

— 字体抗锯齿技术

— 提高网页版式可读性的10项原则

— 设置网页字体的5项原则和方法

— 源自瑞士风格平面设计的经验教训

— 提升版式设计的8个简单技巧

— 版式设计样式探究

— 版式设计秘诀：选择和使用字体的原则

— 学习字体搭配的最佳途径方法

— CSS 字体堆栈设计指南：技巧和语法

— CSS3的版式设计指南

— 经典@Font-Face 规则回顾

— 现有的网页字体格式

— 回顾流行的网页字体嵌入服务

— 如何在网页中嵌入字体

— 版变通技巧

— 有用的字体排版工具

THE  
SMASHING  
BOOK

# 众妙之门 网页排版设计 制胜秘诀

[德] Smashing Magazine 著  
侯景艳 范辰 译



人民邮电出版社  
POSTS & TELECOM PRESS



# 众妙之门

## 网页排版设计 制胜秘诀

[德] Smashing Magazine 著

侯景艳 范辰 译

人民邮电出版社  
北京

## 图书在版编目 ( C I P ) 数据

众妙之门：网页排版设计制胜秘诀 / 德国  
Smashing杂志著；侯景艳，范辰译. — 北京：人民邮  
电出版社，2013.10  
ISBN 978-7-115-32829-8

I. ①众… II. ①德… ②侯… ③范… III. ①网页制  
作工具 IV. ①TP393.092

中国版本图书馆CIP数据核字(2013)第182546号

## 版权声明

*Getting the Hang of Web Typography*

Copyright © 2013 by Smashing Media GmbH

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, by photocopying, recording or otherwise, without the prior permission in writing from Smashing Media GmbH.

CHINESE SIMPLIFIED language edition published by POSTS &  
TELECOMMUNICATIONS PRESS Copyright ©2013.

本书中文简体版由德国 **Smashing Media** 公司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。

- 
- ◆ 著 [德] Smashing Magazine
  - 译 侯景艳 范辰
  - 责任编辑 赵轩
  - 责任印制 王玮
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号  
邮编 100061 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京捷迅佳彩印刷有限公司印刷
  - ◆ 开本：880×1230 1/32  
印张：6.5  
字数：204千字 2013年10月第1版  
印数：1-3500册 2013年10月北京第1次印刷  
著作权合同登记号 图字：01-2012-7261号
- 

定价：39.00元

读者服务热线：(010)67132692 印装质量热线：(010)67129223

反盗版热线：(010)67171154

广告经营许可证：京崇工商广字第0021号

# 内容提要

本书是 Smashing Magazine 网站的精选文章合辑，旨在帮助读者提升网页版式设计水平。本书囊括了字体抗锯齿技术，网页排版的原则，网页字体的设置技巧，CSS 字体堆栈指南，CSS3 的新特性，网页字体服务推荐以及排版工具介绍，等等。

还在为网页排版苦恼吗？想设计出别具一格无与伦比的网页吗？想了解基本的网页编程知识吗？想成为设计与开发的双面手吗？本书将是你的最佳选择。

# 目 录

第 1 章 字体抗锯齿技术 .....	1
1.1 技术 .....	3
1.2 输入 .....	6
1.3 输出 .....	12
1.4 总结 .....	17
第 2 章 提高网页版式可读性的 10 项原则 .....	19
2.1 专业术语及其内涵 .....	20
2.2 影响版式可读性的 10 项原则 .....	23
2.3 赋予版式风格的各种效果 .....	28
第 3 章 设置网页字体的 5 项原则和方法 .....	31
3.1 系统地进行网页版式设计 .....	32
3.2 利用信息层级 .....	34
3.3 为流畅地浏览而设计 .....	36
3.4 展现品牌视觉的同时保持网页的可读性 .....	37
3.5 将文字看作用户界面 .....	40
3.6 网页版式高级教程 .....	41
3.7 总结 .....	44
第 4 章 源自瑞士风格平面设计的经验教训 .....	45
4.1 如何将瑞士风格应用到网页设计上 .....	47
4.2 留白：让文字呼吸 .....	49
4.3 网格系统 .....	49
4.4 网格的内涵 .....	50
4.5 输入自定义列表 .....	53
4.6 关于元素 .....	53
4.7 关于衬线 .....	55
4.8 关于字号 .....	57

4.9 关于照片 .....	58
<b>第5章 提升版式设计的8个简单技巧 .....</b>	<b>61</b>
5.1 行宽 .....	62
5.2 行高 .....	63
5.3 引号的悬挂式排版 .....	64
5.4 垂直的韵律 .....	65
5.5 结句和起承句 .....	66
5.6 强调 .....	67
5.7 比例 .....	68
5.8 对齐字行 .....	69
<b>第6章 版式设计样式探究 .....</b>	<b>71</b>
6.1 常见的排版问题 .....	72
6.2 总结 .....	84
<b>第7章 版式设计秘诀：选择和使用字体的原则 .....</b>	<b>87</b>
7.1 适合才是最重要的 .....	88
7.2 为字体编组 .....	89
7.3 对比的使用原则：大胆而奔放 .....	95
7.4 少即是多 .....	97
7.5 没有规则即是规则 .....	102
7.6 总结 .....	102
<b>第8章 学习字体搭配的最佳实践方法 .....</b>	<b>103</b>
8.1 无衬线字体搭配衬线字体 .....	104
8.2 避免使用同类字体 .....	105
8.3 划分字体的角色 .....	106
8.4 对比字体粗细 .....	107
8.5 为版式创建丰富的色彩 .....	108
8.6 选择合适的字体进行搭配 .....	108
8.7 独特与中性的对比 .....	110

8.8	避免将差异太大的字体进行搭配 .....	111
8.9	字体搭配的“简”原则——字族两种就够 .....	112
8.10	使用不同的字号 .....	113
8.11	总结 .....	114
<b>第 9 章</b>	<b>CSS 字体堆栈设计指南：技巧和资源 .....</b>	<b>115</b>
9.1	创建自己的字体堆栈 .....	116
9.2	常见字体堆栈 .....	117
9.3	工具 .....	120
<b>第 10 章</b>	<b>CSS3 的版式设计潜能 .....</b>	<b>123</b>
10.1	关于基础知识 .....	124
10.2	字体属性 .....	124
10.3	其他字体属性 .....	125
10.4	关于留白、换行符和文本绕排 .....	125
10.5	单词间距和字母间距 .....	127
10.6	缩进和标点悬挂 .....	128
10.7	网页字体和字体装饰 .....	129
10.8	新文本装饰属性 .....	132
10.9	溢出控制 .....	133
10.10	对齐和断字 / 字符连接 .....	134
10.11	统一码范围和语言 .....	135
10.12	总结 .....	136
<b>第 11 章</b>	<b>经典 @Font-Face 规则回顾 .....</b>	<b>137</b>
11.1	@font-face——源自 20 世纪 90 年代 .....	138
11.2	多亏次像素渲染，字体嵌入再次回归 .....	139
11.3	字体制造商的关注点 .....	140
<b>第 12 章</b>	<b>现有的网页字体格式 .....</b>	<b>143</b>
12.1	True Type .....	144
12.2	OpenType .....	144

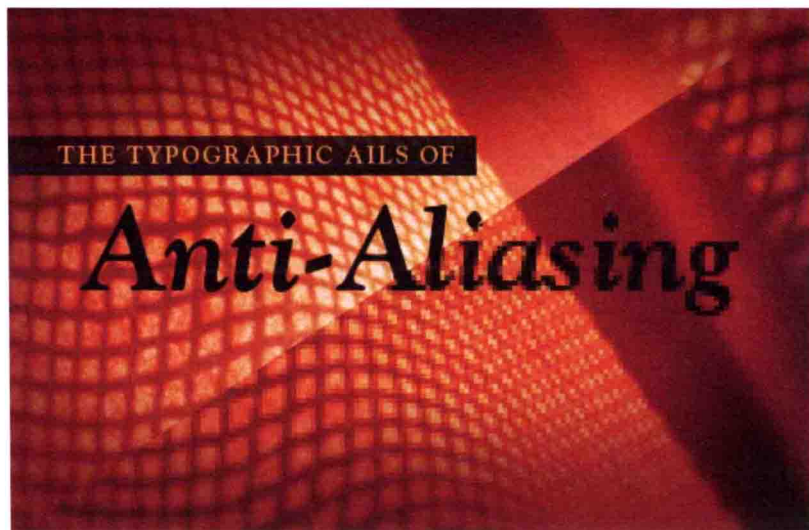
12.3	EOT	145
12.4	WOFF	146
12.5	SVG	147
<b>第 13 章</b>	<b>回顾流行的网页字体嵌入服务</b>	<b>149</b>
13.1	十一种流行的网页字体服务对比	150
13.2	Font-Face	168
13.3	如何选择服务	169
13.4	总结	170
<b>第 14 章</b>	<b>如何在网页中嵌入字体</b>	<b>171</b>
14.1	字体的取舍	172
14.2	嵌入免费和开源的字体	172
14.3	使用 @font-face Kit Generator	173
<b>第 15 章</b>	<b>网页排版变通技巧</b>	<b>179</b>
15.1	文字排版	181
15.2	OpenType 排版功能	181
<b>第 16 章</b>	<b>10 种有用的字体排版工具</b>	<b>183</b>
16.1	TypeChart——预览和对比网页字体	184
16.2	Web Font Specimen——展示字体的网页显示效果	184
16.3	Typografix——修复字体	185
16.4	TYPEIT——轻松营造独特版式	186
16.5	Font Finder——轻松检测网站元素中的字体	187
16.6	TypeSet——文字自动换行算法	188
16.7	Perfect Typography——网上首个行高计算器	189
16.8	EMCHART——眼找到正确的比例	190
16.9	Em Calculator——使 CSS 设计更具扩展性, 更易实现	191
<b>第 17 章</b>	<b>术语表</b>	<b>193</b>



字体抗锯齿技术 第 1 章

Tom Giannattasio

得益于高分辨率带来的优势，印刷字体可以傲人地展现出漂亮的曲线和细节。然而一旦到了屏幕上，却因为精度不够而出现锯齿边缘，形变并丢失细节。在 200 或 300PPI（分辨率）的显示器大规模普及之前，我们将不得不依赖软件技术来解决这些问题。



首先让我们来看抗锯齿技术。除了更高分辨率的显示器，这是最令人振奋的消息了。它的原理相当简单：沿着字体的边缘增加半透明的像素点，从而达到平滑阶梯状锯齿的效果。

当然，在这个过程中，有很多因素会决定最后的呈现效果，如字体微调、次像素渲染、软件性能、操作系统规范等，此处不一一列举。

现在，让我们看看作为设计师的你，如何用 Photoshop、Flash 和 CSS 来改善抗锯齿技术的效果。另外，我们也会解释硬件、浏览器和操作系统所带来的限制。

## 1.1 技术

### 1.1.1 锯齿 ( aliased ) Vs. 抗锯齿 ( anti-aliased )

只需稍加注意就能发现，抗锯齿技术对于文字的清晰度有重要的影响。除了极少的例外，经过抗锯齿处理的文字能显著减轻用户的眼部负担，而且渲染后的字形更接近预期设计。

因此，设计师如果想得到好的设计效果，就应该有意识地使用抗锯齿技术。此外，还需考虑从设计到输出整个过程中的诸多因素。



42pt “Goudy Oldstyle Bold 字体” 的锯齿和抗锯齿版本。

### 1.1.2 字体微调 ( font hinting )

大多数字体渲染引擎都是依靠微调来确定哪些字形区域应该平滑处理。字体微调使用数学指令表格将字形与像素网格对齐，并确定哪些像素应该采用哪个灰度。尽管多数软件都提供了标准算法的自动微调，但是理想的做法还是由字体设计师手工调整，并嵌入文件中。

简单来说，这些指令是通过修改一些结构上重要节点的位置来实现功能的。例如，当指令找到那些在字形曲线或在字体主干 ( stem ) 上的点，就令它们在像素边界上对齐。次要的点则基于重要节点的位置来重新定位。而诸如 FontFore 这样的开源字体编辑工具，则允许你查看并编辑字体的微调信息。在了解了清晰字形的形成需要投入那么多的工作之后，你一定对字体设计师和开发人员增加了几分感激。

经过微调

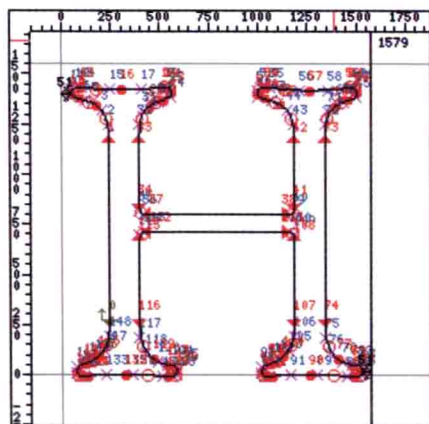
不一致

未经微调

模糊

零散的像素

经过微调和未经微调的字体各有其利弊，设计师要根据字体的易读性和完整性来做出选择。



0	40	NPUSHB
1	42	66
2	00	0
3	73	115
4	6c	108
5	6b	107
6	04	4
7	6f	111
8	65	101
9	79	121
10	4f	79
11	51	81
12	4a	74
13	22	34
14	29	41
15	25	37
16	1b	27
17	42	66

用 FontFore 字体编辑器查看对 Goudy Oldstyle 字体的“H”所做的微调。

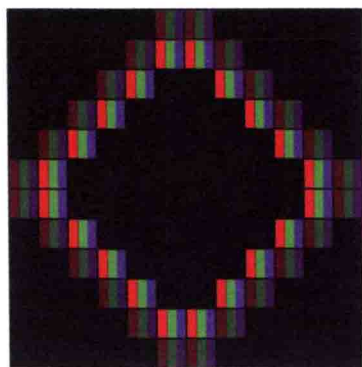
### 1.1.3 次像素渲染 ( subpixel rendering )

每一个像素在标准显示器上都由三部分组成：一个红次像素点、一个绿次像素点和一个蓝次像素点。每一个次像素点的亮度都可以单独控制。因为它们的尺寸非常小，所以我们的肉眼会将这三个次像素点视为一个纯色像素点。

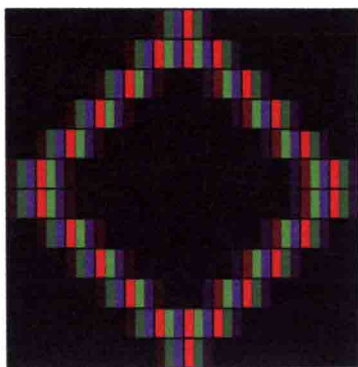
标准的抗锯齿技术是精确到为每一个次像素点设置值，从而获得全灰度像素图 ( full grayscale pixel )。次像素渲染利用每个单色组成的独立性，来使感知

到的显示器分辨率进一步增加。

这使所有的像素点都可根据其临近的像素点来平衡视觉权重，从而使字体能以更小的增量被平滑。用这种方式渲染的字体可在字形边缘看到微弱的色彩变化。



全像素渲染



次像素渲染

次像素渲染需要依靠完美的像素网格对齐，这使得液晶平板显示器（LCD）成了这项技术唯一可选的显示器。显像管显示器（CRT）因为不够精准和像素分布不规则，致使次像素渲染技术难以实现。

值得一提的是，不同液晶显示器的次像素排列也不相同：有些显示器是按照 RGB 的顺序来排列，而有些则是按 BGR 的顺序来排列。

次像素渲染通过对每个单色组成进行单独设置，可使感知到的分辨率提升 3 倍。

标准渲染

Render

次像素渲染

Render

次像素渲染比标准渲染的效果更令人满意，但会产生彩色边缘。

## 1.2 输入

作为设计师和开发人员，我们很难控制终端用户最终以何种方式来查看字体，但是通过使用适当的传输方法，还是可以确保表现效果最佳。现在最常用的三种将文字输出到用户的方式为：HTML、图片和 sIFR。只要应用得当，每一种方式都能获得理想的效果，大幅提高可读性，从而提升整体的用户体验。

### 1.2.1 HTML 文本

HTML 文本无疑是互联网上最主要的文本形式。之前设计师们完全不能用客户端技术来实现抗锯齿技术，现在可以了。CSS3 引入了两种新方法来控制 HTML 文本的传送：`font-smooth` 和 `@font-face`。

### 1.2.2 font-smooth

`Font-smooth` 允许你控制使用平滑的时间，但无法控制使用平滑的方式；这种抗锯齿方法仍然受控于用户的使用环境并且尚未得到广泛支持。但它有一项功能是允许关闭抗锯齿，这在我们遇到小号字体时很有用，因为小号字体经过抗锯齿处理经常会变得模糊不清。当碰到更复杂的、非浏览器安全字体时，结合新的 `@font-face` 规则可使效果加倍。

### 1.2.3 @font-face

`@font-face` 规则是一个令人兴奋的 CSS3 新特性，设计师们对它期待已久。尽管我们随时可往字体堆栈中添加高级字体，但绝大多数的客户端设备并没有高端字体，最终只能显示为标准网页安全字体（例如 Times New Roman 字体替换 Adobe Garamond Pro 字体）。

通过允许浏览器从 URL 导入字体，我们可以将任何字体呈现给用户，而不再局限于用户的字库。这意味着我们不但能表现更加独特的字体，而且字体效果比经过微调的更好、更易辨识。

尽管这一特性有望创建一个更完美的线上字体环境，我们仍会发现设计师选择了像 Verdana 这样的字体，其专为屏幕显示做过特殊设计和微调。一些我

们钟爱的印刷字体放到屏幕上一经渲染看起来却很糟糕，尤其是小号字体。

对 `@font-face` 特性的支持，以及通过 `typekit` 等工具实现的浏览器字体嵌入，无疑将使我们看到新的行业前景。

#### SAFARI 4:

The image shows the text "Bickham Script Pro" in a highly decorative, cursive script font. The letters are black and have a classic, elegant feel with many flourishes and connections between letters.

#### FIREFOX 3:

The image shows the text "Bickham Script Pro" in the same decorative, cursive script font as above. The rendering appears slightly different, possibly due to the browser's font handling, but it maintains the same overall style and readability.

Bickham Script Pro 字体通过 `@font-face` 嵌入 Safari 4 和 Firefox 3 浏览器。注意 OpenType 字体的斜体（swash）和连字（ligature）在 Firefox 3 浏览器上得到支持！

### 1.2.4 以图片呈现的文字

以图片形式来呈现文字的情况可能很少见，但它允许你在必要时调整每一个字母。Photoshop 提供了 5 种抗锯齿预设，这些设置通过不同算法可确定连接到文件像素网格的像素值。遗憾的是没有一种预设允许像素渲染，但你可通过自由变形工具（Free Transform）来微调图层位置，从而有效地加强算法使渲染更清晰。

每种设置都允许不同数量的原点，有些仅在沿着  $x$  轴方向转化时才会发生变化。下面的表格列举了可转化的范围。

	x 轴	y 轴
NONE (无)	1	1
SHARP (锐利)	2	1
CRISP (犀利)	4	1
STRONG (浑厚)	32	16
SMOOTH (平滑)	4	4
SHAPE LAYER (图形图层)	32	32

### 1.2.5 无 (None)

原始轮廓

*Anti-Aliasing*

Photoshop 的“无”设置选项

*Anti-Aliasing*

用“无”设置选项生成的锯齿字体使用非常有限，通常 9 ~ 18 号的字体显示效果最好。更小的字体将无法识别，而更大的字体又会被加粗并出现明显的锯齿边缘。有时两个大小不同的字体以相同高度被渲染，会造成字母间距 (letter spacing)、字母宽度和 x 字高 (x-height) 的改变。

例如，Arial 的 14pt 字体是按照 10 像素高和 8 像素 x 字高来渲染，而同种 13pt 字体也是 10 像素高，但 x 字高却只有 7 像素，虽差异细微但感觉很不同。当字体紧排的时候，这项设置可能还需要手工调整，否则一些字母会挤成一团。

13pt Arial

14pt Arial

13pt Arial 14pt Arial



13pt 和 14pt 的 Arial 字体以相同的字母高度( cap height )但不同的 x 字高来渲染。

### 1.2.6 锐利 ( Sharp )

原始轮廓

Photoshop 的“锐利”设置选项

*Anti-Aliasing*

*Anti-Aliasing*

“锐利”设置选项使用了非常紧密的网格对齐来达到锐化字体的效果。如果处理得当的话,这种方法和“无”设置选项的绘制方法非常相似,但增添了一些平滑效果。事实上,如果将像素点重叠,也能看到大多数纯色像素点从“无”变成了“锐利”效果。

虽然大写字母高度和 x 字高一般保持不变,但字符粗细和宽度会改变。

注意,“锐利”选项在渲染过程中往往趋向于完全去掉细节,所以有时会造成字体变形,如果字体的完整性对你来说很重要的话,就要考虑采用别的设置选项。

### 1.2.7 犀利 ( Crisp )

原始轮廓

Photoshop 的“犀利”设置选项

*Anti-Aliasing*

*Anti-Aliasing*

“犀利”设置选项能很好地保持字体原有的粗细和曲线,而清除由细衬线或细描边产生的多余像素,这对大号字体特别有用。

不过用“犀利”选项的话,你就不能沿 y 轴做图层微调了。

### 1.2.8 浑厚 ( Strong )

原始轮廓

Photoshop 的“浑厚”设置选项

*Anti-Aliasing*

*Anti-Aliasing*

“浑厚”设置选项会给字体笔画增加不必要的粗细,但它却为原点变化提供了最大的调整自由度——32 格的 x 轴和 16 格的 y 轴变化范围。