

# 14.4.4 设计定位页面

定位布局存在自身的优势和缺陷，优点就是比较精确，缺点就是无法适应网页内容的变化而自动调整自身区域大小。很多设计师一般不喜欢使用绝对定位布局，仅把它作为一种小技巧用在页面局部细节设计中。当不过适当使用定位布局，也能轻松应对复杂的定位问题。

本节将通过一个示例讲解定位布局的实战应用技巧。这里尝试使用定位法设计三行三列版布局页面。在浮动布局中，如果让主信息区域显示在页面中间栏，而次要信息和功能服务版块放置在页面左右两侧会比较麻烦，但是如果使用定位布局就很简单。

## 【操作步骤】

第 1 步，启动 Dreamweaver，新建网页，保存为 index.html，在<body>标签内输入以下代码，设计网页模板结构。

```
<div id="model">
  <div id="header">
    <h1>网页标题</h1>
    <h2>网页副标题</h2>
  </div>
  <div id="main">
    <div id="content">
      <h3>主信息区域</h3>
    </div>
    <div id="subplot">
      <h3>次信息区域</h3>
    </div>
    <div id="serve">
      <dl>
        <dt>功能服务区域</dt>
        <dd>服务列表项</dd>
      </dl>
    </div>
  </div>
  <div id="footer">
    <p>版权信息区域</p>
  </div>
</div>
```

第 2 步，在<head>标签内添加<style type="text/css">标签，定义一个内部样式表，然后准备定义样式。

第 3 步，梳理定位布局的设计思路：

设计标题行、版权信息行流动显示，中间行定义为包含块。然后就可以使用绝对定位设置次信息栏、功能服务栏定位到页面的左右两侧，而主信息栏显示在中间。

第 4 步，编写 CSS 布局的核心样式代码，所得效果如图 E14.1 所示。

```
<style type="text/css">
#main {/position:relative; /* 相对定位 */} * 定义定位包含框 */
#content {/* 主信息栏样式 */
margin-left:25%; /* 左侧边界，腾出空间为左栏显示 */
margin-right:20%; /* 右侧边界，腾出空间为右栏显示 */}
#subplot {/* 次信息栏样式 */
width:25%; /* 左栏宽度 */
position:absolute; /* 绝对定位 */
left:0; /* 靠左显示 */
top:0; /* 靠顶显示 */}
#serve {/* 服务栏样式 */
width:20%; /* 右栏宽度 */}
```

```
position:absolute;           /* 绝对定位 */
right:0;                     /* 靠右显示 */
top:0;                       /* 靠顶显示*/}
</style>
```

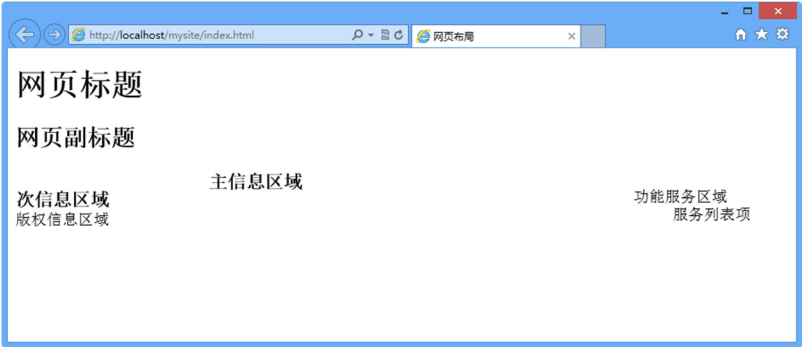


图 E14.1 网页定位布局初步样式

第 5 步，在上面初步设计思路的基础上，借助上节网页布局居中样式码，并适当为每个栏目定义背景色，以方便观察效果。详细代码就不再演示，效果如图 E14.2 所示。

```
<style type="text/css">
* {margin: 0;padding: 0;}
body { text-align: center; }
#model {width: 800px; margin-left: auto; margin-right: auto; text-align: left;}
#header { background: #FF00FF; }
#content {background: #FFCC00;}
#subplot {background: #00CCCC;}
#serve {background: #99CCFF;}
#footer { background: #FF99FF; }
</style>
```

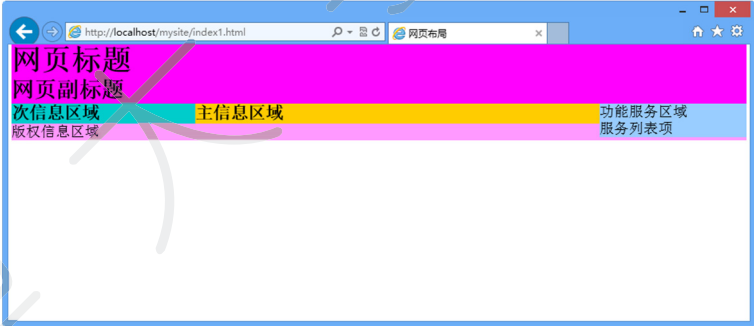


图 E14.2 美化网页定位布局

【提示】

通过观察，发现定位布局存在一个问题：当绝对定位的栏目高度延伸时，由于它已经脱离了文档流，所以就不会对文档流中相邻结构块产生影响，于是就出现了图中绝对定位栏目覆盖其他栏目的现象，如图 E14.3 所示。

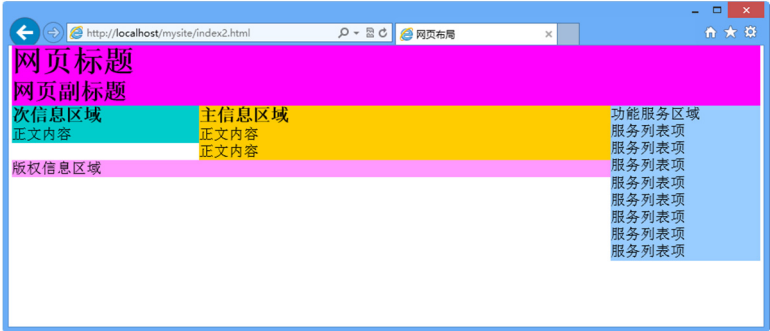


图 E14.3 定位布局的缺陷

解决方法:

方法一，被动预防。在预知绝对定位的栏目高度情况，也就是说绝对定位栏目的高度是固定的，则可以事先固定住绝对定位栏目的高度。

方法二，借助 JavaScript 脚本来动态调整绝对定位元素的高度。代码如下所示:

```
<script type="text/javascript">
window.onload = function(){
    var main = document.getElementById("main");
    var left = document.getElementById("subplot").offsetHeight;
    var right = document.getElementById("serve").offsetHeight;
    var middle = document.getElementById("content").offsetHeight;
    var height = 0;
    height = left - right > 0 ? left : right;
    height = middle - height > 0 ? middle : height;
    main.style.height = height + "px";
}
</script>
```

//定义即将控制的外框 div  
//获得 ID=subplo 的 div 高度  
//获得 ID=serve 的 div 高度  
//ID=content 的 div 高度  
// 定义变量来储存最大值  
// 数值比较  
// 数值比较  
// 设定外框 div 的高度为最大高度

最后所得的效果如图 E14.4 所示。

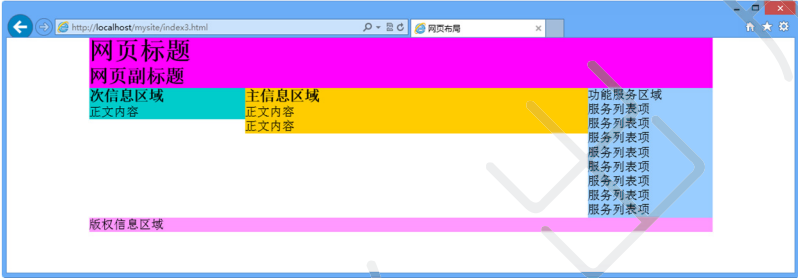


图 E14.4 使用脚本自动拉伸定位布局栏目

【提示】

使用定位布局可以轻松的把左右栏位置进行调换，仅需要调整栏目的定位方向和宽度，这样所得的效果如图 E14.5 所示。

```
#subplot {
    width:20%;
    right:0;
#serve {
    width:25%;
    left:0;
```

/\* 调整宽度\*/  
/\* 右对齐\*/  
/\* 调整宽度\*/  
/\* 左对齐\*/

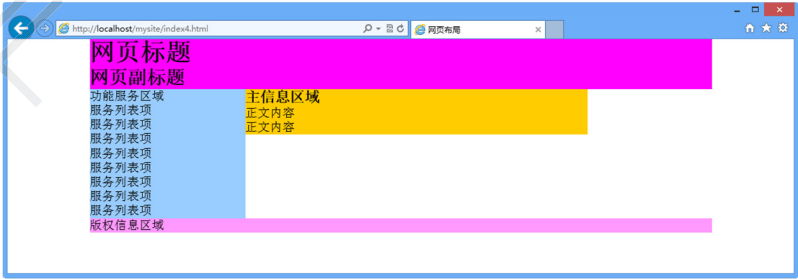


图 E14.5 轻松调整定位布局中栏目的位置

这对于浮动布局来说，难度就非常大的，因为改动一点就会影响整个页面的布局效果。