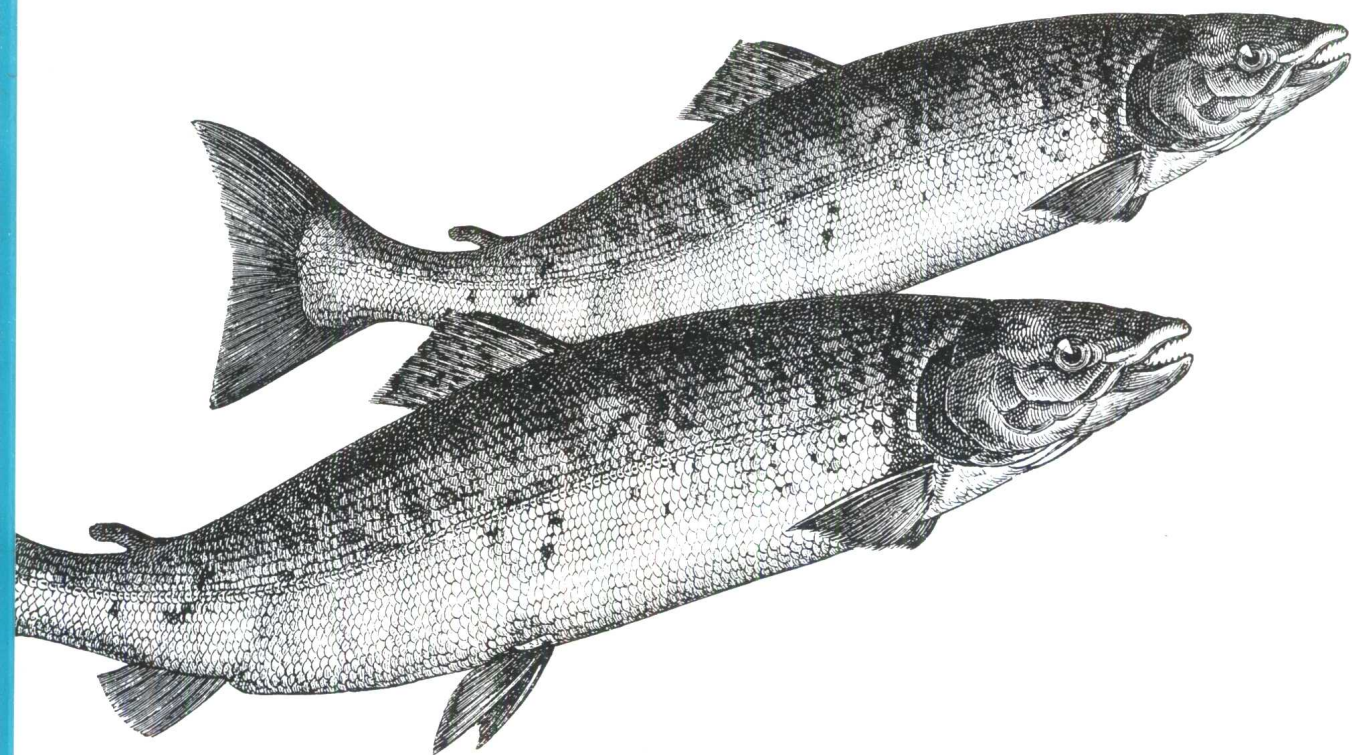


Cascading Style Sheets: The Definitive Guide

CSS权威指南



O'REILLY®
中国电力出版社

Eric A. Meyer 著

许勇 齐宁 译

CSS 权威指南

Eric A. Meyer 著

许勇 齐宁 译

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Paris • Sebastopol • Taipei • Tokyo

中国电力出版社

图书在版编目 (CIP) 数据

CSS 权威指南 / (美) 迈耶 (Meyer, E.) 编著; 许勇, 齐宁译. - 北京: 中国电力出版社, 2001

书名原文: Cascading Style Sheets: The Definitive Guide

ISBN 7-5083-0560-4

I .C... II .①迈... ②许... ③齐... III .主页制作 - 软件工具, CSS - 指南
IV .TP393.092-62

中国版本图书馆 CIP 数据核字 (2001) 第 12491 号

北京市版权局著作权合同登记

图字: 01-2001-2084 号

© 2000 by O'Reilly & Associates, Inc.

Simplified Chinese Edition, jointly published by O'Reilly & Associates, Inc. and China Electric Power Press, 2001. Authorized translation of the English edition, 2000 O'Reilly & Associates, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly & Associates, Inc. 出版 2000。

简体中文版由中国电力出版社出版 2001。英文原版的翻译得到 O'Reilly & Associates, Inc. 的授权。此简体中文版的出版和销售得到出版权和销售权的所有者——O'Reilly & Associates, Inc. 的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式复制。

书 名 / CSS 权威指南

书 号 / ISBN 7-5083-0560-4

责任编辑 / 关敏

封面设计 / Ellie Volckhausen, 张健

出版发行 / 中国电力出版社 (www.infopower.com.cn)

地 址 / 北京三里河路 6 号 (邮政编码 100044)

经 销 / 全国新华书店

印 刷 / 北京市地矿印刷厂

开 本 / 787 毫米 × 1092 毫米 16 开本 30.5 印张 450 千字

版 次 / 2001 年 5 月第一版 2001 年 5 月第一次印刷

印 数 / 0001-5000 册

定 价 / 55.00 元 (册)

译者序

自制定和发布以来，CSS规范便颇受世人的青睐。特别是对于专业网页设计者来说，CSS无疑是给他们的设计带来了新的希望。传统的HTML语言只是一种简单的结构标记语言，还不能够很好地控制网页的外观。而CSS的引入，正好弥补了这一缺陷。运用CSS，不仅能够设计出精美的网页效果，而且还能提高网页的可访问性、可维护性，从而为网页设计者节省大量的时间和精力。

本书的特点在于，它是第一本将CSS同当前浏览器的支持情况结合起来进行描述的读物，而不仅仅是讲述CSS理论上应该如何工作。它向Web创作者和编辑人员提供了高效使用CSS的全面指南。整本书以CSS的属性为主线，理论加实践，深入浅出地讲解了CSS的强大功能及最新特性，并配合适当的代码和警告提示，对某些重点难点或容易被忽略的地方加以强调和说明。而且在最后的附录部分，还列出了各种资源信息以方便读者查阅。当然在进行本书的学习之前，最好是对HTML有所了解，这样会更容易上手。

原书的作者Eric A. Meyer是万维网联合会(W3C)CSS&FP工作组的特邀专家和成员之一，他不仅对CSS规范相当熟悉，而且还亲自编写代码进行测试，积累了大量的实践经验。同时他还广泛听取了其他一些资深CSS专家的建议，从而使得本书的内容更全面，更有针对性。本书不仅可以作为专业网页制作人员和文档编辑人员的参考用书，而且可以当作一本很好的网页制作初学者的教科书。

本书由邓劲生组织翻译，由许勇、齐宁等进行主要章节的翻译，由孙兆林统校全书，灯芯工作室的其他工作人员完成全书的录入、校对和审稿等工作。由于书中涉及的知识和技术范围较广，尽管我们为此付出了许多努力，但由于时间紧迫，加上译者的水平有限，翻译中的不妥和错误之处在所难免，殷切希望广大读者和同仁批评指教。

译者

2001年3月

目录

前言	1
第一章 HTML 与 CSS	7
网络的由盛到衰	7
CSS 是拯救之方	10
CSS 的局限	15
CSS 和 HTML	17
小结	26
第二章 选择符与结构	27
基本规则	28
分组	32
类选择符和 ID 选择符	38
伪类和伪元素	43
结构	50
继承	55
特殊性	57
层叠	61

元素分类	65
小结	69
第三章 单位和值	71
颜色	71
长度单位	83
百分比值	89
URL	89
CSS2 中的单位	91
小结	92
第四章 文本属性	93
使用文本	93
小结	131
第五章 字体属性	133
字体系列	134
字体加粗	141
字体尺寸	149
样式和变形	157
使用缩略方式：字体属性	161
字体匹配	164
小结	166
第六章 颜色和背景	167
颜色	169
复杂背景	183
小结	215

第七章 框与边框	217
基本元素框	218
是边界，还是补白？	221
边界	222
边框	238
补白	256
浮动与清除	261
列表	273
小结	280
第八章 可视化格式编排	282
基本框	282
块级元素	283
浮动元素	299
内联元素	310
小结	325
第九章 定位	326
基本概念	327
相对定位	348
绝对定位	352
固定定位	356
层叠定位元素	357
小结	363
第十章 CSS2 展望	364
与 CSS1 的区别	364
CSS2 选择符	366
字体与文本	380

产生内容	382
适应环境	386
边框	387
表格	388
媒体类型与 @ 规则	389
小结	391
第十一章 CSS 应用	394
转换方案	394
提示与技巧	414
附录一 CSS 资源	423
附录二 HTML 2.0 样式表	428
附录三 CSS1 属性	430
附录四 CSS 支持表	455
词汇表	473

前言

正如从封面可以猜到的那样,本书的主题是层叠样式表(Cascading Style Sheets, CSS)。CSS有两个“标准”,即CSS1和CSS2。它们之间的区别是:CSS2包含CSS1,而且还加入了很多其他的内容。本书涵盖了所有CSS1的内容,以及CSS2的一部分,即CSS定位(positioning)。而CSS2的其余部分则未包括在内,因为在写本书时,大部分CSS2的标准都还没有实现,所以与其包含太多的理论叙述,还不如将精力集中在当前实用的东西上。

对于网页设计者和文档作者来说,要想获得精美的页面风格、高的可访问性,并尽量节省时间和精力,那么本书是很适用的。在开始本书的学习之前,所有需要了解的相关知识就是HTML 4.0。当然,对HTML越熟悉,学习本书就会越得心应手。至于其他的相关知识,只需了解一点点就可以了。

请记住,对于网络标准和相关书籍而言,前者是在不停地发展变化的,然而后者则相对固定(至少在下一个版本还未出现之前是不变的)。就拿HTML和CSS来说,有很多变化正以文字的形式记录下来。例如,最近万维网联盟(W3C)推出的XHTML 1.0规范,就是万维网发展的一个重要的里程碑。对于CSS来说,可能还会出现更多的标准以扩展其样式化文档的能力;一些主要的网页浏览器也在向完全支持CSS1靠近,而且健壮的CSS2实现也初见端倪。所以,对于网页设计者来说,这是个令人兴奋的时刻,而且现在学习CSS也会为将来的工作打下一个很好的基础。

排版约定

本书使用下列排版约定：

等宽字体 (*Constant width*)

用于代码示例，HTML 标签和 CSS 元素。

等宽斜体 (*Constant width*)

用于文本中的可替换元素。

斜体 (*Italic*)

用于引入新的术语、文件名及路径名。

属性约定

在本书中，对于每个给定的 CSS 属性都用一个框将其取值或者相关的约定加以细化和说明。这些说明都已按照 CSS 规范逐字逐句重新编排过，但还是有必要对某些语法元素进行相应的解释。

每个属性所允许的取值都用下面的语法加以列举：

允许值：`[<长度> | thick | thin]{1,4}`

允许值：`[<系列名称> ,]* <系列名称>`

允许值：`<url>? <颜色> [/ <颜色>]?`

允许值：`<url> || <颜色>`

任何位于符号“<”和“>”中的文字都给定了一种类型的值，或者是对另一属性的引用。例如，`font` 属性可以接受任何属于 `font-family` 属性的值，这是通过取值“<字体系列>”来表示的。任何以固定宽度文本出现的文字都是关键字，它必须按照实际的形式出现，不能引用。正斜杠（/）和逗号（,）也不能引用。

几个关键字排在一起意味着它们都必须同时出现——而且按照给定的顺序。例如，“`help me`”意味着这个属性必须按这样的前后顺序使用这些关键字。

如果一条单竖线 ($X|Y$) 分隔了两个选项, 则它们中之一必须出现。双竖线 ($X||Y$) 意味着 X 或者 Y , 或者两者都出现, 但它们可以按任何顺序出现。中括号 [...] 用于分组。分组优先于双竖线, 而双竖线又优先于单竖线。这样, “ $VW|X||YZ$ ” 就等同于 “[VW]| [$X||YZ$]”。

每个文字或括号必须紧跟以下修饰符中的一个:

- 星号 (*) 表示前面的值或者分组被重复零次或多次。这样, $bucket^*$ 表示文字 “bucket” 可以使用任意次, 包括零次。对于被使用的次数上限没有明确的规定。
- 加号 (+) 表示前面的值或者分组被重复一次或多次。这样, mop^+ 就表示文字 “mop” 至少必须使用一次, 也隐含可以使用多次。
- 问号 (?) 表示前面的值或者分组是可选的。例如, $[pine\ tree]^?$ 表示文字 “pine tree” 可以不使用 (也可以使用, 但它们必须按正确的顺序出现)。
- 大括号中的一对数字 ($\{M,N\}$) 表示前面的值或分组重复至少 M 次, 最多 N 次。例如, $ha\{1,3\}$ 表示可以有一个, 两个或者三个文字 “ha” 出现。

下面是一些例子:

`give || me || liberty`

至少必须使用三者之一, 而它们可以按任何次序出现。例如, `give liberty`, `give me`, `liberty me` `give` 和 `give me liberty` 都是有效的。

`[I | am] ? the || walrus`

文字 `I` 或者 `am` 可以使用, 使用哪一个为可选的。而且, `the` 或者 `walrus`, 或者它们一起都可以跟在后面。这样, 可以组合成 `I the walrus`, `am walrus the`, `am the`, `I walrus`, `walrus the` 等等。

`koo+ka-choo`

一个或者多个 `koo` 的后面必须跟一个 `ka-choo`。当然, `koo koo ka-choo`, `koo koo koo ka-choo` 和 `koo ka-choo` 都是合法的。`koo` 数量可以是无限的, 但在实现时总是有限的。

```
I really{1,4}* [love | hate][ Microsoft | Netscape]
```

这是网页设计者发表意见的通用表达器。可以翻译成 I love Netscape, I really love Microsoft, 以及相似的一些句子。可以使用零到四个 really。但必须在 love 和 hate 中选择其一, 在本例中选择了 love。

```
[[ Alpha || Baker || Cray],,]{2,3} and Delphi
```

这是个很长很复杂的表达式。可能的结果是 Alpha, Cray, and Delphi。另一个可以是 Alpha Baker, Cray Alpha, Baker Cray Alpha, and Delphi。逗号因其位置的特殊而放在中括号内。

建议与评论

本书的内容都经过测试, 尽管我们做了最大的努力, 但错误和疏忽仍然是在所难免的。如果你发现有什么错误, 或者是对将来的版本有什么建议, 请通过下面的地址告诉我们:

美国:

O'Reilly & Associates, Inc.
101 Morris Street
Sebastopol, CA 95472

中国:

100080 北京市海淀区知春路 49 号希格玛公寓 B 座 809 室
奥莱理软件 (北京) 有限公司

询问技术问题或对本书的评论, 请发电子邮件到:

info@mail.oreilly.com.cn

可以访问本书的网站, 这里有书中的实例、勘误表和对将来版本的计划:

<http://www.oreilly.com/catalog/css/>

最后, 你可以在 WWW 上找到我们:

<http://www.oreilly.com>

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

致谢

写这本书是出于一个目标：就是使 CSS 易学、易懂、易于使用。当然，我们为此做了很多的工作试图将复杂的算法简要地翻译成易于理解的语言并不像听起来那么容易。

在编写本书的过程中，我发现自己好像是在 CSS 中不引人注意的角落里独自爬行，创建测试程序，苦思冥想所有的问题，而且还要兼顾网页浏览器的一些限制。在这期间，我遇到了很多热心的人，我应该感谢他们。

当然，首先是 Opera Software 的 Håkon wium Lie 和 W3C 的 Bert Bos，由于他们努力创建了 CSS，而且回答了我关于 WWW 样式邮件列表的很多问题，甚至是一些很愚蠢的问题。我同样要感谢 W3C 的 Chris Lilley，他鼓励我将网页变得更加样式化。他对我先前的努力所给予的赞扬，对于我来说真是一副强心剂，而且也正是他使我加入 W3C 的 CSS&FP 工作组成为可能。

Tim O'Reilly 给了我向专业出版物迈进的机会，为此我感激不尽。O'Reilly 的编辑，Richard Koman，在我写作的过程中非常耐心，或许我该送去更多的稿件以表示我对他的感激。Tara McGoldrick，同样在 O'Reilly 工作，他在图形整理、语言沟通方面给了我许多帮助，使我的生活变得更容易。Songline 工作室的 Dale Dougherty 给了我最初在专业写作方面的突破，而且 Chuck Toporek（现在在 O'Reilly 工作）是我的第一个长期在《Web Review》的编辑，他认真负责地为我安排所有文章的提交日程。我的朋友、同事 Peter Murray，非常大方地献出他的宝贵时间及精力帮助我创作其中的一个案例研究，以及 Ron Ryan，他是我所知道的最棒的管理人员，基于此他获得了一颗金星和很多好评。

在过去的几年，来自 Usenet 组 comp.infosystems.www.authoring.stylesheets 的一群热心人既回答了我许多的问题又提供了反馈信息，给了我巨大的帮助。在这个组里，Todd Fahrner，Alan Flavell，Anthony Boyd 和 Jan Roland Eriksson 为我提供了许多有用的线索。特别值得一提的是，Sue Sims 勤勤恳恳地帮助（有时是保

护)她从未遇见过的人——那就是我。Ian Hickson 和 David Baro 作为 CSS 方面的专家,奉献出了他们宝贵的精力以解释规范,寻找浏览器的 bug 和传播 CSS 带来的好处,同样要感谢他们。他们的评论及洞察力给本书的写作带来了极大的帮助。

当然,衷心的感谢要献给 CSS 的勇士们: Todd Fahrner, Liam Quin, David Baron, Ian Hickson, Sue Sims, Jan Roland Eriksson, John Alsopp 和 Braden McDaniel。是的,许多名字都有重复。毕竟,这只是一个小小的团体,但我还能有什么可说的呢?再一次感谢他们的大力支持。

请注意,他们中不应该有任何人会因本书中的错误而受到责备。相反,由于他们为我纠正了许多的错误而应受到我的赞许。

当然,为 Tim Berners-Lee 发明了 WWW 而感谢他,对我来说是合适的。我虽不认识他,但我觉得凡是和万维网相关的书籍的作者都有必要感谢 Tim 的巨大贡献。我认为这或许可以成为一项国际上的惯例;不妨就叫它 Berners Convention。

最后,我个人还要致谢: Michelle, 他是我患难与共的朋友; Randy 总是很乐意充当我的老师和密友; Steve 在大学及以后的几年里给了我很多的帮助; Dave 在这些年里给我带来了欢笑,特别是当我最需要它的时候,在我头脑发热时,又给了我许多让我清醒的言语;还有 Tina,在我想放弃的时候帮助我站了起来。

同时还要感谢我的妻子, Kathryn, 她给了我无尽的支持,特别在紧急关头,她对我和我的能力的信任和肯定;还有我的父母, Art 和 Carol, 姐姐, Julie, 他们总是一如既往地给我支持。

再次感谢他们。

— Eric A. Meyer
2000年2月23日

第一章

HTML 与 CSS

从很多方面来说，层叠样式表（Cascading Style Sheets，CSS）规范代表了万维网历史上一个独特的发展阶段。从最大限度地使结构文档样式化的根本能力上来说，CSS 算是前进了一步，也算是后退了一步——但这是很好的一步后退，而且是必需的一步。要想知道其原因，首先有必要理解网络为何需要 CSS 这样的东西，以及 CSS 是怎样使网站更适合于网页制作者和网上冲浪者的。

网络的由盛到衰

追溯到网络发展的早期（1990 - 1993），HTML 还是一种很贫乏的微型语言。它几乎全由结构化元素组成，这些元素对于描述段落、超链接、列表及标题非常有用。但像表格、框架或者是我们认为在创建网页时很有必要的一些复杂的标记却没有包含在内，甚至连边儿都未沾上。通常认为，HTML 是一种结构化标记语言，用于描述文档的不同部分。至于这些部分应该怎样去显示，HTML 却很少去关注。所以这种语言并不关心文档的外观，它仅仅是一种纯粹的小的标记系统。

然后 Mosaic 出现了。

突然，对于几乎所有上网超过十分钟的人来说，万维网的力量是显而易见的。从一个文档跳到另一个文档，只需让鼠标指向一个特殊颜色的文本或者是图像，然

后再轻轻点击鼠标即可。甚至，文本和图像可以在一起显示，而且要创建网页所需的所有工具只是一个纯文本编辑器。这就显得非常自由、非常开放，而且网页的外观也非常漂亮。

于是网站开始到处萌生。个人杂志网站、大学网站、公司网站及其他各种各样的网站纷至沓来。然而，随着网站数量的增加，对产生新的网页效果的HTML标签的需求也相应增加了。网页制作者也就开始有了使用粗体或者斜体文本的需求。

当时，HTML并不具备处理这类需求的能力。读者可能需要强调一小部分文本，但你无法保证它一定变为斜体——或许粗体，甚至是不同颜色的正常字体，这依赖于用户的浏览器及其设置。但是没有任何措施能保障制作者所创作的就是读者所看到的。

由于种种压力，像和<I>这样的标记元素开始引入到语言中。突然，一种结构化语言开始出现了。

乱作一团

几年后，我们同时也继承了这个过程所带来的内在缺陷。例如，HTML 3.2和HTML 4.0的大部分内容都专注于外观上的考虑。至于通过FONT元素来控制文本颜色和大小，对文档和表格应用背景颜色和图像，对表格单元内容的分隔和填充以及使用闪烁的文本字符，所有这些都成了最初要求“尽量控制！”所带来的产物。

想知道为什么这是一件坏事吗？不妨对某公司网站的网页中的标记进行观察。把那些大量的标记同有用的信息进行比较，可能会令人大吃一惊。更坏的是，对于大多数网站来说，表格和FONT标签几乎组成了整个标记网页，但它们没有表达任何真正的意义。而从结构的立场来看，这些网页仅比随机的字符串好那么一点点。

例如，假设有个页标题，制作者使用的是FONT标签而不是像H1这样的标题标签，如下：

```
<FONT SIZE="+3" FACE="Helvetica" COLOR="red">Page Title</FONT>
```


从结构的观点来说，FONT 标签毫无意义，这使得文档几乎无用。比如，FONT 标签会给支持语音的浏览器带来什么好处呢？如果制作者使用标题标签而非FONT 标签，那么语音浏览器还可以用某种语音格式来读取标题文本。但是对于 FONT 标签，浏览器却没有办法区分不同的文本。

制作者为什么要这样使用呢？因为他们想让读者看到的网页跟他们设计的一样。使用结构化的HTML 标记将会放弃对网页外观的许多控制，而且理所当然地不允许过去几年中曾流行的那种密密麻麻的网页设计。

那么问题出在哪里呢？考虑下面的因素：

- 非结构化的网页使内容索引变得很难。一个真正强大的搜索引擎应该允许用户按页面标题搜索，或者是页面内的子标题，或者仅仅是段落文本，或仅仅是那些标记为重要的段落。然而，为了做到这一点，网页的内容必须包含在某些结构化标记内——这正是大多数网页缺乏的那种标记。
- 结构的缺乏降低了可访问性。设想一下，盲人主要通过语音浏览器来浏览网页。那他将如何做选择呢：是允许浏览器阅读章节标题，进而对感兴趣的章节进行选择呢？还是那种缺乏结构的，强迫阅读所有内容，而且对于哪些是标题，哪些是段落，哪些东西是重要的都毫无指示的网页呢？
- 高级的页面表现力只可能通过某种文档结构来达到。设想某个页面只显示了章节标题，而每个章节旁都带有一个箭头，那么用户就能够决定哪些章节适合他，从而点击它，打开相应章节的文本。
- 结构化标记易于维护。设想为了在某个浏览器中找寻弄乱网页的某个小错误，会花掉多少时间去搜索人家的或自己的HTML 页？为了一个带白色超链的边框，会花多少时间去书写嵌套表格和 FONT 标签呢？为了在标题和其后的文本之间得到一个理想的间隔距离，又会插入多少的换行标签呢？通过使用结构化标记，就能使代码变得清晰，而很容易找到想要的东西。

诚然，全结构化的文档的确有些平淡。这都是基于这样一个事实：在 20 世纪末，结构化标记还不能动摇市场对时下流行的HTML 页面的钟爱，真正需要的是寻找一种可以将结构化标记同丰富的页面表现相结合的方式。