

《响应式Web设计实践》

图书基本信息

书名：《响应式Web设计实践》

13位ISBN编号：9787115304094

10位ISBN编号：7115304092

出版时间：2013-3-1

出版社：人民邮电出版社

作者：[美] Tim Kadlec

页数：256

译者：侯鸿儒

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《响应式Web设计实践》

内容概要

随着各种各样的移动设备不断地涌现到使用者面前，Web设计的适应性已经成为设计师们所面临的最为艰巨的挑战。你设计出的网站不仅要在桌面计算机的大尺寸屏幕上可以为用户提供友好的UI和用户体验，同时在小尺寸屏幕上也应该可以提供一致的用户体验，并可以让用户能够在桌面大屏幕上和移动小屏幕上平滑切换，同时没有任何的不适应感觉。

本书作者是一位出色的开发者，在本书中，他将诸多技术和设计理念杂糅在一起，再辅以大量详细的配图、注释、引用等加以说明，从理论到实践，对响应式设计进行了详尽的解读。

本书适合从事Web设计的前端工程师和开发人员阅读参考。

《响应式Web设计实践》

作者简介

Tim Kadlec，是一位资深的Web开发者，也是Breaking Development联合创始人。Tim通过应用各类Web技术，满足了出版业、制造业、金融业等不同行业诸多用户对于跨设备完成各种规模任务的需求。侯鸿儒，本科期间自学了HTML、CSS、JavaScript和PHP，在清华大学计算机系完成了有关HTML5应用的硕士毕业设计。之后又顺其自然地接触到了UI/UX，对响应式设计有着浓厚兴趣，并做过“ Responsive Web Design & Workflow ” 的演讲。

书籍目录

第1章 无处不在的Web

第2章 流动布局

第3章 媒介查询

第4章 响应式多媒体

第5章 计划

第6章 设计流程

第7章 响应式内容

第8章 RESS

第9章 响应式体验

后记 展望未来

精彩短评

- 1、 简明易懂
- 2、 很棒的书，对RWD(responsive web design)的各方面都覆盖到，理论比较多，浅谈即止 ~ 高大全的书容易被写砸，这本小薄书面很全，接下来看实际应用逐点击破吧 ~
- 3、 设计理论的东西太多了，不是我想要的，有点像《CSS Garden》。
- 4、 不错 ~
- 5、 书中介绍了一些不错的开源小项目，后面几章侧重思想，就css目前来说，最佳实践还是bootstrap3，foundation等框架
- 6、 中英版本都看过，推荐。响应式设计绝不仅仅是开发者的事情，需要产品一同来决定。
- 7、 响应式设计入门，阅读体验很好。
- 8、 @20130425 开始阅读。第一章才是最重要的。一定好好好看第一章。这是总章。类似于独孤九剑的总诀式。ps：这本书似乎是校友翻译的？看译者序最后说是2012年x月于xx大学。
@20130902 终于把最后2章看完了.得出的教训就是看书最好是一口气看完,不然隔得时间久了,真的会失去连贯性.不管是书的内容还是看书的热情.

这本书对我来说,稍微有些超前,部分技术已经在用.但是对于我这个web新人来说,还不是太实用.可以留作以后参考.

- 9、 Mark Boulton在一篇博客种谈到响应式设计包含有3种不同的东西：传感器，系统，执行器。
- 10、 一半讲代码，一半讲设计，作者涉及面很广
- 11、 标原文啊啊啊
- 12、 作者语言幽默，但是实质性内容不多。
- 13、 还不错。
- 14、 从布局、media query、渐进式增强 3个方面揭露了响应式布局的本质和方向，非常好。后面几章（5~9）则略显枯燥
- 15、 精读了前5章，后面的粗略的翻了过去，就响应式来说比图灵的那一本好很多
- 16、 给一个概念
- 17、 基本思想：最基础用用户需求需要被满足
基本方法：查到当前设备的属性，针对性编辑，以最佳方式动态呈现
技术：开放性平台与SDK的应用简化业务
- 18、 和另外几本讲web响应式设计/HTML5的书大同小异啊
- 19、 这本书主要是从大处着手，对读者需要一定的css基础功。
- 20、 做项目的时候拿来应急。
- 21、 花了三天时间把这本书看完了。之前知道RWD但是没有具体的操作过，真的是很好的实战指南啊。另外当一开始就用RWD的思想去设计站点时，以移动优先为原则，那么它会颠覆你之前的思维和设计模式。
- 22、 前面几章讲媒介查询、布局、移动设计优先、渐进增强等，对RWD有了进一步的了解，后面几章很枯燥就没看了。毕竟是2013年的书，前端开发变化日新月异，广泛阅读还是挺有好处的。
- 23、 对响应式的介绍比较具体，完整，比较系统，让人容易接受和理解。
- 24、 比较认真了看了此书，比较系统的介绍，看完对响应式设计的一些完全的认知，不过确实更适合开发同学，但作为设计师，还是需要一些自己的思考与转译。
- 25、 2013-06-13亚马逊27.5¥；粗略一翻。
- 26、 适合前端开发看
- 27、 一些细节上的东西，以及，响应式不止是响应式布局
- 28、 方法论太多，意思不大。
- 29、 在书店一中午翻完，概念是不错，不少东西是抄的别的书，而流式布局，媒体查询等其它书上都讲了很多，svg，设备识别这些业界使用已久。如果一两年前出版，这是本好书，但现在太迟了，书里讲的都是大家用过的东西，且不够深入，说是“实践”实则“基础”。对于有移动端开发经验的人来说，这本书可有可无。

- 30、不错的设计思维传导
- 31、弥补了响应式的好多观念

1、从方法上系统的介绍了响应式设计，较之于从概念上简单介绍的书来说帮助很大，不仅仅是对响应式设计的一个深入理解，更学习了相关方法。译者保持了笔者简洁诙谐的语言风格，使全书可读性非常高~力荐^o^^o^^o^^o^^o^^o^

2、1.渐进增强 insteadof 优雅降级 —— mobile first 2.相应的限制 —— 定宽元素 3.自适应布局 (1).display:table-cell (2).flexbox (亦可用作水平垂直居中) (3).css3多列 4.media query 内容引导断点 not 设备引导断点 5.响应式加载媒体 (或使用SVG), display:none并不能节省性能 6.Adobe Shadow —— 远程调试测试移动设备 7.matchMedia —— media query for js 8.解决方案 RESS —— 客户端响应式布局+服务器用户代理检测

3、[《响应式Web设计实践》](http://book.douban.com/subject/21263576/)是一本关于响应式Web设计的书，虽然之前也有构建响应式站点的经验（比如http://z.qzone.com）仔细读完全书，才知道响应式不只是media query这么简单。本书内容非常丰富，下面介绍一些对我有所帮助的章节：3.4 媒介查询顺序---我之前的制作经验是“优雅降级”型的，先制作一个完整的站点，在Chrome、IE中都能完整查看了，然后再往下增加一些断点，比如640px、320px等，在这些断点中新增样式来覆盖之前的完整样式，或者隐藏一些元素，但这样的缺点是不支持媒体查询的手机浏览器就只能读取完整站点样式了。而且因为做了restyle，整体代码量会偏大。所以本书的经验“渐进增强”，先制作一个核心站点样式，这个样式中没有负责布局的样式，没有左右栏，就是简单的核心样式。然后在媒体查询中往上增加一些断点，比如320px、640px等，在这些断点中新增样式来增加布局样式，这样就免掉了之前的缺点。最后桌面端的IE678不认识media query，所以需要IE条件注释加载一个单独的样式：`<!--[if (lt IE 9) & (!IE Mobile)]><link rel="stylesheet" type="text/css" href="/css/ie.css" /><![endif]-->`我根据这一原则和方法修改了我的[博客](http://yuguo.us)的响应式实现方法，大家可以看看源码。

3.6 确定断点---这一节的观点是，新增的断点不应该由主流设备宽度来决定，应该由内容来决定，也就是说新增一个480的站点不应该是因为手机屏幕宽度很多都是480px，而应该是在481px的时候就能够有所调整了。这个观点很好，让我们从被动地设置断点，到拥有主动权。P102 响应式设计对性能的影响---Guy介绍了一下“前端的响应式设计”的弊端，大多数简单的前端处理的响应式有以下弊端：1. 下载并隐藏 2. 下载并缩小 3. 额外的DOM 这三个问题在比如http://z.qzone.com中几乎都存在，手机端下载了跟桌面端一样的HTML代码【问题3】，然后隐藏了一些下载功能【问题2】，最后把幻灯片缩小到手机屏幕大小【问题1】。但是这样也是有原因的，因为首页图经常需要运营来修改，设计师也没有资源出一系列大图的时候，再专门针对手机做一系列小图，所以当时的方案是简单缩小。当然以后有机会一定会去优化，做出更好的体验。4.5 背景图片---利用媒体查询，在屏幕大于一定宽度的浏览器中才加载自定义字体。4.7.2 广告---广告真的是一个很大的阻力，空间改版的时候想修改边栏宽度，就遇到广告位的阻力.....也许以后可以跟商户签订更灵活的协议。7.5.1 代码汤---不知道原文是什么，这里翻译成代码汤，有点奇怪，但意思大概是懂了，就是页面上的富文本CMS生成了很多冗余标签，冗余标签储存在数据库中，然后在移动端生成内容的时候就有诸多困难。解决办法是简化CMS，同时在编辑器中做一个简单内容拷贝，或者过滤器过滤标签。8.2.2 服务器端Modernizr---浏览器端的Modernizr我们都了解，是通过在浏览器里运行JavaScript来判断浏览器的能力，然后在html标签上附加一些标记class。服务器端Modernizr的原理有所不同，访问者第一次访问页面时（不带cookie），JavaScript会立刻执行并取得测试的结果，然后这些结果被添加到cookie，页面立刻刷新。当下一次加载页面时，服务器端会读取cookie值，然后直接输出html。缺点是JavaScript可能禁用，或者浏览器压根就不支持JavaScript，而且要加载两次（虽然第一次数据量很少，也是一次请求）。优点是不必依赖UA字符串，更稳定。9.2 网络---测试网络的方法之一是发起某个图片的请求，然后测试下载时间，根据这个来考虑是否需要加载更高分辨率的图片，方法之2是网速API，但其实并不太稳定，一方面不是所有设备都有这个API，另一方面不一定3G就更快。最后---响应式不简单是前端的责任（media query），也不简单是后台的责任（单独的m.站点），而是二者统一，后端来输出不同的内容，前台来按需加载图片，来做断点。设计师也需要了解前端知识，因为未来的主流趋势是设计师在浏览器中设计。本书有一些小问题，比如：P38边栏的链接中有空格，其实url中不能有空格的吧。第8章原章节名字可能确实是RESS，但中文书里这样写太奇怪了，这并不算约定通俗的缩写，让人云里雾里，简单的“服务器端检测”就非常好了

。另外不喜欢（译注：）这样的文字插入到正文中，应该用脚注或者边栏注可能更好。整体来说翻译的不赖，译者的自我评价（原文5星，翻译3星）过谦了，我打5星

。 <http://yuguo.us/weblog/responsive-web-design/>

4、去年年中一次很偶然的的机会，在微博里看到人民邮电出版社在寻人翻译这本《Implementing Responsive Design》。由于之前在百度实习时为项目组里做过相关的演讲，而且对RWD（Responsive Web Design）也很有兴趣，于是便有了自己翻译的第一本书——《响应式Web设计实践》。在我看来，似乎很难将RWD归到现有的特定职位的工作范围中去，因为它里面即有设计又有开发，这倒是正好中了想要学设计的前端工程师（例如我）或者想要学开发的设计师们的下怀。我想响应式Web设计的出现，不仅会影响到Web应用的站点模式（不同平台不同站点vs不同平台相同站点）以及项目流程，更重要的是它将给那些双面手们更大的舞台。在Startups, this is how design works[1]网站中，这些双面手们被称之为独角兽（Unicorn）。我们暂且不在这里讨论专业设计师与专业前端工程师的搭配与“独角兽”小组孰优孰劣，作为前端工程师学一些设计的东西至少能在实施RWD的过程中有很大的帮助。下面是我作为一名计算机研究生学习设计的一些经验，希望能对各位志同道合的前端工程师有所帮助。首先，要认识到你不是一个人在战斗。一开始，我曾以为有这种跨界想法的人不多，但后来发现其实非常多，尤其是国外的一些独立开发者或设计师。甚至有网站为这些人起了新的名字——Design Technologist[2]。其次，最好能订阅一些知名的设计网站或博客，比如dribbble[3]（如果招程序员要看github的话，那招设计师就要看他的dribbble了）。要想能设计出优秀的设计，就要先多看优秀的设计，从而提高自身的审美水平。此外，网上还有大量专门针对程序员的设计教程（例如hackdesign[4]）。当然，购买一些设计类书籍（例如《设计心理学》，你也可以参考我的设计类图书豆列[5]）进行学习也是很有必要的。然后，就要动手画了。一开始你可以不去计较工具，随处可见的铅笔+A4纸就是很好的起点。如果你能负担得起樱花针管笔和Moleskin的本子当然更好，因为更好的工具确实能给你营造更好的气氛甚至给你带来更多的灵感。到后来你还可以购买一些诸如数位板这样的高级工具辅助创作。此外，如果你不知道设计师们是如何进行创作的话，那就要主动去跟他们接触——去找同公司的设计师交流，或者参与线下的设计师分享会（iconfans[6]就经常举办这样的分享活动）都是非常好的途径。最后，要从手中的项目开始。如果是公司的项目，那你可以主动参加有关设计的讨论；如果是你自己的项目那就更方便了。经过一次次的修改之后，你会发现自己的设计水平也会一次比一次更好。罗马不是一天建成的，独角兽也是，所以要有耐心。虽然我们还远不是独角兽，但至少我们在磨砺自己的角。PS：由于还是学生，因此翻译功力还很稚嫩，本书中翻译得不妥之处还请大家多多包涵。书的内容值得给5星，但给自己的翻译3星，所以打了4颗星。最后附上一些自己总结的RWD相关资源，希望能对诸位读者朋友有所帮助：1. RWD灵感来源：<http://mediaqueri.es/>2. RWD设计模式

：<http://bradfrost.github.com/this-is-responsive/patterns.html>3. RWD开发浏览器窗口调整工具

：<http://resizemybrowser.com/>4. RWD Weekly：<http://responsivedesignweekly.com/>参考资料：[1]

<http://startupsthisishowdesignworks.com/>[2]

<http://adaptivepath.com/ideas/what-makes-a-design-technologist>[3] <http://dribbble.com>[4]

<http://hackdesign.org/>[5] <http://book.douban.com/doulist/1391423/>[6] <http://e.weibo.com/iconfans>

5、好书，翻译简洁明了，很喜欢~《响应式Web设计实践》比《响应式Web设计：HTML5和CSS3实战》讲得更深入，但是作者已经默认读者懂得基础的HTML5和CSS3知识，而没有介绍例如<aside>、<section>、radius这些标签或者属性的意义。所以个人觉得这两本书能够很好的互补。

6、做responsive web design，如果需要看什么书籍的话，这一本就够了，其他的自己google上搜搜，在项目中积累问题的解决方法 and 经验，这样就够了。这是一本很负责任的书，里面的旁白中有好的想法和插件的介绍，还有在线的sample供参考和练习，真的是一本不可多得的好书。可以先粗略的看一遍，列出一个主脉，然后在实践中依次去验证和思考，把知识化为自己的能力，又强大不是，嘿嘿。我推荐这本书，因为它让我相信响应式设计，我能搞定！所以你也一样！

7、原书很好，烂在翻译，必须拿原版英文对照着看。如果问google在线翻译与译者翻译哪个好，我会说一样。出现此原因，最大可能是因为译者根本不是相关从业者，第二，因为时间关系译者根本没时间去思考而直接用google翻译。看了下译者简介，原来是个研究生，自学了HTML5等等，还在百度做过演讲？！原来是一点从业经验都没有的人。再看看现在大公司都找什么211名校实习，反而错过了不少大专院校优秀学生。名校为什么会是名校，因为政治思想过硬啊！以后买书要看清楚译者。

章节试读

1、《响应式Web设计实践》的笔记-第103页

这章主要是讲性能的。。

虽然每个网站的情况各不相同，但是3种原因几乎是所有过度下载网站所共有的：

下载并隐藏

下载并缩小

额外的DOM

下载并隐藏：可惜display:none帮不上性能任何忙

下载并缩小：其实在小屏幕上用户是无法欣赏那些高质量的图片的，那些多余的字节就这样被白白浪费掉了。

然后还是有一些方法可以帮助我们将站点的性能维持在一个可控的范围之内：

使用响应式图片

优先为移动平台创建

测量

测量

可以用来测试的工具很多，但我推荐在真实的设备上做测试(<http://akamai.com/mobitest>)，同时用WebPageTest在桌面浏览器上做测试，因为你可以使用setviewportsize命令来缩放它们响应式图片的2种实现方法：

1.Sencha.io Src

2.自适应图片

第二个可能是比较好的选择，因为它不需要依赖任何第三方资源

在一篇讨论两种流行的解决方案的冲突的博客文章中，Jason Grigsby在开头处便一语中的：为了提高性能，浏览器希望在知道页面的布局前，就可能的快地加载网页；但另一方面开发者们却要依靠她们关于页面布局的知识，来确定要下载哪张图片。这真是一块难啃的骨头

2、《响应式Web设计实践》的笔记-第221页

苹果公司建议:一个触摸目标的高度至少为44px。为链接增加一些padding。

3、《响应式Web设计实践》的笔记-第184页

$(c^2=a^2+b^2)$

(τ ϵ χ). χ [6pt]

100~m(χ^3) of water [6pt]

移动设备上的站点应该是简单而易于使用的，而不是简化了的。

4、《响应式Web设计实践》的笔记-第73页

内嵌式媒介查询的不足之处：创建了一个较难维护的大型文件。这个问题现在应该有很多解决方案，例如bootstrap3 多less文件@import, grunt-contrib-watch监听recess，始终生成一个css文件，维护模块化的less文件。

惊讶，外部媒介查询也会下载所有的样式。.....

5、《响应式Web设计实践》的笔记-第28页

字体大小单位 <https://developer.mozilla.org/zh-CN/docs/Web/CSS/length>

6、《响应式Web设计实践》的笔记-第70页

p63

然而从视口元标签目前的发展情况来看，有关它将会被从浏览器中剔除的猜想应该是合理的，取而代之的是浏览器对于@ViewPort规则的整合。

P74

从桌面端向下设计还是从移动端向上设计

P76

先用一个较为简单的布局作为浏览器的默认样式，然后再在上面搭建各种其他样式，这样做所需的CSS代码更少，同时代码结构也更为清晰。

7、《响应式Web设计实践》的笔记-第1页

最好的相机就是你手里的那部

p5

在这个网络连接无处不在的时代，最好的浏览器就是你手里的那个。

p7

键盘始于打字机，鼠标始于苹果1984年推出的Macintosh不像那么非触摸的操作,这里可没有鼠标悬浮状态可供使用。此外，触屏设备还有很多固有的交互方式，例如滑动、下拉刷新以及拖动等等。开发独立站点并不是一种未来友好型的方法，为了能在即将到来的海量设备大潮中生存下来，我们必须拥抱Web的灵活性。

Marcotte在他的博客文章中"Toffee-Nosed"(趾高气昂)中明确地指出了这一点：“每当我谈论或者撰写有关响应式设计的東西的时候，我总是希望强调非常重要的一点：响应式设计不是‘为移动设备而设计’，它也不是‘为台式机而设计’，它是关于采用灵活的、设备无关的方法来为Web进行的设计。”

响应式设计的核心内容：

媒介查询 (Media Query)、流动式布局(fluid grids)、自适应图片(scalable images)

但它们只是冰山一角.

后来人们发现,一个成功的响应式的设计方法其实是建立在与渐进增强非常相似的原则之上的。其实,响应式设计就是打了激素的渐进增强。

p15

这里有关于 "渐进增强"和"优雅降级"的观念的定义和对比.

p20 关于是不是要使用js的库我想澄清的一点是,我不反对使用库,事实上你将在本书中见到几个很有用的jQuery插件。我想说的其实是:你一定要仔细考虑要将什么东西包含到你的页面中来。如果你需要某个库,那就去用它;如果你不需要,那么不如自己写一段代码来节省一些页面。

8、《响应式Web设计实践》的笔记-第149页

这一页提到的Adobe Shadow.

现在已经停止更新了(最终版本似乎是2012年下半年的4.x的版本)

然后改名叫做 Adobe Edge Inspect CC

下载地址：<https://creative.adobe.com/inspect>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu000.com