

# 基于 CSS 与 JavaScript 技术的 Tab 面板的设计与实现

李 冲,熊淑华,魏颖颖

(四川大学 电子信息学院,四川 成都 610065)

**摘 要:**随着因特网技术的发展,各式各样具有交互性能的网页应运而生。CSS 与 JavaScript 技术的结合使用可以设计出许多个性化动态网页效果。该方法设计与实现的网页具有规范、简洁、易于维护等特点,被广泛应用于 Web2.0 网页设计与开发中。传统方法设计的 Tab 菜单由于在切换过程中要重新加载整个页面,增加了服务器的负担。在分析传统的 Tab 菜单实现方法的不足的基础上,设计并实现了一种基于 CSS 与 JavaScript 技术的 Tab 面板效果。实践证明,该方法设计与实现的 Tab 面板不但简捷有效,而且可以降低因网页频繁刷新带来的负担。

**关键词:**CSS; JavaScript; Tab 面板; 网页特效

**中图分类号:**TP391.9

**文献标识码:**A

**文章编号:**1673-629X(2011)03-0028-03

## Design and Implementation of Tab Panel Based on CSS and JavaScript

LI Chong, XIONG Shu-hua, WEI Ying-ying

(Electronic Information Institute of Sichuan University, Chengdu 610065, China)

**Abstract:** With the development of Internet technology, various Web pages with interactive performance have emerged. Using the combination of CSS and JavaScript technologies, many personalized effects of dynamic pages can be designed. Because of standard, conciseness and easy maintenance, CSS and JavaScript are widely used in Web2.0 design and development. Because traditional methods of design Tab menu have to reload all the Web page in the process of switching, it increases the burden of server. In this paper, the disadvantages of traditional Tab menu design are discussed firstly, and then a CSS and JavaScript-based Tab panel is designed and implemented. It is proved that this design method of Tab panel is simple and efficient and releases the burden of Web page.

**Key words:** CSS; JavaScript; Tab panel; Web effects

## 0 引 言

互联网发展到今天,网页的设计与开发除了注重功能以外,更多的还要关注网页的效果,尤其是个性化动态效果<sup>[1]</sup>。传统的网页制作方法能实现的功能非常有限,缺乏动态性、个性化以及和用户的交互性能。CSS 与 JavaScript 技术的出现,使网页上能够实现许多单纯使用 HTML 语言无法实现的个性化动态效果。

Web 界面的设计趋势是缩短页面屏长,降低信息的显示密度,但同时又不能牺牲可视的信息量。在这种趋势下,Tab 这种交互元素成为了一个越来越普遍的应用<sup>[2]</sup>。Tab 是一个常见的交互元素——将不同的内容重叠放置在某一布局区块内,重叠的内容区里每次只有一层是可见的,用户通过鼠标点击或移动到内

容区所对应的标签上,来请求显示该层内容区<sup>[3]</sup>。Tab 效果可分为 Tab 菜单和 Tab 面板两种。在切换各个 Tab 页中的内容时不刷新浏览器窗口,各个页面上的内容实际上都已经装载到页面当中,只是有些部分被隐藏了起来,只有被选中的 Tab 页中的内容被显示出来,这种方式制作的效果称为“Tab 面板”;在切换各个 Tab 页中的内容时会刷新浏览器的窗口,实际上是更换到了另一个新的 HTML 页面,这种方式制作的效果称为“Tab 菜单”<sup>[4]</sup>。

传统方法实现的 Tab 菜单存在诸多不足之处,如网页加载速度慢等,CSS 与 JavaScript 技术的出现,克服了以上缺点。CSS(Cascading Style Sheet)层叠样式表,是一组格式设置规则,用于控制 Web 页面的外观,通过使用 CSS 样式设置页面的格式,可以将页面的内容与表现形式相分离<sup>[5]</sup>。JavaScript 是一种客户端脚本语言,它可以增强静态 Web 应用的功能,从而为 Web 页面提供动态的、个性化的内容,通过 JavaScript 还可以与用户进行交互<sup>[6]</sup>。文中基于 CSS 与 JavaS-

收稿日期:2010-08-02;修回日期:2010-11-19

基金项目:四川省科技支撑计划项目(002050 5501111)

作者简介:李 冲(1985-),男,广西柳州人,硕士研究生,研究方向为多媒体通信;熊淑华,副教授,硕士研究生导师,研究方向为多媒体通信。

cript 技术设计与实现 Tab 面板效果。

## 1 基于 CSS 与 JavaScript 技术的 Tab 面板的设计

### 1.1 总体设计思路

由于 Tab 菜单在切换过程中要刷新整个页面,当网页的内容比较多时,加载的时间会比较长,而且网页中大部分保持不变的内容也重新加载,加重了服务器的负担,因此其应用受到了限制<sup>[6]</sup>。而 Tab 面板是在页面中已经装载了全部内容,只是有些部分被隐藏起来,只有被选中的显示出来,切换过程中不刷新页面,不需要经过服务器的处理,减少了服务器的负担,因此得到广泛的应用<sup>[7]</sup>。要实现 Tab 面板效果,仅仅通过 CSS 来实现是非常困难的,而且实现的功能也非常有限,必须通过与 JavaScript 的结合使用<sup>[8]</sup>。

首先,搭建 TTML 构架,建立两个区块容器分别用于盛放标签和内容,设计了四个标签分别为:NBA 球队、网站设计、著名大学和湖人球员,分别对应四个不同的内容重叠放置在内容区块中,重叠的内容区块中每次只有一层是可见的,其他三层都被隐藏起来,通过鼠标移动到内容区所对应的标签上来请求显示该内容。其次,通过设置 CSS 来改变页面的样式,设计了当鼠标经过某一个标签时对应的标签的背景变为蓝色,字体变为红色,其他三个标签背景都为蓝色,字体为黑色。最后,通过设计 JavaScript 来实现内容的切换,定义了一个数组分别用于盛放四个不同的内容,当鼠标经过某一个标签时即触发 tab(n) 函数,将相应的 n 值传入,在函数体内实现内容的切换,被触发标签相应的内容被显示出来,其他内容隐藏,从而实现了 Tab 面板效果。

### 1.2 搭建 HTML 结构

首先建立一个

区块容器,用于设置该容器的整体样式,在该容器内再设置两个小

区块容器,上面一块是标题菜单栏,建立四个子菜单分别为:NBA 球队、网站设计、著名大学和湖人球员;下面一块为内容区,用于盛放对应菜单的内容,并为每一个容器和项目都定义一个 CSS 类型,CSS 的引用类型采用外联引用方式。代码实现过程中,class(id) = “样式名”和 onmouseover = “JavaScript 函数名”分别表示调用 CSS 和 JavaScript 的内容<sup>[9]</sup>,具体代码如下所示:

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html charset=gb2312">
<title>tab 面板的实现</title>
<link rel="stylesheet" href="style1.css" type="
```

```
text/css">
</head>
<body>
<div class="container"> // 定义一个区块容器
<div id="container1"> // 用于盛放菜单栏的块容器
<ul id="menu"> // 定义一个四项目的列表
<li onmouseover="tab(0)" class="color2">NBA 球队</li>
<li onmouseover="tab(1)" class="color1">网站设计</li>
<li onmouseover="tab(2)" class="color1">著名大学</li>
<li onmouseover="tab(3)" class="color1">湖人球员</li>
</ul></div>
<div id="container2"> <ul><li>洛杉矶湖人队</li><li>休斯
顿火箭队</li><li>波士顿凯尔特人队</li></ul></div></div>
</body>
</html>
```

### 1.3 设置 CSS 样式

文中 CSS 的引用类型采用外联引用方式,在外部的 style1.css 文件中定义了各个 CSS 的属性<sup>[10]</sup>,具体实现代码及注释如下所示:

```
body {
text-align:center; // 定义整体内容居中
}
.container { // 定义大容器的长和宽
width:400px;
height:150px;
}
#container1 { // 定义容器的整体样式
padding:0px;
}
ul#menu { // 项目列表的整体样式
list-style-type: none; // 去掉项目符号
margin: 0px;
padding-left: 0px;
padding-bottom: 24px;
border-bottom: 1px solid #11a3ff;
font-size: bold 12px verdana, arial;
}
#menu li { // 设置各个项目列表的样式
display: block; // 区块元素
height:25px;
text-align: center; // 内容居中
float: left;
margin: 0px 2px 0px 0px;
cursor: hand; // 鼠标经过时指针效果为手型
padding:3px 10px 0px 10px;
}
.color1 { // 设置正常情况时菜单样式
background-color:a3dbff;
border: 1px solid #11a3ff;
}
.color2 { // 鼠标经过时菜单样式
```

```
background-color: #fff;
border-top: 1px solid #11a3ff;
border-left: 1px solid #11a3ff;
border-right: 1px solid #11a3ff;
color: red; // 字体变为红色
}

#container2 { // 定义第二个容器的内容显示样式
border-left: 1px solid #11a3ff;
border-right: 1px solid #11a3ff;
border-bottom: 1px solid #11a3ff;
height: 150px;
text-align: left;
padding: 20px 0px 0px 0px;
font-size: 14px;
}
```

#### 1.4 JavaScript 设计

JavaScript 函数是实现整个效果的核心部分, 函数设计的整体思想是: 在 JavaScript 中实例化一个数组 description, description[n] 分别用于存放四个菜单所对应的内容, 例如 description[0] 为对应第一个菜单 NBA 球队所对应的项目内容, description[1] 为对应第二个菜单网站设计所对应的项目内容等等。用 on-MouseOver="tab(n)" 实现在鼠标经过时触发 JavaScript 中 tab 函数并将 n 值传入函数, 用 container2.innerHTML = description[n] 语句将 container2 所对应容器的内容设置为 description[n] 的内容, 从而实现了各个内容的切换。具体实现代码及注释如下所示:

```
<script language="JavaScript">
var description = new Array(); // 实例化数组 description
description[0] = "<ul><li>洛杉矶湖人队</li><li>休斯顿火箭队</li><li>波士顿凯尔特人队</li></ul>";
description[1] = "<ul><li>PHP 即 Hypertext Preprocessor (超文本预处理器), 是当今最为火热的脚本语言</li><li>ASP 即 Active Server Pages, 它是微软开发的一种类似 HTML、Script 与 CGI 的结合体</li><li>JSP 即 Java Server Pages, 它是由 Sun Microsystems 公司于 1999 年 6 月推出的新技术</li></ul>";
description[2] = "<ul><li>清华大学</li><li>北京大学</li><li>四川大学</li></ul>";
description[3] = "<ul><li>科比-布莱恩特</li><li>安德鲁-拜纳姆</li><li>保罗-加索尔</li></ul>";
function tab(n) // 定义函数 tab
{
for (i=0; i<menu.childNodes.length; i++)
// 使用 for 循环遍历整个数组
menu.childNodes[i].className = "color1"; // 未被触发时菜单的样式
menu.childNodes[n].className = "color2"; // 鼠标经过时被触发菜单的样式
container2.innerHTML = description[n]; // -- 将内容区显示的内容变为数组 description[n] 所对应的内容 --//
}
```

```
</script>
```

## 2 基于 CSS 与 JavaScript 技术的 Tab 面板的实现

得到的整体 Tab 面板效果如图 1、图 2 和图 3 所示, 其中图 1 为初始打开页面或者鼠标放在第一个菜单“NBA 球队”时的效果, 图 2 为鼠标经过第二个菜单“网站设计”时的效果。图 3 为鼠标经过第三个菜单“著名大学”时的效果。

基于 CSS 与 JavaScript 技术采用显示/隐藏的方法实现了 Tab 面板效果, 鼠标经过某一标签时, 对应的内容被显示出来, 其他三个标签对应的内容隐藏。由于所有的内容事先都已加载到页面中, 因此, 在实现各个 Tab 切换时不需要刷新页面, 从而降低了服务器的压力<sup>[11]</sup>。实现的效果如图 1 与图 2 及图 3 所示。

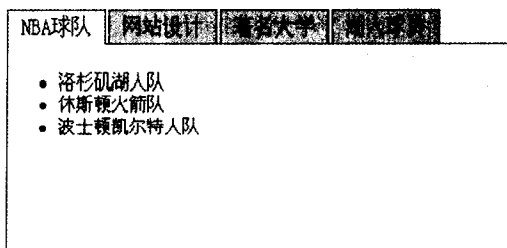


图 1 Tab 面板效果图 1

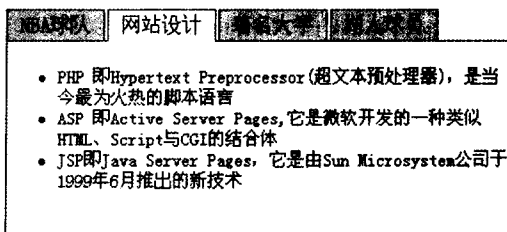


图 2 Tab 面板效果图 2

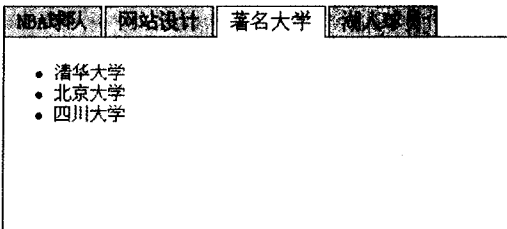


图 3 Tab 面板效果图 3

## 3 结束语

CSS 与 JavaScript 技术的出现, 给网站的设计与开发带来了活力和发展<sup>[12]</sup>。分析了传统的 Tab 菜单实现方法的缺陷, 并在此基础上, 详细介绍了一种基于 CSS 与 JavaScript 技术实现了 Tab 面板效果的设计方

(下转第 35 页)

一个长度为 10000 的文本对应词汇量为 1000 的词汇表中,后缀树的深度小于 20 的概率很大。

同样,从这些概率论分析来看,当  $l \geq 0$  时,随机变化值大于  $l$  的概率之和是随机变化的平均值,因此,重复子串的平均长度是

$$\sum_{l=1}^{L-1} \frac{(L-l)^2}{M^l} < \sum_{l=1}^{L-1} \frac{L^2}{M^l}$$

它也是一个上限值。当  $l < \log_M^{L^2}$  时,有  $\frac{L^2}{M^l} > 1$ ,但是概率值永远不会大于 1,因此,重复子串的平均长度可以精确地表达为:

$$\sum_{l=1}^{\log_M^{L^2}} 1 + \sum_{l=\log_M^{L^2}}^{L-1} \frac{L^2}{M^l} < \sum_{l=1}^{\log_M^{L^2}} 1 + \sum_{l=\log_M^{L^2}}^{\infty} \frac{L^2}{M^l} = \log_M^{L^2} + \frac{M}{M-1}$$

正如  $\frac{M}{M-1}$  一样,长度的平均值呈现对数性的小,

因此,搜索请求能够在确定的距离内得到回答。

## 5 结束论

DST 方法反映了词汇之间的语义顺序,因此它支持拥有大规模分布式文档的 P2P 网络的高效搜索,DST 覆盖网具有可扩展性、快速性和负载均衡性。它进行关键字序列搜索的时间花费与被搜索的串的长度呈准线性关系。由于它只通过关键字-节点函数与 DHT 覆盖网进行交互,因此它不依赖任何特殊的 DHT 覆盖网。DST 方式适用于任何提供绘图函数的下层重叠网。这些工作是我们努力建立支持知识网络应用的分布式语义覆盖网的一部分。

### 参考文献:

- [1] 何丽,韩文秀.一种基于后缀树的 Web 访问模式挖掘算法[J].计算机应用,2004,24(11):68-70.

(上接第 30 页)

法。实践证明用该方法设计 Tab 面板是可行的,该方法简单实用,为网站的页面节约了空间,减轻了服务器的负担。

### 参考文献:

- [1] 三扬科技飞思科技产品研发中心.大道 PHP 开发与实战[M].北京:电子工业出版社,2009.
- [2] Jensen S H, Moller A. Type Analysis for JavaScript[J]. STATIC ANALYSIS,2009,5673:238-255.
- [3] 温谦.CSS 设计彻底研究[M].北京:人民邮电出版社,2008.
- [4] Wilton P, McPeak J. JavaScript 入门经典[M].第3版.北京:清华大学出版社,2009.
- [5] 胡孟杰.JavaScript 动态网页开发案例指导[M].北京:电子工业出版社,2009.

- [2] Steinmetz R, Wehrle K. P2P 系统及其应用[M].北京:机械工业出版社,2008:80-85.
- [3] Wen Yuanfeng. A Novel Distributed Index Approach for Service Discovery in MANETs[C]//IEEE International Conference on Parallel and Distributed Systems. [s. l.]: [s. n.], 2008:415-421.
- [4] Jiang Yan. A Suffix Tree Based Handwritten Chinese Address Recognition System[J]. IEEE, 2007,3(1):292-296.
- [5] Kale A. A New Suffix Tree Similarity Measure and Labeling for Web Search Results Clustering[J]. Engineering and Technology, 2009,24(19):856-860.
- [6] 包小源,宋再生,唐世渭,等. SuffIndex——一种基于后缀树的 XML 索引结构[J]. 计算机研究与发展,2004,41(10):1793-1799.
- [7] 吴国庆. 对等网络技术研究[J]. 计算机技术与发展,2006,16(7):100-103.
- [8] 张建宇,马皓,廖唯浆,等. P2P 系统测量关键技术研究[C]//全国网络与信息安全技术研讨会论文集(下册),全国网络与信息安全技术研讨会. 中国山东青岛,2007:619-625.
- [9] 冯冰洁,杨天奇. 后缀树聚类算法在元搜索引擎中的应用[J]. 微计算机信息,2010,27(11):1-3.
- [10] 葛健,王国仁,于戈. 后缀树的并行构造算法[J]. 计算机科学,2004,31(5):96-99.
- [11] 曾本胜,廉玉忠,李世取. 后缀树的平均高度[J]. 运筹与管理,1996,8(4):9-13.
- [12] Harren M, Hellerstein J M, Huebsch R, et al. Complex Queries in DHT-Based Peer-to-Peer Networks[C]//Proc. First Int'l Workshop Peer-to-Peer Systems, Lecture Notes in Computer Science, US. (IPTPS '02). [s. l.]: [s. n.], 2002:242-250.
- [13] 孙名松,刘杰,李胜利. 基于语义划分的 P2P 搜索技术研究[J]. 计算机技术与发展,2010,20(8):67-71.

- [6] 李焯. 别具光芒 DIV+CSS 网页布局与美化[M]. 北京:人民邮电出版社,2007.
- [7] 郑俊生,姜敏. 一种基于 Div+CSS+JavaScript 的网页布局特效研究[J]. 电脑知识与技术,2008(17):1556-1558.
- [8] 周利荣. 基于 JavaScript 的动态网页制作[J]. 电脑学习,2005(3):11-13.
- [9] 赵清华. 基于 DIV+CSS 的网页布局技术应用研究[J]. 现代计算机,2010(5):140-142.
- [10] 胡波. JavaScript 的动态网页制作探讨[J]. 软件导刊,2006(9):71-73.
- [11] 熊迪. 结合 CSS 和 JavaScript 技术实现动态选项卡菜单[J]. 湖北广播电视大学学报,2009,29(2):155-156.
- [12] 周广田. 用 JavaScript 创建交互式 web 页面[J]. 信息技术,2001(2):35-36.