

## 第2章 使用Web标准设计和制作

Web 标准掀起了国内网页制作的革命,网页设计与制作人员都开始重新审视自己过去的网页作品,并且惊讶地发现那些充满了嵌套表格的 HTML 代码臃肿而且难以修改,于是一场清理 HTML 的行动开始了。本章中主要讲解了使用 Web 标准设计和制作网页的原理和优点。

### 2.1 HTML 基础

表格布局已经有很多年的历史了,在 HTML 和浏览器还不是很完善的时候,要想让页面内的元素能有一个比较好的格局几乎是不可能的事,由于表格不仅可以控制单元格的宽度和高度,而且可以互相嵌套,所以为了让各个网页元素能够放在预设的位置,表格就成为网页制作者的必备工具。

#### 2.1.1 表格布局的优点

目前仍有很多的网站在使用表格布局,表格布局使用简单,制作者只要将内容按照行和列拆分,用表格组装起来即可实现版面布局。

由于对网站外观“美化”要求的不断提高,设计者开始用各种图片来装饰网页。由于大的图片下载速度缓慢,一般制作者会将大图片切分成若干个小图片,浏览器会同时下载这些小图片,就可以在浏览器上尽快的将大图片打开。因此表格成为了把这些小图片组装成一张完整图片的有力工具,如图 2-1 所示。



#### 本章知识点

- 了解表格布局的特点及方式
- 了解什么是 Web 标准以及 Web 标准的优势
- 理解为什么要建立网站标准
- 运用网站标准的优缺点
- 理解 Web 标准
- Web 标准的争论与思考
- 了解 Web 标准的误区

🏠 DIV+CSS 3.0 > 网页样式与布局全程揭秘



图 2-1 预览效果与表格布局

2.1.2 表格布局的缺点

采用表格布局的页面内，为了实现设计的布局，制作者往往在单元格标签<td>内设置高度、宽度和对齐等属性，有时还要加入装饰性的图片，图片和内容混杂在一起。使代码视图显得非常臃肿。

因此当页面布局需要调整时，往往都要重新制作表格。尤其当有很多页面需要修改时，工作量将变得难以想象。

表格在版面布局上很容易掌控，通过表格的嵌套可以很轻易地实现各种版式布局，但是即使是一个 1 行 1 列的表格，也需要<table>、<tr>和<td>这 3 个标签，最简单的表格代码如下所示。

```

1. <table>
2.   <tr>
3.     <td>这里是内容</td>
4.   </tr>
5. </table>
    
```

如果需要完成一个比较复杂的页面时，HTML 文档内将充满了<tr>和<td>标签。由于浏览器需要把整个表格下载完成后才会显示，因此如果一个表格过长、内容过多的话，那么访问者往往需要等很长的时间才能看到页面中的内容。

同时，由于浏览器对于 HTML 的兼容，因此就算嵌套错误甚至不完整的标签都能显示出来。有时仅仅为了实现一条细线而插入一个表格，表格充斥着文档，使得 HTML 文档的字节数直线上升。对于使用宽带或专线来浏览页面的访问者来说，这些字节也许不算什么，但是当访问者使用手持设备(如手机)浏览网页时，这些代码往往会占据很多的流量和等待时间。

如此多的冗余代码，对于服务器端也是一个不小的压力，也许一个只有几个页面、每天只有十几个人访问的个人站点对流量不会太在意，但是对于一个每天都有几千人甚至上万人在线的大型网站来说，服务器的流量就是一个必须关注的问题了。

一方面，浏览器各自开发属于自己的标签和标准，使得制作者常常要针对不同的浏览器而开发不同的版本，这无疑就增加了开发的难度和成本。

另一方面,在不支持图片的浏览设备上(如屏幕阅读器),这种表格布局的页面将变得一团糟。正是由于上述的种种弊病,使得制作者们开始关注 Web 标准。

## 2.2 了解 Web 标准

Web 标准似乎是前几年刚刚在国内出现的一个新兴名词。大概从 2003 年左右开始,有关 Web 标准的各类文章与讨论便伴随着网络上大大小小的设计与技术论坛开始展开,也掀起了一股学习 Web 标准的热潮,从此之后众多网站开始打上了“符合 Web 标准”的字样。

### 2.2.1 什么是 Web 标准

Web 标准,即网站标准。目前通常所说的 Web 标准一般指进行网站建设所采用的基于 XHTML 语言的网站设计语言。Web 标准中典型的应用模式是 DIV+CSS。实际上,Web 标准并不是某一个标准,而是一系列标准的集合。

Web 标准由一系列的规范组成。由于 Web 设计越来越趋向于整体与结构化,对于网页设计制作者来说,理解 Web 标准首先要理解结构和表现分离的意义。刚开始的时候理解结构和表现的不同之处可能很困难,特别是如果不习惯思考文档的语义结构的话。但是,理解这点是很重要的,因为当结构和表现分离后,用 CSS 样式表来控制表现就是很容易的一件事了。

### 2.2.2 什么是 W3C

W3C 组织是对网络标准制定的一个非赢利组织,W3C 是 World Wide Web Consortium(万维网联盟)的缩写,像 HTML、XHTML、CSS、XML 的标准就是由 W3C 制定的。它创建于 1994 年,研究 Web 规范和指导方针,致力于推动 Web 发展,保证各种 Web 技术能很好地协同工作。

根据 W3C 官方网站的介绍,W3C 会员包括生产技术产品及服务的厂商、内容供应商、团体用户、研究实验室、标准制定机构和政府部门,大家一起协同工作,致力在万维网发展方向上达成共识。自 1998 年开始,“Web 标准组织”将 W3C 的“推荐”重新定义为“Web 标准”,这是一种商业手法,目的是让制造商重视并重新定位规范,在新的浏览器和网络设备中完全支持那些规范。

### 2.2.3 W3C 发布的标准

网页主要由三部分组成:结构(structure)、表现(Presentation)和行为(Behavior)。对应的网站标准也分三个方面。

- 结构化标准语言,主要包括 HTML、XHTML 和 XML。
- 表现标准语言,主要包括 CSS。



◎ 行为标准，主要包括对象模型(如 W3C DOM)、ECMAScript 等。

这些标准大部分由 W3C 组织起草和发布，也有一些是其他标准组织制定的标准，比如 ECMA(European Computer Manufacturers Association)的 ECMAScript 标准。下面对它们详细介绍。

## 1. HTML

Hyper Text Markup Language(HTML, 超文本标识语言)被广泛用于现在的网页, HTML 的目的是为文档增加结构信息, 例如表示标题、表示段落。浏览器可以解析这些文档的结构, 并用相应的表现形式表现出来, 例如浏览器会将<strong> </strong>之间的内容用粗体显示。并且设计师也可以通过 CSS(Cascading Style Sheets)来定义某种结构以什么形式表现出来。

HTML 元素构成了 HTML 文件, 这些元素是由 HTML 标签(Tags)所定义的。HTML 文件是一种包含了很多标签的纯文本文件, 标签告诉浏览器如何去显示页面。

从结构上讲, HTML 文件由元素(Element)组成, 组成 HTML 文件的元素有许多种, 用于组织文件的内容和指导文件的输出格式。绝大多数元素有起始标签和结束标签, 在起始标签和结束标签之间的部分称为元素体, 例如<body></body>。每一个元素都有名称和可选择的属性, 元素的名称和属性都在起始标签内标明。HTML 的标签有两种标记: 一般标记和空标记。

## 2. XML

XML 是 Extensible Markup Language(可扩展标识语言)的简写。XML 与 HTML 类似, 也是标识语言, 不同的地方是 HTML 有固定的标签, 而 XML 允许自定义自己的标签, 甚至允许通过 XML namespaces 为一个文档定义多套设定, 看一个 XML 例子。

```
1. <addressbook>
2.   <entry>
3.     <name>AJIE</name><email>ajie33@hotmail.com</email>
4.   </entry>
5.   <entry>
6.     <name>ALLAN</name>
7.     <email>neo_n@21cn.com</email>
8.   </entry>
9.   <entry>
10.    <name>YAHOO</name>
11.    <email>tingpeng@msn.com</email>
12.  </entry>
13. </addressbook>
```

在该 XML 程序代码中每个自定义的标签都是一一对应的, 有开始标签就必须有结束标签。在<entry>标签中显示了 1 组相关内容, 共显示了 3 组相关内容, 文档结构和内容清晰明确。

一些 XML 的应用, 例如 XHTML 和 MathML, 已经成为 W3C 推荐的规范。用户同样可以通过样式规范(CSS 和 XSL), 来定义 XML 标签的表现形式。XML 文档目前还不能直接用浏览器显示, 页面展现依然采用 HTML 或者 XHTML, XML 现在大多用于服务器与服务器(系

统与系统)之间的数据交换。

### 3. CSS

CSS 是 Cascading Style Sheets(层叠样式表)的缩写,是用于控制网页样式并允许将样式信息与网页内容分离的一种标记性语言。通过 CSS 可以控制 HTML 或者 XML 标签的表现形式。W3C 推荐使用 CSS 布局方法,使得 Web 更加简单,结构更加清晰。

### 4. XHTML

XHTML 实际上就是将 HTML 根据 XML 规范重新定义一遍。它的标签与 HTML 4.0 一致,而格式严格遵循 XML 规范。因此,虽然 XHTML 与 HTML 在浏览器中一样显示,但是如果需要转换成 PDF,那么 XHTML 会容易得多。

XHTML 有三种 DTD 定义:严格的(Strict)、过渡的(Transitional)、框架的(Frameset)。DTD 是 Document Type Definition(文档类型定义)的缩写。它写在 XHTML 文件的最开始,告诉浏览器这个文档符合什么规范,用什么规范来解析。

### 5. DOM

DOM 是 Document Object Model(文档对象模型)的缩写。DOM 给了脚本语言(类似 ECMAScript)无限发挥的能力。它使脚本语言很容易访问到整个文档的结构、内容和表现。

DOM 是 HTML 和 XML 文档的编程基础,它定义了处理执行文档的途径。编程者可以使用 DOM 增加文档、定位文档结构、添加修改删除文档元素。W3C 的重要目标是利用 DOM 提供一个使用于多个平台的编程接口。W3C DOM 被设计成适合多个平台,可使用任意编程语言实现的方法。

## 2.2.4 Web 标准的优势

Web 标准是由 W3C 和其他标准化组织制定的一套规范集合,包含一系列标准,包含我们所熟悉的 HTML、XHTML、JavaScript 以及 CSS 等。Web 标准的目的在于创建一个统一的用于 Web 表现层(即普通浏览者所看到的页面)的技术标准,以便于通过不同浏览器或终端设备向最终用户展示信息内容。相对于目前的网页制作方法,使用 Web 标准的优势有以下几点。

#### 1. 加速开发

在 20 世纪 90 年代后期,当互联网和 Web 逐渐成为主流时,Web 浏览器的开发商还没有完全支持 CSS(层叠式样式表),而对于 Web 开发人员来说,他们可以用 CSS 来控制 HTML 文档的表现。

由于浏览器对 CSS 的支持不够,再加上一些设计师的需求,导致他们为了控制网页的美观性而滥用 HTML。一个典型的例子就是,当设计师可以用 `border="0"` 来隐藏表格的边框时,用隐藏表格来控制布局的方法同样被使用。另一个例子是对 Transparent(透明)的使用,同样是不可见,设计者却使用空白的 GIF 图片来控制布局。





虽然 HTML 页面的简单大大促进了 Web 的发展，但是它也为将来留下了隐患。因为它们对代码格式非常的“宽容”，助长了一些浏览器私有代码的发展，这些私有代码造成无数用户无法得到最佳体验。由于 HTML 从来就没被用来控制一个文档的表现，导致大量的乱码、非法代码、浏览器的专用代码和属性被随意地使用。“校验”这个词也很少被人问津。

新版本的 Web 浏览器的发布，使得对 CSS 的支持得到了加强与扩展，但是并没有达到它应有的水平。尽管浏览器的开发商对 CSS 支持的步伐很缓慢，但是现在已经有许多浏览器选择了支持 CSS，此时不应该再有任何理由像以前那样使用 HTML 了，应该让它恢复本来的面貌，去描述文档的结构，而不是它的表现。正因如此，我们现在才能够用 CSS 达到其本来的目的了。

Web 标准强制用户进行错误校验。简单地声明用户的 HTML 是什么版本，校验程序将按用户声明的标准来校验用户的页面。校验器将严格校验并详细地告诉用户有哪些错误，这样缩短了开发者花费在质量上的时间，并保证用户的站点在不同浏览器上保持高度的一致性。即使现在的浏览器依然有某些显示 Bug，但是这已经比之前的情况好多了。

## 2. 易于维护

多年以来，Web 标准团体一直推荐“保持视觉设计和内容相分离”的优点，这意味着 HTML 变得非常简单，大部分的 HTML 页面只有一些<div>和<p>标签，以及一个指向强大的 CSS 文件的链接。这种完全的分离使得你的页面开发和维护变得简单，开发团队之间能够更好的协调，例如编辑和设计师可以分开工作。

## 3. 拓展访问渠道

干净的代码带来更多的利益。不支持 CSS 的浏览器通常忽略样式表，换句话说，符合标准的 XHTML 可以被任何浏览器呈现，包括非传统的客户端，例如手机、语音阅读器和屏幕阅读器等，任何支持这些简单标签的设备都可以。

一个符合 Web 标准的站点可以支持移动访问，以及兼容以前老版本的浏览器。用户可以获得所有好处而且更加容易开发和维护，甚至于在这个过程中能节省一些硬件成本。

## 4. 节约带宽成本

从页面上剥离了<font>、<table>标签和一些用于装饰的图片，从而使页面尺寸从 20.9k 缩减到 9.2k。虽然这些缩减看起来微不足道，但是当所有页面访问聚集起来就相当多了，使得站点的流量不堪重负。

普通的站点一天大约有几千的页面访问量，可以节约 56% 的带宽。而大型商业站点可能一两分钟内就达到这样的访问量，流行的热门站点更常常一天就达到几千万的页面访问量。如果每页节约 30k~40k，加上缓存的样式表不需要再次下载，每月能为这些站点节约上千美元。

## 5. 提高用户体验

在 Web 初期，巨大的图形界面设计使得拨号上网的用户忍受缓慢的访问速度，随着宽带

的普及这种情况有些好转。但是出差的商务人士仍然可能通过旅馆的电话拨号上网成为你的新用户，采用干净、标准化的代码可以帮助你的用户快捷方便地在网站上完成他们的目的。

## 2.3 网络设计的标准时代

网络技术的更新从来没有停止过，当我们觉得网站设计技术已经非常简单和熟悉的时候，国外网站设计标准化的概念早已悄悄的形成和迅速“蔓延”。自 2000 年起，大部分新发行的浏览器版本都开始支持网站标准。主流的网页编辑工具也开始全面支持网站标准，甚至一些软件几乎完全由 XML 文件组成，例如 Dreamweaver。一些著名的大型商业网站开始采用网站标准来重新构建，另外一些则拒绝非标准浏览器浏览它们的网站。众多的设计网站和个人网站更是标准推广的先行者，纷纷转向采用 XHTML+CSS 来建立。

### 2.3.1 为什么要建立网站标准

每当主流浏览器的版本升级，用户刚建立的网站就可能变得过时，这就需要升级或者重新建造一遍网站。例如 1996~1999 年典型的“浏览器大战”，为了兼容 Netscape 和 IE，网站不得不为这两种浏览器写不同的代码。同样的，每当新的网络技术和交互设备的出现，用户也需要制作一个新版本来支持这种新技术或新设备，例如支持手机上网的 WAP 技术。类似的问题不胜举：臃肿、繁杂的网站代码浪费了大量的带宽；针对某种浏览器的 DHTML 特效，屏蔽了部分潜在的客户；不易用的代码，残障人士无法浏览网站等等。这是一种恶性循环，是一种巨大的浪费。

如何解决这些问题呢？有识之士早已开始思考，需要建立一种普遍认同的标准来结束这种无序和混乱。商业公司(Netscape、Microsoft 等)也终于认识到统一标准的好处，因此在 W3C(W3C.org)的组织下，网站标准开始被建立(1998 年 2 月 10 日发布 XML1.0 为标志)，并在网站标准组织(webstandards.org)的督促下推广执行。



网站标准的目的是：提供最多的利益给最多的网站用户；确保任何网站文档都能够长期有效；简化代码、降低建设成本；让网站更容易使用，能适应更多不同用户和更多网络设备；当浏览器版本更新，或者出现新的网络交互设备时，确保所有应用能够继续正确执行。

对于网站设计和开发人员来说，遵循网站标准就是使用标准；对于网站的用户来说，网站标准就是最佳体验。

### 2.3.2 遵循标准的网站与传统网站的区别

传统网站只是印刷媒体的延伸，设计目标是保证在 4~6 个主流浏览器版本中看起来一致。

通常的特征如下。

- ⊙ 以表格为基础的布局；
- ⊙ 内容与表现方式混杂在一起。典型的例子是<font>标签；
- ⊙ 垃圾代码(非标准代码)；
- ⊙ 不易用的代码；
- ⊙ 语义(即语法意义)不正确的代码，例如<b>不解释的话，用户根本不明白这是字体加粗的意思。

而采用网站标准建立的网站是一个能够接受各种用户和各种设备的广泛的交流沟通工具。其一般特征如下。

- ⊙ 语义正确的标识。即使使用能够表达含义的标签，也能保证代码可以在文本浏览器、PDAs、搜索引擎中被正确理解；
- ⊙ 正确有效的代码。通过 W3C 代码校验的就是正确的代码；
- ⊙ 对人、机都易用的代码。能够接受广泛的用户和访问的设备；
- ⊙ 用 CSS 分离表现层和内容。使代码更简洁、下载速度更快，批量修改和定制表现形式更容易。

### 2.3.3 采用网站标准的优缺点

针对以上所讲，使用 Web 标准的优势有很多。总结归纳一下，主要可以从以下几点来体现，如表 2-1 所示。

表 2-1 使用 Web 标准的优点

对浏览器	<ul style="list-style-type: none"> <li>⊙ 文件下载速度更快，浏览器显示页面速度也更快；</li> <li>⊙ 有清晰的语义结构，所有内容能被更多的用户访问；</li> <li>⊙ 由于实现了结构与表现的分离，使得内容能被更多的设备访问；</li> <li>⊙ 有独立的样式表，用户可以很方便地选择自己的页面外观</li> <li>⊙ 可以调用独立的打印样式文件，方便页面的打印。</li> </ul>
对网站拥有者	<ul style="list-style-type: none"> <li>⊙ 由于代码更简洁，相关组件用得更少，所以使得维护变得更容易；</li> <li>⊙ 宽带成本减低，节约了成本；</li> <li>⊙ 页面具有结构清晰的语法意义，使得更容易被搜索引擎搜到网站的信息；</li> <li>⊙ 页面可以调用不同的 CSS 样式文件，使得更新或改版变得更加容易；</li> <li>⊙ 有清晰合理的结构，提高了网站的易用性。</li> </ul>
兼容性	<ul style="list-style-type: none"> <li>⊙ 由于实现向后兼容，所以当浏览器版本更新或者出现新的网络交互设备时所有应用能够被继续正确执行。</li> </ul>



使用 Web 的缺点如下。

- ◎ 需要花费更多时间来学习标准；
- ◎ 依然需要注意浏览器的兼容性问题；
- ◎ 用 CSS 来实现某些表现反而比表格更为麻烦。

## 2.4 Web 标准三剑客

Web 标准是由一系列规范组成，由于 Web 设计越来越趋向于整体化与结构化，此前的 Web 标准也逐步成为由三大部分组成的标准集：结构(Structure)、表现(Presentation)和行为(Behavior)。对应的网站标准也分三方面：结构化标准语言、表现标准语言(主要包括 CSS)和行为标准。这些标准大部分由 W3C 组织起草和发布，也有一些是其他标准组织制定的标准。

### 2.4.1 什么是内容、结构、表现和行为

#### 1. 内容

“内容”就是制作者放在页面内真正想要访问者浏览的信息，可以包含数据、文档或者图片等。注意这里强调的“真正”，是指纯粹的数据信息本身，而不包含辅助的信息(如导航菜单、装饰性图片等)，内容是网页的基础，在网页中具有重要的地位。

#### 2. 结构

##### ◎ XML——可扩展标记语言

XML(Extensible Markup Language)目前遵循的是 W3C 于 2000 年 10 月发布的 XML 1.0。与 HTML 一样，XML 同样来源于标准通用标记语言(Standard Generalized Markup Language)，但 XML 是一种能定义其他语言的语言。XML 最初设计的目的是弥补 HTML 的不足，以强大的扩展性满足网络信息发布的需要，后来逐渐用于网络数据的转换和描述。

##### ◎ XHTML——可扩展超文本标记语言

XML 虽然数据转换能力强大，但是完全可以替代为时过早。因此，在 HTML 的基础上，用 XML 的规则对其进行扩展，得到了 XHTML。简单而言，建立 XHTML 的目的就是实现 HTML 向 XML 的过渡。

XHTML(The Extensible HyperText Markup Language)，在 2000 年年底国际 W3C 组织公布发行了 XHTML 1.0 版本，这是一种在 HTML 基础上优化和改进的新语言，目的是基于 XML 的应用。



### 3. 表现

#### ◎ CSS——层叠样式表

W3C 创建 CSS 标准的目的是用 CSS 取代 HTML 表格布局, 以及其他基于表现层的语言, 使站点的访问与维护更加容易。

### 4. 行为

#### ◎ DOM——文档对象模型

根据 W3C DOM 规范, DOM(Document Object Model)是一种 W3C 颁布的标准, 用于对结构化文档建立对象模型, 从而使得用户可以通过程序语言(包括脚本)来控制其内部结构。简单地理解, DOM 解决了 Netscape 的 JavaScript 和 Microsoft 的 JavaScript 之间的冲突, 给予 Web 设计师和开发者一个标准的方法, 让他们来访问站点中的数据、脚本和表现对象。

#### ◎ ECMAScript

ECMAScript 是 ECMA(European Computer Manufacturers Association)制定的标准脚本语言 (JavaScript), 目前遵循的是 ECMAScript262 标准。

## 2.4.2 DIV 与 CSS 结合的优势

传统的 HTML 标签里既有控制结构的标签(如<p>), 又有控制表现的标签(如<font>), 还有本意用于结构后来被滥用于控制表现的标签(如<table>)。结构标签与表现标签混杂在一起。使用传统方法制作网页的制作者往往会遇到如下问题。

### 1. 改版问题

例如, 需要把标题文字替换成红色, 下边线变成 1px 灰色的虚线, 那么制作者可能就要一页一页地修改。既增加了工作量, 又增大了出错的机率。

CSS 的出现, 一开始似乎就是用来解决“批量修改表现”的问题, 被制作者广泛接受了 CSS 属性包括控制字体的大小颜色、超链接的效果、表格的背景色等等。

### 2. 数据的利用问题

本质上讲, 所有页面信息都是数据。由于结构和表现混杂在一起, 所以装饰图片、内容被层层嵌套的表格拆分。

用 CSS 制作的话就只需要修改 CSS 即可, 而不用修改 HTML 文档, 可以使结构清晰化, 将内容、结构与表现分离。

## 2.4.3 如何改善现有网站

目前大部分的网页设计师依旧在采用传统的表格布局、表现与结构混杂在一起的方式来建立网站。学习使用 XHTML+CSS 的方法需要一个过程, 使现有网站符合网站标准也不可能

一步到位。最好的方法是循序渐进，分阶段来逐步达到完全符合网站标准的目标。如果你是新手，或者对代码不是很熟悉，也可以采用遵循标准的编辑工具，例如 Dreamweaver，它是目前支持 CSS 标准最完善的工具。

Web 标准的目标是实现网页结构、表现、行为的分离，达到最佳架构，提高网络可用性与用户体验。Web 标准中包括了众多技术标准，包括相似的 HTML 与 XHTML 等。以目前 Web 标准的技术框架看，用以下几个标准及方法进行网站构建是目前最为理想的选择。

### 1. 从 HTML 转向 XHTML

面向结构的 XHTML 设计语言在面向结构的设计思想上能带给用户超越 HTML 的实质性内容。面向结构的设计能够帮助页面适应更多终端的需要，对于不同的应用终端，如 PC、PDA、手机及其他产品，只要这些设备能够接受结构语言 XHTML，那就能对信息进行再设计重新发布以适用不同的终端需要。

### 2. 发挥 CSS 的作用

CSS 样式的设计意味着用户对网站整体风格的把握需要重新考虑，从视觉设计上为了达到最大限度的重用与合理的结构，需要统一的字体字号及排版形式，这些统一性的设计都有助于视觉设计上可用性的提升。对于网站的细节表现，如链接改变提示、链接区域、导航的操作感等，也是 CSS 在可用性上设计的目标，最终目的都是希望通过良好的设计创造更好的交互式网站以方便用户使用，为网站及用户创造价值。

对于现有的网站，要认真理解并履行以上两点的要求。将编程习惯改为 XHTML，工作中能适用 CSS 控制的部分就不要使用其他方法。这样对于整个网站的重构有很大的帮助。

## 2.5 关于 Web 标准的思考与争论

要将 Web 标准普及也不是件容易的事情，目前在互联网上还有许多的网站页面是采用 HTML 或者更老规范建立的网页需要转换到 XHTML，并且目前主流浏览器 IE 对 Web 标准的支持还不太完善，再加上众多设计师还不理解 Web 标准，依然在观望甚至反对。这些问题使得 Web 标准的推广工作任重而道远。

### 2.5.1 关于 Web 标准的优点

首先，最为明显的就是用 Web 标准制作的页面代码量小，可以节省带宽。这只是 Web 标准附带的好处，因为 DIV 的结构本身就比较简单，Table 布局的层层嵌套造成代码臃肿，文件尺寸膨胀。在通常情况下，相同表现的页面用 DIV+CSS 比用 Table 布局要节省 2/3 的代码，这是 Web 标准直接的好处。



一些测试表明，通过内容和设计分离的结构进行页面设计，使浏览器对网页的解析速度大大提高，相对老式的内容和设计混合编码而言，浏览器在解析过程中可以更好地分析结构元素和设计元素，良好的网页浏览速度使来访者更容易接受。



提示

在很多西方国家，由于 Web 标准页面的清晰结构、语义完整，利用一些相关设备能很容易的正确提取信息给残障人士。因此，方便盲人阅读信息也成为 Web 标准的好处之一。

## 2.5.2 关于布局

页面使用 Table 布局是现在比较常见的制作方法。但是由于 Table 布局将表现和内容混杂在一起，结构不清晰、内容不完整，不利于内容的重用。虽然用 Table 布局改版也很快，但这不是长远之计。透过现象看本质，Web 标准将内容与表现相剥离，所有样式、风格、布局等表现的东西独立出来，由 CSS 或者 XSLT 来单独控制，这样剥离后，改版才是真正的方便。而且“改版”并不仅仅是浏览器上的改版，同样的页面如果需要发布到手机上，符合 Web 标准的页面就只需要修改样式文件，而 Table 布局则需要完全重做，如果未来还需要再发布到网络电视或者其他新设备上，CSS 的效率一定比 Table 高。

由于一开始研究和推广 Web 标准的人士做的页面都比较“朴素”，因此引起了大家的误解，以为 Web 标准的页面就是简洁、轻图形、轻视觉效果的。实际上，用 Table 布局能够实现的页面效果，用 CSS 也基本上能实现。这个问题不需要多解释，看看国内外新建立的 Web 标准站点就清楚了。

## 2.5.3 关于浏览器兼容

Web 的目标是使用统一的技术与模式进行网站设计，从而提升网站可用性、改善网站结构、降低网站成本。使用统一的标准为未来的兼容打下基础。但是事事并非一帆风顺，Web 标准是 2000 年以后提出的新方法模式，W3C 希望以此标准改善网络表现层的面貌。标准的提出结束了 Netscape 与 IE 的争端，使它们在后来的版本中都开始使用统一的 Web 标准进行网页的表现。一直到现在，所有新推出的浏览器产品，包括 Mozilla、Firefox 及 IE 8.0 等无所不用 Web 标准作为浏览器的基础，但是也暴露了很多隐藏的问题。

首先是旧版本兼容的问题。IE 浏览器大概从 IE 5.0 开始才对 CSS 2.0 有了较好的支持，但是并非绝对完美。对于 CSS 中的盒模型原理、浮动模式等标准的定义并非严格执行，及部分 CSS 2.0 的属性在 IE 5.0 中完全没有效果，IE 5.5 时已经修复了一些问题，但直到 IE 6.0

才得到明显改善。截止到 2005 年底,一些网络调查的数据显示,网络浏览者中还有约 5% 的用户在使用 IE 5.0/IE 5.5 这些老式浏览器,虽然数量很少,但是如果真正做到对所有用户的完全兼容,还需要下一些功夫。

还有一个问题就是不同浏览器的表现。例如目前较流行的浏览器 IE 8.0 与 Firefox,二者由两家公司分别开发,在浏览器的核心架构上有着明显的区别,二者虽然都是以 Web 标准作为开发基础,但是由于架构及开发方式的不同,最终对于 Web 标准的展现上,难免会有部分区别,这也导致了一小部分 CSS 设计方式在两个浏览器中的表现会有所不同。这些问题都使得一些设计师对 Web 标准产生质疑,以及学习上的困难。不过随着时间的推移,一些问题已经开始得到改善,毕竟由浏览器带来的问题只是极少数,这些问题并不能掩盖住 Web 标准开发的优势。

将内容和设计分开,不用再担心未来的技术变革,无论是结构还是设计,都可随时替换和修改,不需要在混杂的信息和设计的代码中进行修改。

## 2.6 关于 Web 标准的误区

Web 标准看上去很诱人,但是对于刚刚接触 Web 的设计师来说却是很容易犯错误的,以至于使自己的学习进度停滞不前。

### 2.6.1 抛弃表格布局的思维方式

刚开始使用 DIV+CSS 布局的网页设计人员往往认为“DIV+CSS 布局就是将原来用表格的地方转换为 DIV,原来是表格嵌套,现在是 DIV 嵌套”,这是一个最大的误区。



提示

Web 标准的目的是分离内容和表现,因此制作者首先要做的事情就是抛弃传统的表格布局方式。

在页面里有的仅仅是内容没有修饰的情况下,它看上去就是一张单调的页面,其中有一些文字和图片(这里的图片是指内容中的图片,如新闻中的照片等)。这些文字和图片仅仅是依次罗列下来,没有任何样式。

将所有修饰的图片作为背景,然后加入表现,用 CSS 来定义每一块内容的位置、字体、颜色等。这样制作的页面才是内容与表现分离的。就是说,当你抽掉 CSS 文件,剩下的就是干净的内容。这样才能在文本浏览器中阅读,才能在手机、PDA 中阅读,才能随时修改 CSS 实现改版。





## 2.6.2 正确使用 XHTML 标签

XHTML 标签是用来定义结构的，不是用来做表现的。因此，建立良好的文档结构对不同的内容使用正确的 XHTML 标签是很重要的。在文档中合理地使用<p>等标签，不仅便于理解文档的内容，而且也便于 CSS 样式的编写。

XHTML 里有很多的标签，但是经常用到的也就是那几个，只要掌握这几个就可以了。下面——列举，如表 2-2 所示。

表 2-2 常用的 XHTML 标签

标签名	用法
<div>标签	没有什么特殊的意义，可以使用在很多地方，也就是说它可以装不同的内容。其正确的写法是大家都用来做布局之用，也有用来作为存放文章形成段落。实际上，这个做法并不是很好，因为作为文章的分段自然有一个特定的标签来用。那就是下面要讲的标签，不过用 DIV 来再整体的包住所有的段落，这是非常实用的。
<p>标签	这是一个有特定语义的标签，表示段落，是用来区分段落的。在大部分的浏览器中对 p 基本上都有一个上下的边距。但是没有首行缩进，因为首行缩进只是表示段落的方式，并不是一定或是必须的。所以在用 p 标签的时候如果需要针对 p 设定一下首行缩进，推荐在文章以外的地方不要使用 p，因为 p 是有一定语义的，如果用在别的地方会不合适。正确的写法是<p></p>必须要有封口。
<span>标签	这也是一个非常常用的标签，这个标签可以说与 DIV 很像，没有什么特定的意义，只是它是一个级联元素，不是块级元素。这个标签与 DIV 正好互补。
<ul><li>标签	这是一个列表，在列表中，除了 ul 还有 li。ul 通过 CSS 定义一样可以有 ol 的数字排序效果，所以一般不推荐使用 ol，有 ul 就可以了。ul 是块级的，它的子级 li 也是块级标签。正确的写法是<ul><li></li></ul>。li 标签是被 ul 标签包裹的，在 ul 标签里可以有无数个 li 标签，li 标签不能独立使用。并且 li 标签一定要封口，这不光是美观问题，对于后期的维护也很有好处。很多程序员都不喜欢把这个 li 封口。ul 列表的用处主要是列举出一维的，同一类型的数据。例如使用在菜单上，文章中列数的一些条例等。在列表中有一种特别的形式与 ul 是不一样的，那就是下面的 dl。
<dl><dt><dd>标签	这是一个很特别的三个标签的组合。这里的 dt 是指标题，dd 是指内容，dl 是包括它们的容器。正确的写法是<dl><dt></dt><dd></dd></dl>。在 dl 里可以有很多组的 dt、dd，当出现很多组的时候尽量是一个 dt 配一个 dd，如果 dd 中内容很多，可以在 dd 里加 p 标签配合使用。dl 列表是一个非常好的列表形式，可以多加利用。

(续表)

标签名	用法
<a>标签	表示链接，是一个特定属性标签，也是网页中最为神奇的标签。因为它才让无数的网页都连在了一起。正确的写法是<a href="" title="">。其中的 href 是表示目标地址，title 是鼠标悬停的提示文字，这是可有可无的，但是有必是利大于无。
<img>标签	这是图片标签，也是个特定属性的标签。正确的写法是<img src="" alt="" title="">。这里的 src 是目标地址，alt 与 title 是替换文字，alt 是 IE 特定的，title 是其他浏览器通用的。不过后面的反斜线一定要有的。
<h>标签	这是一个标题系列的标签，从 h1 到 h6，一共 6 个，主要是用来存放标题，也有一些把它用在其他地方，但是建议只使用这个标签控制标题。
<strong>标签	这个意思是着重，这是有语义的，作用也很简单。至于样式，是加粗着重，还是用色彩表明着重，那都是自行选择。正确的写法是<strong></strong>。
<em>标签	这个与 strong 很像，是表示强调。一般浏览器的默认值是斜体，使用方式与 strong 一样。正确的写法是<em></em>。

### 2.6.3 善于利用 CSS

灵活地运用 CSS，根据不同的选择器来进行 CSS 定义，将通用的样式写在外部 CSS 文件中，然后在页面内调用，同时还可以将不同的 CSS 定义分在几个文件中。

对于多次引用的样式可以用 class 来定义，不需要每个元素都定义 id；也不是一定要用 <div>，有的内容完全可以用 <p> 来代替，同样都是块级元素，一样有盒模型的 7 个参数，<div> 仅仅方便浮动。

### 2.6.4 “通过验证”并不是最终目的

Web 标准的本意是实现内容(结构)和表现分离，就是将样式剥离出来放在单独的 CSS 文件中。这样做的好处是可以分别处理内容和表现，也方便搜索和内容的再利用。

W3C 验证仅仅是帮助制作者检查 XHTML 代码的书写是否规范，CSS 的属性是否都在规范内。代码的标准化指的是 W3C 验证的第一步，不是说通过的校验，网页就标准化了。使用很多方法都可以骗过校验程序而得到通过验证的那个小图标。

让网页具有良好的结构、更快的浏览速度、更友好的界面以及对更多设备的支持才是最终目的。



## 2.7 本章小结

本章中介绍了使用 Web 标准制作页面的理由和特点,使用表格布局和使用 Web 标准布局的不同和联系。通过本章的学习读者要对为什么使用 Web 标准有所了解,了解使用 Web 标准制作页面的必要性,使用 Web 标准的优点和缺点,以及目前使用 Web 标准时会遇到的问题等等,为后面的具体制作学习打下基础。