

PROGRAMMER TO PROGRAMMER™



Professional CSS
Cascading Style Sheets for Web Design, 2nd Edition

CSS Web设计 高级教程 (第2版)

(美) Christopher Schmitt
Todd Dominey
窦朝晖

等著
译



清华大学出版社

CSS Web

设计高级教程(第2版)

(美) Christopher Schmitt 等著
Todd Dominey
窦朝晖 译

清华大学出版社

北 京

Christopher Schmitt, Todd Dominey, et al

Professional CSS: Cascading Style Sheets for Web Design, 2nd Edition

EISBN: 978-0-470-17708-2

Copyright © 2008 by Wiley Publishing, Inc.

All Rights Reserved. This translation published under license.

本书中文简体字版由Wiley Publishing, Inc.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: 01-2009-3246

本书封面贴有Wiley公司防伪标签, 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

CSS Web设计高级教程(第2版)/(美)施密特(Schmitt, C.), 多米尼(Dominey, T.) 等著; 窦朝晖 译.

—北京: 清华大学出版社, 2009.7

书名原文: Professional CSS: Cascading Style Sheets for Web Design, 2nd Edition

ISBN 978-7-302-20311-7

I. C… II. ①施… ②多… ③窦… III. 主页制作—软件工具, CSS—程序设计 IV. TP393.092

中国版本图书馆CIP数据核字(2009)第092684号

责任编辑: 王 军 于 平

装帧设计: 孔祥丰

责任校对: 成凤进

责任印制: 李红英

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: 北京市世界知识印刷厂

装 订 者: 北京市密云县京文制本装订厂

经 销: 全国新华书店

开 本: 185×230 印 张: 19 字 数: 380 千字

版 次: 2009 年 7 月第 1 版 印 次: 2009 年 7 月第 1 次印刷

定 价: 68.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770177 转 3103 产品编号: 030091-01

前言

设计人员是典型的创造型人才，偏爱通过右脑思考，而注重技术细节的程序员则偏爱于用左脑思考。

但当面对Web设计的挑战时，设计人员则会运用称为CSS(Cascading Style Sheet，层叠样式表)的设计技术。

CSS是由Worldwide Web Consortium (W3C)制定的一个Web标记标准集，该标准集用于定义Web页面中的一致样式，并把该模板应用于多个页面。CSS本质上是一种技术，这种技术在很大程度上必须通过手工编写以创建一些强制性规则。这样大多数设计人员要完成自己的工作就必须编写大量的代码。

我们不需要只会手写PostScript的设计人员，而是需要能利用Adobe Illustrator提供的可视化编程环境(代码隐藏在背景中)来编写PostScript文件的设计人员。

当WYSIWYG(What you see is what you get，所见即所得)Web页面编辑人员的可视化编程经验越来越丰富时，这些应用软件不再是真正的专业CSS设计工具，如Illustrator中的PostScriptv那样。

CSS还存在一个与浏览器兼容的问题，但PostScript不存在这样的问题。浏览器开发商在其浏览器中没有及时支持这种技术。即使对CSS的支持越来越好(特别是在IE 7 for Windows中对CSS的良好支持)，但当设计人员试图在老的或过时的浏览器上支持他们的设计时仍会遇到问题。这意味着有必要深入研究CSS，并编写一些hack(在本书指一些修复浏览器错误的代码)和一些迂回解决方案。

也就意味着应该花更多的时间来编写和修改代码，而不应该把过多时间花在WYSIWYG工具的使用上。

即使了解CSS的基本元素(属性、取值、选择符等)，要充分利用这项技术也是很困难的，甚至会使人感到沮丧。CSS会妨碍我们中的大部分人，不论是对W3C规范有很好了解的人，还是那些怀着畏惧心理浏览这些规范的人。

从正面角度看，CSS确实是一种工具。

一旦设计人员掌握了该技术的基本元素，了解了其目的，并获得了该技术的足够经验，几乎任何一个画在餐巾纸上或用Adobe Photoshop完成的设计思想都可以被实现。

要达到这一点，您必须阅读本书。

对于大规模的、设计良好的多页面网站，强调以集成方式运用CSS设计出基于标准的CSS，而本书就是当今市场上倡导这种思想的为数不多的几本书之一。

针对网站开发中遇到的设计问题，本书通过网站开发的最佳实践经验和大量的网站实例，提供了独特的、CSS驱动的解决方案。

本书读者对象

本书适用于对CSS有所了解的中级和高级设计人员。如果您对如何以专业水平高效地开发CSS驱动的设计不是很清楚，学习本书将受益匪浅。特别地，本书对下列读者非常有用。

- 刚刚接触CSS并有一定HTML编程经验的中级用户——对任何一个专业的Web开发人员而言，掌握CSS都是很容易的，但要用CSS创建专业水平的Web站点则需要对CSS有非常深入的了解。
- 专业设计人员——学习过CSS的专业Web开发人员(没有20世纪90年代传统设计经验知识)并希望了解使用这种技术的最佳实例的专业Web开发人员。

本书主要内容

本书每一章都集中介绍一位设计人员以及他所设计的Web站点。每一章都提供一些容易接受的CSS技巧演示和开发网站所用到的技术。另外，设计人员通过介绍开发过程中的不同做法向读者提供了一些更深邃的见解。

下面简单介绍一下本书主要内容和合著者贡献的深刻见解。

- 第1章：“有关XHTML和CSS的最佳实例”——Ethan Marcotte，WSP(Web Standards Project)的权威委员、基于标准的Web设计运动的组织领导者。他将在该章与大家分享一些使用含CSS的XHTML(eXtended HTML，XHTML)的深刻见解。
- 第2章：“Google的blogger.com：翻转器和设计思想”——Dunstan Orchard，也是WSP的一位委员，他对blogger.com(一个Google Web站点)外观的实现内幕进行了深入研究。Douglas Bowman(资深设计人员，极力宣扬对一些Web站点进行重新设计，并取得了巨大成功，这把他推到设计与标准相兼容的Web的前沿)是该项目的

负责人之一，Orchard对他进行了访谈，给我们提供了一些极其有价值的见解。

- 第3章：“经典的美国职业高尔夫联盟锦标赛网站”——Todd Dominey，TSI(Turner Sports Interactive)的高级互动设计人员，他为大多数职业高尔夫联盟(Professional Golf Association, PGA)巡回赛设计和开发了相应的Web站点，包括PGA冠军杯和莱德杯(Ryder Cup)。该章将讲述如何根据世界各地球迷需求进行网站设计的细节，Dominey将为读者讲述独家观点。该章着重介绍的关键技术包括阴影、下拉菜单和如何把Flash内容嵌入到一个Web站点。
- 第4章：“佛罗里达大学主页UFL.edu”——Mark Trammell，负责指导美国一所著名大学的Web互动网站。他介绍了佛罗里达大学是如何从开发Web站点的过程中使学生和机构受益的。该章着重介绍的关键问题包括处理与浏览器兼容的问题和导航功能结构。
- 第5章：“Stuff and Nonsense: CSS切换策略”——除与Andy Clarke(富有创造性的Stuff and Nonsense项目组负责人)的访谈外，Ethan Marcotte还对如何改进Web站点的可访问性进行了探讨，以进一步保证了所有用户能以统一的方式访问Web站点。在该章，Marcotte仔细研究了CSS切换和如何克服令人讨厌的浏览器兼容问题。在Stuff and Nonsense站点所体现出来的创新是运用这些技术的优秀范例。
- 第6章：“CindyLi.com的风险投资：博客修改”——Cindy Li介绍了她是如何通过她的设想和CSS编码定制自己的Web站点的。
- 第7章：“辛辛那提的AIGA: HTML email模板”——Christopher Schmitt先简单介绍了创建email模板的过程，接着介绍了如何创建一个基本的HTML布局表，并说明了CSS所起的关键作用。
- 第8章：“专业CSS图书网站：透明PNG图像的使用”——Christopher Schmitt介绍了如何用PNG图像创建该图书网站，以及如何解决IE 6对PNG图像的alpha透明缺乏良好支持的问题。
- 第9章：“构建CSS布局”——Christopher Schmitt介绍了网格和布局在设计中的重要性，Ethan Marcotte探讨了如何创建一个三栏布局的稳定架构。

另外，本书附录还提供了有关HTML 4.01元素、HTML到XHTML的转换规则、CSS 2.1属性的参考资料以及一个排除常见问题的帮助指南。

源代码

在读者学习本书中的示例时，可以手工输入所有的代码，也可以使用本书附带的源代码文件。本书使用的所有源代码都可以从本书合作站点<http://www.wrox.com/>(或<http://www.tupwk.com.cn/downpage>)上下载。登录到站点<http://www.wrox.com/>，使用Search工具或使用书名列表就可以找到本书。接着单击本书详情页面上的Download Code链接，就可以获得所有的源代码。

注释：

由于许多图书的标题都很类似，所以按ISBN搜索是最简单的，本书英文版的ISBN是978-0-470-17708-2。

在下载了代码后，只需用自己喜欢的解压缩软件对它进行解压缩即可。另外，也可以进入<http://www.wrox.com/dynamic/books/download.aspx>上的Wrox代码下载主页，查看本书和其他Wrox图书的所有代码。

勘误表

尽管我们已经尽了各种努力来保证文章或代码中不出现错误，但错误总是难免的，如果您在本书中找到了错误(例如拼写错误或代码错误)，请告诉我们，我们将非常感激。通过勘误表，可以让其他读者避免受挫，当然，这还有助于提供更高质量的信息。

请给wkservice@vip.163.com发电子邮件，我们就会检查您的反馈信息，如果是正确的，我们将在本书的后续版本中采用。

要在网站上找到本书英文版的勘误表，可以登录<http://www.wrox.com>，通过Search工具或书名列表查找本书，然后在本书的详情页面上，单击Book Errata链接。在这个页面上可以查看到Wrox编辑已提交的所有勘误项。在完整的图书列表中还包括每本书的勘误表，网址是<http://www.wrox.com/misc-pages/booklist.shtml>。

p2p.wrox.com

要与作者和同行进行讨论，请加入p2p.wrox.com上的P2P论坛。这个论坛是一个基于Web的系统，便于您发布与Wrox图书相关的消息和相关技术，与其他读者和技术用户交流

心得。该论坛提供了订阅功能，当论坛上有新的消息时，它可以给您传送感兴趣的论题。Wrox作者、编辑、其他业界专家和读者都会到这个论坛上来探讨问题。

在<http://p2p.wrox.com>上，有许多不同的论坛，它们不仅有助于阅读本书，还有助于读者开发自己的应用程序。要加入论坛，可以遵循下面的步骤：

- (1) 进入p2p.wrox.com，单击**Register**链接。
- (2) 阅读使用协议，并单击**Agree**按钮。
- (3) 填写加入该论坛所需要的信息和自己希望提供的其他信息，单击**Submit**按钮。
- (4) 您会收到一封电子邮件，其中的信息描述了如何验证账户，完成加入过程。

注释：

不加入P2P也可以阅读论坛上的消息，但要发布自己的消息，就必须加入该论坛。

加入论坛后，就可以发布新消息，回复其他用户发布的消息。可以随时在Web上阅读消息。如果要让该网站给自己发送特定论坛中的消息，可以单击论坛列表中该论坛名旁边的Subscribe to this Forum图标。

关于使用Wrox P2P的更多信息，可阅读P2P FAQ，了解论坛软件的工作情况以及P2P和Wrox图书的许多常见问题。要阅读FAQ，可以在任意P2P页面上单击FAQ链接。

目 录

第1章 有关XHTML和CSS的最佳实例	1
1.1 把结构和表现标记硬挤在一起	2
1.2 学习并热衷于使用标记	7
1.2.1 XHTML: 新热点	8
1.2.2 从结构提取样式	13
1.3 CSS: 添加样式层	21
1.3.1 更好地了解选择符	21
1.3.2 其他选择符	25
1.3.3 多重声明组合	28
1.3.4 对选择符进行分组	29
1.3.5 继承	31
1.3.6 综合应用	33
1.4 了解层叠	38
1.4.1 探寻样式来源	38
1.4.2 根据优先级排序	41
1.4.3 根据顺序排序	42
1.5 把理论应用于实践	43
1.5.1 基于可靠浏览器进行构建	43
1.5.2 理性对待hack	44
1.5.3 与hack有关的问题	48
1.5.4 编写hack的技巧	49
1.6 小结	52
第2章 Google的blogger.com: 翻转器和设计思想	53
2.1 设计人员访谈	54
2.2 CSS驱动的翻转器	57
2.2.1 改变链接的颜色和背景色(简单)	58

2.2.2	改变链接的颜色和背景色(复杂).....	59
2.2.3	所要完成的工作.....	63
2.2.4	改变表格行的背景色.....	67
2.2.5	改变文本颜色.....	71
2.2.6	改变链接的背景位置.....	75
2.3	小结.....	84
第3章	经典的美国职业高尔夫联盟锦标赛网站.....	85
3.1	阴影效果的实现.....	86
3.1.1	创建幻觉效果.....	87
3.1.2	使阴影更真实.....	92
3.2	创建CSS下拉菜单.....	95
3.2.1	定制下拉菜单位置.....	96
3.2.2	定制下拉菜单的样式.....	98
3.3	与Web标准兼容的Flash嵌入.....	102
3.3.1	使用Flash Satay方法.....	102
3.3.2	用JavaScript编写object/embed标签.....	103
3.3.3	SWFObject.....	103
3.4	小结.....	104
第4章	佛罗里达大学主页UFL.edu.....	105
4.1	UF第一个Web站点回顾.....	105
4.1.1	对修改版本的反思.....	106
4.1.2	对目前网站的分析.....	108
4.2	网站定义.....	109
4.2.1	组建开发团队.....	109
4.2.2	从用户研究着手.....	110
4.2.3	自我检查.....	110
4.2.4	定义技术规范.....	111
4.3	构建主导航结构.....	112
4.3.1	XHTML.....	112
4.3.2	CSS.....	114

4.3.3 图像	116
4.3.4 实现细节	116
4.3.5 边框的构建	117
4.3.6 段落题头样式	118
4.3.7 列表样式	119
4.4 实现辅助导航	120
4.4.1 XHTML	120
4.4.2 CSS	122
4.5 再论Flash的嵌入	129
4.5.1 Flash Satay方法回顾	130
4.5.2 服务器端的Flash Satay检测	132
4.6 寻找设计中的失误	133
4.6.1 仅凭示例引导	133
4.6.2 “习惯势力”和“谁移动了我的输入框”	133
4.7 小结	134
第5章 Stuff and Nonsense: CSS切换策略	135
5.1 基础准备	136
5.2 CSS切换	142
5.3 工作机制	144
5.3.1 永久样式表	144
5.3.2 首选样式表	145
5.3.3 备选样式表	145
5.3.4 又一个(几乎)完全不能用的方案	148
5.4 目前的解决方案	149
5.4.1 转向JavaScript	150
5.4.2 PHP方案	159
5.5 超越浏览器的CSS	162
5.5.1 媒体类型: 康复的开始	164
5.5.2 选择问题	168
5.6 Stuff and Nonsense: 创建一个更好的切换器	168
5.7 设计师Andy Clarke访谈	171

5.8 小结	176
第6章 CindyLi.com的风险投资: 博客修改	177
6.1 博客	177
6.2 CSS: Cindy Li 开博客	179
6.3 设计要素	179
6.3.1 创建布局	179
6.3.2 对设计进行布局	180
6.4 创建站点	182
6.4.1 设计导航条	182
6.4.2 创建翻转图形	184
6.4.3 设置导航标记和CSS	185
6.4.4 整合翻转器	188
6.5 设置说话框	190
6.5.1 说话框编码	191
6.5.2 再次应用这种效果	193
6.5.3 添加Flickr徽章	193
6.6 复选框样式	198
6.7 小结	204
第7章 AIGA辛辛那提分会: HTML email模板	205
7.1 HTML Email简介	205
7.2 制造模板	206
7.2.1 打印设计	206
7.2.2 创建HTML表格布局	207
7.2.3 对设计进行调整	215
7.2.4 对HTML email模板的CSS规则的效果分析	216
7.3 嵌入样式	218
7.4 为HTML email进行预处理	219
7.5 小结	220

第8章 专业CSS图书网站：透明PNG图像的使用	221
8.1 PNG和浏览器支持	222
8.1.1 在IE 6中使用PNG的图像过滤方案	223
8.1.2 在IE 6中使用PNG的HTC脚本方案	223
8.1.3 PNG图像的颜色问题	224
8.2 使用Alpha透明	225
8.2.1 更好的阴影	225
8.2.2 使用彩色阴影	227
8.3 小结	230
第9章 构建CSS布局	231
9.1 网格与布局	231
9.2 做打印所不能做的事	235
9.3 CSS定位基本原理	237
9.3.1 功能强大的绝对定位	238
9.3.2 在相对定位容器内的绝对定位	241
9.4 创建三栏：布局的基础	243
9.4.1 编写XHTML：从模式到标记	244
9.4.2 样式层	247
9.4.3 解决浏览器错误	255
9.4.4 IE 5.x+/Win	260
9.5 设置边界：max-width属性	262
9.6 小结	265
附录A HTML 4.01元素	267
附录B 从HTML到XHTML的转换规则	271
附录C CSS 2.1属性	277
附录D CSS故障排除指南	285

第 1 章

有关XHTML和CSS的 最佳实例

在早期，Web并不是最吸引人的事物。由核物理学家创建和使用的超文本只不过是开放的分布式网络中共享文档的一种手段，而且这些文档的主要内容是文本。不必讳言，高品质的设计并不是早期Web开发者所优先考虑的目标。事实上，常用的HTML表格元素(也是经常被滥用的，在后面将介绍)是为一个目的而创建的：显示表格数据。

20世纪90年代后期是Web设计的全盛时期。在该时期，HTML中的“L(Language)”常常被忽视。许多专业人士觉得构建Web页面的代码在本质上并不是一种语言，因此不受真正编程语言的规则和约束的限制。此外，用户并不会为兼容的、可读性好的或防过时的代码支付费用。事实上，许多站点是根据“向后兼容”的要求开发的。这可能用词不当，因为曾经有这样的站点，只支持IE 4.0或以上版本的浏览器。

至少可以说，那个时期的浏览器性能是不稳定的。由于对浏览器W3C(<http://www.w3.org>)制定的规范支持很差，可能同一个页面在浏览器A和浏览器B中的显示结果不一样。因此，当许多程序员朦朦胧胧知道W3C制定的“标准”时，他们必须支持的浏览器还很少能容忍与标准兼容的标记。在这种意义上，把Web内容和形式进行分离的要求很迫切。对于HTML中与浏览器相关的专有标记，程序员故意使它无效以保证在目标浏览器中“看起来是好的”。所以一切暂时相安无事。程序员抱怨需求规格说明不准确、工期很紧，而且他们可能不是按钟点计酬，总之，他们有很多借口。

当然，对于以前学习过如何把表格的单元、间距和边框设置为零的设计人员，可以构建基于网格的复杂布局，使他们的站点在美学感染力方面到达一个新的高度。不可否认，由于在这个时期的浏览器对CSS的支持很差，程序员除了删除目前不用的表现代码来减小页面大小外别无选择。这导致Web自身的标记太多，页面大小以千字节计。对这样的页面而言，维护困难、重新设计的成本非常高，而且还会浪费用户的带宽。

令人欣慰的是，这样的时代过去了。扩展超文本标记语言(eXtensible Hypertext Markup Language, XHTML)和CSS这两个标准技术将消除页面中的混乱，使页面的构建工作变得越来越轻松。页面占用空间也越来越小、可读性越来越好、越来越易于维护。当然，只有在完全掌握这两个工具后才能充分发挥它们的作用。本章对引入XHTML和CSS的必要性进行了分析。为了把它们灵活运用设计中，还介绍了几个实用策略。

1.1 把结构和表现标记硬挤在一起

现在屏住呼吸，诚实地问一下自己：熟悉下面一行HTML代码吗？

```
<body marginwidth="0" marginheight="0" leftmargin="0" topmargin="0" >
```

在Web设计的全盛时期，这是把页面内容放入整个浏览器窗口的方法。没有这四个属性，设计的页面将被10个像素左右的边界所环绕——是的，类似这样的事有点太挑剔了。

强调“看起来是对的”在早期的Web设计中非常普遍，这个方案说明了这种普及程度。尽管HTML源于一种结构良好的语言，但页面已发展为一种“标签汤(tag soup)”——把结构和表现标记混在一起的页面(就像一种味道不是很好的菜炖牛肉)。由于当时的浏览器不支持CSS或支持不够，因此程序员只有依靠透明图形、字体元素和多重嵌套表来控制站点的设计。body元素包含了大量属性，很能说明这种在标记中结构和样式不匹配的问题。虽然body元素本身主要用于结构设计(它包含Web页面的内容)，但在其开始标签中的少量属性只是使结构看起来像那么回事。

必须承认，小小的body元素似乎没有如此神奇——真的值得我们关注标记的这一行吗？为了讲述标记中的问题，我们来看看哈佛大学主页(<http://www.harvard.edu>)这样的具体范例。该网站的设计(如图1-1所示)很好地反映了哈佛大学已树立的良好品牌：保守、土色调的色条着重强调了独特的哈佛红，同时中央的两列布局增加了内容的可读性。据大家反映，这是一个成功的站点设计——承载的信息非常丰富。



图1-1 哈佛大学主页

显然这是一个直观有效的设计。但是，如果把HTML中所有表格元素的边框都显示出来，表现的效果就差多了(如图1-2所示)。

说明：

现在有很多浏览器实用工具。安装这些工具后可影响页面的显示，正如这里的例子一样。对Molliza浏览器，Web Developer Toolbar (<http://chrispederick.com/work/firefox/webdeveloper>) 就是这样的工具，而且是一个非常优秀的工具。它是CSS工具集中极其有价值的一部分。设计人员可用它激活不同页面元素的边框、快速编辑页面的CSS、很容易地获得各种在线代码验证软件。对于Mac平台的Web开发人员，可在下列网址中找到相应插件：<http://hicksdesign.co.uk/journal/web-development-with-safari>。

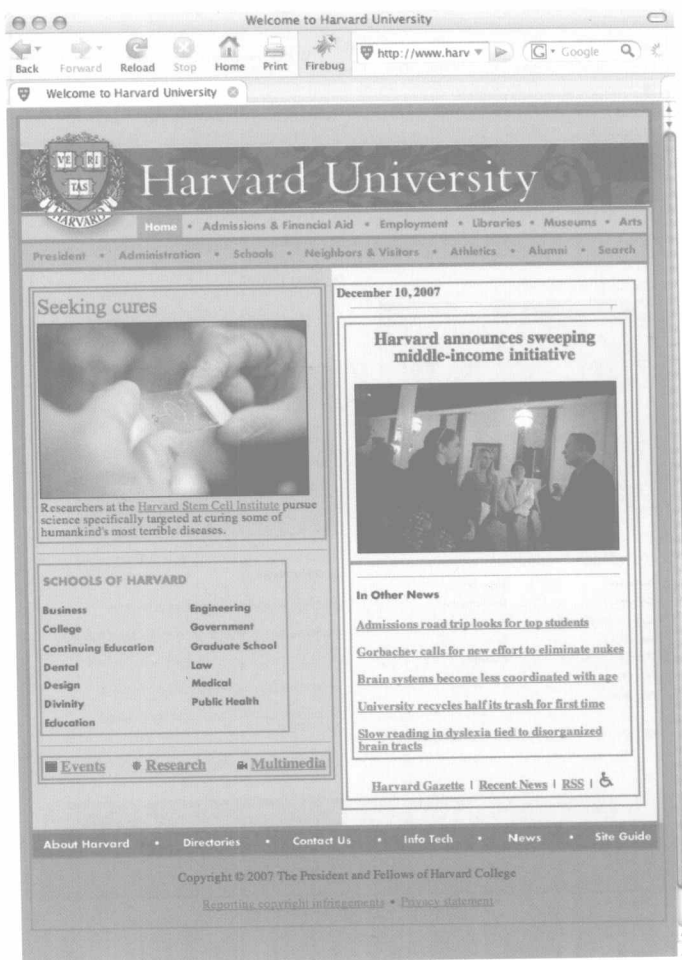


图1-2 显示表格边框的哈佛大学主页

变化很大，不是吗？在如此简单的布局中就有如此多的标记：表格嵌套深度为五层；logo图标被分解到多个文件并散布在多个表格行中。即使查看一下简单的导航条，也有人晕眩。

```
<table bgcolor="#cdd397" border="0" cellpadding="0" cellspacing="0"
width="650">
  <tbody><tr>
    <td valign="top"></td>
```