

DHTML

动态网站建设

高 成 主编 董长虹 姚德忠 副主编

杨连池 编 著

NATIONAL DEFENSE INDUSTRY PRESS

国防工业出版社



动态网站建设丛书

DHTML动态网站建设

高 成 主 编

董长虹 姚德忠 副主编

杨连池 编 著

国防工业出版社

·北京·

图书在版编目(CIP)数据

DHTML 动态网站建设/高成主编.—北京:国防工业出版社,2002.6

(动态网站建设丛书)

ISBN 7-118-02793-6

I.D... II.高... III.超文本标记语言,DHTML - 主页制作-程序设计 IV.TP393.092

中国版本图书馆 CIP 数据核字(2002)第 004516 号

国防工业出版社出版发行

(北京市海淀区紫竹院南路 23 号)

(邮政编码 100044)

北京奥隆印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 31¼ 725 千字

2002 年 6 月第 1 版 2002 年 6 月北京第 1 次印刷

印数:1—3000 册 定价:42.00 元

(本书如有印装错误,我社负责调换)

编 委 会

主 编： 高 成

副主编： 董长虹 姚德忠

编委会成员：

潘时龙 高海滨 朱 奇 马 迪 黄志正

郭兴成 陈 洁 崔益亮 张乐乐 张华弟

杨连池 贾立峰 王 震 余 杰 崔 桦

谢 华 郭明玄 张 龙 张建宁 张雪莲

廖 俊 李耀华 李品一 李元瑾 李 瑾

何 熙 王 剑 杨广军 赵 阳 汪 洋

前 言

我们曾经赋予互联网太多的价值，太多的梦想，甚至希望有朝一日能够建立自己的网络帝国，成为新一代的杨致远，成为网络时代的比尔·盖茨。但自从纳斯达克股市跌宕开始，梦想似乎已经破灭，网络时代似乎进入了冬季。

然而，错的并非互联网，而在于操作者。互联网仍然是互联网，网络并没有消亡，而且随着其他产业的发展而不断进步。网络已经渗透到社会中，影响着生活的方方面面。

网络最基本的功能在于信息的传输。网页正是充当着信息传输的角色之一。信息的发布、交流均可以在网页这样的平台上进行。

与其他的技术一样，随着社会整体水平的提高，原有的静态的信息发布方式很难满足人们的要求。动态的网页表达成为新的需求。而 DHTML 正是在这样的背景下产生的。它能动态地展现内容，使表达手段更加饱满，表达效果更加完美，并能满足交互式的信息传送。更加人性化的信息表达效果使得动态网页技术得到广泛的认可和推广。

在平常，大多数人还是用“所见即所得”的主页编辑软件进行主页的制作。无可否认，这些对于大家快速完成主页制作起到了不可磨灭的作用。简单的操作即可做出漂亮的页面，甚至是十分美妙的——这当然令人鼓舞。然而，毕竟软件只是封装了一些现成的功能，数量和灵活性都十分有限，很难满足进一步的发展需要。而且庞大的代码往往导致 web 文档变得臃肿不堪。这就需要手工修改代码。

真正的高手期待着手工编程的挑战，这样的工作将更加具有创造性。你可以随心所欲地编写自己的网页，充分发挥想象力，更能充分体会手工编程的乐趣！

正是基于这样的想法，我们编写了本书。目的在于让广大的读者早日掌握动态网页技术，从中体会手工编程的乐趣！在此，对给予本书编写提供大量帮助的海龙工作室表示感谢。

在书中我们尽量采用实例进行讲解，以期达到更好的效果。

主页作为一种艺术自然离不开设计！依靠纯粹的技术组合不可能制作出赏心悦目的主页，设计意识的融入将使您的主页制作水平提高到一个新的档次！所以在最后我们加入了有关设计艺术中的一些常识和基础知识。

总之，希望这本书能给您带来收获！

编 者

目 录

第 1 章 DHTML 简介(概述)

第 2 章 HTML4.0

2.1 新特点	6	2.9.1 用 frame 实现分帧	46
2.2 基本概念	7	2.9.2 链接指向	46
2.2.1 URI	7	2.9.3 分帧的不足之处	48
2.2.2 相对、绝对路径	8	2.9.4 浮动帧	50
2.2.3 HTML 结构	10	2.10 表单	53
2.3 头元素<HEAD>	11	2.10.1 创建表单	54
2.3.1 Link 元素	11	2.10.2 文字和密码的输入	55
2.3.2 Meta 元素	12	2.10.3 单选框和复选框	56
2.3.3 Title 元素	14	2.10.4 隐藏表单	57
2.3.4 Base 元素	14	2.10.5 图像坐标	58
2.4 文字与页面设计	15	2.10.6 列表框	58
2.4.1 字体控制	15	2.10.7 其他标记	60
2.4.2 颜色设置	17	2.10.8 表单的提交与处理	63
2.4.3 字符样式	18	2.10.9 表单技巧	64
2.4.4 段落格式化	18	2.11 媒体对象	67
2.4.5 预先格式化	21	2.11.1 嵌入多媒体文本	
2.4.6 Marquee	23	(EMBED)	67
2.4.7 背景设置	24	2.11.2 插入背景音乐	68
2.5 列表	26	2.11.3 插入视频剪辑	68
2.6 使用图像	31	2.12 前景展望	70
2.6.1 基本格式	31		
2.6.2 图形与文字的对齐			
排列	31		
2.6.3 图文之间的距离设置	32		
2.7 超链接的设置	33		
2.7.1 超链接基础	33		
2.7.2 图像映像	34		
2.8 使用表格	36		
2.9 使用框架	44		

第 3 章 层叠样式单 CSS

3.1 CSS 概述	72
3.2 定义 CSS 样式	74
3.2.1 选择器	74
3.2.2 DIV 和 SPAN	77
3.2.3 伪类和伪元素	78
3.3 运用 CSS 方式	79
3.4 文字属性	83
3.5 文本属性	84
3.5.1 行距	85
3.5.2 对齐方式	85
3.5.3 段落的首行缩进	86

3.5.4 字母的大小写	86	4.1.3 JavaScript 程序运行 环境	145
3.5.5 文字装饰	87	4.2 JavaScript 与 HTML	146
3.5.6 鼠标技巧	88	4.2.1 与 HTML 的结合	146
3.5.7 滚动条技巧	91	4.2.2 JavaScript 的运行	147
3.6 背景和颜色	93	4.3 基本知识	150
3.6.1 文字的颜色	93	4.3.1 变量	150
3.6.2 背景的颜色	93	4.3.2 数据类型	152
3.7 文字列表	95	4.3.3 运算符和表达式	152
3.7.1 list-style-position	95	4.4 流程控制	153
3.7.2 list-style-image	95	4.4.1 条件语句	153
3.7.3 list-style-type	95	4.4.2 循环语句	160
3.8 布局	96	4.5 函数	164
3.8.1 背景图像	96	4.5.1 函数的定义	165
3.8.2 边距及边框属性(容器 属性)	96	4.5.2 函数的调用	165
3.8.3 浮动元素	99	4.5.3 内置函数	168
3.8.4 元素定位	100	4.6 JavaScript 对象	170
3.9 滤镜	108	4.6.1 什么是对象	170
3.9.1 调整透明度	109	4.6.2 对象操作语句	171
3.9.2 移动模糊(Blur)	113	4.6.3 对象的属性和方法	172
3.9.3 颜色过滤(Chroma)	115	4.6.4 定义对象	173
3.9.4 设置阴影(DropShadow)	115	4.6.5 对象方法的引用	174
3.9.5 光晕(Glow)	117	4.6.6 Date()日期对象	177
3.9.6 灰度、X 光和反转滤镜	118	4.6.7 String()字符对象	178
3.9.7 遮蔽效应(Mask)	121	4.6.8 Math()数学对象	183
3.9.8 灯光效果(Light)	123	4.6.9 Array()数组对象	192
3.9.9 对称变换(FlipH、FlipV)	124	4.7 文档对象模型(DOM)	202
3.9.10 Wave 滤镜	124	4.7.1 Navigator 对象	203
3.10 动态滤镜	127	4.7.2 Window 对象	206
3.10.1 RevealTrans 滤镜	127	4.7.3 位置和历史属性	213
3.10.2 BlendTrans 滤镜	129	4.7.4 文档对象	216
3.10.3 动态滤镜效果综合	130	4.8 事件响应	221
3.11 跨浏览器编程	140	4.8.1 什么是事件	221
第 4 章 脚本语言		4.8.2 事件处理	221
4.1 JavaScript 的特点	143	4.8.3 事件运用	223
4.1.1 什么是 JavaScript	143	4.9 高级技巧:Cookie	229
4.1.2 JavaScript 和 Java 的 区别	144	4.9.1 什么是 Cookie	230
		4.9.2 Cookie 的特性	230
		4.9.3 Cookie 限制	231

4.9.4 创建 Cookie	231	5.3 文字效果	295
4.9.5 读取 Cookie	232	5.4 图片特效	338
4.9.6 删除 Cookie	234	5.5 鼠标特效	360
4.9.7 设置“有效期”	234	5.6 信息类	380
4.9.8 Cookie 路径和域	235	5.7 浏览器相关	390
4.9.9 Cookie 的应用举例	235	5.8 颜色相关	410
第 5 章 实例展示		5.9 代码生成、调试类	418
5.1 时间日期	241	5.10 超酷实例	431
5.2 页面导航	263	5.11 其他	476
		附录 快速参考	485

第 1 章 DHTML 简介（概述）

HTML 以其语法简单精练、易学性和通用性强而被广泛接受，继而开创了一个全新的网络时代。但是链接容易断开，格式有限，缺少灵活性等缺点也使得人们开始寻求解决的办法。其中比较常见的就是对 HTML 进行修改扩充。以更少、更快捷的页面下载，提供更加丰富多彩的图形和数据为目标的 DHTML 应运而生。它在最新的 HTML4.0 的帮助下崭露头角，逐渐被人们接受。

那何为 DHTML？DHTML 就是动态的 HTML（Dynamic HTML），它是 IE4.0 上增加的一组新特性，通过它可以来动态地改变 web 页面上各元素的位置、内容及显示风格，尤其是可以像在 Photoshop 中一样对图像动态地实施滤镜（Filter）及转换（Transition），从而使 Web 页面表现出多媒体的显示效果。它具有如下几个特点：

- 高效：所有这些 DHTML 代码都是在浏览器上解释执行，避免了频繁地与服务器连接，减少了网络传输。
- 交互性能加强：对用户的许多操作在本地就可作实时处理，从而得到更快的用户响应。
- 界面更丰富：用代码可控制 Web 页面上的所有元素，使页面制作者可随心所欲地表达自己的构思。

☞ 注意：

DHTML 并不是一种新的语言，也不是一种专门的技术，而是多种技术的组合。

DHTML 在保持了一些熟悉的语言的同时扩展了 HTML，Netscape 和微软分别在发布了各自的 4.x 浏览器的同时推出了所谓的动态 HTML（DHTML）。DHTML 有三个最主要的优点，即动态样式、动态内容和动态定位。动态样式能使开发商改变内容的外部特征而不强制用户再次下载全部内容。动态内容可以使开发人员改变显示在一页上的文本或图像，以便内容能够交互式地对用户的鼠标和键盘操作做出响应。动态定位则让页面制作者以自动方式或对用户的操作做出响应的方式移动页面上的文本和图像。

在页面上的表现手法主要包括以下几部分：HTML4.0、层叠样式表（CSS）文档对象模式（DOM）和脚本。

1. HTML4.0

1997 年 12 月，W3C 颁布了 HTML4.0 的正式规范。其中包括许多增强特性，如大表格的分段显示，带有固定标题的可以滚动的表格，以及对长表格打印的更好支持。对 HTML 格式的增强侧重于使这些格式更灵活。例如新的 Button 标记使得 Form 可以拥有

除 Submit 和 Reset 按钮之外更多的按钮;“accesskey”属性可以支持快捷键;Input 标记的“accept”属性允许制作者指定有效内容。另外,字符集也有所增强,正规的 HTML4.0 字符集在仍旧使 HTML 文档保持与 SGML 的一致性的同时,向非西欧语言扩展。

2. CSS

CSS 可以控制 HTML 语言编写的文档的显示,CSS 可以使页面布局和格式比 HTML 更精确。CSS 的新版本(CSS2)已经发布,W3C 关于 CSS2 的报告草案增加了专门介绍多媒体样式表部分。HTML 文档的听觉描绘将有助于为弱视用户存取 Web 内容提供方便。它还可以用于其他场合,比如在汽车内使用,在家庭娱乐系统上表演和教授单词的发音等。

CSS 的有关视觉描绘的规范介绍了相对和绝对位置问题。这些规范规定了 HTML 文档中内容的二维布局的规则,以及如何在三维中排列内容。

3. DOM

文档对象模式(Document Object Model),它给 HTML 文档定义了一个与平台无关的程序接口。使用该接口可以控制文档的内容、结构和样式。Web 开发人员借助 DOM 可以在其 Web 页中引入动态和交互式内容,而不必依赖于 Web 服务器来提供新的内容或改变现有内容的显示方式。

4. 脚本

脚本可以说是 DHTML 的核心。DHTML 所有的动作的实现都是靠脚本来完成的。我们知道动态网页的最大特点,就是将浏览器和网页内容对象化,让设计者制作网页时,可以在 HTML 文件中加入脚本程序。这样,当网页下载到浏览器时,浏览器就会根据脚本程序所定义的条件,调用浏览器对象或是控制网页的内容,产生动态的效果。在 DHTML 中常见的脚本语言主要是 Netscape 的 JavaScript 与微软的 JScript 和 VBScript。

为了使 DHTML 的各个部分成功地彼此交互作用,这些部分必须相互兼容。浏览器必须识别 HTML 和 CSS 语法,DOM 必须展示 HTML 和层叠样式表单元。脚本语言必须把浏览器作为主机,而且脚本必须能够在控制对象和集合属性以及调用其方法的同时对对象事件做出响应。

微软使样式、位置和内容产生动感的方式是把所有的 HTML 标记作为元素处理,让标记制作者直接控制其属性,并允许页面上的文件和图像动态回流。Netscape 的方式是展示较少的元素,并着重依赖于 HTML 内容层。制作者用层标记或 CSS 位置坐标来建立这些内容层。通过用 JavaScript 改变各层的属性,页面制作者在给页面装入内容后可以获得动态效果。开发人员采用 Netscape 的 Visual JavaScriptPro,可以从元素模板向 Web 拖放 HTML、Java 和 JavaScript 元素。他们也可以从元素模板拖放第三方 JavaBeans 和 COBRA 服务。开发人员还能够以虚拟的方式为两种元素建立基于事件的链接或界定属性值以使它们协调一致。程序包通过使用包含在软件包中的 JavaScript 部件支持 NetscapeEnterpriseServer3.0 中的数据库连接性,支持 Oracle、Informix、Sybase 或 ODBC 数据源。定制属性编辑器能使开发人员为特定数据提取,以交互方式建立一个 SQL 陈述。

字体是浏览器之战的又一个方面。可下载的字体系允许制作者为页面上的文本确定精确的字体,而不管在用户的电脑上是否已装入这种字体。Netscape 的方法是依靠一个与 Web 页相配合的字体定义文件。微软的方法是将 CSS 表示法扩展为参考字体样式,此方法还能支持微软的 TrueType 字体。

数据绑定和多媒体效果是另一个问题。只有微软的 DHTML 操作方式支持这些功能。数据绑定便于 Web 冲浪者简单快速地与页面中的高速数据缓存器交互作用，因为本地高速缓存器中的筛选和排序操作不要求向服务器往返传送。并且，页面作者可以通过样式表、内置样式属性设置和脚本应用实现筛选和变换功能。一共有 14 个筛选程序，如 **Blur** 和 **FlipH**，可以给 HTML 内容增加多媒体效果。Web 开发人员还可以为输入或消除 URL 调用变换功能。标准的目的在于为各部分建立一个参考，这样浏览器的制造商就需要至少满足这些要求。至少对于某些核心功能，DHTML 代码需要使所有的浏览器遵从标准要求。随着浏览器制造商致力于为客户提供最好的浏览器，我们可以使语言标准不断扩展，在核心功能以外演变出不同的功能。

遗憾的是这两个公司对选用的部分都有其专用扩展方式。尽管他们声称自己的产品绝对服从标准，但事实上两家浏览器却经常互不兼容。

因此，为了表达的信息和形式更能遵循原意，有时还得针对不同的浏览器进行编程。下面便是很好的例子。见图 1-1。

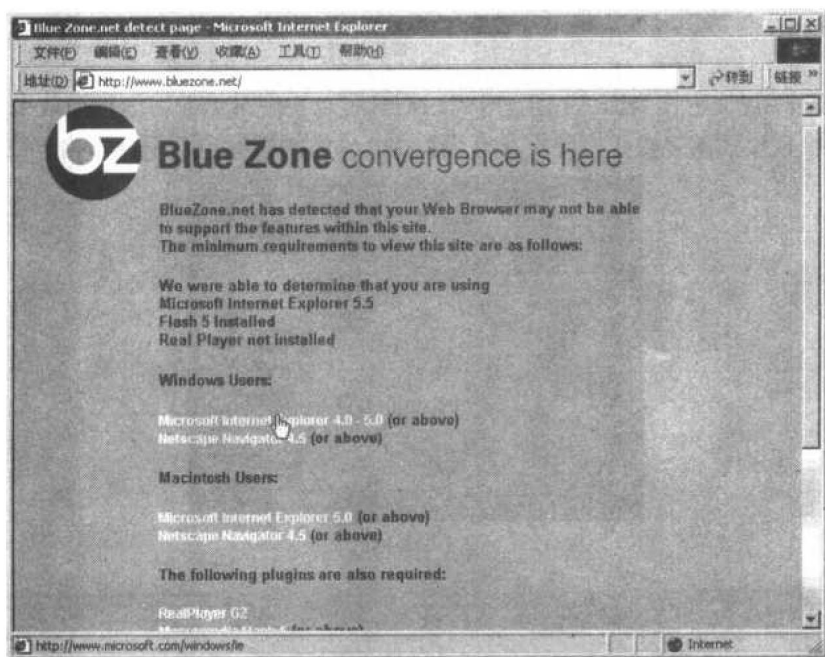


图 1-1 针对性主页编程

该主页设置了好几个“入口”：适合 IE 和 NETSCAPE 的浏览器的，适合 WINDOWS 操作系统的，适合 Macintosh 操作系统的。

目前，W3C 正着手发布 DHTML 部件的规范。当它完成其工作时，浏览器制造商或许就会有一套可以遵循的开放式标准，从而使各种浏览器实现兼容。

作为对 HTML 的发展，DHTML 的功能当然更加丰富，更加强大。它使得设计出复杂的、动态的、交互式的主页成为可能。下面请看一个综合使用 DHTML 的例子。该例子中的导航条模拟微软操作系统中常见的菜单形式。见图 1-2。

在网页中利用动态滤镜技术很容易实现多媒体效果。例如对一张星球图片（如图 1-3 所

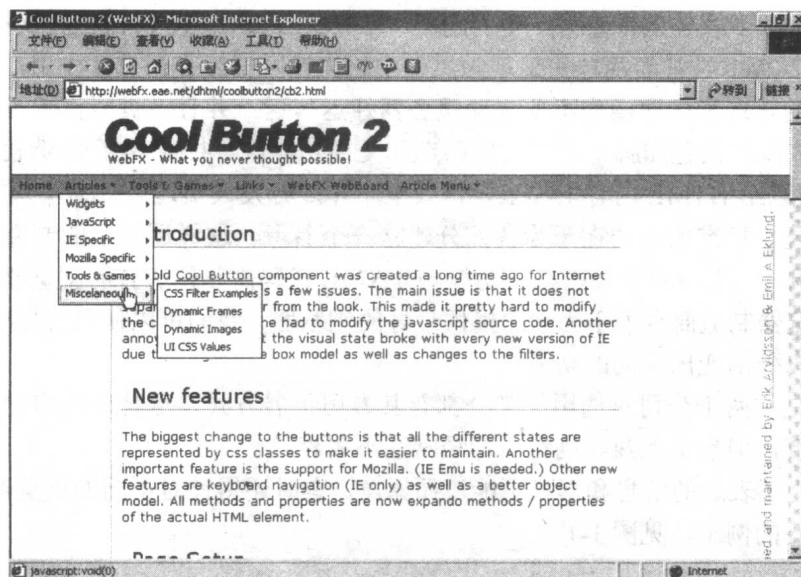


图 1-2 模仿微软菜单

示)进行滤镜作用, 让它作为雷达扫描背景, 这样就可以实现常见的雷达扫描效果(见图 1-4)。这比在页面中直接插入多媒体文件更具有优势: 文件占的空间小, 不会影响下载速度。



图 1-3 原始图片

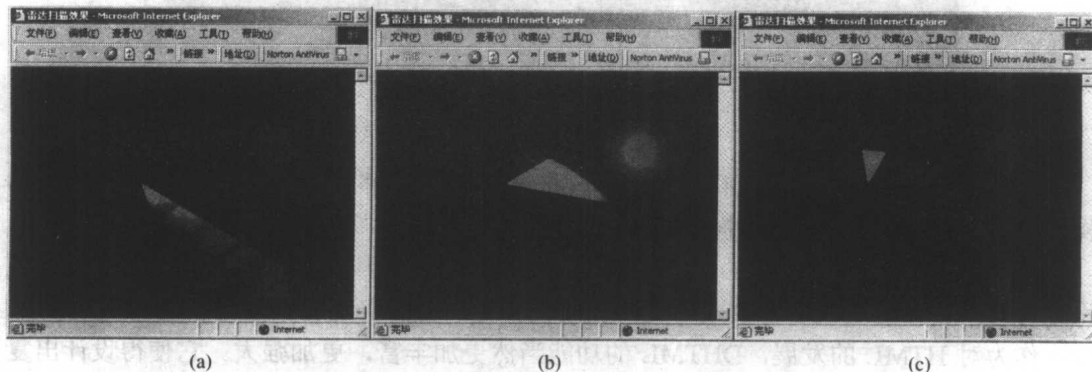


图 1-4 类似多媒体的滤镜效果

总之, 在 DHTML 世界中, 有着无穷无尽的奇妙效果。它将给你的网页带来质的变化。

第 2 章 HTML4.0

当前，主页制作软件越来越“傻瓜化”，“所见即所得”的主页制作方式似乎使得网页制作除了创作的灵感以外，不需要任何技术，甚至连主页制作的基础——“HTML”都可以抛开。但是，这种观点忽视了一个很简单的事实：初学算术时，学生得背“乘法口诀”，而非一下子拿起计算器，甚至运用电脑进行计算。同样，我们也有必要学好 HTML。况且 HTML 的语法简单易学。

正如古人所言，“授之鱼不如授之渔”。唯有掌握了 HTML，才能够更加主动地了解到自己能够做什么、怎么做，就可以尽量避免无谓的失误，易于检查出失误所在。毕竟，软件不能代替人的思考，软件也有不足之处，比如容易产生一些无用的代码。为了使网页的下载更加迅速，代码更加简洁，有时就得手动修改代码，而这种能力的具备的前提就是掌握 HTML 知识。

掌握了 HTML，就可以对其内幕更加了解，这也是一种乐趣。另外，制作主页的乐趣，也许不是用傻瓜化的软件做出一个主页，而是手写代码的创作过程。对 HTML 的不屑一顾，将是一种不明智的选择。

本章将介绍作为 DHTML 的重要组成部分的 HTML4.0 的相关知识。

在制作主页之前，要先考虑一些问题，比如对象定位：是儿童，还是年轻人；确定主题：是娱乐性质的，还是学习性质；风格定位：是欢快的，还是严肃的；版面设计：是否分帧，怎样分。比如面向儿童的网站，风格自然要欢快的，色调要丰富多彩。下面的网站（见图 2-1）很能体现这一点。

该网站针对儿童活泼的特点，采用丰富的色彩和儿童喜欢的卡通形象来表达主题。

网页设计的有关技巧将融入平时的讲解中，并在附录中进行专门论述。

在写代码之前，请注意不同的浏览器对 HTML 的支持问题。这其中问题可能是浏览器本身滞后于 HTML 地发展步伐，也可能是不同的浏览器所采用的标准不同。比如你使用了新的 HTML 元素，那么，旧的浏览器将不能识别而将它看作文本内容；或者我们编写主页时只根据微软的 Internet Explorer 浏览器支持的标准进行，而没有注意到所写的代码在 Netscape 的浏览器中是否也正常显示，那么，就会给 Netscape 的浏览器用户带来麻烦。所以，编写时一定要弄清这些问题，多在不同的浏览器，不同的操作系统（如果可能的话）中测试网页。虽然，HTML 的发展也尽量考虑到这些问题，以期“世界大同”；但现在毕竟存在这样的问题，所以还得注意这一点。当然，如果对主页的适应性要求较低，这些问题可以不考虑。

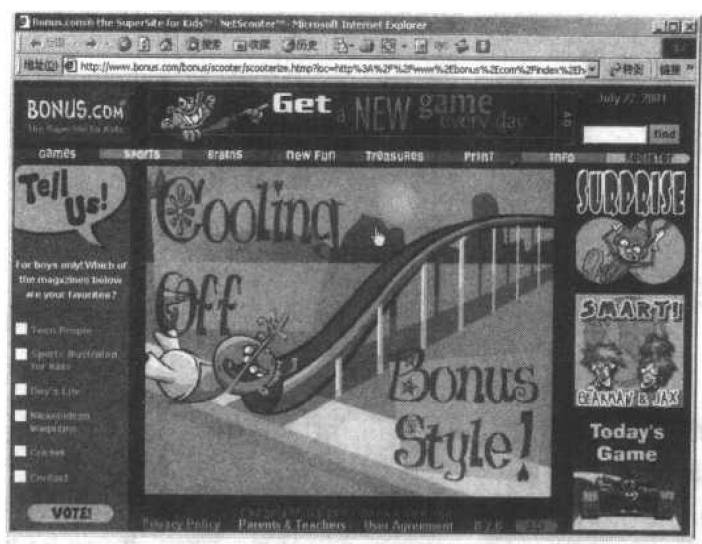


图 2-1 网站设计定位

2.1 新特点

作为新的版本，HTML4.0 并非以前版本（HTML3.2）的修修补补，而是有了质的飞跃。它将 HTML 语言扩展到了全新的领域，例如增加了对层样式菜单（CSS）、脚本语言（javascript 和 VBscript）、框架结构、内嵌对象、丰富的表格、多国语言文字、形式（比如从右到左的希伯来语）的支持；丰富了表格形式，增强表单功能，赋予残疾人使用网络的权利。

1. 国际化

这个版本的 HTML 在设计过程中得到了国际化领域专家的帮助，因此可以用各种语言编写文档并易于在世界范围内发布。

设计过程中的一个重要步骤是采用了 ISO/IEC:10646 标准作为 HTML 的字符集。它是处理国际化字符、文本方向、发音和其他语言问题的最具包容性的标准。

HTML 现在在文档内提供对不同语言的更强大的支持。这就为搜索引擎提供了更有效的文档索引功能、更高质量的排版格式、更好的文本到发音的转换、更好的连字符连接等。

2. 可访问性

既然网络社区在增长，而且它的成员的能力和技术各不相同，那么底层技术满足他们的特殊需要是很重要的。HTML 已经被设计成可以被有身体缺陷的人访问的网页。HTML 4.0 涉及到的便于访问的方面包括：

- 文档的结构和表现更好地区分开来，这就鼓励使用样式表而不用 HTML 表现元素和属性。
- 更好的表单，包括附加的存取键、按语意分类表单控件、SELECT 选项和活动标签。
- 标记作为内嵌对象的文本描述的能力（使用 OBJECT 元素）。
- 一个新的客户端图像映射机制（MAP），允许把图像和文本链接集成在一起。
- 替代文字与图像（包含在 IMG 元素内）和图像映射（包含在 AREA 元素内）一起出现。

- 支持所有元素的 title 和 lang 属性。
- 支持 ABBR 和 ACRONYM 元素。
- 与样式表一起使用的更多的目标媒体(tty, braille, 等)。
- 更好的表格, 包括标题、列组、和实现不可见渲染的机制。
- 表格、图像、框架等的长名描述。

在设计中考虑到可访问性因素不仅可以使信息交流更加顺畅还有其他的益处: 设计良好的、把结构和表现分开的 HTML 文档会更容易地采用新技术。

3. 表格

通过表格, 设计者可以更好地控制结构和布局。定义列宽度的能力可以逐渐显示表格, 而不必等到全部表格被绘制完。

4. 混合文档

HTML 现在提供在 HTML 文档中嵌入常见媒体对象和应用程序的标准机制。OBJECT 元素(与 IMG 和 APPLET 一起)提供了在文档内包含图像、视频、声音、数学、特殊应用程序和其他对象的功能。

5. 样式表

样式表简化了 HTML 标记并且减轻了 HTML 对表现形式的责任。它们使设计者和用户都可以控制文档的表现形式 - 字体信息、对齐方式、颜色等。

样式信息可以为单独的元素或一组元素定制。样式信息可以定制在 HTML 文档内或以外部样式表的形式出现。

把样式表与文档联系起来的机制独立于样式表语言。

在样式表出现之前, 制作者很少能控制页面绘制过程。HTML 3.2 包括一些属性和元素, 它们提供了对对齐方式、字体大小和文本颜色的控制。制作者也使用表格和图像进行页面布局。用户需要较长时间才能升级他们的浏览器, 意味着对这些特征的使用还要持续一段时间。然而, 既然样式表提供更强大的表现机制, WWW 协会将最终逐渐淘汰很多 HTML 的表现元素和属性。

6. 脚本

通过脚本, 制作者可以制作动态网页(例如, 反应用户填充内容的"smart forms")和使用 HTML 建立网络应用程序。

把脚本包含在 HTML 文档中的机制独立于脚本语言。

7. 打印

有时, 制作者希望读者不限于只打印当前文档。当文档成为一个工程的一部分时, 它们之间的关系可以被描述为使用 HTML LINK 元素或使用 W3C 的资源描述语言(Resource Description Language - RDF)。

2.2 基本概念

2.2.1 URI

它是 Universal Resource Identifier 的缩写, 意为“通用资源标识”。网络上的所有资源,

包括 HTML 文档、图片、视频剪辑、程序等都有一个能够被 URI 编码的地址。URI 包含三个部分：

- 被用来访问资源的统一命名规则分配体系。
- 资源宿主机器的名称。
- 以路径的形式给出的资源名称。

常见的命名分配体系有 HTTP 协议、mailto、FTP 协议三种。在访问一个 HTML 文档时，在地址栏输入的 URI 地址正是利用这三部分。比如要访问清华大学主页中的中文版，只要输入网址：<http://www.tsinghua.edu.cn/chn/index.htm>，它说明通过 HTTP 协议，能够在主机名为：www.tsinghua.edu.cn 通过路径“/chn”可以访问到命名为 index.html 的文档（中文版）。如果访问一个网站的 ftp 资源，只需把“http”改为“ftp”，即更换传输协议。又比如要定义一个电子邮件的地址，可以这样写：

<mailto:xiaoyang6@263.net>

“mailto”作为 E-MAIL 命名分配体系。

在 HTML 中，以上三种命名分配体系可以用<A>来实现，比如：

Please mail to me

or visit my homepage
if you have any questions.

☞ 注意：

URI 区别于 URL：URL（Uniform Resource Locator）就是统一资源地址，是 URI 命名体系规范中的一个分支，范围比 URI 小。顾名思义，URL 就是在网络上找到需要的资源的“路径”，上面所举正是 URL 的例子。

另外，链接不一定非得是 web 数据（如扩展名为.html，.asp 等等的文件）！比如.doc（word 文档）也是可以的。比如事先在 C:\Documents and Settings\Administrator\My Documents\目录下建立一个链接：中国农业 vs WTO的链接。那么，它打开时将显示如图 2-2 所示的效果。



图 2.2 非 WEB 文档链接

2.2.2 相对、绝对路径

1. 相对路径

假设主页目录在 H 盘（如图 2-3 所示），其中 MyHomepage 中有文件：index.html；interest 文件夹中有文件：in1.html 和 in2.html；文件夹 photo 中有文件：ph1.html。现在，就下面出现的几种情况进行分析：

(1) 从 MyHomepage 中的 index.html 的页面中建立链接到 interest 文件夹中的 in2.html：因为 interest 为 MyHomepage 的下一级文件夹，所以，可以这样定义：

```
<a href="interest/in2.html">链接到 interest 中的 in2.html</a>
```

在文件夹 interest 中，如果要从 in1.html 链接到 in2.html：

因为两个文件在同一个文件夹中，所以可以这样定义：

```
<a href="in2.html">从 in1.html 链接到 in2.html </a>
```

(2) 从 interest 文件夹中的 in1.html 的页面建立一个链接到 MyHomepage 中的 index.html：因为所要访问的目标文件在上一级的文件夹中，则定义如下：

```
<a href=" ../index.html">访问上一级文件夹中的文件</a>
```

(3) 从 in2.html 链接到 ph1.html,那该怎么做呢？其实，道理很简单。因为 interest 和 photo 同级，所以只需先从 interest 目录返回到上一级目录，即 MyHomepage，然后再到 MyHomepage 的下一级目录 photo。所以，可如下定义：

```
<a href=" ../photo/ph1.html">从这里链接到 photo 中的 ph1.html</a>
```

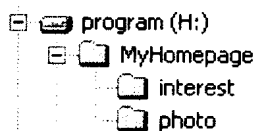


图 2-3 目录结构

2. 绝对路径

看图 2-3，如果要访问 interest 中的 index.html，则可以这样定义：

```
<a href="H:/MyHomepage/interest/index.html">采用绝对路径访问 index.html</a>
```

另外，绝对路径也用于网站之间的链接。比如：

```
<a href="http://www.sina.com.cn">新浪网</a>
```

提示：相对路径和绝对路径孰优孰劣：在主页上传时最好采用相对路径。在编写 html 文件时，对于关系明确的一组资源（比如在同一个目录中）建议采用相对地址，这件简化链接的标记，而且便于维护。比如需要将某个目录整个搬到另外一个地方，如果采用相对 URL 写的 html 文件用不着更新其中的 URL。但如果用绝对 URL 编写 html，那就不得不逐字修改每个链接中的 URL，这是一件很乏味也很容易出错的工作。对于各个资源之间没有固定的关系，比如所编写的 html 文件是介绍各免费资源的，它所指向的目标（网站）分布广泛，这时只能用绝对 URL 了。

3. URL 片断

在 HTML 文档的写作中，可以让 URI 使用“#”符号在同一份 HTML 文档中指向一个锚点，从而进行定位。这就是“URL 片断”。

比如有一个定位名称为“ComeHere”，那么可以通过以下代码可以实现定位：