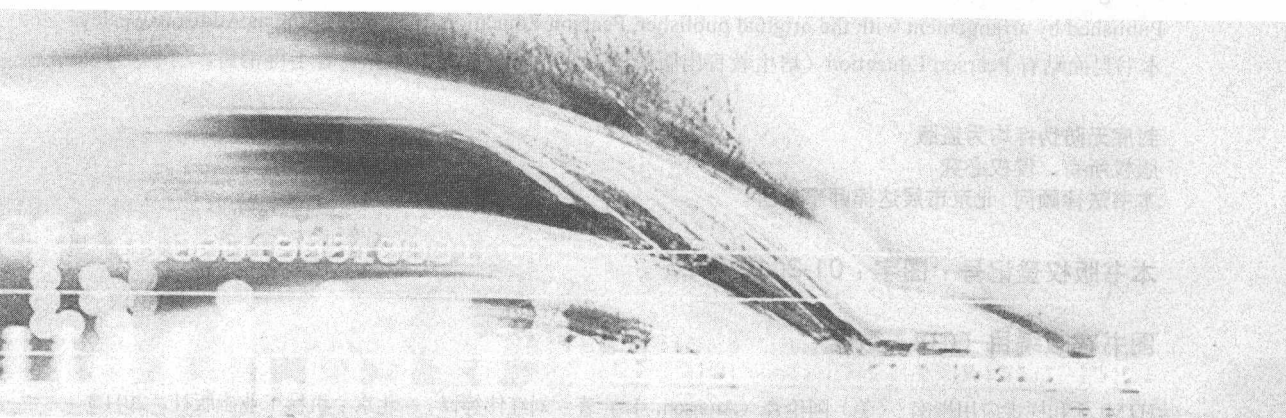
(INCLUDING HTML 5) Fourth Edition



机械工业出版社  
China Machine Press

# HTML5

## 和样式应用指南



### HTML MANUAL OF STYLE

A CLEAR, CONCISE REFERENCE FOR  
HYPERTEXT MARKUP LANGUAGE

( INCLUDING HTML 5 ) Fourth Edition

( 美 ) Larry Aronson 著  
刘红伟 等译



NLIC 2970656254



机械工业出版社  
China Machine Press

HTML 是用来构建 Web 结构和编写 Web 内容的语言, 目前已经成为构建 Web 站点的事实上的标准语言。本书主要内容有: Web 和 HTML 的基础知识、HTML 语言的核心内容和各种重要元素的用法、对 HTML 实现样式化、HTML 构建 Web 站点的技术和可能遇到的问题等。本书结合 5 种流行的应用和服务, 一一介绍 HTML 标记的用法和特点, 帮助读者实际应用 HTML。本书还分别给出了 HTML 元素和 CSS 属性的列表和说明, 其中包括最新的 HTML 5 和 CSS 3 的新增内容。

本书是初学者了解 HTML 的很好的入门材料, 也是想进一步了解 HTML 知识和应用的 Web 开发者的参考指南。

Simplified Chinese edition copyright © 2011 by Pearson Education Asia Limited and China Machine Press.

Original English language title: *HTML Manual of Style: A Clear, Concise Reference for Hypertext Markup Language (including HTML5) (Fourth Edition)* (ISBN 978-0-321-71208-0) by Larry Aronson, Copyright © 2011.

All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison-Wesley. 本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签, 无标签者不得销售。

封底无防伪标均为盗版

版权所有, 侵权必究

本书法律顾问 北京市展达律师事务所

本书版权登记号: 图字: 01-2010-7825

## 图书在版编目 (CIP) 数据

HTML 5 和样式应用指南 / (美) 阿伦森 (Aronson, L.) 著; 刘红伟等译. —北京: 机械工业出版社, 2011.3

书名原文: *HTML Manual of Style: A Clear, Concise Reference for Hypertext Markup Language (Including HTML 5) (Fourth Edition)*

ISBN 978-7-111-33195-7

I. H… II. ①阿… ②刘… III. 超文本标记语言 HTML 5—指南 IV. TP312-62

中国版本图书馆 CIP 数据核字 (2011) 第 012338 号

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑: 秦 健

北京市荣盛彩色印刷有限公司印刷

2011 年 3 月第 1 版第 1 次印刷

186mm×240mm·15 印张

标准书号: ISBN 978-7-111-33195-7

定价: 45.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991; 88361066

购书热线: (010) 68326294; 88376949; 68995259

投稿热线: (010) 88379604

读者信箱: hzjsj@hzbook.com

## 译者序

HTML (HyperText Markup Language, 超文本标记语言) 是用来构建 Web 结构和编写 Web 内容的语言。在过去的 20 年里, HTML 几乎随着 Web 的发展而同步发展, 已经成为构建 Web 站点事实上的标准语言。当今, 在 Web 朝深入影响人类生活方式和社会结构变化的过程中, HTML 再次扮演着举足轻重的角色。随着 HTML 5 的发展, 浏览器开发厂商、Web 开发者、Web 内容编写者都对 Web 的未来充满了憧憬和期望。

本书是一本经典的 Web 编写指南。它首先介绍了 Web 和 HTML 的基础知识, 然后介绍了 HTML 语言的核心内容和各种重要元素的用法, 接下来, 讲解了如何结合 CSS 来对 HTML 实现样式化。在当前的 Web 上, 有很多应用和服务支持编写者加入 HTML 标记来构建自己的内容, 本书结合 5 种这方面的流行应用和服务, 一一介绍其用法和特点, 帮助读者将学习到的 HTML 很快付诸应用。最后, 本书从宏观的角度介绍了使用 HTML 构建 Web 站点的技术和可能遇到的问题。本书末尾有两个附录, 分别给出了 HTML 元素和 CSS 属性的列表和说明, 其中包括最新的 HTML5 和 CSS3 的内容。

本书历经数版而畅销不衰。第 1 版在 1994 年出版, 印刷了 5 次并且翻译成 7 种语言, 第 2 版《HTML 3 Manual of Style》也同样成功。同时, 作者在全国范围内在线教授 HTML, 并且在 CompuServe 为 Ziff-Davis University 创办了课程。现在, 本书针对 HTML 5 进行了全面更新, 权威的内容加上最新的技术, 能够满足各层次读者学习 HTML 5 的需求。

本书由刘红伟、关志兴、王建勇、毛立涛、闫柳青、姜巧生、沈海峰、李振胜、李秋强等翻译。译文如有不妥之处, 还请读者批评指正, 可通过 [liuhongwei198004@sina.com](mailto:liuhongwei198004@sina.com) 与译者交流。

# 前言

## 本书主要内容以及为什么有用

本书介绍如何使用 HTML 将你的内容放置到 Web 上。HTML (HyperText Markup Language, 超文本标记语言) 是一种语言, 它告诉 Web 浏览器对于文本、图像和其他媒体 (你希望其他人看到的内容) 做些什么。有使用这些工具的好方法, 也有不好的方法。Web 浏览器是智能的应用程序。它们可以接受书写糟糕的 HTML, 并且仍然可以显示看得过去的 Web 页面。然而, 仍然有非常好的理由来学习如何编写好的标记。本书介绍如何创建满足如下条件的 Web 页面:

- 让人们乐于查看并且有兴趣访问。

- 对搜索引擎机器人友好。

- 易于随着时间而更新和维护。

可以通过多种比喻来理解 Web, 这些比喻允许我们把 Web 站点看做是我们探险领域中的一个地方。我们甚至在其“空间”中社会化。但这只是一种有用的假象。实际上, Web 根本不是那样的。第 1 章介绍了 Web 编写者和开发者用来创建这一假象的客户端/服务器技术。即便你认为自己是一位有经验的 Web 用户, 第 1 章也值得一看。

第 2 章完全介绍 HTML 元素, 包括一些更加有趣的 HTML 5 添加内容。它用很多实例说明了如何语义性地标记文档, 以便最终的 Web 页面为访问者 (既包括人类也包括机器人) 提供所有正确的信息, 并且易于更新。

我们在设计中的首要任务是让自己愉悦。有了好的文档结构, 可以很容易地将一个 Web 站点的所有页面都以一致的方式样式化。第 3 章通过很多实例说明了如何使用 CSS 语句来对文档元素应用样式化, 并且创建人们喜爱的 Web 页面。

第 4 章介绍使用 HTML 作为其他接受标记内容的 Web 站点的内容提供者。给出了 5 个示例, 分别是博客、GDoc、eBay、Wikipedia 编辑和 HTML 邮件。尽管将内容放到线上有很多选择, 但是有时候, 你的组织的目标或你的个人的目标制定了要创建和运行自己的 Web 站点。第 5 章



介绍了涉及的很多问题，包括 Web 站点的结构、组织和导航，以及搜索引擎优化。

在本书的末尾，你将会看到 HTML 元素和 CSS 属性的快速参考指南，包括 HTML 5 和 CSS3 草案规范中的新元素和属性。这里没有给出参考资源的列表。W3C 的 Web 站点 w3.org，以及关于 HTML 和 CSS 的 Wikipedia 文章，都从技术的观点介绍了所有的内容。你可以搜索 Web 以找到其他的指南、教程和示例。

最后，本书也是关于你的，因为你从为了信息和服务而使用 Web 的人变成了为 Web 贡献内容的人。与那些乐于创新协作的人相比，通过在线购物发现乐趣的人要逊色多了。在这个新的 Web 上，你的内容有了一席之地，并且，本书是关于如何创建带有样式的内容的。我希望这本书对你有用。

## 本书没有介绍什么以及原因

本书并不想成为介绍 HTML 5 的完全参考手册。这样的图书至少要比现在的篇幅多 3 倍，并且，可能出版不久就过时了。这是因为 Web 技术更新太快。本书中的信息都是基于 W3C 草案建议的一个 HTML 5 提案标准。尽管这可能听起来有点空洞，但很多草案规范已经由我们喜爱的浏览器所采用了（尽管在正式采用标准的时候，可能有很多已经改变了）。实际情况如此，我并不敢说是自己是 HTML 5 的权威，我只不过是一本介绍 HTML 5 的图书的作者。

除了 HTML 和 CSS，JavaScript 在本书中的某些示例中也占有一席之地。教授 JavaScript 超出了本书的范围，但是将其纳入本书有两个原因。首先，HTML 5 规范了文档元素使用 JavaScript 语法和方法来响应用户操作的行为。其次，jQuery 这样的 JavaScript 库提供了元素行为的新词汇表，这些对之前的 Web 编写者和开发者不可用。

其他的技术在一些站点的运行中起到了重要的作用，但本书中并没有真正讨论这些。如果你想要学习关于使用 Microsoft 的 Silverlight 和 Adobe 的 Flash 平台开发 Web 页面的技术，那么你选错书了。作为一位自由开发者，我喜欢免费且带有社区支持的工具。并不是说，我认为我使用的工具比其他的技术好，我只是没有用过其他的，因此，也就没有比较的基础。

## 本书是如何促成的以及将会带来什么

在 20 多年前，万维网在瑞士边境就诞生了。当我初次接触 Web 时，我的工作是在华尔街那些大型投资公司之一的顾问，不过这家公司现在已经不存在了。那是在 1993 年秋季，我为了建模抵押贷款衍生产品而将一个基于大型机的系统转换为在有一台微机上运行。我在网络管理员的办公室中，网管和我成为朋友，并且，他炫耀了最近从伊利诺伊大学的 FTP 站点上下载的一个很

酷的应用程序。这个程序叫做 Mosaic。我的生活由此开始改变，并且我那时候已经准备好要迎接这一改变。

那时候，作为 Apple 计算机的 HyperCard 应用程序爱好者和用户，我已经进入了超文本应用程序的世界。我已经为自己和他人创建了多个“栈”（HyperCard 程序的称呼）。我通过参与 Usenet 用户组 alt.hypertext 以及 Panix 的本地讨论组（这是总部在纽约市的一个较早的互联网服务提供商），继续跟进这一领域。

从 Mosaic 对 FTP 和超文本导航的无缝整合中，我立即看到了 Mosaic 和 Web 的潜能。在可以使用 Mosaic 之前，要访问一个特定的文档，首先必须知道文档位于互联网上何处。然后，登录到 FTP 服务器，下载文件，并打开它以便阅读。Mosaic 不仅将这些步骤自动化，而且帮助你找到你感兴趣的下一个文档。

那是一个令人激动的时刻。每周出现数十个新的站点。Web 浏览器的更新版本也可用了，随后 Mosaic、MacWeb、WinWeb 和 Arena 频繁地发布，支持更多的 HTML 标记元素和新的编写功能，例如居中的文本和内联的图像。每一天，在新闻组讨论和用户组会议中，都会发现和分享新的技术。

1994 年初夏，在美国政府允许互联网商用后很短的时间内，另外一位新闻组活跃撰稿人 Clay Shirky 联系了我。他问我是否感兴趣和他的出版商见面，这家出版商在寻找一位有经验的作者来编写一本关于 HTML 的图书。Clay 有其他的事务，因此，我成了关于 Web 发布的第一本图书的作者。Clay 是一位优秀的作者，他的图书《Voices from the Net》、《Here Comes Everybody》和《Cognitive Surplus》是任何想要探索新技术的文化影响的人的必读之作。

本书的第 1 版在 1994 年年底面世，并且第 2 版《HTML3 Manual of Style》在一年之后出版<sup>①</sup>。十四年弹指一挥间，到了 2009 年秋天，当我正在参加关于 HTML 5 的会议和博客时，另一位编程书作者 Elliotte Rusty Harold 给我发邮件，问我是否感兴趣和他的出版商谈谈出版《HTML Manual of Style》的新版。谈话的结果是一个正式的图书策划案和一份合同，现在，我再一次成为作者。

在回顾第 2 版的过程中，我读到了前言中的这么一段：

本书处于中间阶段。本书第 1 版在 HTML 2 完成之前编写。如今，HTML 处在向第 3 版转变的过程中。Web 本身从学术中心向商业中心过渡，并且你真的正处在从程序员 / 分析师到作者 / 讲师的职业转变中。这里介绍的一些主题是采用仍处在 beta 测试阶段的产品来说明的，这意味着，

① 本书第 3 版《HTML3.2 Manual of Style》在 1997 年出版，在 ZD 出版社被一个较大的出版公司收购之后，我没有参与其中。因此，从技术上讲，本书是《HTML Manual of Style》的第 4 版。

我今天的最佳猜测可能无法准确地描述 Web 明天将会是什么样子。本书将带你开始 Web 发布，剩下的培训学习将在网上进行。

这一段话的意义在今天仍然成立。Web 是正在发展的一项重要技术，它正在经历从商业关注向包围和塑造我们的社会活动的方向升级。主流的浏览器厂商对于新兴的 HTML 5 和 CSS 3 规范的支持，使得 Web 编写者和开发者创建令人激动的新 Web 站点和应用程序成为可能。Web 在未来几年的改变，将超越过去十多年的变化，这么说是有把握的。这也是本书为之激动的原因。

## 致谢

我想要对策划编辑 Trina MacDonald 表达感谢和感激之情，他耐心的指导和对我的理解，使得编写本书的工作充满乐趣，尽管有着交稿日期的压力。我还要感谢我的开发编辑 Songlin Qiu，他指出了我的错误，并且飞速地检查完各章，还要感谢 Elliotte Rusty Harold 联系了我，并由此启动了本书出版的过程。最后，我深深地感谢我的妻子 Heidi Cohen，她帮助我找到我的写作风格，并且对于我在在线营销和搜索引擎优化方面的知识作出了显著的贡献。



## 作者简介

Larry Aronson 出生于伊利诺伊州的 Evanston，毕业于 Evanston Township 高中，并且进入了伊利诺伊大学厄巴纳分校。在取得计算机学学士学位和物理学学士学位之后毕业之前，他有两个暑假作为 IBM 芝加哥生产办公室的助力系统工程师而工作。

毕业之后，Aronson 来到纽约市，坠入爱河，并且决定在这里生活。他在纽约市广播电台、剧院和录音行业干过一些工作，然后，开始在哥伦比亚大学计算机活动中心担任用户服务工作。四年后，他成为教员并且负责电子工程和计算机科学的研究生工作，但是，薪水微薄，随后他离开了学校去为 Boeing Computer Services 工作，开始成为一名技术销售代表，并且一直做到 BCS 纽约办公室的技术经理。

在看到自己的第一台个人计算机之后，Aronson 离开 Boeing 开始自己的业务——作为一名独立顾问。他的第一个客户是 CBS News 选举单位，在那里，他编写了 House Race Analysis Model 及其选举系统的其他部分。他其他的主要客户还包括 Mobil Oil 公司的 Product Safety Information Systems 分部。Aronson 是名骨干程序员，负责将 Mobil 的安全数据发布系统转变为一个关系数据库管理系统，并且转变为图形化、全屏幕、数据输入和显示的技术。

在 1993 年下半年，Aronson 从自己的母校下载了 Mosaic，并且认识了万维网。他开始在讨论 Web 编写和发布的新闻组和讨论列表中活跃起来，编写了关于 Web 发布的第一本书——《HTML Manual of Style》，并在 Ziff-Davis Press 出版。第 1 版印刷了 5 次并且翻译成 7 种语言，第 2 版《HTML3 Manual of Style》也同样成功。到了 1995 年，他在全国范围内在线教授 HTML，并且在 CompuServe 为 Ziff-Davis University 创办了课程。

Aronson 居住在“Silicon Alley”中心的 Manhattan loft 社区。他将自己的时间花在为个人和小公司构建 Web 应用程序上，帮助人们在互联网上工作和生活。他是 World Wide Web Artists' Consortium 的创始人，是纽约软件行业协会的董事成员，也是 Social Media Club 的创建成员。

# 目 录

|                       |     |  |
|-----------------------|-----|--|
| 译者序                   |     |  |
| 前言                    |     |  |
| 作者简介                  |     |  |
| 第 1 章 HTML 和 Web      | 1   |  |
| HTML : Web 的语言        | 1   |  |
| Web 简史                | 4   |  |
| 超文本内容和在线媒体            | 7   |  |
| 统一资源定位符               | 8   |  |
| Web 浏览器和服务器           | 9   |  |
| Web 术语                | 11  |  |
| HTML 5 和 Web 标准       | 14  |  |
| 我们都必须现在学习 HTML 5 吗    | 15  |  |
| 小结                    | 16  |  |
| 第 2 章 HTML 语言         | 17  |  |
| 语言概览                  | 17  |  |
| 页面结构和 DOM             | 21  |  |
| HTML 5 语法             | 25  |  |
| 注释                    | 25  |  |
| 字符实体                  | 26  |  |
| 标记元素                  | 26  |  |
| HTML 5 语义             | 29  |  |
| HTML 属性               | 37  |  |
| 事件处理程序                | 40  |  |
| 块级元素                  | 41  |  |
| 标题                    | 42  |  |
| 段落、块引用和地址块            | 44  |  |
| 列表                    | 49  |  |
| division 和 section 元素 | 53  |  |
| 表格                    | 56  |  |
| 链接和锚点                 | 62  |  |
| 统一资源定位符               | 63  |  |
| 锚点状态                  | 65  |  |
| 锚点属性                  | 66  |  |
| 内联图像                  | 67  |  |
| 音频和视频                 | 71  |  |
| 输入表单                  | 74  |  |
| HTML 5 画布             | 82  |  |
| 小结                    | 84  |  |
| 第 3 章 样式的元素           | 85  |  |
| 层叠样式表                 | 85  |  |
| CSS 选择器               | 88  |  |
| 伪类和伪元素                | 92  |  |
| 版式                    | 95  |  |
| 字体样式                  | 97  |  |
| 字体粗细                  | 97  |  |
| 字体变体                  | 98  |  |
| 字体大小                  | 98  |  |
| 字体                    | 101 |  |
| 颜色                    | 103 |  |
| 背景属性                  | 104 |  |
| 文本属性                  | 107 |  |
| text-align            | 107 |  |
| text-decoration       | 107 |  |

|                            |     |                        |     |
|----------------------------|-----|------------------------|-----|
| text-indent .....          | 109 | 资金 .....               | 159 |
| text-transform .....       | 109 | 未来 .....               | 160 |
| line-height .....          | 109 | Web 站点 .....           | 161 |
| 字母间距和单词间距 .....            | 110 | cgi-bin .....          | 163 |
| 空白 .....                   | 112 | 日志 .....               | 163 |
| vertical-align .....       | 112 | public_html .....      | 164 |
| 盒子属性 .....                 | 114 | 其他 Web 站点文件 .....      | 165 |
| 高度和宽度 .....                | 114 | 组织和导航 .....            | 167 |
| 边距和补白 .....                | 115 | 文件和目录 .....            | 167 |
| 边框 .....                   | 117 | 页面布局 .....             | 168 |
| 列表样式 .....                 | 120 | 导航 .....               | 171 |
| CSS 定位 .....               | 124 | 图像映射 .....             | 177 |
| 其他 CSS 属性 .....            | 128 | 开关和手风琴 .....           | 178 |
| display 和 visibility ..... | 129 | 标签页的内容节 .....          | 180 |
| overflow .....             | 133 | 打开新窗口 .....            | 185 |
| 浮动和清除 .....                | 133 | 页面头部信息 .....           | 187 |
| 小结 .....                   | 134 | meta 元素 .....          | 187 |
| 第 4 章 使用 HTML .....        | 135 | link 元素 .....          | 189 |
| 常用工具 .....                 | 135 | 其他的头部元素 .....          | 191 |
| 博客 .....                   | 138 | 搜索引擎优化 .....           | 193 |
| GDoc .....                 | 143 | 避免常见错误 .....           | 197 |
| eBay 销售 .....              | 147 | 在信息架构之前设计表现 .....      | 197 |
| 维基百科 .....                 | 149 | 使用过时的工具和构建方法 .....     | 198 |
| HTML Email .....           | 151 | 不验证 HTML 和 CSS .....   | 198 |
| 小结 .....                   | 154 | 没有在不同浏览器中测试 .....      | 199 |
| 第 5 章 构建 Web 站点 .....      | 156 | 没有使用足够的注释 .....        | 199 |
| 开发方法 .....                 | 156 | 小结 .....               | 200 |
| 内容站点或服务站点 .....            | 157 | 附录 A HTML 5 快速参考 ..... | 201 |
| 静态内容还是动态内容 .....           | 158 | 附录 B CSS 属性 .....      | 211 |
| 目标受众 .....                 | 158 |                        |     |

# 第 1 章 HTML 和 Web

HTML 是 Web 的框架。本章介绍了 Web 是如何工作的，以及 Web 的一点历史背景。你将了解到 Web 的客户端 / 服务器架构，以及它是如何进行超链接的。我将给出经典 Web 缩略语和定义的集合，并讨论 HTML 5 的原理和内涵。

尽管本章是关于 Web 和 HTML 的，但它实际上包含了非常少的 HTML 知识。如果你想直接开始学习 HTML 语言，可以跳过本章并阅读后面的内容。稍后你可以回顾这里，以帮助巩固所学到的知识。

## HTML : Web 的语言

超文本标记语言 (HyperText Markup Language, HTML) 是 Web 的语言。如果你能听得到你的计算机和你所访问的 Web 站点之间的对话，你将会听到它们说的是 HTML 语言。当你访问 Web 服务器所寄存的站点并与其交互时，Web 服务器会接收到来自你的浏览器的请求。作为回应，服务器会返回标记内容，你的浏览器会将这些内容格式化为你所看到的 Web 页面。Web 服务器还会彼此发送请求，收集并交换数据，这些数据为搜索引擎提供支持，并且使得各种广泛的社会和商务事务成为可能。

HTML 与 C、Perl 或 Ruby 等语言不同。HTML 是一种标记文本的语义性语言。这些标记为 Web 浏览器用来构建相应的 Web 页面的内容提供了一种描述。链接用 HTML 来定义。链接这种能够在一个文档中动态引用到其他文档的功能（而不管文档的物理位置在哪里）非常强大。所有的 Web 资源都使用统一资源定位符 (Uniform Resource Locator, URL) 来表示地址。任何信息都可以很容易地定位，并且与相关内容链接起来，从而创建了流畅的连接性。

Web 包含了多种协议和实践，但是，HTML 是基础，它提供了一种基本的语言，通过描述各种元素的角色和属性，将文本内容标记到一个结构化的文档中。一种相关的技术叫做层叠样式表 (Cascading Style Sheets, CSS)，允许你选择文档元素并对其应用样式规则以实现外观。CSS 规则可以混合到 HTML 代码中，或者可以放在一个外部文件中，这个文件可以应用到整个站点中，从而使得内容创建者和站点设计者都不必查遍彼此所有的工作。HTML 描述了页面的内容元素，而 CSS 告诉浏览器它们看上去（或者听上去）应该是什么样子。浏览器可以覆盖 CSS 指令

或忽略它们。

示例 1-1 创建了一个非常简单的 Web 页面。你可以将这段 HTML 代码复制到计算机上的一个纯文本文件中，并且在任何浏览器中打开它。给它一个以扩展名 .html 结尾的文件名。

示例 1-1 一个非常简单的 Web 页面的 HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>Example 1.1</title>
  <style type="text/css">
    h1 { text-align: center; }
  </style>
</head>

<body>
  <h1>Hello World Wide Web</h1>
  <p>
    Welcome to the first of many webpages.
    I promise they will get more interesting than this.
  </p>
</body>
</html>
```

示例 1-1 中的代码（粗体显示的部分）包含了两个部分：包含了页面内容的一个文档主体（body），前面的一个头部（head）部分包含了文档的相关信息。在这个示例中，头部部分包含了文档的标题（title）和将页面标题居中的一条 CSS 规则。主体部分包含一个层级 1 的标题（h1），后面跟着一个段落。其结果如图 1-1 所示。

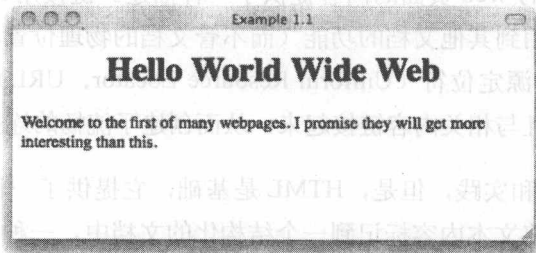


图 1-1 一个简单的 Web 页面

这涉及 Web 如何工作的一个基本原理：Web 编写者不应该对其访问者做出假设，包括他们的显示设备的特征或者他们的格式偏好。对于移动 Web 用户以及具有视觉障碍的人来说，这一点尤为重要。Web 编写者或开发者甚至不应该假设一个站点的访问者是人类。Web 站点常常是由



收集和分类与 Web 相关的信息的自动化程序来访问的。通用术语“用户代理”(user agent)，用来描述可以与一个 Web 服务器通信的任何软件应用或程序。一个现代的 Web 站点，将来自各种用户代理的访问看待得与使用 Web 浏览器的人类访问者同等的重要。最好的方法是保持 HTML 简单，以便为各种内容元素提供一个语义的描述，并且将显示细节留给访问者。

Web 编程团队中另一个主要的角色是 JavaScript，这是在浏览器中运行的一种编程语言，它操作 HTML 元素以响应用户动作或其他事件。除了 JavaScript 之外，也有其他的脚本编程语言，但是，JavaScript 是最为流行的。此外，HTML 5 规范使用的也是 JavaScript 语法和术语。和 CSS 一样，JavaScript 代码可以嵌入到一个 Web 页面的 HTML 源代码中，或者可以从一个单独的文件来导入。浏览器以外的用户代理，通常会忽略 JavaScript 和其他的嵌入式可执行代码。这对于机器人来说是很危险的。



**机器人？！** 机器人是非常重要的一类 Web 用户。它们是在 Internet 服务器上运行的自动化计算机程序，并且像真正使用浏览器的人一样去访问 Web 页面。但是，机器人不会显示页面，而是分析 Web 页面，它将页面的相关信息存储到一个数据库中，并且使用这些信息来决定接下来访问什么页面。这正是 Yahoo!、Bing 和其他搜索引擎的工作方式。另一些机器人为了市场营销和学术研究的目的而进行一些类似的数据收集。机器人通常叫做“蜘蛛”，因为它们就像蜘蛛一样从 Web 上的一个链接“爬行”到下一个链接。此外，还有一些恶意的机器人。这些自动程序在博客上留下垃圾评论，或者查找安全漏洞以获取对那些不该染指的资源的控制。坏蛋机器人！

当为 Web 创建内容的时候，你通常不会关心任何这些内容。大多数的处理浏览器、机器人和小挂件的 HTML 结构，是由你使用的 Web 编辑软件提供的，或者是由服务器端脚本和模板系统提供的。如果你直接在线编辑内容，只需要理解如何使用简单的 HTML 元素来标记内容就可以了。Web 开发者，也就是程序员，相对于 Web 编写者来说，需要完整地理解这 3 个基本的部件（HTML、CSS 和脚本编程）如何协同工作以构成 Web 的框架（见图 1-2）。

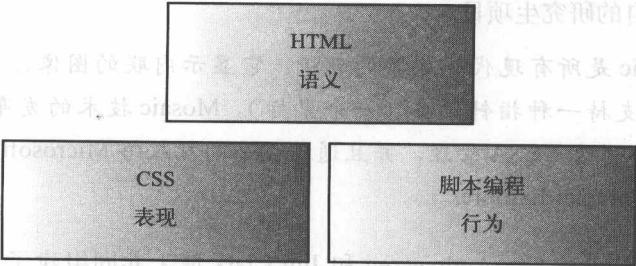


图 1-2 Web 页面的 3 个部件

顺便提一下，我说过所有这些基本上都是 free 的吗？free 在这里有两种意思。它是免费的，因为没有获取成本；它是自由的，因为你可以将其用于自己的目的。Web 页面中所有的 HTML、CSS 和脚本都可以供你查看、复制和重用，只有很少的一些例外。作为 HTML、URL 以及用户代理和 Web 服务器用来彼此通信的 HTTP 协议的发明者，Tim Berners-Lee 将所有这些放入到公共的领域中。在 CERN（European Center for Nuclear Research，欧洲原子能研究中心）工作的时候，他试图为较大的研究者团队找到一种较好的方式以快速地发布研究论文，而这些研究者不同的国家使用不同的字处理程序工作。专利权和诺贝尔奖遭遇危险。1991 年 8 月 6 日 alt.hypertext 新闻组的一篇帖子，实际上是 Web 的诞生宣言，Berners-Lee 写道：

WWW 项目开始允许高能物理学家分享数据、新闻和文档。我们非常有兴趣将 Web 扩展到其他的领域，并且为其他的数据提供网关服务器。欢迎合编写者！

二十年后，作为万维网联盟（World Wide Web Consortium，W3C）的领导者，Berners-Lee 仍然极力推动 Web 的发展。我在这里强调“发展”，是要指出，尽管 Web 已经改变了社会，使我们能够自由地在信息的海洋中工作和玩乐，但还是有很多意外情况发生。HTML 仍然是一项进行中的工作。

## Web 简史

早期的 Web 只有文本，没有图像或颜色，并且浏览器也是以命令行的模式工作的。换句话说，是通过光标和键盘在顺序链接的页面中浏览，这就好像在一部低端手机上浏览一样。直到 1993 年，位于伊利诺伊大学厄本那 - 香槟分校的国家超级计算应用中心（National Center for Supercomputing Applications，NCSA）才开发了一款叫做 Mosaic 的图形化浏览器。Mosaic 能很容易在 Windows、Macintosh 和 UNIX 上安装和使用。

Mosaic 由一群研究生编写，其中的主力是 Marc Andreessen 和 Eric Bina。之所以开发 Mosaic，是因为他们对于超文本的可能性兴奋不已，并且对当时可用的浏览器很不满意。人们本以为他们应该忙于各自的研究生项目。



Mosaic 是所有现代浏览器的鼻祖。它显示内联的图像、多种字体族、粗细和样式，还支持一种指针设备（一个鼠标）。Mosaic 技术的发布和 Mosaic 商标都由 Spyglass 公司为 NCSA 管理，并且通过该公司授权给 Microsoft，后者重写了源代码并称之为 Internet Explorer。

从伊利诺伊大学毕业之后，Andreessen 和 Jim Clark 博士共同组建了 Netscape 公司。Clark 博士则是 Silicon Graphics 公司的前 CEO，Silicon Graphics 的漂亮、强大的图形计算机 / 工作站

使得好莱坞的电影制作发生了翻天覆地的变化。Netscape Navigator 浏览器引入了重大的创新, 并且, 由于 Netscape 公司的工作, 该浏览器变得极为流行, 而 Netscape 公司做出了令当时软件行业相当震惊的事情, 它卖掉了 Navigator。在鼎盛时期, Netscape 占据了浏览器市场近 90% 的份额。

1994 年, 发生了一些美妙的事情。作为克林顿的“政府再造”计划的管理者, 美国副总统 Al Gore 安排国家科学基金会 (National Science Foundation, NSF) 将互联网销售给电信公司的一个联盟。这终结了 NSF 对于“非商业用途”政策的限制, 并且催生了 .com 时代, 人们开玩笑说, 是 Al Gore 发明了互联网。在 1994 年年中的时候, 有 2738 个 Web 站点, 而到了 1994 年年末, 有了大约 10 000 个 Web 站点<sup>①</sup>。

刚开始的时候, 互联网商业化的竞争很激烈。在 20 世纪 90 年代中期, 技术社区激烈地讨论着所谓的“浏览器战争”, 浏览器制造商在他们的软件中投入了数十种功能, 向 HTML 添加了许多新的元素, 以扩大他们各自的市场。Netscape 也添加了功能来吸引图形设计师, 包括对 jpeg 图像、页面背景颜色的支持, 甚至支持有争议的 FONT 标记以允许 Web 设计师指定文本的大小和颜色。Microsoft 将 Internet Explorer 绑定到其 Windows 操作系统中, 并且将 Web 内容发布绑定到其 Microsoft Office 产品线中。这些举措给 Microsoft 带来了相当多的法律诉讼麻烦。这些问题一直持续到 2001 年, 当 George W. Bush 刚刚开始担任总统的时候, 美国政府突然放弃对微软的反垄断诉讼。



其他的公司也向浏览器中引入了一些有趣的思想, 但是, 并没有从 Netscape 和 Microsoft 夺取到任何显著的市场份额。惠普 (Hewlett-Packard, HP) 的 Dave Raggett 所编写的一款 HTML 3 测试床浏览器 Arena, 引入了对表格、环绕图形的文本流以及内联数学表达式的支持。Sun Microsystems 发布了一款叫做 HotJava 的浏览器, 它带来了很多有趣的事情。HotJava 使用 Java 语言编写, Java 是 Sun 开发出来最初用于控制 TV 机顶盒的编程语言。Sun 针对互联网重新设定了该语言的目的, 梦想着将浏览器带到一个平台中, 该平台针对叫做 applets 的较小的、交互式的应用程序, 这些程序将在你的 PC 的 Java 虚拟机中运行。Sun 将 Java 放入到公开领域, 从而促进其采用, 并使得 Microsoft 能够制作并推广其自己的 Java 版本。Microsoft 的 Java 与 Sun 的版本有着显著的不同, 它使得使用 applet 很困难 (更不要说编写它们了)。尽管 Java 语言最终在构建内部协作应用程序中得到了广泛普及, 但 HotJava 成为 Sun 的互联网之梦的殉葬品。

在一条相关报道中, 一家叫做 WebTV Networks 的公司提供了一种低成本的互联网工具和服务, 用户可以使用一个无线键盘和遥控器来在电视机上浏览 Web 和收发 Email。尽管由

① Wikipedia: [http://en.wikipedia.org/wiki/List\\_of\\_websites\\_founded\\_before\\_1995](http://en.wikipedia.org/wiki/List_of_websites_founded_before_1995).

于面临投资困难以及与 Sony 公司之间分分合合的关系，几乎扼杀了这个项目，但 WebTV 成功地将 Web 和 Email 带给了近一百万的客户，而这些人主要是想避免拥有个人计算机的成本和复杂性。

为了说明与 Web 相关的事情会变得如何怪异，根据维基百科的介绍，WebTV 很快就被美国政府认定为一款军事武器并且禁止出口，因为它使用了很强大的加密系统。1997 年，Microsoft 收购了 WebTV，并且将其重新打造为 MSN TV 以扩展其 Web 功能。由于没有做好推广和客户服务，MSN TV 几年后死掉了。但是 WebTV 技术保存了下来，最终在 Microsoft 的 Xbox 游戏控制器中重新露面。

我所喜欢的一款 Web 浏览器是 Virtual Places，它是由以色列的 Ubique 公司开发的。Virtual Places 将 Web 浏览与互联网聊天软件结合了起来，并且允许协作式的 Web 冲浪。它将任何 Web 页面变成了一个虚拟的聊天室，其中，你和其他的访问者都通过化身来表示，化身是一个可以在整个页面中移动的、较小的个人图标。你在一个浮动的窗口中输入的任何内容，都将会出现在化身头上的一个卡通气球中。它还有一个“观光巴士”的功能，例如，允许一位教师带领一群学生浏览世界各地的 Web 站点并返回。

遗憾的是，保持开放连接的服务器以及跟踪化身位置的开销，使得 Virtual Places 无法像 Web 站点数目的爆炸式增长那样普及开来。当时，Netscape 每隔几周就更新 Navigator。由于 Ubique 无法及时更新，没有人使用 Virtual Places 作为默认的 Web 浏览器。AOL 无缘由地收购了 Ubique，几年之后又将其卖给了 IBM。IBM 将其软件中的一些技术用于公司通信和协作。在 20 世纪之初的 .com 泡沫破灭过程中，Virtual Places 死掉了，但是，化身保留了下来。

尽管 Java 很热门，Netscape 开发了 JavaScript，这是在 Netscape Navigator 浏览器中运行的一种脚本语言，并且它允许 Web 开发者为 Web 页面的 HTML 元素添加动态行为。尽管与 Java 的前四个字母相同，JavaScript 和 Java 编程语言却有着很大的不同。据推测，Netscape 将其名称从 LiveScript 改为 JavaScript，只是为了围绕 Java 展开营销。表面上看起来，二者的代码相似，因为它们都是面向对象编程系统，并且具有相似的语法。

美国在线 (AOL) 于 1988 年收购了 Netscape，并且公开了浏览器的源代码。最终，这成为 Mozilla 组织构建 Firefox 浏览器的基础。其他的公司如法炮制，在随后的几年里，各种基于 Netscape 的图形化浏览器进入到市场中。Microsoft 的 Internet Explorer (IE) 浏览器随着每个新的版本而改进，并且最终由于与 Windows 操作系统绑定而成为最流行的浏览器。

浏览器战争以 .com 泡沫的破灭而告终，厂商开始让自己的浏览器符合新兴的标准。在 W3C 的指导下，HTML 语言进展缓慢，并且在 HTML 4 规范上保持稳定。CSS 的应用促使 Web 开发