

构建跨平台APP

Building Cross-Platform Apps with Bootstrap

响应式UI设计入门

杨旺功 赵荣娇 编著



是广受欢迎的HTML、CSS和JS框架，用于开发响应式布局、移动设备优先的Web项目

- ❖ 页面需要自动响应用户所有设备的屏幕大小
- ❖ 小例子多，不讲究很深的理论，只要求快速上手
- ❖ 结合响应式框架Bootstrap，详解跨平台Web开发
- ❖ 提供两个完整的网站响应式Web设计案例

3 示例项目源代码下载

清华大学出版社

构建跨平台APP

响应式UI设计入门

杨旺功 赵荣娇 编著



清华大学出版社
北京

内 容 简 介

基于“响应式内容”的本质,页面应该在任何合理屏幕大小的设备上看上去都很舒服,我们做设计的就应该站在用户的角度考虑,一切设计都应该看上去很美、很实用。本书的响应式设计正是基于这一理念,从小到一个图标,大到完整的页面布局,都从响应式UI设计入手,适合所有响应式设计的入门人员和中小企业的网站搭建人员学习。

本书分为3篇共11章,第一篇是响应式设计基础,我们首先介绍清楚到底什么是响应式设计,然后介绍分解响应式设计页面,包括页面中的元素、页面的布局、导航栏、多媒体等;第二篇是响应式设计框架,详细介绍了目前使用最广泛的Bootstrap框架,包括它的各种样式设计和特效设计;最后一篇是实战,用一个Bootstrap贴吧和HTML5扁平化公司主页,实践前面所学的内容。

本书技术新颖、内容精练、结构清晰、注重实战,适合广大网页设计或移动设计初学人员学习,同时也非常适合大中专院校师生学习阅读,也可作为高等院校计算机及相关专业的教材使用。

本书封面贴有清华大学出版社防伪标签,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

构建跨平台APP:响应式UI设计入门/杨旺功,赵荣娇编著. —北京:清华大学出版社,2016
ISBN 978-7-302-43797-0

I. ①构… II. ①杨… ②赵… III. ①移动终端—应用程序—人机界面—程序设计 IV. ①TN929.53

中国版本图书馆CIP数据核字(2016)第100157号

责任编辑:夏非彼

封面设计:王 翔

责任校对:闫秀华

责任印制:何 芊

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 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 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:190mm×260mm 印 张:12.75 字 数:326 千字

版 次:2016 年 7 月第 1 版 印 次:2016 年 7 月第 1 次印刷

印 数:1~3000

定 价:49.00 元

产品编号:067444-01

前言

响应式Web设计（Responsive Web design）的理念是：页面的设计与开发应当根据用户行为以及设备环境（系统平台、屏幕尺寸、屏幕定向等）进行相应的响应和调整。具体的实践方式由多方面组成，包括弹性网格和布局、图片、CSS media query的使用等。无论用户正在使用笔记本还是iPad，我们的页面都应该能够自动切换分辨率、图片尺寸及相关脚本功能等，以适应不同设备。换句话说，页面应该有能力去自动响应用户的设备环境。

目前，响应式设计仍然在不断变化，不断创新。比如，新的设备不断出来（iPad Pro、iPad Mini 4、小米Note），这让以前的设计想法土崩瓦解。而各种Web的响应式设计也获得了越来越多的注意，“让人们忘记设备尺寸”的理念将更快地驱动响应式设计，所以Web设计也将迎来更多的响应式设计元素。随着各种移动设备的不断出现，响应式设计不再新鲜，但却成为我们UI身边不可缺少的元素。

本书就是一本梳理响应式设计元素的图书，它的目的是让所有UI设计人员找到设计方向，完善设计理念和办法。从而使自己设计的网站能够兼容多个终端——而不是为每个终端做一个特定的版本。这样，我们就可以不必为不断到来的新设备做专门的版本设计和开发了。

与其他书的区别

（1）细分响应式页面中的各个元素，从小地方入手，了解响应式设计的精髓。

详解响应式设计涉及的文字、表单、框架、图片、视频等基础元素，帮读者一步一步打下响应式设计的基础。

（2）实例佐证技术的讲解方式，每个技术点都伴有实例，让读者能看完就学会。

全书的技术点不仅仅有理论知识，还有大量的HTML 5和CSS 3代码，让读者可以看、可以动手，可以通过一点一滴的操作逐步掌握所有技术。

（3）紧贴流行的设计趋势，让读者不仅能学会现在的设计方式，还能看清什么样的设计才是好的设计和未来的设计。

大部分例子都提供了宽屏和窄屏的设计效果，并讲解了各种设计下的书写技巧。

进阶路线

本书共3篇11章，主要章节规划如下。

第一篇（第1章~第5章）响应式设计基础篇

讲述了响应式设计的由来和简单发展，通过响应式网页中的元素、响应式布局、响应式导航、响应式多媒体等4个模块，详细介绍了响应式设计中我们需要学习和关注的技术点。

第二篇（第6章~第9章）响应式设计框架篇

讲述了如何使用响应式框架Bootstrap，包括它的样式设计、组件设计、特效设计三大模块，每个模块都辅助了很多小例子，让读者能一看就会。



第三篇（第10章~第11章）响应式网站设计实战篇

本篇包含两个完整案例，一个是利用框架实现的响应式后台管理，一个是利用扁平式结构实现的公司宣传主页。

本书适合的读者

本书适合任何对 Web 设计感兴趣的读者进行阅读，示例中会涉及一些 HTML 和 CSS 代码，理解这些代码会极大地帮助你深入了解响应式 Web 设计开发中的技巧。

- 前端工程师
- 设计师
- 产品经理
- 架构师
- 项目经理
- 大中专院校的学生
- 可作为各种培训学校的入门教程

作者简介

杨旺功，男，毕业于重庆邮电大学，硕士，现任职于北京印刷学院设计艺术学院教师。具有丰富的数字媒体交互产品设计和移动应用UI设计的教学经验。在数字化艺术与设计领域成果显著，著有《给设计师看的交互程序设计书——FLASH ACTIONSCRIPT 3.0溢彩编程》《Flex 4.0 RIA开发宝典》等教材，并发表多篇数字化艺术与设计的论文。主要方向是信息与交互设计产品创新设计，对信息与交互设计有深入的研究。

赵荣娇，女，毕业于中国传媒大学，工学硕士。目前就职于阿里去啊，担任前端开发工程师，曾参与旅游特卖首页、1688订单等项目开发。热爱技术，喜欢分享。中国传媒大学新媒体研究院主办《信息科技周刊》总编辑、《新媒体技术动态》发起人。

本书第1~5章由阿里去啊的赵荣娇编写，第6~11章北京印刷学院设计艺术学院的杨旺功编写，参与创作的还有陈宇、刘轶、姜永艳、马飞、王琳、张鑫、张喆、赵海波、肖俊宇、欧阳薇、周瑞、李为民、陈超、杜礼、孔峰。

本书代码下载地址（注意数字和字母大小写）如下：

<http://pan.baidu.com/s/1boCkSkZ>

如果下载有问题，请电子邮件联系booksaga@163.com，邮件主题为“构建跨平台APP：响应式UI设计入门源码”。

编者
2016年5月

目 录

第1章 传说中的响应式设计	1
1.1 支持跨平台设备的响应式设计	1
1.1.1 什么样的设计是响应式的	1
1.1.2 响应式设计坚守的4大原则	3
1.2 响应式设计与其他页面设计的对比	4
1.2.1 固定布局	5
1.2.2 流式布局	7
1.3 实战：创建一个响应式网页	9
1.3.1 设置HTML文档的Meta标签	9
1.3.2 设计HTML文档结构	9
1.3.3 使用CSS 3媒介查询	10
1.3.4 运行第一个响应式页面	12
1.4 小结	14
第2章 响应式网页中的元素	15
2.1 文字	15
2.2 表单	18
2.2.1 自定义Radiobox	19
2.2.2 自定义Checkbox动画	21
2.2.3 美化输入框	24
2.2.4 下拉框	27
2.3 框架	28
2.3.1 传统的iFrame框架	29
2.3.2 响应式的iFrame框架	29
2.4 实战：实现一个响应式登录表单	31
2.4.1 设置登录表单的HTML结构	31
2.4.2 设计登录表单的通用样式	33
2.4.3 使用CSS 3媒介查询实现响应式登录表单	35
2.5 小结	37
第3章 响应式布局	38
3.1 布局切换	38
3.2 侧边栏	40
3.3 宽高等比例变化	44
3.4 列表	46
3.4.1 定义列表分级菜单	46
3.4.2 列表切换效果	48
3.5 表格	52
3.5.1 简单自适应表格	52



3.5.2 翻转滚动表格.....	55
3.5.3 隐藏表格栏目.....	60
3.6 实战：响应式商品展示列表.....	61
3.7 小结.....	64
第4章 响应式导航.....	65
4.1 响应式导航菜单设计五大原则.....	65
4.1.1 按照优先级显示内容.....	65
4.1.2 用创造力来处理有限的空间.....	66
4.1.3 下拉菜单.....	66
4.1.4 给导航菜单换位置.....	66
4.1.5 放弃导航菜单.....	66
4.2 导航类型.....	66
4.2.1 单层导航.....	67
4.2.2 多层导航.....	70
4.2.3 面包屑导航.....	72
4.3 页码设计.....	73
4.4 小结.....	75
第5章 响应式多媒体.....	76
5.1 图标的响应式.....	76
5.2 图像.....	78
5.2.1 可适配的图像.....	78
5.2.2 图像网格.....	80
5.3 视频.....	84
5.3.1 内嵌视频响应式的难点.....	85
5.3.2 从其他网站中手动嵌入视频.....	85
5.4 响应式图表.....	86
5.4.1 一款响应式图表库.....	86
5.4.2 带Tooltip提示的线形图.....	88
5.4.3 简单的饼图.....	91
5.5 小结.....	93
第6章 Bootstrap入门.....	94
6.1 初次接触Bootstrap.....	94
6.1.1 Bootstrap的优势.....	94
6.1.2 下载Bootstrap.....	96
6.2 在网站中引入Bootstrap.....	97
6.3 调用Bootstrap的样式.....	98
6.4 调用Bootstrap的组件.....	100
6.5 调用Bootstrap的js特效.....	101
6.6 实战：一个Bootstrap实现的响应式页面V1.0.....	102
6.7 小结.....	104
第7章 Bootstrap的样式设计.....	105
7.1 字体.....	105
7.1.1 标题.....	105
7.1.2 全局字体和段落.....	106
7.2 表格.....	107

7.2.1 基本用法.....	108
7.2.2 表格的附加样式.....	108
7.2.3 为表格行或单元格添加状态标识.....	110
7.2.4 响应式表格.....	112
7.3 表单	112
7.4 按钮	115
7.4.1 按钮的基本样式.....	115
7.4.2 调节按钮大小.....	116
7.4.3 块级按钮.....	117
7.4.4 为按钮设置不可点击样式.....	118
7.5 图片.....	118
7.5.1 图片类.....	118
7.5.2 响应式图片.....	119
7.6 Bootstrap工具类	119
7.6.1 响应式工具	119
7.6.2 工具类.....	120
7.7 实战: Bootstrap响应式页面V2.0	122
7.8 小结.....	123
第8章 Bootstrap的组件设计	124
8.1 下拉菜单	124
8.2 按钮组	125
8.2.1 垂直排列的按钮组.....	126
8.2.2 两端对齐的按钮组.....	126
8.2.3 嵌套按钮组.....	127
8.3 input控件组	128
8.3.1 最常见的搜索框.....	128
8.3.2 带提示的搜索框.....	128
8.4 导航	129
8.4.1 胶囊式导航.....	129
8.4.2 面包屑导航.....	130
8.4.3 头部导航.....	130
8.5 列表组.....	133
8.6 分页	134
8.6.1 普通的分页.....	134
8.6.2 上一页/下一页.....	135
8.7 标签	136
8.8 面板	137
8.9 进度条	139
8.10 缩略图.....	140
8.11 实战: Bootstrap响应式页面V3.0	142
8.12 小结.....	143
第9章 Bootstrap的特效设计	144
9.1 模态对话框	144
9.2 标签页切换	146
9.3 Tooltip	147
9.4 弹出框	147
9.5 折叠	148



9.6 幻灯片.....	149
9.7 实战：Bootstrap响应式页面V4.0.....	151
9.8 小结.....	153
第10章 使用Bootstrap实现一个百度贴吧后台.....	154
10.1 确定后台管理的需求.....	154
10.2 设计页面布局.....	155
10.2.1 引入Bootstrap 3框架.....	155
10.2.2 实现页面布局代码.....	156
10.3 设计导航栏.....	157
10.3.1 构建导航的整体架构.....	157
10.3.2 设计标题和导航链接.....	158
10.3.3 实现搜索框和通知系统.....	158
10.3.4 实现管理员的登录信息.....	159
10.3.5 构建响应式导航.....	160
10.4 设计左侧边栏.....	162
10.5 设计主功能部分.....	163
10.5.1 主功能的头部.....	163
10.5.2 主功能的帖子列表.....	165
10.6 小结.....	168
第11章 使用HTML 5设计扁平化的公司主页.....	169
11.1 响应式设计的关键.....	169
11.2 导航栏的设计.....	170
11.2.1 列表的实现.....	170
11.2.2 弹出式菜单的实现.....	174
11.3 主功能部分的设计.....	176
11.3.1 什么是视差滚动效果.....	176
11.3.2 视差效果的实现.....	176
11.3.3 图文列表的实现.....	178
11.4 小结.....	183
附录 CSS 3选择器使用一览.....	184
f1.1 标签选择器.....	184
f1.2 类选择器.....	184
f1.3 id选择器.....	185
f1.4 通配符选择器.....	186
f1.5 子元素选择器.....	186
f1.6 后代元素选择器.....	187
f1.7 相邻元素选择器.....	187
f1.8 属性选择器.....	188
f1.9 组选择器.....	188
f1.10 复合选择器.....	189
f1.11 结构化伪类.....	190
f1.12 目标伪类:target.....	195
f1.13 状态伪类.....	195
f1.14 否定伪类:not(S).....	196



传说中的响应式设计

第 1 章

响应式设计已成为最新的Web设计趋势，并且已成为人们热议的话题。许多网站都已采用响应式来设计网站，响应式设计也正在改变人们的网站设计方式。不仅网站的设计方式改变了，作为Web设计师也不得不与时俱进，积极接受新事物，不断改善和提升自己。

曾几何时，打开新浪或搜狐的主页，我们已被满屏的文字刺得眼花缭乱，而这些内容显示在手机上，则让人无所适从，更无从下手。响应式设计让我们可以在电脑上、iPad上和手机上实现看似相同实则不同的设计，而且让一切自动实现转换。

本章的主要内容是：

- 了解什么是响应式设计
- 知道响应式设计的设计原则
- 响应式设计与网页其他布局的区别
- 创建一个简单的响应式网页



支持跨平台设备的响应式设计

响应式Web设计（Responsive Web design）的理念是：集中创建页面的图片排版大小，可以智能地根据用户行为以及使用的设备环境（系统平台、屏幕尺寸、屏幕定向等）进行相对应的布局。

1.1.1 什么样的设计是响应式的

页面可以根据用户的设备尺寸或浏览器的窗口尺寸来自动地进行布局的调整，这就是响应式布局。在这个移动互联兴起的时代，响应式布局占据着越来越重要的地位。图1.1是一个直观的响应式布局设计示意图。

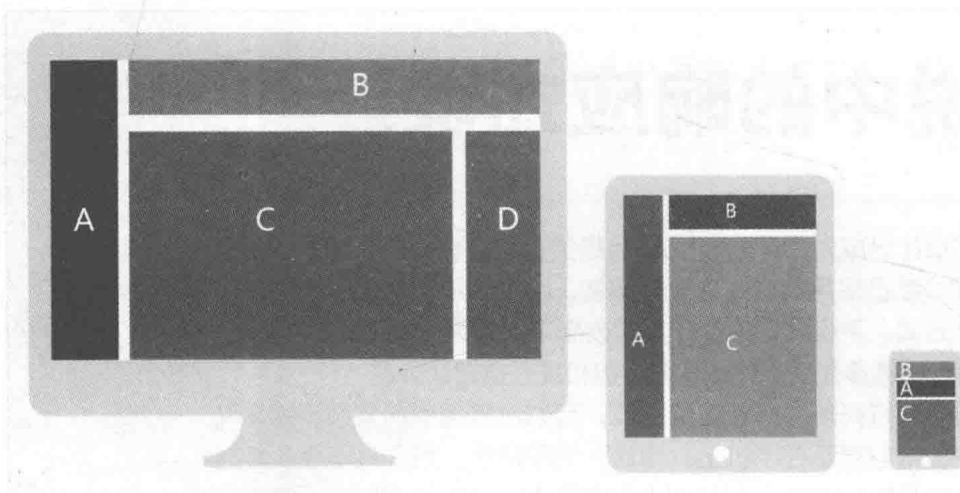


图1.1 响应式布局设计示意图

我们来看图1.2和图1.3，这是两个典型的响应式设计案例，读者可以直观地感受一下。



图1.2 响应式设计案例1



图1.3 响应式设计案例2

近年来，移动互联网发展势头迅猛，尤其是高性能智能手机和平板的普及，使得在移动设备上浏览绚丽的页面成为可能（相对于曾经的WAP手机站来说）。响应式设计越来越流行，预计在不久的将来，大部分的网站都会拥抱移动，响应式页面将会成为主流选择。

1.1.2 响应式设计坚守的4大原则

响应式设计对于解决这么多类型的屏幕问题来说是个不错的方案，但作为一门专业的设计，我们需要注意一些基本的设计原则：

- 移动优先还是PC优先
- 内容流
- 位图还是矢量图
- 相对单位和固定单位

1. 移动优先还是PC优先

随着移动互联网的发展，很多小型创业企业甚至没有了自己的网站，只有一个APP应用。在这个时代，网站项目是从小屏幕入手过渡到大屏幕（移动优先），还是从大屏幕入手过渡到小屏幕（PC优先），成为企业考虑的首要问题。

传统的大企业改造型网站，大部分是从大屏幕逐步过渡到小屏幕，而且在过渡到小屏幕时会碰到一些额外的限制，如没法在第一页面显示更多的内容，要更简洁，具体到要放哪些标签就需要决策的内容。

在新兴的创业公司中，通常情况下都会从两方面同时着手，所以具体哪个优先还是要看哪种方式最适合你。

2. 内容流

随着移动屏幕尺寸越来越小，内容所占的垂直空间也越来越多。也就是说，内容会向下方延伸，这被称为“内容流”。早先的Web设计师习惯了使用像素和点来设计页面，可能会觉得这有点难以掌握。不过好在它很简单，多多练习就习惯了。图1.4展示了两种设计状态下页面内容变宽后的效果。

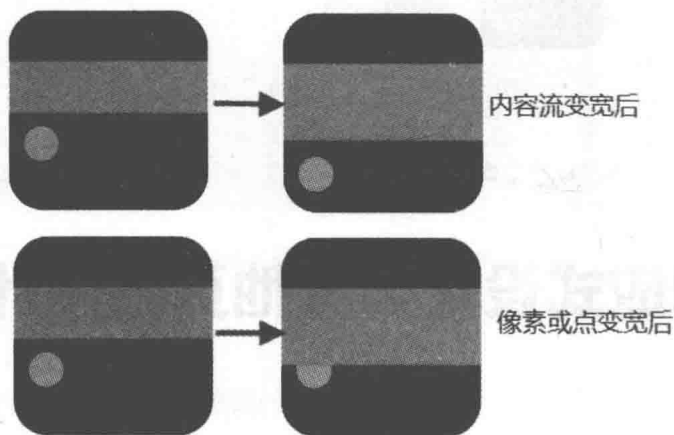


图1.4 内容流对比



3. 位图还是矢量图

以前我们知道，当一张图片被放大后就会出现比较“虚”的情况，这种图是位图；而放大后不变“虚”的则是矢量图。我们先来了解一下两者的概念。

矢量图使用线段和曲线描述图像，所以称为矢量，同时图形也包含了色彩和位置信息。

位图使用像素（一格一格的小点）来描述图像，计算机屏幕其实就是一张包含大量像素点的网格，在位图中，图像由每一个网格中的像素点的位置和色彩值来决定，每一点的色彩是固定的，所以放大后观看图像时，每一个小点看上去就像是一个马赛克，这就是我们常说的“虚”。

在响应式设计中，图标或图像都会涉及这个问题。如果我们的图标有很多细节，并且应用了很多华丽的效果，那就不用位图；否则，考虑使用矢量图。如果是位图，使用jpg、png或gif；矢量图则最好使用SVG或图标字体。位图和矢量图两者各有利弊。矢量图通常比较小，很适合移动端来展示，但部分比较老的浏览器可能不支持矢量图。还有，有些图标有很多曲线，可能导致它的大小比位图还大，所以要根据实际情况明智取舍。

4. 相对单位和固定单位

对于设计师而言，我们的设计对象可能是桌面屏幕，也可能是移动端屏幕或者介于两者之间的任意屏幕类型。不同的终端像素密度也会不同，所以我们需要使用灵活可变且能够适应各种设备的单位。

传统的设计单位有px、pt、cm等等，但他们都是固定单位，没法实现跨平台展示；那么，在这种情况下，百分比等相对单位就到了发挥作用的时候了。使用百分比时，我们所说的宽度50%是表示宽度占屏幕大小（或者叫视区，即所打开浏览器窗口的大小）的一半，如图1.5所示。

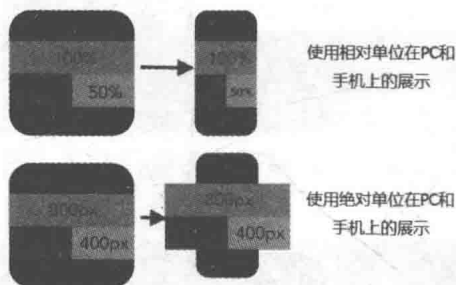


图1.5 绝对单位和相对单位对比



响应式设计与其他页面设计的对比

为了适应移动互联网大潮，传统的网页技术也在不断发展，还有一定的生存空间，如果不知道传统的技术，我们也不会了解响应式设计为什么这么流行，也不知道之前网页设计是

如何实现的。本节介绍两种非常传统的网页布局方式：固定布局和流式布局。

1.2.1 固定布局

固定布局，顾名思义就是各个部分都采用固定宽度的页面布局，如果缩放页面到窗口宽度小于页面宽度时，就会导致部分内容不可见，必须通过滚动条的拖动才可以浏览全部内容。

虽然移动互联网来势汹汹，响应式设计、流式布局开始逐渐流行，但是在很多应用场景下，固定布局仍是最合适的。例如，B/S结构的企业应用、海报宣传性质的页面等等。而固定布局的稳定、简单、成熟也是前端技术选型中重要的考量。

在开发流程的表达上，固定布局也是成熟而稳定的，从产品经理的草图到设计师的PSD设计稿，再到前端页面，全都是静态的，思路的传递简单明晰、成本低廉。相对来说，响应式界面不仅在HTML/CSS编写上更为复杂，对产品经理和设计师的能力素质、沟通表达都有更高的要求，需要更多的沟通成本。

开发者不应当盲目追求概念，更需要根据团队的情况、产品的需求、成本的考量来综合考虑技术的选型。因此笔者认为即使移动风潮来势汹汹，固定布局在很多场合下仍然不失为合适的选择。

如图1.6所示，一个固定列宽的网页布局主要由3个部分组成：列（Column）、槽（Gutter）、外边距（Margin）组成。列的宽度决定了容器内部的宽度，槽的宽度决定了列与列之间的固定间距，外边距则表示container边界和实际内容之间的间距。

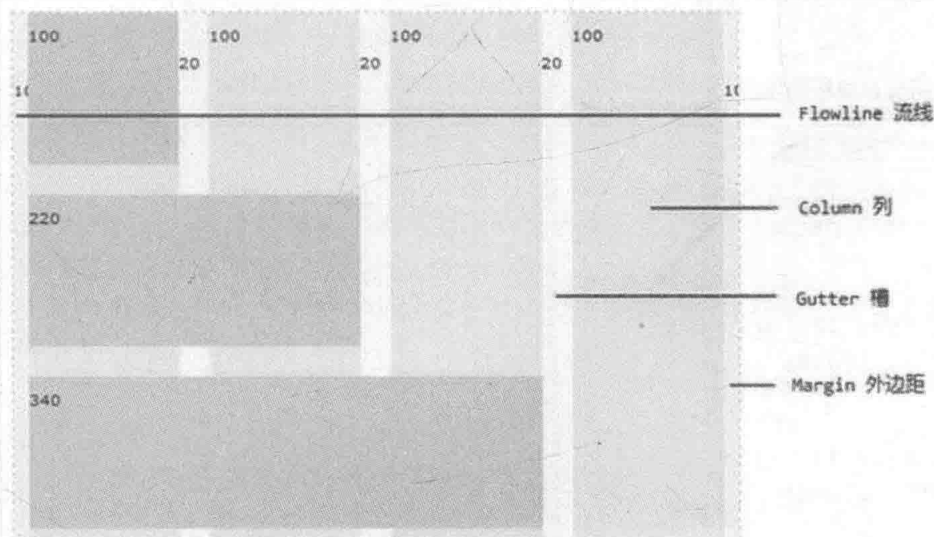


图1.6 固定栅格系统示意图

【示例1-1】

现在我们来看一段固定布局的源码：

```
/* 定义整个容器
```




```
-----*/  
.container_12 {  
    margin-left: auto;  
    margin-right: auto;  
    width: 960px;  
}
```

```
/* 全局设置
```

```
-----*/  
.grid_1,  
.grid_2,  
.grid_3,  
.grid_4,  
.grid_5,  
.grid_6,  
.grid_7,  
.grid_8,  
.grid_9,  
.grid_10,  
.grid_11,  
.grid_12 {  
    display: inline;  
    float: left;  
    margin-left: 10px;  
    margin-right: 10px;  
}
```

```
/* 分别设置栅格的宽度
```

```
-----*/  
.container_12 .grid_1 {  
    width: 60px;  
}  
  
.container_12 .grid_2 {  
    width: 140px;  
}  
  
.container_12 .grid_3 {  
    width: 220px;  
}  
  
.container_12 .grid_4 {  
    width: 300px;  
}  
  
.container_12 .grid_5 {  
    width: 380px;  
}
```

```
.container_12 .grid_6 {  
    width: 460px;  
}  
  
.container_12 .grid_7 {  
    width: 540px;  
}  
  
.container_12 .grid_8 {  
    width: 620px;  
}  
  
.container_12 .grid_9 {  
    width: 700px;  
}  
  
.container_12 .grid_10 {  
    width: 780px;  
}  
  
.container_12 .grid_11 {  
    width: 860px;  
}  
  
.container_12 .grid_12 {  
    width: 940px;  
}
```

这段代码涵盖了固定列宽的页面布局的主要内容，有以下几点：

- 设置容器，容器采用了960像素宽度居中的设置，需要设置width: 960px;并且左右两侧的外边距设置为auto，也可以简单地设置为margin: auto;。
- 将页面设计成一段段的栅格，栅格命名一般采用grid（格子）或者column（列）作为前缀进行表意，数字作为后缀表示栅格的列数。
- 设置页面全局属性，左浮动（float: left;）属性是必需的，此外还需要设置栅格之间的间隔宽度。此处设置左右两侧的外边距均为10像素。

1.2.2 流式布局

在响应式布局成熟之前，某些需要元素随窗口大小变化的需求往往采用Table进行布局，但是Table布局相对来说又有着不够灵活、结构复杂、语义性差的问题；而流式布局在某种程度上可以帮助开发者解决这个难题。对于流式布局来说，我们可以通过直接定义模块和模块间距的百分比的方式来实现。我们在前面讲解响应式设计原则的时候也提到过，使用相对单位而不是固定单位。

【示例1-2】

如下面给出一段流式布局的代码：



```

.row-fluid [class*="span"] {
    display: block;
    float: left;
    width: 100%;
    min-height: 30px;
    margin-left: 2.127659574468085%;
    *margin-left: 2.074468085106383%;
    -webkit-box-sizing: border-box;
    -moz-box-sizing: border-box;
    box-sizing: border-box;
}

.row-fluid [class*="span"]:first-child {
    margin-left: 0; /* 设置第一个子元素的左外边距为0 */
}

.row-fluid .controls-row [class*="span"] + [class*="span"] {
    margin-left: 2.127659574468085%;
}

.row-fluid .span12 {
    width: 100%;
    *width: 99.94680851063829%; /* 老版本的IE浏览器兼容代码 */
}

```

【代码解析】

类似*width: 23.351063829787233%; 这样的属性定义, 原因是IE 6/7下宽度100%时是包含了外层滚动条的宽度的, 因此需要针对性地做出兼容设置。

在流式布局页面中引入图片时会发现图片大小是固定的, 怎样让图片随着窗口的大小调整显示大小呢?

我们只需要为图片元素添加max-width: 100%; 和height: auto; 属性, 可以让图片按比例缩放不超过其父元素的尺寸。如果想让图片和父元素一直等宽的话, 则将max-width: 100%; 改为width: 100%; 即可, 例如下面的代码:

```

<html>
<head>
<style>
    .respond_img{
        height:auto;
        width: 100%;
    }
</style>
</head>
<div style="width:30%">
    
</div>
<body>
</html>

```