



本书内容

- XHTML 语法剖析
- 滤镜与转场效果的应用
- ScriptLet 控件的设计
- HTC 的设计方法与内置行为
- XHTML+TIME 2.0 的应用
- 多媒体控制项的设置方法
- VML 的设计与应用

XHTML 高级应用 —— 网页动画与特效

何东隆 李美真 编著



清华大学出版社
<http://www.tup.tsinghua.edu.cn>



XHTML 高级应用 ——网页动画与特效

何东隆 李美真 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

XHTML 编程语言是 HTML 4.0 的改良版本,它以 XML 的规范来应用 HTML 4.01 的标记,现在已逐渐成为网页设计语言的标准。本书以大量的实例程序详细介绍了 XML 的语法和 HTML 的标记在网页动画与特效中的应用。主要包括:XHTML 语法剖析、滤镜的设置与应用、图片的展示方法、动画的原理及环境、HTC 的设计方法与内置行为、HTML+TIME 的应用、多媒体的应用、VML 的设计与应用以及 IE5.5 新增功能的详细范例和解说。

本书适用于网页设计人员、网页编辑与维护人员,也适用于略懂 XHTML 还想深入研究网页动画与特效的读者。

本书繁体字版书名为《进阶应用 XHTML 网页动画与特效》,由文魁资讯股份有限公司授权出版,版权属何东隆、李美真所有。本书简体中文版由文魁资讯股份有限公司授权清华大学出版社独家出版。未经本书原版出版者和本书出版者书面许可,任何单位和个人不得以任何形式或任何手段复制或传播本书的部分或全部内容。

北京市版权局著作权合同登记号:图字 01-2000-2245 号

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名:XHTML 高级应用——网页动画与特效

作 者:何东隆 李美真

译 者:清华文源

责任编辑:石宏宇

出 版 者:清华大学出版社(北京清华大学学研大厦,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印 刷 者:世界知识印刷厂

发 行 者:新华书店总店北京发行所

开 本:787×1092 1/16 印张:29.25 字数:711 千字

版 次:2001 年 8 月第 1 版 2001 年 8 月第 1 次印刷

书 号:ISBN 7-900635-45-9

印 数:0001~5000

定 价:49.00 元(含 1 张光盘)

序 言

1998 年 W3C 通过 XML 的审查后，迅速地改变了网页设计的概念与方法，各大软件厂商也陆续推出相关的产品，IE5.5 所支持的 CDF、TIME、VML 等等都是典型的例子。

XML 的制定是以数据的描述为主要的目标，而不是数据的显示，所以不能取代 HTML。为了彻底解决 HTML 的缺陷与限制，W3C 在 2000 年 1 月正式通过 XHTML 的推荐标准。

XHTML 以 XML 的规范，来应用 HTML4.01 的标记，可以看成是 XML 与 HTML 的综合体。除了比较严谨的语法规则外，与 HTML 的设计方法几乎完全相同，所以只要会用 HTML，就可以轻松地进入 XHTML 的领域，难怪大家都说 XHTML =HTML 5.0。

本书的主题包括 XHTML 语法规则的说明，IE5.5 添加功能的详细范例及解说。为你提供 一个开阔的“网页设计的新视野”。

何东隆 李美真

目 录

第 1 章 XHTML 概述	1
1.1 认识 XHTML	2
1.1.1 声明区	2
1.1.2 文件格式的定义	3
1.1.3 名称空间	4
1.1.4 文件的主体	5
1.2 文件的格式	6
1.2.1 XML	6
1.2.2 HTM	8
1.2.3 HTA	9
1.2.4 转换 HTML 文件	10
1.3 XHTML 的语法规则	12
1.3.1 字体与字号	12
1.3.2 标记	14
1.3.3 属性的设置方法	16
1.3.4 特殊符号的应用	18
1.4 自定义标记的应用	20
1.4.1 自定义标记的设计方法	20
1.4.2 自定义标记的样式	23
1.4.3 Style 的声明	25
1.4.4 自定义标记的对象模型	27
第 2 章 滤镜的设置与应用	29
2.1 滤镜的设置方法	30
2.1.1 在标记中声明滤镜	30
2.1.2 滤镜与自定义样式	32
2.1.3 参数的设置方法	34
2.1.4 控件的属性	35
2.1.5 滤镜的命名方法	38
2.1.6 交互式的滤镜	40
2.2 滤镜与背景	43

2.2.1 渐变背景	43
2.2.2 渐变的方向	45
2.2.3 图片与背景	47
2.3 变形滤镜的应用方法	50
2.3.1 灰度效果	50
2.3.2 反白效果	51
2.3.3 屏蔽效果	52
2.3.4 镜像效果	53
2.3.5 透明效果	55
2.3.6 旋转效果	56
2.3.7 透视效果	57
2.4 其他滤镜的应用方法	58
2.4.1 透明效果	58
2.4.2 动态模糊效果	60
2.4.3 模糊效果	63
2.4.4 滤色效果	65
2.4.5 阴影效果	66
2.4.6 浮雕效果	68
2.4.7 雕刻效果	69
2.4.8 辉光效果	70
2.4.9 马赛克效果	73
2.4.10 渐变的立体效果	75
2.4.11 波浪效果	76
2.5 灯光的应用	79
2.5.1 环境光	81
2.5.2 锥形光	83
2.5.3 点光源	88
2.5.4 移动灯光	90
2.5.5 改变灯光的颜色	93
2.5.6 改变灯光的强度	96
第3章 图片的展示方法	99
3.1 转场效果的设置方法	100
3.1.1 滤镜与转场	100
3.1.2 属性与转场	101
3.1.3 文字的转换	103
3.1.4 区块控件的转换	104
3.2 转场滤镜的应用	105
3.2.1 掀开的效果	105

3.2.2 百页窗效果	108
3.2.3 棋盘效果	111
3.2.4 淡入效果	113
3.2.5 渐变擦拭效果	115
3.2.6 对角扩张	118
3.2.7 其他形状的扩张效果	119
3.2.8 磁砖效果	122
3.2.9 辐状擦拭	124
3.2.10 乱数效果	126
3.2.11 交织效果	127
3.2.12 螺旋效果	131
3.2.13 伸展效果	133
3.2.14 阶梯效果	136
3.2.15 回转效果	137
3.2.16 锯齿效果	139
第 4 章 ScriptLet 控件的应用	141
4.1 认识 ScriptLet	142
4.1.1 何谓 ScriptLet	142
4.1.2 ScriptLet 的应用方法	143
4.1.3 嵌套的 ScriptLet	146
4.1.4 ScriptLet 的外观	147
4.1.5 ScriptLet 的名称	148
4.2 ScriptLet 的设计原理	150
4.2.1 内部变量	150
4.2.2 外部变量	152
4.2.3 子程序的设计	154
4.2.4 SetTimeout() 的用法	157
4.2.5 改变 ScriptLet 的控件	159
4.3 事件的处理与传递	163
4.3.1 事件的传递	163
4.3.2 事件的检查	166
4.3.3 EVENT 的应用	168
4.3.4 自定义的事件	171
4.4 ScriptLet 的其他属性	175
4.4.1 ScriptLet 的地址	175
4.4.2 ScriptLet 的载入状态	177
4.4.3 滚动条	178
4.4.4 滚动条的类型	182

4.4.5	Frozen.....	184
4.4.6	SelectableContent.....	186
第 5 章	行为与 HTC.....	189
5.1	行为的使用方法.....	190
5.1.1	认识行为.....	190
5.1.2	整页应用的设置方法.....	191
5.1.3	个别应用的设置方法.....	193
5.1.4	局部应用的设置方法.....	194
5.1.5	在程序中应用行为.....	195
5.2	HTC 的设计方法.....	196
5.2.1	HTC 的文件结构.....	196
5.2.2	控件的样式.....	199
5.2.3	事件的处理顺序.....	202
5.2.4	HTC 的应用.....	206
5.3	HTC 的高级设计.....	210
5.3.1	Attach 的应用.....	210
5.3.2	自定义的属性.....	212
5.3.3	属性的应用.....	215
5.3.4	自定义的方法.....	217
5.4	内置行为的应用.....	219
5.4.1	ClientCaps.....	219
5.4.2	DownLoad.....	221
5.4.3	HomePage.....	223
5.4.4	SaveFavorite.....	225
5.4.5	SaveHistory.....	226
第 6 章	HTML+TIME.....	231
6.1	HTML+TIME 的设置方法.....	232
6.1.1	认识 HTML+TIME.....	232
6.1.2	时间的表示方法.....	233
6.1.3	时间的设置方法.....	234
6.1.4	区块控件与时间.....	236
6.2	显示时间的控制方法.....	238
6.2.1	Begin、End.....	238
6.2.2	Dur.....	240
6.2.3	RepeatCount.....	243
6.2.4	RepeatDur.....	245
6.2.5	Restart.....	247

6.3	改变控件的外观	248
6.3.1	改变样式	248
6.3.2	改变类的名称	251
6.3.3	改变显示状态	253
6.3.4	改变控件的位置	255
6.4	多重控件的控制方法	256
6.4.1	<t:seq>	257
6.4.2	<t:par>	260
6.4.3	<t:excl>	263
6.4.4	Time Container	265
6.5	动画效果的设置方法	266
6.5.1	<t:Animate>	266
6.5.2	<t:AnimateColor>	269
6.5.3	<t:AnimateMotion>	272
6.5.4	<t:Set>	274
6.6	动画控件的控制	277
6.6.1	显示状态	277
6.6.2	播放次数	280
6.6.3	TargetElement 的设置方法	282
6.6.4	位移的设置方法	285
第 7 章	多媒体的应用	289
7.1	路径控件	290
7.1.1	路径控件的使用方法	290
7.1.2	控制播放的流程	293
7.1.3	移动的方向	296
7.1.4	移动的速度	298
7.1.5	路径的设置方法	301
7.1.6	控件的控制	307
7.2	结构化图形控件	310
7.2.1	图形控件的使用方法	310
7.2.2	坐标系	313
7.2.3	线条与样式	315
7.2.4	绘图的方法	318
7.2.5	变形处理	327
7.3	定序器控件	332
7.3.1	认识定序器控件	332
7.3.2	Onintt() 的设置方法	334
7.3.3	播放流程的控制	336

7.3.4	定序器控件的属性	337
7.3.5	改变播放状态	339
7.4	子图形控件	342
7.4.1	子图形控件的使用方法	342
7.4.2	播放的方法	345
7.4.3	调整播放速度	347
7.4.4	背景颜色的处理	348
7.4.5	改变播放的流程	350
7.4.6	动画文件的控制	351
第 8 章	VML 的应用	355
8.1	认识 VML	356
8.1.1	VML 的设计方法	356
8.1.2	VML 的样式	358
8.1.3	VML 与动画	360
8.1.4	VML 的对象模型	361
8.2	基本的图形控件	363
8.2.1	Rect	363
8.2.2	RoundRect	364
8.2.3	Line	366
8.2.4	PolyLine	367
8.2.5	Oval	368
8.2.6	Curve	369
8.2.7	Arc	371
8.3	属性的设置方法	373
8.3.1	填色	373
8.3.2	边线的设置	375
8.3.3	旋转图形	377
8.3.4	镜像	379
8.4	复合式的图形	382
8.4.1	复合图形的设置方法	382
8.4.2	嵌套的复合控件	384
8.4.3	<ShapeType> 的应用	386
8.4.4	坐标单位	389
8.5	图片的应用	390
8.5.1	图片的设置方法	390
8.5.2	图片的剪切方法	392
8.5.3	调整色彩	394
8.5.4	反白处理	396

第9章 VML 的其他元素.....	399
9.1 <Stroke>.....	400
9.1.1 线条的样式.....	400
9.1.2 端点的形式.....	404
9.1.3 箭头的设置方法.....	406
9.1.4 线段的连接方式.....	410
9.1.5 线条的填色方法.....	411
9.1.6 填色的透明度.....	415
9.2 <Fill>.....	416
9.2.1 线性渐变的设置方法.....	416
9.2.2 渐变的方向.....	419
9.2.3 渐变色的分布方法.....	421
9.2.4 辐射渐变的设置方法.....	423
9.2.5 图片与填色.....	427
9.2.6 透明度与背景颜色.....	430
9.3 <Shadow>.....	432
9.3.1 阴影的设置方法.....	432
9.3.2 阴影的颜色.....	435
9.3.3 双层阴影的设置方法.....	437
9.4 <Extrusion>.....	439
9.4.1 <Extrusion>的设置方法.....	439
9.4.2 厚度.....	442
9.4.3 立体控件的外观.....	444
9.4.4 颜色与亮度.....	445
9.4.5 旋转控件.....	447
9.5 <TextPath>.....	449
9.5.1 文字的设置方法.....	449
9.5.2 文字与路径.....	451
9.5.3 文字的分布方法.....	453

1 XHTML 概述



本章主要内容

- 认识 XHTML
- 文件的格式
- XHTML 的语法规则
- 自定义标记的应用

1.1 认识 XHTML

XHTML 的全名是 eXtensible HyperText Markup Language(可扩展的超文本标记语言), 可以看成是 XML 与 HTML 的综合体。XHTML 文件先用 XML 的语法来声明, 再用 HTML 的标记来设计网页的内容。

```
<!-- 声明区 -->
<?xml version="1.0" ?>
<!-- 文件格式 -->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<!-- 命名方法 -->
<html xmlns="http://www.w3.org/1999/xhtml">
<!-- 文件的主体 -->
<head><title>ex110.1</title></head>
<body>
  <center>
    <h2 style="color:red">XHTML & </h2>
    <h2 style="color:red">IE 5.5</h2>
    <h1 style="color:blue">网页设计的新视野</h1>
  </center>
</body>
</html>
```



1.1.1 声明区

XHTML 文件属于 XML 的领域, 所以要用 `<?xml version="1.0" ?>` 来声明。系统默认的字体是 UTF-8 或 UTF-16, 如果要改用其他的字体, 必须在声明区指定。例如:

```
<?xml version="1.0" encoding="GB2312"?>    // 指定字体
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head><title>ex110.1</title></head>
<body>
  <center>
    <h2 style="color:red">XHTML & </h2>
```

```

<h2 style="color:red">IE 5.5</h2>
<h1 style="color:blue">网页设计的新视野</h1>
</center>
</body>
</html>

```

如果只是在浏览器上显示 XHTML 文件的内容, 可以省略<XML>的声明。

<!-- 省略 XML 的声明 -->

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head><title>ex11.2</title></head>
<body>
  <center>
    <h2 style="color:red">XHTML and IE5.5 </h2>
    <h1 style="color:blue">网页设计的新视野</h1>
  </center>
</body>
</html>

```



1.1.2 文件格式的定义

XML 的自定义标记, 必须用<!DOCTYPE>来定义, 或是将标记的属性及相关信息存储在*.dtd 文件中, 再用<!DOCTYPE>来引用, 例如:

```

<?xml version="1.0" encoding="GB2312" ?>
<!DOCTYPE 唐诗 SYSTEM "ex112.1.dtd" [      <!-- 引用外部的 DTD -->
  <!-- ELEMENT 解析 (#PCDATA)>              <!-- 加入新的声明 -->
  <!-- ELEMENT 说明 (#PCDATA)>
]>
<唐诗>
  <作者> 杜甫    </作者>
  <标题> 八阵图  </标题>
  <内容> 功盖三分国, 名成八阵图。
    <newline/>
    江流石不转, 遗恨失吞吴。
  </内容>
</唐诗>

```

XHTML 则只能使用 W3C 所制定的 3 种 DTD:

【Strict DTD】

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3c.org/TR/xhtml1/DTD/xhtml1-strict.dtd">.
```

【Transitional DTD】

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional //EN"
"http://www.w3c.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">.
```

【Framesett DTD】

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Framesett //EN"
"http://www.w3c.org/TR/xhtml1/DTD/xhtml1-framesett.dtd">.
```

因为浏览器会用自己的 DTD 来解读 XHTML 文件，而不会去下载指定的 DTD，所以用 IE5.5 显示 XHTML 文件的时候，`<!DOCTYPE>` 的声明没有任何作用，通常就用 `<!DOCTYPE html>` 来代替。XHTML 文件如果要用其他的软件来处理，就一定要加入正确的 `<!DOCTYPE>` 才能顺利地读取。

1.1.3 名称空间

xmlns 用来指定标记名称的来源，默认的使用方法是：

```
<?xml version="1.0" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-Transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head><title>ex113.1</title></head>
<body>
  <center>
    XHTML NameSpace
  </center>
</body>
</html>
```

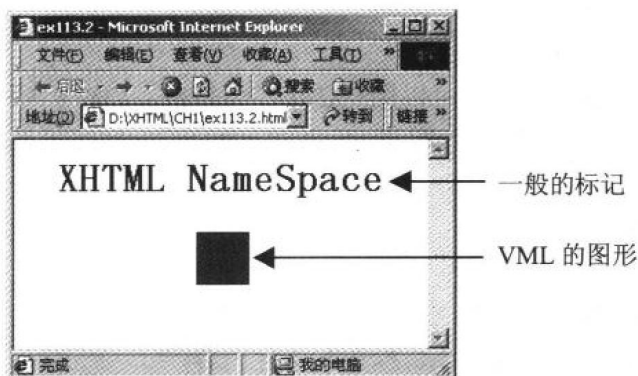
在同一份 XHTML 文件中指定不同的名称空间，就可以使用不同的标记，例如：

```
<?xml version="1.0" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-Transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" // 第1组名称空间
      xmlns:v="urn:schemas-microsoft-com:vml" // 第2组名称空间
<head>
  <title>ex113.2</title>
  <STYLE>
    v\:* { behavior: url(#default#VML); }
  </STYLE>
</head>
<body>
  <center>
    <h1>
      XHTML NameSpace
    </h1>
    <v:shape
      fillcolor="red"
      style="position:relative;top:1;left:1;width:200;height:200"
      path = "m 1,1 l 1,200, 200,200, 200,1 x e">
    </v:shape>
```

```

<center>
</body>
</html>

```



1.1.4 文件的主体

XHTML 的文件主体区，用 HTML4.01 的标记来设计网页的内容。

```

<?xml version="1.0" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-Transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
    xmlns:v="urn:schemas-microsoft-com:vml">
<!-- 文件的主体 -->
<head>
    <title>ex114.1</title>
<style>
    .type1 {position:absolute;
        top:60; left:20; color:blue; font-size:36;
        filter:shadow(color=red,direction=135)}
    </style>
</head>
<body>
    <p class="type1">网页设计的新视野</p>
</body>
</html>

```



除了比较严谨的语法规则外（请参考 1.3 节的说明），XHTML 与 HTML 的设计方法几乎完全相同。所以只要会用 HTML，就能轻松地进入 XHTML 的领域，难怪大家都说 XHTML=HTML5.0。

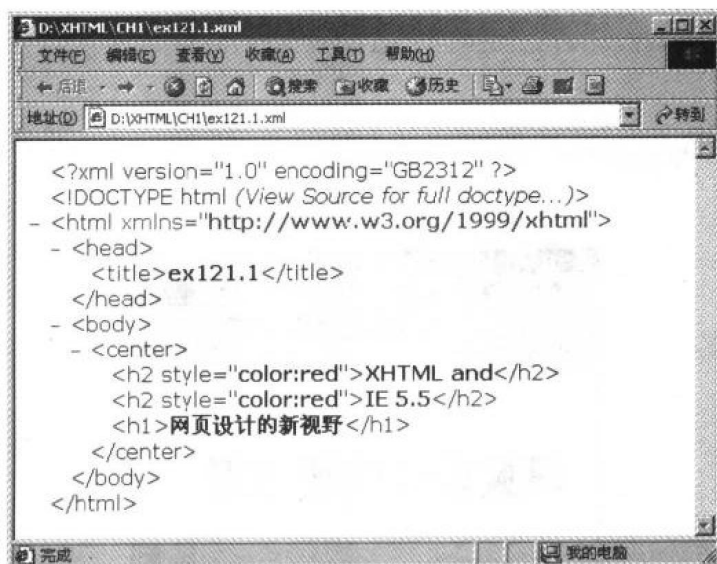
1.2 文件的格式

同一份 XHTML 文件，可以用*.XML、*.HTM、*.HTA 3 种扩展名来存储，IE5.5 对这 3 种文件格式各有不同的处理方法。

1.2.1 XML

将 XHTML 文件存储为 XML 的格式，然后用 IE5.5 直接打开文件，窗口上会显示树状的数据结构，包括所有的指令、标记和文件内容。

```
<!-- 范例文件: Chap01\ex121.1.xml -->
<?xml version="1.0" encoding="GB2312" ?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>ex121.1</title>
</head>
<body>
  <center>
    <h2 style="color:red">XHTML and </h2>
    <h2 style="color:red">IE 5.5</h2>
    <h1>网页设计的新视野</h1>
  </center>
</body>
</html>
```



IE5.5 所显示的数据结构类似“可折叠大纲”。在“-”号上面单击鼠标的左键，就可以隐藏内部的数据，标示符号则变成“+”。