

关于作者

译者序

前言

第1章 HTML5、CSS3和响应式网页设计入门

- [1.1 为什么智能手机很重要（而老版的 IE 不再重要）](#)
- [1.2 响应式设计一定是最佳选择吗](#)
- [1.3 响应式网页设计的定义](#)
- [1.4 为什么要在响应式设计上停滞不前](#)

1.5 响应式网页设计示例

- [1.5.1 下载视口调试工具](#)
- [1.5.2 在线创意源泉](#)

1.6 为什么 HTML5很优秀

- [1.6.1 省时省力](#)
- [1.6.2 新增了语义化标签元素](#)

1.7 CSS3为响应式设计和更多创新奠定了基础

- [1.7.1 底线：CSS3不破坏任何东西](#)
- [1.7.2 CSS3如何解决日常设计问题](#)

1.8 看呐，不用图片

- [CSS3还带来了什么](#)
- [1.9 HTML5和 CSS3现在就能用吗](#)
- [1.10 响应式网页设计不是灵丹妙药](#)
- [1.11 引导客户：网站不必在所有浏览器中表现一致](#)

- [1.12 小结](#)

第2章 媒体查询：支持不同的视口

- 2.1 现在就能使用媒体查询
- 2.2 为什么响应式设计需要媒体查询
 - 2.2.1 媒体查询语法
 - 2.2.2 媒体查询能检测那些特性
 - 2.2.3 用媒体查询改造我们的设计
 - 2.2.4 加载媒体查询的最佳方法
- 2.3 我们的第一个响应式设计
 - 2.3.1 我们的设计是固定宽度的，不要惊讶
 - 2.3.2 响应式设计中要保证图片尽可能精简
 - 2.3.3 小视口下的内容剪切
- 2.4 阻止移动浏览器自动调整页面大小
- 2.5 针对不同视口宽度修正设计
- 2.6 响应式设计中内容始终优先
- 2.7 媒体查询只是必要条件之一
 - 我们需要流动布局
- 2.8 小结

第3章 拥抱流式布局

- 3.1 固定布局经不起未来考验
- 3.2 为什么响应式设计需要百分比布局
- 3.3 将网页从固定布局修改为百分比布局

- 3.3.1 需要牢记的公式
- 3.3.2 设置百分比元素的上下文
- 3.3.3 必须时刻牢记上下文
- 3.4 用 `em` 替换 `px`
- 3.5 弹性图片
 - 3.5.1 让图片随视口缩放
 - 3.5.2 为特定图片指定特定规则
 - 3.5.3 给弹性图片设置阈值
 - 3.5.4 超级全能的 `max-width` 属性
- 3.6 为不同的屏幕尺寸提供不同的图片
 - 设置自适应图片
 - …把背景图片放在其他地方
- 3.7 流动网格布局和媒体查询的默契配合
- 3.8 CSS 网格系统
 - 使用网格系统快速搭建网站
- 3.9 小结

第4章 响应式设计中的 HTML5

- 4.1 HTML5的哪些部分现在就能用
 - 4.1.1 大多数网站可以用 HTML5编写
 - 4.1.2 腻子脚本和 Modernizr
- 4.2 如何编写 HTML5网页
 - 4.2.1 HTML5的精简之道
 - 4.2.2 HTML5标签的合理写法

- 4.2.3 伟大的<a>标签万岁
- 4.2.4 HTML 的废弃零件
- 4.3 HTML5的全新语义化元素
 - 4.3.1 <section>
 - 4.3.2 <nav>
 - 4.3.3 <article>
 - 4.3.4 <aside>
 - 4.3.5 <hgroup>
 - ……HTML5的大纲结构算法
 - 4.3.6 <header>
 - 4.3.7 <footer>
 - 4.3.8 <address>
- 4.4 HTML5结构元素的实际用法
 - 网站的主体内容怎么办？
- 4.5 HTML5的文本级语义元素
 - 4.5.1
 - 4.5.2
 - 4.5.3 <i>
 - 4.5.4 在页面中应用文本层语义元素
- 4.6 遵循 WAI-ARIA 实现无障碍站点
 - ARIA 的地标角色
- 4.7 在 HTML5中嵌入媒体
- 4.8 用 HTML5的方法为页面添加视频或音频

- 4.8.1 提供备用的媒体源文件
- 4.8.2 针对老版本浏览器的备用方案
- 4.8.3 `<audio>`和`<video>`标签的用法基本一致
- 4.9 响应式视频
- 4.10 离线 Web 应用
 - 4.10.1 离线 Web 应用概述
 - 4.10.2 让网页可离线使用
 - 4.10.3 理解 manifest 文件
 - 4.10.4 页面被自动加载到离线缓存
 - 4.10.5 版本注释的用途
 - 4.10.6 离线访问网站
 - 4.10.7 离线 Web 应用的故障诊断
- 4.11 小结

第5章 CSS3：选择器、字体和颜色模式

- 5.1 CSS3给前端开发人员带来了什么
 - 5.1.1 Internet Explorer 6到8对 CSS3的支持
 - 5.1.2 使用 CSS3设计和开发页面
- 5.2 CSS 规则解析
- 5.3 私有前缀及其用法
- 5.4 快速而有效的 CSS 技巧
 - 5.4.1 CSS3多栏布局
 - ……增加栏位间隙和分割线

- 5.4.2 文字换行
- 5.5 CSS3的新增选择器及其用法
 - 5.5.1 CSS3属性选择器
 - 1. CSS3的子字符串匹配属性选择器
 - 2. 一个活生生的例子
 - 5.5.2 CSS3结构伪类
 - 1. :last-child 选择器
 - 2. nth-child 选择器
 - 3. 理解 nth 规则的作用
 - 4. 否定 (:not) 选择器
 - 5.5.3 对伪元素的修正
 - :first-line 对响应式设计来说好用吗
- 5.6 自定义网页字体
 - 5.6.1 @font-face 规则
 - 5.6.2 使用@font-face 嵌入网页字体
- 5.7 帮帮我，标题模糊怎么办
 - 在响应式设计中自定义@font-face 字体的注意事项
- 5.8 新的 CSS3颜色格式和透明度
 - 5.8.1 RGB 颜色
 - 5.8.2 HSL 颜色
 - 5.8.3 针对 IE6、IE7和 IE8提供备用颜色值
 - 5.8.4 透明通道

- 5.9 小结

第6章 用 CSS3创造令人惊艳的美

- 6.1 文字阴影
 - 6.1.1 HEX、HSL 或 RGB 颜色都可以
 - 6.1.2 px、em 或 rem 都行
 - 6.1.3 取消文字阴影
 - ... 左上方阴影
 - 6.1.4 制作浮雕文字阴影效果
 - 6.1.5 多重文字阴影
- 6.2 盒阴影
 - 6.2.1 内阴影
 - 6.2.2 多重阴影
- 6.3 背景渐变
 - 6.3.1 线性背景渐变
 - ... 分解线性渐变语法
 - 6.3.2 径向背景渐变
 - ... 分解径向渐变语法
 - 6.3.3 重复渐变
- 6.4 背景渐变图案
- 6.5 CSS3的响应性
- 6.6 组合使用 CSS3属性
- 6.7 多重背景图片
 - 6.7.1 背景图片大小

- 6.7.2 背景图片位置
- 6.7.3 背景属性的缩写语法
- 6.8 更多 CSS 特性
- 6.9 可缩放图标：响应式设计中的完美选择
- 6.10 小结

第7章 CSS3过渡、变形和动画

- 7.1 什么是 CSS3过渡以及如何使用它
 - 7.1.1 过渡相关的属性
 - 1. 过渡的简写语法
 - 2. 在不同时间段内过渡不同属性
 - 3. 理解过渡调速函数
 - 7.1.2 响应式网站中的有趣过渡
- 7.2 CSS3的2D 变形
 - 我们能做哪些变形
 - 1. scale
 - 2. translate
 - 3. rotate
 - 4. skew
 - 5. matrix
 - 6. transform-origin 属性
- 7.3 尝试 CSS3的3D 变形
 - 7.3.1 分析3D 变形效果

- 7.3.2 3D 变形尚未成熟
- 7.4 CSS3动画效果
 - 组合使用 CSS3变形和动画
- 7.5 小结

第8章 用 HTML5和 CSS3征服表单

- 8.1 HTML5表单
 - 8.1.1 理解 HTML5表单中的元素
 - 8.1.2 placeholder
 - 8.1.3 required
 - 8.1.4 autofocus
 - 8.1.5 autocomplete
 - 8.1.6 list（及对应的 `datalist` 元素）
 - 8.1.7 HTML5的新输入类型
 - 1. email
 - 2. number
 - 3. url
 - 4. tel
 - 5. search
 - 6. pattern
 - 7. color
 - 8.1.8 日期和时间输入类型
 - 1. date

- 2. month
 - 3. week
 - 4. time
 - 5. datetime 和 datetime-local
 - 6. range
- 8.2 如何给不支持新特性的浏览器打补丁
 - 8.3 使用 CSS3美化 HTML5表单
 - 针对表单的 CSS3伪类选择器
 - 8.4 小结

第9章 解决跨浏览器问题

- 9.1 渐进增强与优雅降级
 - 现状
- 9.2 该不该修复老版本 IE
 - 9.2.1 统计数据（再看看世界的变化）
 - 9.2.2 个人选择
- 9.3 前端的瑞士军刀：Modernizr
 - 9.3.1 使用 Modernizr 辅助修正样式问题
 - 9.3.2 使用 Modernizr 让老版本 IE 支持 HTML5元素
 - 9.3.3 给 IE6、7、8追加 min/max 媒体查询功能
 - 9.3.4 使用 Modernizr 按需加载资源
- 9.4 必要时将导航链接转换为下拉菜单
- 9.5 高分辨率设备（未来趋势）

- 9.6 小结

附录

- [附录1: 响应式 Web 设计 \(HTML5和 CSS3\) 工具集](#)
- [附录2: 响应式 Web 设计 \(HTML5和 CSS3\) 范例网站](#)
- [附录3: 本书涉及的 HTML5及 CSS3标准索引](#)
- [附录4: 本书提到的电影索引](#)

关于作者

Ben Frain 是一名具有十多年经验的网页设计师和前端工程师，直接与世界各地的客户和设计机构并肩工作。同时他还是一名技术记者，定期为一些关注 Mac 平台、前沿科技、网页设计和航空技术的刊物撰稿。

在此之前，他曾是一名怀才不遇的（而且谦虚的）电视演员，毕业于索尔福德大学的媒体与表演专业。他写了四部（个人认为）同样被低估的剧本，而且始终心怀能卖出一部的信念（尽管不像最初那么强烈了）。工作之余，在身体（和妻子）允许的情况下，他喜欢玩室内足球。

他的个人网站是 www.benfrain.com，Twitter 地址是 twitter.com/benfrain。

首先要感谢网络社区。没有大家集思广益的才智、慷慨大方的文档和开放共享的解决方案，我不可能在互联网行业做出现在这点还稍稍令我自豪的事情。

其次，我要感谢响应式网页设计之父 Ethan Marcotte。他和我从未谋面或交谈过，但他的方法论现在每时每刻都在影响着我制作网页的方法。毫无疑问，假如有哪些地方造成读者对响应式方法论的误解，那都要怪我自己表达不好。

最后，感谢我的家人。看过（同样被低估的）《义海倾情》¹的人都知道：“血浓于水。其他人都是陌生人。”

1 《义海倾情》 Wyatt Earp: <http://movie.douban.com/subject/1293055/>) 。——译者注

译者序

蒂姆·伯纳斯-李在1991年制作并发布了第一个网站，如今刚刚过去21个年头。在这21年里，计算机和互联网快速发展，这个世界的面貌也日新月异。在这个过程中，网页设计从无到有，从简单渐至专业，从可有可无变为广受关注。网页设计方法也在跟随时代不断创新，从最初简单的文字排版，到表格布局，再到 **DIV+CSS**，直到现在广为流行的网格布局、流式布局等，设计师和开发者们一直致力于为全球网民提供更好的设计观感和使用体验。

iOS 和 Android 的发布，掀起了移动互联网的浪潮，智能手机、平板电脑、智能家电等新设备层出不穷，我们的世界变得更加精彩纷呈。但这却给网页设计带来了新的挑战，在面对形形色色的终端设备、千差万别的屏幕分辨率，以及良莠不齐的网络连接质量时，目前的设计方法显得力不从心。我们无法预料用户的设备和网络状况，更不可能为每种设备专门设计一套网站，那么在移动互联网时代，如何创新为用户提供更好的设计和体验呢？

2010年5月25日，伊桑·马科特发表在 A List Apart 上的一篇文章，为我们打开了思路。在这

篇名为《Responsive Web Design》的文章中，伊桑·马科特将三种已有的技术整合在一起，提出了响应式网页设计的概念，用以解决我们当前遇到的设计难题。响应式网页设计概念一经提出就大受欢迎（当然也有争议），很多设计师和开发者纷纷实践并完善这种理念。两年多的时间里，涌现出了越来越多的响应式网站，针对响应式设计的工具和资源也日渐丰富。截至目前，响应式设计是使用一套代码为各类设备提供良好设计效果和使用体验的最佳设计方法。

你肯定对响应式设计有所耳闻，也可能看过一些响应式设计的技巧或相关文章，但它们都零零散散不成体系，无法让你真正理解并掌握响应式设计，也无法指导你立即开始响应式设计的实践。这本书将会带你系统地学习响应式网页设计的方法论，书中涵盖了响应式设计的思想、方法、工具、技巧、HTML5、CSS3、相关资源，以及针对老版本浏览器的兼容方案等内容，并以实际案例循序渐进地讲解了如何创建一个优雅高效易于维护的响应式网站。希望这些正是你所需要的。

“得功有法，秉一应万”，这是太极拳修炼的目标，我想也可以作为响应式设计的目标。修炼好响应式设计这门功夫，能够让你安然自在，以一应万。元芳，潜心修炼吧！

感谢裕波、杨海玲老师及朱巍老师，让我有机会翻译本书。非常感谢图灵各位编辑的辛勤工作，尤其感谢李松峰老师的细心指导。另外要感谢图灵社区，我在这里受益匪浅。最后，感谢我的老婆，本书的翻译离不开她的支持、监督与参谋。

我在翻译本书时尽力保证信与达，雅则不敢奢望。翻译中的错误在所难免，欢迎广大读者指

正。如果对本书有任何意见、建议或想法，请发送邮件至 wyqbailey@gmail.com 或反馈至图灵社区。

王永强 2012年10月 成都

前言

如果你想给自己的网站做一个单独的“手机版”，请三思而后行！响应式网页设计提供了一种设计方法，可以使同一网站在智能手机、桌面电脑，以及介于这两者之间的任意设备上完美显示。这种方法能够根据用户的屏幕尺寸，合理地为用户及将来的各种设备提供最佳的浏览体验。

本书提供了一整套方法，用来将一个现有的固定宽度的网站设计变成响应式的。此外，本书应用 **HTML5** 和 **CSS3** 提供的最新最有用的技术，扩展了响应式网页设计的方法论，以便网站更简洁、更易于维护。本书还讲解了编写和发布代码、图片、文件的最佳实践。

只要你懂 **HTML** 和 **CSS**，你就能制作响应式网站。

本书内容

第1章，**HTML5**、**CSS3**和响应式网页设计入门，定义了什么是响应式网页设计，展示了一些响应式设计的网站示例，重点强调了使用 **HTML5** 和 **CSS3** 的优势。

第2章，媒体查询：支持不同的视口，讲解了什么是媒体查询，如何实现媒体查询，以及如何针对设备能力匹配 **CSS** 样式，将其应用于任意设计。

第3章，拥抱流式布局，讲解了流式布局的优点，以及如何将一个现有的固定宽度设计轻松地转换为流式布局，怎样使用 **CSS** 框架快速搭建响应式网页。

第4章，响应式设计中的 **HTML5**，探讨了使用 **HTML5** 技术的诸多好处，比如更简洁的代码、语义化标签、离线存储，以及无障碍网页应用辅助技术。

第5章，CSS3：选择器、字体和颜色模式，展示了 CSS3选择器的强大威力，可以让你轻松地指定和改变任何元素。还讲解了通过 `@font-face` 声明来使用漂亮的网络字体，另外讲解了新的 CSS3颜色模式如 RGB(A)和 HSL(A)。

第6章，用 CSS3创造惊艳的美，展示了如何使用纯粹的 CSS3代码实现文字阴影、盒阴影和渐变效果。还涵盖了如何使用多重背景图片，以及如何通过字体文件创建图标。

第7章，CSS3的过渡、变形和动画，讲解如何仅使用 CSS3来创建和转换屏幕上的元素，并制作动画效果。

第8章，用 HTML5和 CSS3征服表单，阐述了在所有设备上（从最新的智能手机到桌面版浏览器）都能良好运行的跨浏览器表单开发技巧。

第9章，解决跨浏览器问题，讲解了如何保证老版本的 Internet Explorer 可响应，如何将一组链接修改成移动设备上的一个菜单，如何为高分辨率显示器提供不同内容，以及如何使用 Modernizr 框架分条件地加载资源文件。

准备工作

你必须对 HTML 和 CSS 很熟悉。有一点 JavaScript 基础会很有帮助。良好的电影品味也很益处。

本书读者

你是否正在开发两套网站，一套给移动设备，一套给大显示器？又或者你已听说过“响应式网页设计”但却不确定如何将 HTML5、CSS3和响应式设计融合在一起？如果是，那么本书可以让你在所有竞争对手之前，将你的网站提升到一个更高层次。

本书面向那些正在使用 HTML 4.01和 CSS 2.1开发固定宽度网站的网页设计师和开发人员，讲解了如何使用 HTML5和 CSS3制作可适应任意屏幕尺寸设备的响应式网站。

本书约定

本书中使用几种不同的文字样式来区分不同类型的信息，具体约定如下。

正文中的代码使用如下格式：

“HTML5接受宽松语法，例如 `<sCRipt SrC=js/jquery-1.6.2.js></script>` 这样的语句也是有效的。”

代码段使用如下格式：

```
<div class="header">    <div class="navigation">        <ul class="nav-list">            <li><a href="#"
title="Home">Home</a></li>            <li><a href="#" title="About">About</a></li>        </ul>    </div>

<!--end of navigation -->
```

```
</div> <!--end of header -->
```

当要专门强调代码块中的某一部分时，则对相关行或条目使用粗体格式：[编者注：网页格式无法实现此效果。]

```
#wrapper {    margin-right: auto;    margin-left: auto;    width: 96%; /* 最外围的 DIV */

}
```

```
#header { margin-right: 10px; margin-left: 10px; width: 97.9166667%; /* 940 ÷ 960 */
```

新术语，以及**重要词汇**也使用粗体。文中提到的那些菜单、对话框中的文字，会使用粗体或大写来标注，如：“导航按钮不再做背景色切换，内容区的 **THESE SHOULD HAVE WON** 按钮和侧边栏的**详细信息**按钮消失了，而字体也与设计文档相差甚远。”

！这个图标表示警告或重要提醒。

¥这个图标表示提示或技巧。

读者反馈

我们时刻欢迎你的反馈，以便了解你对本书的看法。你的宝贵意见有助于我们提升书籍的质量。

一般的阅读反馈，可直接发送电子邮件至 feedback@packtpub.com，请在邮件标题中注明书名。

如果你在某个领域内有专长且有兴趣编写相关书籍，请访问 www.packtpub.com/authors 查看作者指南。

客户支持

现在你已是 **Packt** 图书的尊贵读者了，我们有一系列的售后支持，保证你的消费物有所值。

错误

尽管我们已经对书籍作了仔细校对以保证内容准确，但错误在所难免。如果在书中发现任何的文字或代码错误，非常欢迎你将这些错误提交给我们，这样可以帮助我们在后续版本中改正错误，避免其他读者产生不必要的误解。一旦发现错误，请登录 <http://www.packtpub.com/support>，选择书名，点击 **errata submission form**（提交勘误）链接，然后填写具体的错误信息即可。只要你提交的勘误通过验证，勘误信息就会上传到我们的网站，或者追加到已有勘误列表中，显示在该书的勘误页面。

盗版

对所有媒体来说，互联网盗版都是一个棘手的问题。**Packt** 很重视版权保护。如果你在互联网上发现我们公司出版物的任何非法复制品，请及时告知我们相关网址或网站名称，以便我们采取补救措施。

如果发现可疑盗版材料，请通过 copyright@packtpub.com 联系我们。对您帮助我们保护作者权益、确保我们持续提供高品质图书的行为表示敬意。

问题

如果你对本书有任何问题，请联系 questions@packtpub.com，我们会尽力解决。

第1章 HTML5、CSS3和响应式网页设计入门

直到最近，网站设计普遍还在使用固定宽度（如960像素），以期给所有终端用户带来较为一致的浏览体验。这种固定宽度设计在笔记本电脑上显示刚刚好，而在高分辨率显示器上却会在两边多出些空白。

但现在有了智能手机。苹果公司的 iPhone 第一次带给我们真正意义上易用的手机上网体验，之后其他公司纷纷效仿。现在人们可以舒服地使用手机上网冲浪，不用再像过去那样需要有“挑圆片”^[1]世界冠军一样的灵活拇指，才能在小屏幕上看看网页。此外，消费者在家中上网时优先使用小屏幕设备（如平板电脑、上网本）正成为趋势。一个不争的事实是，使用小屏幕设备上网的人数正在以前所未有的速度增长。与此同时，27英寸和30英寸的大显示器也在快速普及。上网设备之间的尺寸差距与日俱增。

[1] 一种游戏，<http://en.wikipedia.org/wiki/Tiddlywinks>。——译者注

幸运的是，面对不断发展的浏览器和上网设备，我们有可行的解决方案。采用 HTML5 和 CSS3 技术的响应式网页设计，可以使网站兼容多种设备和屏幕。而这种方法的最佳之处，在于不需要什么服务器端方案，也完全可以实现。

本章内容：

- 支持小屏幕设备的重要性
- 什么是移动网站设计
- 什么是响应式网页设计
- 优秀响应式网页实例赏析
- 视口和屏幕的区别
- 安装和使用修改视口的浏览器扩展程序
- 使用 HTML5 编写更简洁的标记
- 使用 CSS3 解决常见的设计问题

1.1 为什么智能手机很重要（而老版的 IE 不再重要）

虽然统计数据一般仅用作粗略参考，但来自 gs.statcounter.com 的统计数据却值得注意。从2010年7月到2011年7月的12个月中，全球手机浏览器的使用量从2.86%上涨至7.02%。

Internet Explorer 6的使用率则是从8.79%下降到3.42%。到2011年7月，Internet Explorer 7的使用率也降到了5.45%。如果客户老是要求：“我们的网站必须兼容 IE6和 IE7！”你可以反驳说：“我们应该把精力花在更有价值的地方。”用手机上网的人比用台式机和笔记本中的 IE6或 IE7上网的人多多了。你可以听到全球前端开发工程师震耳欲聋的欢呼声！

越来越多的人使用小屏幕设备上网，这些设备上的浏览器在设计时都考虑到了如何显示好现有网站。手机浏览器会将一个标准网页缩小至与设备可视区域（标准技术术语叫做“视口”）恰好匹配。然后用户在自己感兴趣的内容区域上放大浏览。这样看起来已经挺好了，那作为前端设计师和工程师的我们，为什么还要在这上面继续优化呢？



在 iPhone 或 Android 手机上浏览的网页越多（如上图所示的那样），就越能深刻体会到为什么我们还需要继续优化。为了看清楚页面内容，需要不停地放大、缩小页面，然后为了看到视口外的文字，再左、右拖动，结果一不小心点了一个链接，你说讨厌不讨厌？我们当然可以做得更好！

1.2 响应式设计一定是最佳选择吗

如果预算充足且形势需要，做一个真正的“手机版”网站是首选。这样就可以基于用户的设备、位置、连接速度，以及包括技术特性在内的其他变量来提供不同的内容、设计和交互。举一个实际的例子，假设有一家比萨连锁店，它有一个标准版的网站和一个手机版网站，手机版网站可以基于增强现实功能、用户当前 **GPS** 位置找到附近的比萨店。单独一个响应式设计远远不能支撑这种解决方案。

然而，不是所有项目都要求如此完美。大多数情况下，根据视口大小为用户提供与之匹配的视觉效果还是优先选择。例如，针对大多数网站，虽然从服务器端请求的内容相同，但我们可以根据不同设备改变网页的显示方式。在小屏幕设备上，可能是将次要内容放在主体内容之下，或者最坏情况下将其直接隐藏；也可能是将导航按钮改造成便于手指操作，而不是只提供一些小需要精确瞄准才能点击的玩意儿！字体大小也应该恰到好处，便于阅读，不再需要不停地在屏幕上拖来拖去。同样，在迎合小屏幕的同时，我们也不想降低笔记本和台式机用户的浏览体验。既然我们意在兼容万物，那给那些配备 **1900** 像素甚至更大屏幕的用户提供一点额外改进又有何不可呢？简而言之，我们需要那些能响应各种设备大小的完美设计。

1.3 响应式网页设计的定义

响应式网页设计（**RWD, Responsive Web Design**）这个术语，由伊桑-马科特（**Ethan Marcotte**）提出。他在 **A List Apart** 发表了一篇开创性的文章，将三种已有的开发技巧（弹性网格布局、弹性图片、媒体和媒体查询）整合起来，并命名为响应式网页设计。这个术语还有一堆表示相同意思的其他叫法，如流式设计、弹性布局、塑料布局、流体设计、自适应布局、跨设备设计以及弹性设计。

上面仅列举了其中一部分！不过，正如马科特等人所说，真正的响应式设计方法不仅仅只是根据视口大小改变网页布局。相反，它是要从整体上颠覆我们当前设计网页的方法。以往我们先是针对桌面电脑进行固定宽度设计，然后将其缩小并针对小屏幕进行内容重排；现在我们应该首先针对小屏幕进行设计，然后逐步增强针对大屏幕的设计和內容。

一句话概括响应式网页设计

如果要用一句话概括响应式网页设计，我觉得它是针对任意设备对网页内容进行完美布局的一种显示机制。相反，如果需要根据不同设备提供特定的内容和功能，那就需要一个真正的

“手机版”网站。这种情况下，手机版网站会提供与桌面版网站完全不同的用户体验。

1.4 为什么要在响应式设计上停滞不前

响应式网页设计能够根据视口变化控制页面文档流，但我们可以走得更远。**HTML5**提供了比 **HTML 4**更多且更加语义化的标签。**CSS3**的媒体查询是响应式设计不可或缺的组成部分，**CSS3**的其他功能给了我们前所未有的灵活性。我们将挣脱背景图像和复杂的 **JavaScript** 代码的羁绊，拥抱简洁的 **CSS3**渐变、投影、字体、动画和变换。

在使用 **HTML5**和 **CSS3**制作响应式网页之前，让我们先来欣赏一些值得学习的实例，看看哪些高手正在使用新潮的响应式 **HTML5**和 **CSS3**绝技创造奇迹，而我们可以从他们的开创性杰作中学到些什么。

1.5 响应式网页设计示例

若要全方位测试你自己或别人的响应式网站，则需要针对每一种设备和不同的屏幕尺寸，分别准备不同的测试系统。尽管这是最完美的办法，但通过改变浏览器窗口大小其实就可以完成大多数测试。除此之外，还有很多第三方插件和浏览器扩展可供选择，通过它们可以将当前浏览器窗口或视口设定为指定像素。必要时，还可以自动将当前浏览器窗口或视口切换成为默认大小(如1024×768像素)。这样你就可以轻松地测试不同屏幕视口尺寸下的网站效果。

迷恋像素？忘了它吧！

进入了响应式网页设计的教堂，就不要再迷恋像素（**px**）这个度量单位，因为大多数情况下我们不会用像素，而会使用相对度量单位（**em** 或百分比）。相对单位更方便我们审查其他响应式设计作品，查看设计的变更之处。

1.5.1 下载视口调试工具

Internet Explorer 用户请下载安装 Microsoft Internet Explorer Developer Toolbar，下载地址如下：

<http://www.microsoft.com/download/en/details.aspx?id=18359>

如果你在使用 Safari，虽然 ResizeMe (http://web.me.com/aaronholla/Safari_Extensions/ResizeMe.html) 的功能类似且免费，但我最爱 Resize (<http://resizeSafari.com>)。

Firefox 用户请下载 Firesizer (<https://addons.mozilla.org/en-US/firefox/addon/firesizer/>)，

Chrome 请下载 Windows Resizer (<https://chrome.google.com/webstore/detail/kkelicaakdanhinjdeammmlcgefonfh>)。

不喜欢使用浏览器扩展？还有一个方法：我写了个简单 HTML 页面来显示浏览器窗口的当前视口高度和宽度。页面用了 jQuery 框架，获取当前的视口的高度和宽度并显示出来。你可以在浏览器新标签页中打开这个页面，调整窗口大小，然后切回你要测试的页面查看效果。这个超级简单的“**What size is my viewport page?**”页面地址如下：

<http://benfrain.com/easily-display-the-viewport-size-of-your-page-for-responsive-designs/>

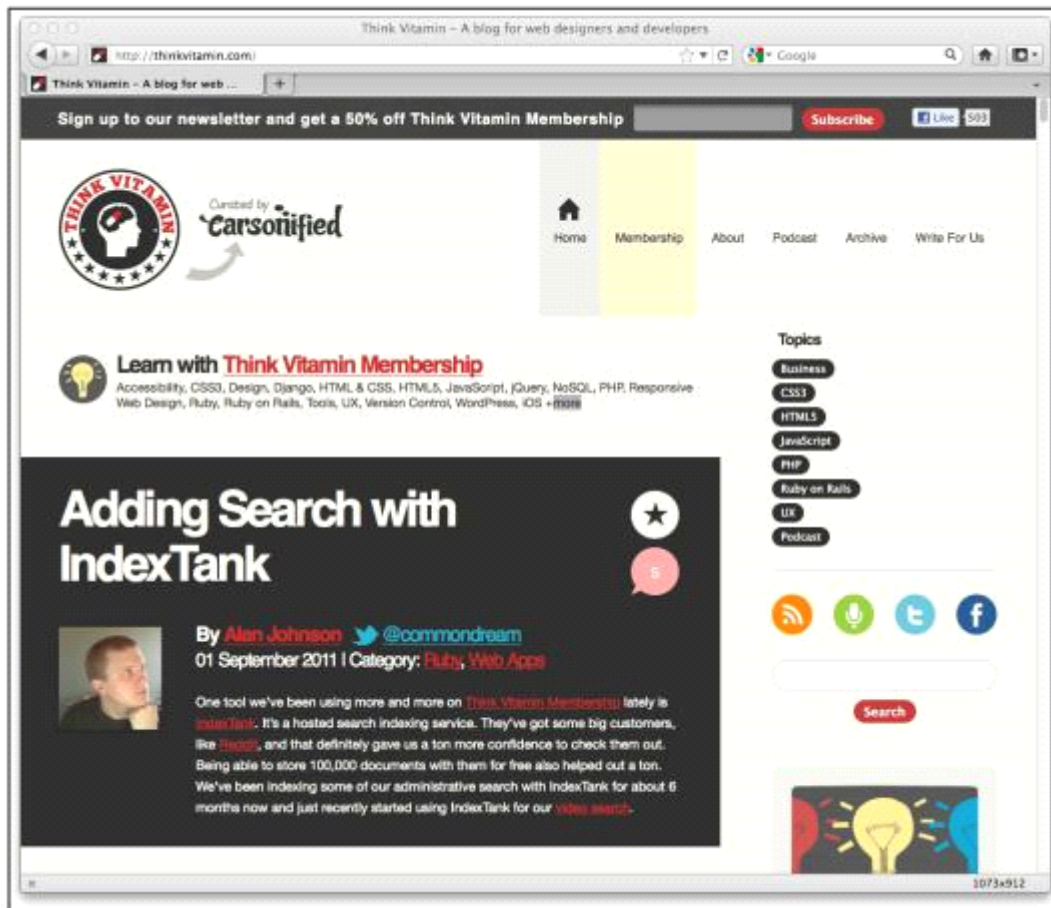
视口和屏幕尺寸

视口和屏幕尺寸不是同一个概念。**视口**是指浏览器窗口内的内容区域，不包含工具栏、标签栏等。也就是网页实际显示的区域。**屏幕尺寸**指的是设备的物理显示区域。需要注意的是有些浏览器调整工具显示的尺寸包含浏览器的其他元素，诸如地址栏、标签栏和搜索栏，而有些工具则不是这样。在下面的截图中，实际的视口尺寸显示在右上角（1156×921像素），同时 Firesizer 插件将窗口尺寸显示在右下角（1171×1023 像素）。

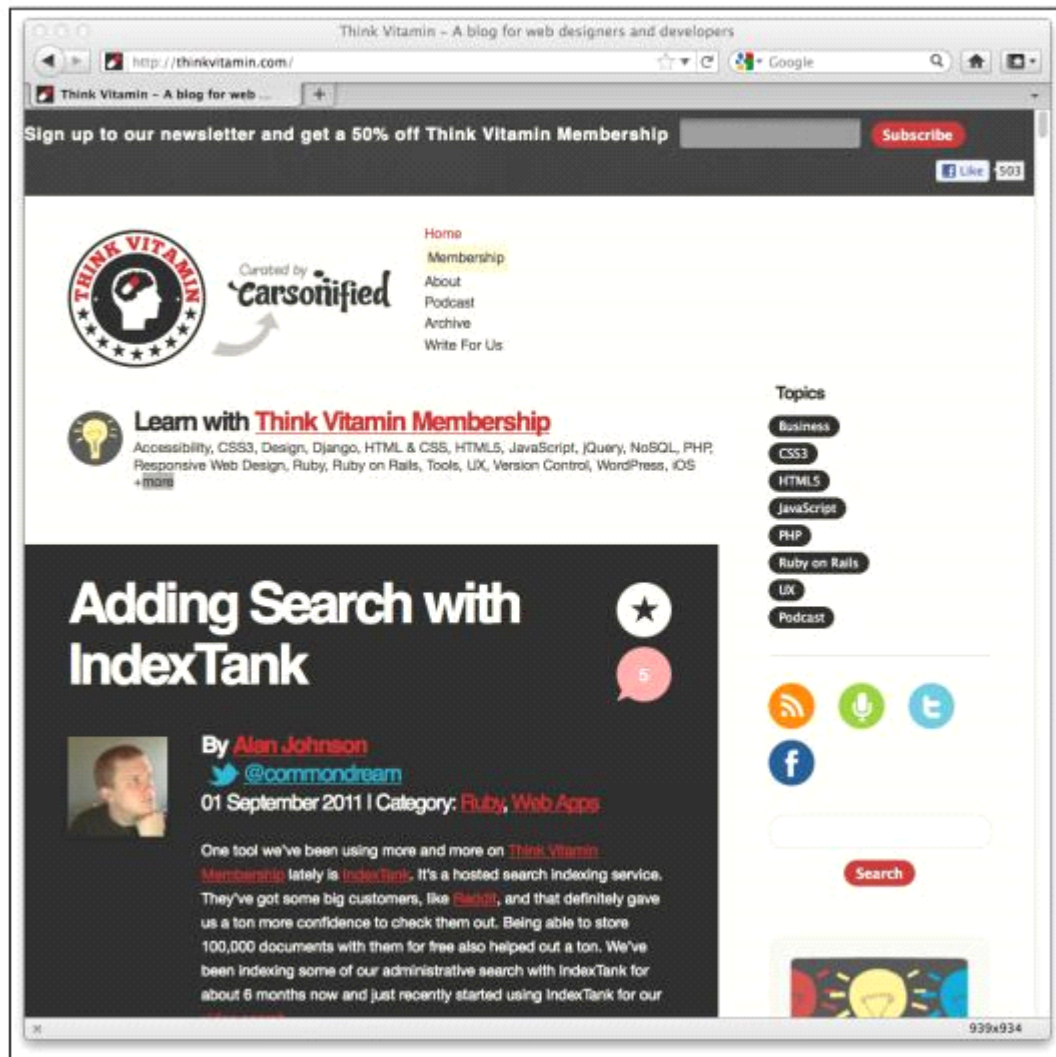


现在，我们带着所有需要的工具，开始鉴赏最好的响应式网站。启动你钟爱的浏览器，打开浏览器调整工具，访问 <http://thinkvitamin.com/> 。

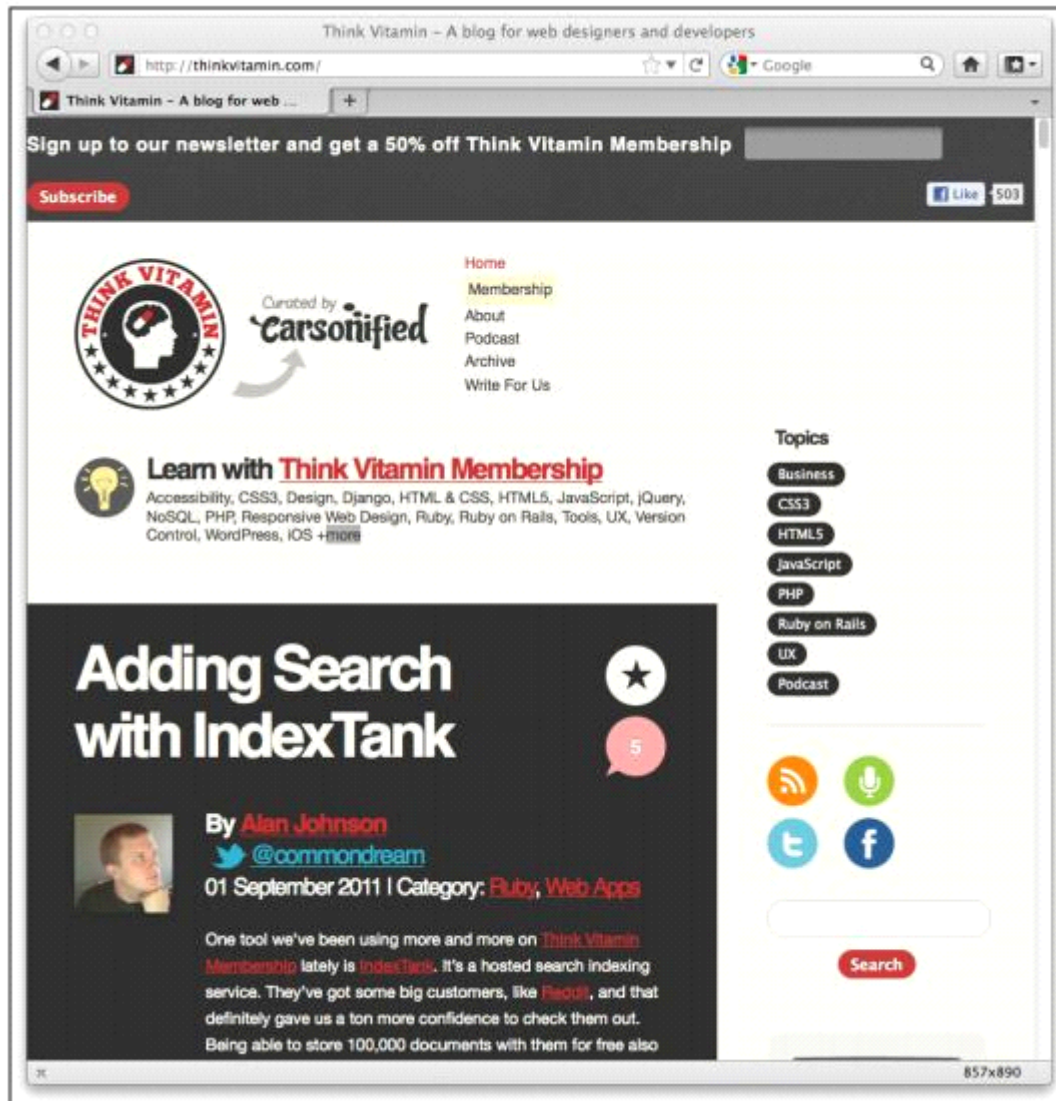
如果你的视口宽度大于1060像素，你会看到如下图所示的布局：



如果你的视口宽度介于930像素到1060像素之间，你会看到如下图所示布局：

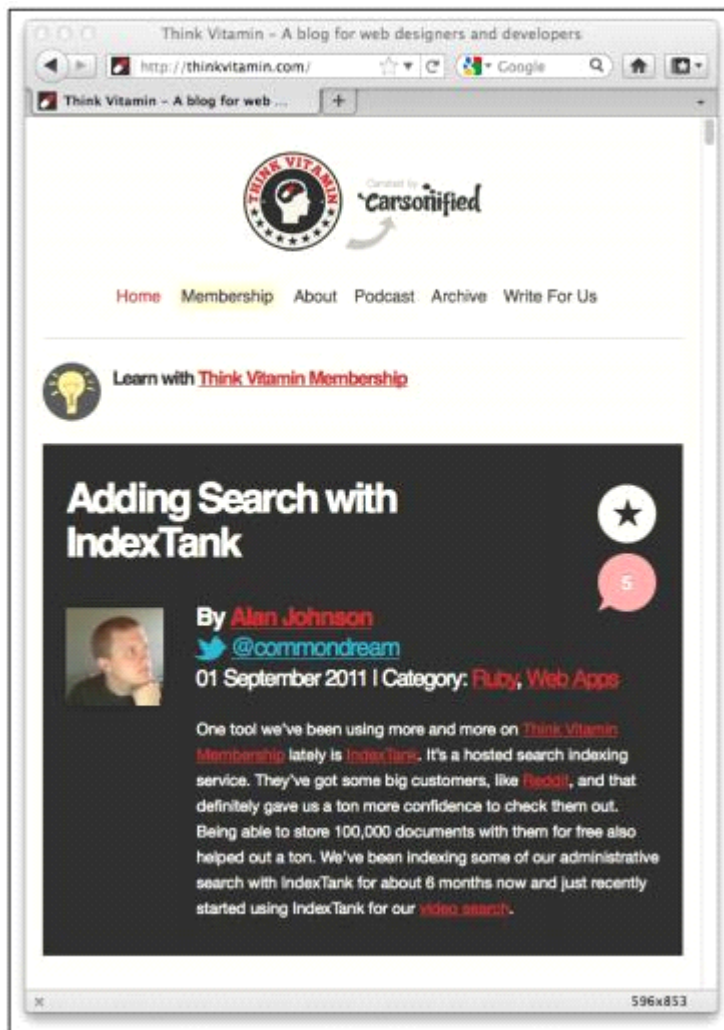


注意看 logo 旁边的主导航菜单是如何变化的。主内容区右侧的图标一个挨着一个排列，界面上的一切都合理可用，最重要的是，没有一样从屏幕上溢出。现在让我们再看看视口宽度小于880像素的效果，截图如下：



头部的效果基本没变，但注意右侧的侧边栏还是变窄了一些。图标排成了2行2列，同时文本块做了适当调整，其中的文本流相应地发生变化。

不过，将视口宽度减小到小于600像素，你就会发现一个重大的变化，如下图所示：



怎么样？整个侧边栏对我们的新视口做出了响应，网站最重要的内容区占据了整个浏览器窗口宽度。注意看头部的导航链接现在是水平排列的，而不是被放在 logo 的旁边。另外，logo 本身的大小也做了调整。以上所有的这些变化有助于根据视口尺寸为用户创建更好的体验。

让我们来看看另一个例子：<http://2011.dconstruct.org/>。视口较宽时（大于1350像素）网站效果如下图所示：



请特别注意那9张图的排列格式。当你减小视口宽度时（小于960像素），注意看界面发生了什么变化。三行三列的图片排列方式变成了三行两列外加下方一行三列，如下图所示：



继续减小视口宽度，在小于720像素的时候我们会遇到另一个设计“断点”。头部导航链接转换成了带图片的导航区域，这为触屏导航提供了更好的操作区域：



继续减小视口宽度，当小于480像素时，图片排列方式又变成了第一行2张图片，第二行3张，第三行4张。这些图片的大小在视口宽度缩小至大约300像素时又发生了变化。下面的截图显示了其在 iPhone 上的效果：



1.5.2 在线创意源泉

推荐一个响应式设计创意收集网站：<http://mediaqueri.es>。虽然这里收集的网站并不是全部都采纳完整的响应式方法论，即先针对小视口进行设计，然后逐步针对大视口进行渐进增强支持。但就目前来看，响应式设计方法兴起的时间不长，再考虑到响应式网页设计的各种可能性，这里确实有很多能让我们汲取创意的范例。尽管调整视口大小来浏览网站能说明响应式网页设计的理念，但这没有展示出 HTML5 的优势。HTML5 的优势事实上在幕后发挥。所以让我们将注意力转移到幕后，来了解一下 HTML5 的优秀之处。

1.6 为什么 HTML5很优秀

在创建可以通过 W3C 标准验证的页面时，HTML5强调简化标签，仅链接那些我们必须的 CSS、JavaScript 和图片文件。智能手机用户只能使用有限的带宽访问我们的页面，而响应式设计的一个主要目标就是，网站不仅要对用户有限的视口做出响应，还要以最快的时间加载网页。虽然移除冗余的标签元素只能节省一点字节，但积少成多！

较之上一个版本的 HTML（HTML 4.0.1），HTML5进行了极大的改进，提供了许多新特性。

前端开发工程师可能主要对 HTML5的新语义元素感兴趣，这些新元素给搜索引擎提供了含义更加丰富的代码。HTML5也可以在表单提交等基本网页交互场景中对用户做出反馈，一般不再需要复杂的 JavaScript 表单处理流程。同样，这为我们的响应式设计带来的好处就是，允许我们创建更加简洁和快速的代码。

获得更多，购买全书：<http://www.ituring.com.cn/book/1055>