

## 第 14 章 怪异的导航模式

本章主要学习内容：

任何一个网站不可缺少的就是导航菜单，缺少了导航菜单用户将会在网站中迷失方向，导航犹如航海员在海中的指南针，必不可少！





## 14.1

## 怪异导航曾经的实现方法

在网页中只有使用热区 `area` 标签才可以绘制奇形怪状的链接热区感应点，但使用热区 `area` 标签有以下几点不方便处理的问题：

- 热区 `area` 标签是只针对图片 `img` 标签才能使用的元素。
- 使用热区 `area` 标签需要了解在图片中热区形成的坐标方式。
- 在后期维护过程中，如果更换了图片，那么就需要重绘热区的坐标点。

不过以上这几个问题并不难处理，我们只需要利用 Dreamweaver 软件中的图片属性面板所拥有的功能绘制热区即可完成，如图 14-1 所示。

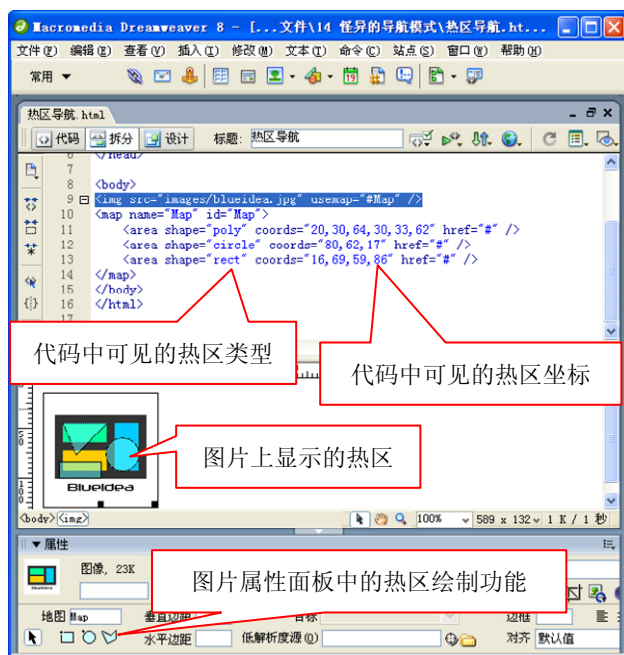


图 14-1 Dreamweaver 中绘制图片热区链接的功能

示例文件：光盘：\示例文件\14 怪异的导航模式\热区导航.html

虽然 Dreamweaver 能帮助我们解决绘制图片热区链接的问题，但图片热区无法实现锚点 `a` 标签所拥有的 `:hover` 伪类特性（对 IE 6 浏览器而言）。假如我们通过其他方式让热区 `area` 标签也拥有了 `:hover` 伪类特性，也还要考虑实现这个特性所需的工作成本。

## 14.2

## 如今实现怪异导航的方法

在追求 XHTML 代码结构语义化的前提下,使用热区 area 标签作为菜单导航也并非明智的选择,但在绘制地图(例如世界地图或者中国地图等)的时候我们可以考虑使用热区 area 标签。合理地使用 XHTML 中的标签,使其语义结构合理,还是需要大家在使用 XHTML 的过程中多多思考的。

网站的导航菜单最好的 XHTML 结构就是通过文字表达,即使在无 CSS 样式修饰(网页裸奔)的情况下我们也能很明确地了解每个锚点 a 链接形成的导航将会指向一个什么类型的网页中,如图 14-2 所示。

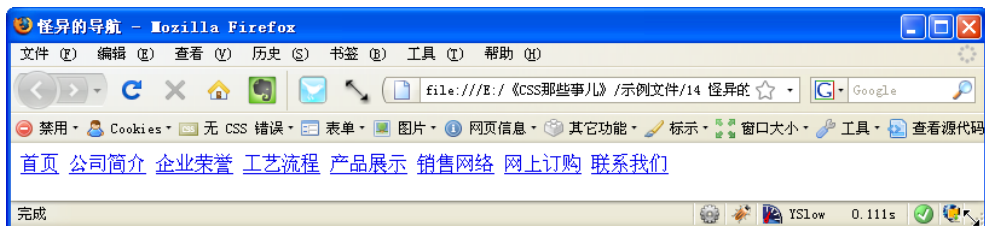


图 14-2 文字导航在无 CSS 样式修饰时的表现方式

对于文字导航,我们可以通过背景图片将其美化,即使是导航菜单的形状怪异(如梯形、菱形等)也还是能完成我们最终所需要的效果,如图 14-3 所示。

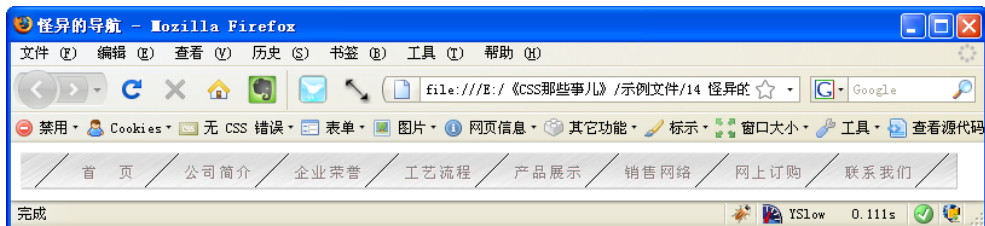


图 14-3 形状怪异的导航菜单

图 14-3 中的导航文字是被斜线包含着的,如果锚点 a 标签的: hover 伪类只是改变了文字的颜色或者样式那倒没什么可以担心的问题。但需要改变的是背景图片的样式,而锚点 a 标签的热区是矩形的,那该怎么处理呢?

关于这个问题不用担心,只要我们能善于利用 CSS 样式的几个属性,就可以实现如图 14-4 所示的效果,当鼠标悬停于文字导航上时,背景图片将会改变,而且是不规则的图片。

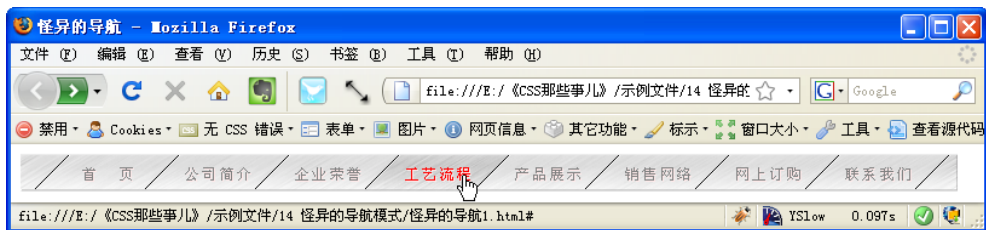


图 14-4 改变形状怪异的导航菜单的背景图片

实现这样一个怪异的导航并不困难，而且 XHTML 结构也十分简单，代码如下：

```
<map id="nav">  
  <div><a href="#" class="index">首页</a> <a href="#" class="intro">  
公司简介</a> <a href="#" class="honor">企业荣誉</a> <a href="#" class="flow">  
工艺流程</a> <a href="#" class="product">产品展示</a> <a href="#" class="sale">  
销售网络</a> <a href="#" class="buy">网上订购</a> <a href="#" class="contact">  
联系我们</a></div>  
</map>
```

将所有的锚点 a 标签写入一个 div 标签中，将 div 标签作为导航的容器。一个页面中的导航可以理解为站点中的地图，因此我们可以在 XHTML 结构中使用 map 标签作为整体结构的语义化要求。

但 map 标签有一个必要属性就是 id，该属性是必不可少的，因此在这个导航结构中我们使用了 nav 作为 id 名。

## 14.3

### 实现怪异导航的背景图片

XHTML 结构只实现了一个页面的部分内容，要将页面表现得更漂亮必不可少的是对图片的修改及 CSS 样式的美化。对于一个简单的 XHTML 结构而言，如何处理其背景图片是需要点小技巧的，处理得当时我们可以用最少的图片实现最好的效果。

本实例中所分解的怪异导航将会使用一张 PNG 格式的图片。因为 PNG 格式的图片不仅比 GIF 格式的图片小一点，而且也支持透明效果。将其处理成一张图片是因为我们可以利用 CSS Sprite 这个小技巧来处理背景图片的定位，而且一张图片也可以减少浏览器对服务器的请求量，减轻服务器的访问压力。如图 14-5 所示的图片是在 Photoshop 中处理时，添加了辅助线后的效果。

在图 14-5 中我们可以看到，导航在默认的情况下其背景图是一张整图，放置在顶端的位置，即背景图片使用的 background-position 是 top left (或者是 0px 0px) 的坐标位置。而当鼠标悬停时变化的背景图片都是不规则的图片，并且是有规律地存放在背景图片的不同位置上的。更需要注意的一点是，这些不规则的图片周围都是透明的。

只有透明才能更好地表现鼠标经过某个导航菜单时的效果。如果不需要透明的效果，那么我们可以将透明的部分与默认导航背景图片（即鼠标未悬停时的背景图片）结

合。具体怎么操作都是可以根据实际情况去分析并实施的，不能一概而论。

例如我们可以将图 14-5 中图片的排列方式改变，只要将透明的部分错开即可，如图 14-6 所示。

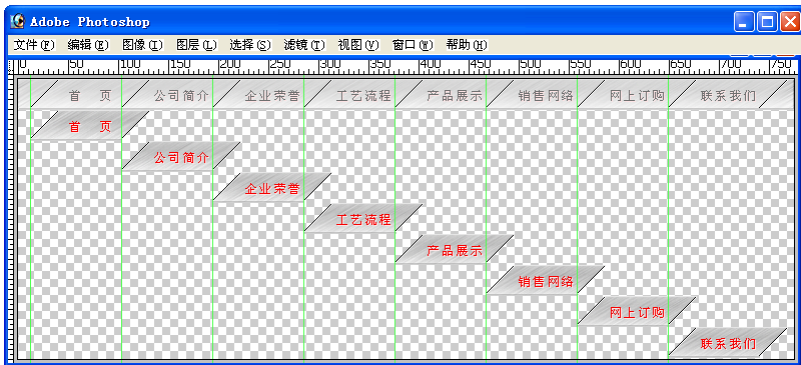


图 14-5 在 Photoshop 中处理怪异导航背景图片

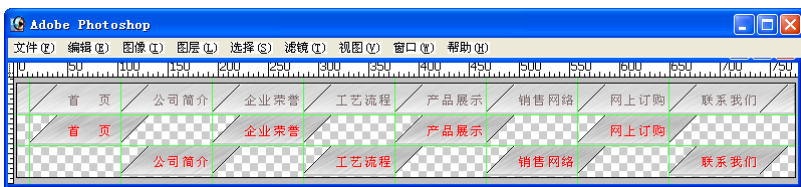


图 14-6 改变背景图的排列方式

利用 CSS Sprite 技巧实现的背景图片，图片的排列方式最终影响的是 CSS 样式的代码，以及背景图片中的坐标值的计算。

## 14.4

### 结合 CSS 样式调用背景图片

万事俱备，只欠东风，为了让读者更清晰地了解背景图片在使用 CSS Sprite 技巧时，其排列方式对 CSS 样式中背景图片坐标（background-position）计算的影响，我们将图 14-5 的最终 PNG 格式的图片命名为 bg\_nav1.png，将图 14-6 的最终 PNG 格式的图片命名为 bg\_nav2.png。

这两张图片无论调用哪一张，对于整体导航在初期的样式都是不会有影响的，都是调用了图片最顶部整体导航背景图片的部分。

看下面一段代码：

```
#nav * {  
    margin:0;  
    padding:0;  
} /* 设置#nav 容器内所有元素的内外补丁为 0，便于后期的细节调整 */
```



```
#nav div {  
    width:754px;  
    height:31px; /* 设置导航区域的宽度和高度属性值 */  
    overflow:hidden; /* 设置导航区域内的内容超过固定的区域时为隐藏 */  
    padding-left:21px; /* 导航的整体宽度是 775px, 为了便于背景图片更好地显示,  
增加左内补丁 21px */  
    background:url(images/bg_nav1.png) no-repeat 0 0; /* 设置导航的背景图  
片 */  
}
```

鉴于 XHTML 中每个标签在浏览器中的解析差别, #nav \* 中的 CSS 样式属性将导航容器内的所有元素内补丁及外补丁设置为 0, 便于后期细节部分的调整。这一步操作是任何一个网站都会使用的, 只是选择符使用的方式不同。本实例为了解决 #nav 容器内元素之间的差异, 使用了 #nav \* 这种组合选择符方式。

#nav div 是将 #nav 容器内的 div 标签作为显示导航内容的区域。在网页中 map 标签不是所有浏览器都能正确将其解析的, 因此不将其作为容器对待, 而是为了让导航更具有语义化才使用 map 标签的。

导航在页面中显示的实际宽度是 775px, 考虑到背景图片中斜线前面非导航菜单感应区的范围有 21px, 所以在 #nav div 中设置 754px 的宽度属性, 并利用内补丁能显示背景的特性增加了 padding-left:21px; 用于显示非导航菜单感应区的范围, 效果如图 14-7 所示。

导航的宽度等于背景图片的宽度, 为 775px, 但导航的高度属性值我们设置的 31px, 因为导航在正常情况下所显示的高度范围本来就应该只有 31px, 如图 14-8 所示。为防止背景图片超出导航的显示范围, 添加 overflow:hidden; 是必要的。

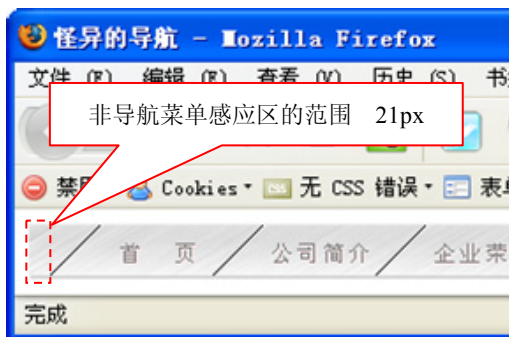


图 14-7 非导航菜单感应区的范围示意图

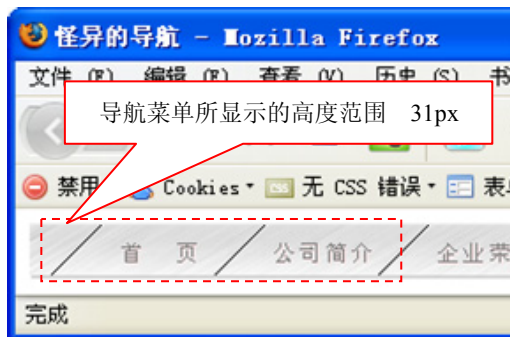


图 14-8 导航菜单所显示的高度范围

完成了怪异导航的雏形后, 在浏览器中我们将会看到, 导航菜单的热点区域还未设置, 并且文字在页面中显得很不好看, 跟背景图片中有阴影效果的文字相差甚远, 如图 14-9 所示。

示例文件: 光盘: \示例文件\14 怪异的导航模式\怪异的导航[雏形].html

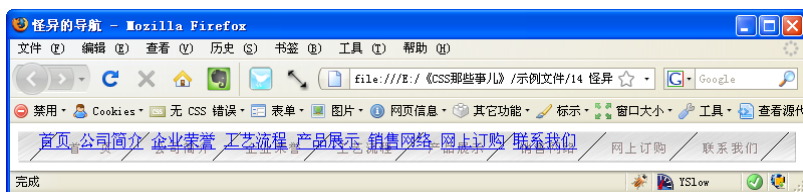


图 14-9 怪异导航的雏形

## 14.5

## 怪异导航细节优化

怪异导航的雏形完成后，需要实现的是为锚点 a 标签的链接感应区设置一个固定的宽度和高度属性值，效果如图 14-10 所示。

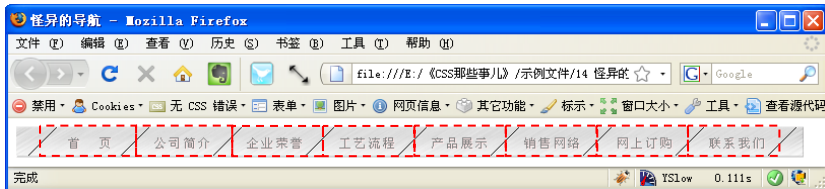


图 14-10 锚点 a 标签的链接感应区示意图

实现以上效果的代码如下：

```
#nav div a {  
    float:left; /* 为导航区域内所有锚点 a 标签设置左浮动，横向排列 */  
    width:92px;  
    height:31px; /* 设置锚点 a 标签的宽度和高度属性值 */  
    overflow:hidden;  
    text-indent:-9999px; /* 利用文本缩进属性将锚点 a 标签的内容全部隐藏 */  
    background:url(images/bg_nav1.png) no-repeat 1000px 1000px; /* 设置  
锚点 a 标签的背景图片，并为背景图片定位一个很大的数值，显示无背景图 */  
}
```

根据如图 14-10 所示的示意图，在设置锚点 a 标签的宽度和高度的属性值的同时，利用文本缩进属性将所有文本内容“推”出，使其消失在热点区域。

在锚点 a 标签中设置背景图片的属性，又将其 X 轴与 Y 轴坐标设置为 1000px，主要作用是便于后期激活该标签的: hover 伪类时，不需要再次调用该图片，而只需要直接修改相应的坐标位置即可。

在 XHTML 结构中我们针对每个锚点 a 标签添加了不同类名的 class，例如 class="index"，目的就是使针对不同类名的链接在激活: hover 伪类时可以调用不同坐标区域的背景图片。



看下面一段代码：

```
#nav div a:hover {
    position:relative;
    width:120px;
} /* 当鼠标悬停锚点 a 标签时，改变其容器的宽度并设置为相对定位，使浮动元素获得层叠属性 */
#nav div a.index:hover {
    margin-left:-10px;
    background-position:-11px -31px;
} /* 当内容为“首页”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
#nav div a.intro:hover {
    margin-left:-10px;
    background-position:-103px -62px;
} /* 当内容为“公司简介”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
#nav div a.honor:hover {
    margin-left:-11px;
    background-position:-194px -93px;
} /* 当内容为“企业荣誉”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
#nav div a.flow:hover {
    margin-left:-12px;
    background-position:-285px -124px;
} /* 当内容为“工艺流程”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
#nav div a.product:hover {
    margin-left:-15px;
    background-position:-374px -155px;
} /* 当内容为“产品展示”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
#nav div a.sale:hover {
    margin-left:-14px;
    background-position:-467px -186px;
} /* 当内容为“销售网络”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
#nav div a.buy:hover {
    margin-left:-15px;
    background-position:-558px -217px;
} /* 当内容为“网上订购”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
#nav div a.contact:hover {
    margin-left:-16px;
    background-position:-649px -248px;
} /* 当内容为“联系我们”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
```

在激活:hover 伪类的同时，设置 position:relative;使锚点 a 标签元素在原有浮动(float)属性的基础上增加层叠特性。只有增加了层叠特性的元素才不会在改变了宽度属性和负边距属性后影响到其他相邻的元素。

其中每个不同内容的锚点 a 标签负边距的属性值是不同的,而且 background-position 的坐标值也是不同的。这些数字都是需要制作人员在调整背景图片的时候精确计算才能设置的,否则会影响导航的最终效果。关于这点读者可以尝试修改其中相应的数值再在浏览器中浏览页面效果,看看数值的改变会对页面产生什么样的影响。

到此为止我们已经能够实现怪异的导航效果了,如图 14-11 所示为鼠标悬停于“工艺流程上”时,导航的变化效果。

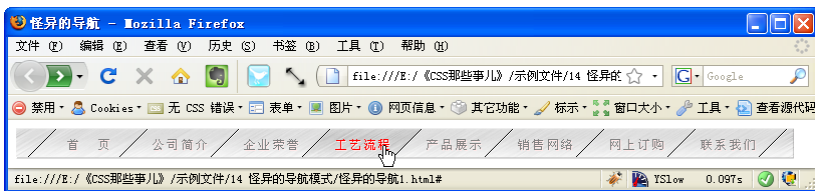


图 14-11 怪异导航的最终效果

示例文件: 光盘: \示例文件\14 怪异的导航模式\怪异的导航 1.html

## 14.6

### 改变思路实现怪异导航

前面我们提过,利用 CSS Sprite 技巧实现背景图片,背景图片中的图片排列方式将直接影响 CSS 样式代码的内容。为了体现这点,我们现在将背景图片改为 bg\_nav2.png,不再使用 bg\_nav1.png。代码如下:

```
<style type="text/css">
#nav * {
    margin:0;
    padding:0;
} /* 设置#nav 容器内所有元素的内外补丁为 0, 便于后期的细节调整 */
#nav div {
    width:754px;
    height:31px; /* 设置导航区域的宽度和高度属性值 */
    overflow:hidden; /* 设置导航区域内的内容超过固定的区域时为隐藏 */
    padding-left:21px; /* 导航的整体宽度是 775px, 为了便于背景图片更好的显示,
增加左内补丁 21px */
    background:url(images/bg_nav2.png) no-repeat 0 0; /* 设置导航的背景图
片 */
}
#nav div a {
    float:left; /* 为导航区域内所有锚点 a 标签设置左浮动, 横向排列 */
    width:92px;
    height:31px; /* 设置锚点 a 标签的宽度和高度属性值 */
    overflow:hidden;
    text-indent:-9999px; /* 利用文本缩进属性将锚点 a 标签的内容全部隐藏 */
```



```
background:url(images/bg_nav2.png) no-repeat 1000px 1000px; /* 设置锚点 a 标签的背景图片，并为背景图片定位一个很大的数值，显示无背景图 */
}
#nav div a:hover {
    position:relative;
    width:120px;
} /* 当鼠标悬停锚点 a 标签时，改变其容器的宽度并设置为相对定位，使浮动元素获得层叠属性 */
#nav div a.index:hover {
    margin-left:-10px;
    background-position:-11px -31px;
} /* 当内容为“首页”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
#nav div a.intro:hover {
    margin-left:-10px;
    background-position:-103px -62px;
} /* 当内容为“公司简介”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
#nav div a.honor:hover {
    margin-left:-11px;
    background-position:-194px -31px;
} /* 当内容为“企业荣誉”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通过 CSS Sprite 技巧显示背景图片 */
.....
</style>
```

最终的页面表现效果是相同的，但 CSS 样式中改变的部分却很多，尤其是 background-position 属性的属性值。

注：代码中省略了部分重复的内容，只是数值的改变，具体的代码读者可参考随书光盘中的实例。

在随书光盘中有相应的实例文件，读者可以参考。通过该实例我们不仅可以了解到如何实现怪异导航的实现方式，而且也明白了背景图片应该如何去运用，尤其是在处理 CSS Sprite 技巧时运用背景图片的方式。

示例文件：光盘：\示例文件\14 怪异的导航模式\怪异的导航 2.html

## 14.7

## 绝对定位方式实现怪异导航

当每个锚点 a 标签都有不同的 class 类名时，我们还可以直接使用定位（position）的方式控制每个导航菜单的位置。原理与使用浮动（float）的方式布局是一样的，不同的是使用定位（position）的方式需要准确表明每个坐标的位置。

看下面一段代码：

```
<style type="text/css">
#nav * {
    margin:0;
    padding:0;
} /* 设置#nav 容器内所有元素的内外补丁为 0，便于后期的细节调整 */
#nav div {
    position:relative; /* 设置相对位置，便于其子级元素有定位参照对象 */
    width:754px;
    height:31px; /* 设置导航区域的宽度和高度属性值 */
    overflow:hidden; /* 设置导航区域内的内容超过固定的区域时为隐藏 */
    padding-left:21px; /* 导航的整体宽度是 775px，为了便于背景图片更好地显示，
增加左内补丁 21px */
    background:url(images/bg_nav2.png) no-repeat 0 0; /* 设置导航的背景图
片 */
}
#nav div a {
    position:absolute;
    top:0; /* 将所有锚点 a 标签定位，顶部为 0 */
    width:92px;
    height:31px; /* 设置锚点 a 标签的宽度和高度属性值 */
    overflow:hidden;
    text-indent:-9999px; /* 利用文本缩进属性将锚点 a 标签的内容全部隐藏 */
    background:url(images/bg_nav2.png) no-repeat 1000px 1000px; /* 设置
锚点 a 标签的背景图片，并为背景图片定位一个很大的数值，显示无背景图 */
}
#nav div a:hover {
    width:120px;
} /* 当鼠标悬停锚点 a 标签时，改变其容器的宽度并设置为相对定位，使浮动元素获得层叠属
性 */
#nav div a.index {
    left:21px; /* 定义内容为“首页”的锚点 a 标签在绝对定位时居左的位置 */
}
#nav div a.index:hover {
    margin-left:-10px;
    background-position:-11px -31px;
} /* 当内容为“首页”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，并通
过 CSS Sprite 技巧显示背景图片 */
#nav div a.intro {
    left:113px; /* 定义内容为“公司简介”的锚点 a 标签在绝对定位时居左的位置 */
}
#nav div a.intro:hover {
    margin-left:-10px;
    background-position:-103px -62px;
} /* 当内容为“公司简介”的锚点 a 标签中的:hover 被激活时，通过负边距改变点的位置，
并通过 CSS Sprite 技巧显示背景图片 */
#nav div a.honor {
    left:205px; /* 定义内容为“企业荣誉”的锚点 a 标签在绝对定位时居左的位置 */
}
.....
</style>
```



设置#nav div 为相对定位,使其具有定位的属性,也可以使其子级元素在使用绝对定位时具备定位参照的对象。

导航中所有的锚点 a 标签都使用绝对定位,并且距顶部的位置都是 0px,因此我们可以将原有的浮动改为 position:absolute;,并设置 top:0;,然后针对每个不同类名的锚点 a 标签设置定位中的 left 属性,那么最终效果也是一样的。

注:代码中省略了部分重复的内容,只是数值的改变,具体的代码读者可参考随书光盘中的实例。

示例文件:光盘:\示例文件\14 怪异的导航模式\怪异的导航 3.html

## 14.8

### 又一种怪异导航模式

在使用 CSS 样式对页面布局进行修饰时,我们更多地需要的是对页面元素的分析及对 CSS 样式的理解。例如,当怪异导航是梯形时,并且相邻导航菜单之间都有叠加效果,当鼠标悬停在导航菜单上时,改变的不仅是导航菜单的图片,而且还有导航的叠加表现效果,如图 14-12 所示。

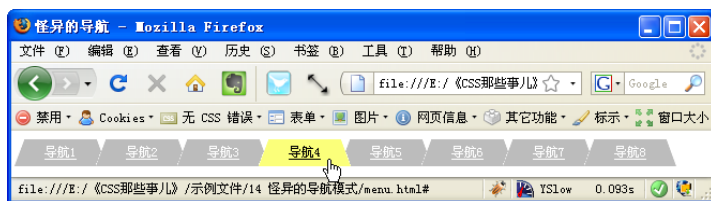


图 14-12 又一种怪异的导航菜单

其代码如下:

```
<map id="nav">
  <ul>
    <li><a href="#">导航 1</a></li>
    <li><a href="#">导航 2</a></li>
    <li><a href="#">导航 3</a></li>
    <li><a href="#">导航 4</a></li>
    <li><a href="#">导航 5</a></li>
    <li><a href="#">导航 6</a></li>
    <li><a href="#">导航 7</a></li>
    <li><a href="#">导航 8</a></li>
  </ul>
</map>
```

这次使用的导航并非由一系列的锚点 a 标签组成,而是由列表 li 标签组成的。这种怪异的导航菜单,其最大的特色就是当鼠标悬停时,最终改变的是将凸起的导航叠加在

周围的导航之上。可以想象一下，实现这样的效果肯定少不了定位（position）及层叠级别（z-index），但 XHTML 结构并没有针对每个锚点 a 标签或者列表 li 标签添加 class 类名，也就是说我们无法使用绝对定位（position:absolute;）而只能通过相对定位（position:relative;）的方式布局。

看下面一段代码：

```
<style type="text/css">
#nav * {
    margin:0;
    padding:0;
    font:normal 12px/25px "宋体";
} /* 设置#nav 容器内所有元素的内外补丁为 0，便于后期的细节调整 */
#nav ul {
    list-style:none;
    height:25px;
} /* 将列表修饰符取消，并设置其高度属性 */
#nav li {
    float:left; /* 为导航区域内所有列表 li 标签设置左浮动，横向排列 */
    width:86px;
    height:25px; /* 设置列表 li 标签的宽度和高度属性值 */
    margin:0 -5px; /* 使用负边距使每个导航之间产生叠加效果 */
    display:inline; /* 避免 IE 6 中的双倍间距 bug */
    text-align:center; /* 设置文字居中显示 */
}
#nav a {
    position:relative; /* 改变其容器的宽度并设置为相对定位，使浮动元素获得层叠属性 */
    float:left; /* 为导航区域内所有锚点 a 标签设置左浮动，横向排列 */
    width:86px;
    height:25px; /* 设置锚点 a 标签的宽度和高度属性值 */
    color:#FFFFFF;
    background:url(images/btn2.gif) center center no-repeat; /* 设置锚点 a 标签的文字颜色及背景图片 */
}
#nav a:hover {
    position:relative;
    width:86px; /* 当鼠标悬停锚点 a 标签时，改变其容器的宽度并设置为相对定位，使浮动元素获得层叠属性 */
    z-index:2; /* 提升锚点 a 标签的层叠级别 */
    color:#000000;
    background:url(images/btn1.gif) 0 0 no-repeat; /* 当鼠标悬停锚点 a 标签时，改变文字的颜色及背景图片 */
}
</style>
```

相比之前所说的怪异导航，这次我们使用的 CSS 样式代码精简了很多，都是可共用的 CSS 样式属性。重点就是 #nav li 中的外补丁（margin）、#nav a 中的相对定位（position:relative;）及 #nav a:hover 中改变相对定位的层叠级别（z-index）。



#nav li 中的外补丁 (margin) 的属性值使用左右负边距, 可以实现导航菜单的叠加效果, 如果没有负边距, 那么我们将看到如图 14-13 所示的效果。

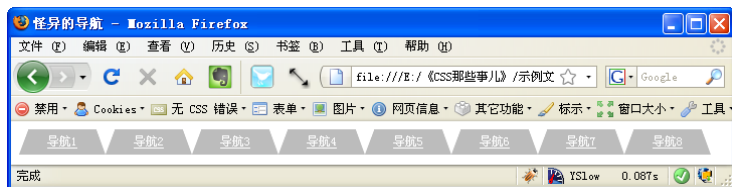


图 14-13 没有负边距时怪异导航的表现效果

在锚点 a 标签中设置相对定位 (position:relative;) 属性后, 只是在其原有浮动的基础上增加了层叠的效果。只有当锚点 a 标签具备了层叠的属性后, 我们才可以在其 :hover 伪类中设置 z-index 属性, 改变当鼠标悬停导航菜单时的层叠次序, 实现我们最终的页面效果。

示例文件: 光盘: \示例文件\14 怪异的导航模式\menu.html

## 14.9

### 小结

怪异的导航菜单无论怎么改变, 都是换汤不换药, 基本的原理都是一样的, 只是在细节的调整上会显得有些烦琐。通过本章的实例分解, 读者需要掌握的是负边距与层叠效果相结合时所带来的页面效果。只要掌握了这两者之间的微妙关系并善于利用, 就可以实现更多更酷的怪异导航菜单。