

第一部分

第一步，第一个网页

第1学时 理解网页的编写

第2学时 从一个网页编写应用程序开始

第3学时 快速完成一个个人网页

第4学时 在其它应用程序中开始编写网页

第1学时 理解网页的编写

我想你已经准备好了，甚至有些迫不及待——要开始创建网页了。但是在建造第一个网页之前，你首先需要对网页是如何产生的有一个基本的理解，并且对所期望的网页外观做一些规划。

在本学时内，你将在很短的时间内了解网页以及网站是由哪些部分组成的。完成本学时的学习后，你将能够回答如下的问题：

- 网页是由什么组成的，它们是怎样工作的？
- HTML 是什么，为什么需要注意它？
- 如何让多媒体(图像、声音、视频和动画)成为网页的一部分？
- 后缀名是什么，它们为什么很有用？
- 用什么方法能够将多个网页组织成一个完整的网站？

1.1 网页的分解

尽管还有其它的一些可选部件，但大多数的网站就是由本部分所描述的这些部件所组成的。知道这些部件是什么是很重要的，因为在网页编写中，主要的任务就是确定为每一个标准的部件选用相应的内容，而且一个主要的难点在于：每一种浏览器在处理不同的部分时有不同的方式(关于此将在以后详细说明)。

1. 你所看到的部分

图1-1显示的是在浏览器内，访问者所能够看到的一个典型网页的各个组成部分。

图1-1 网页中的一些基本组成部分



- **主题** 一般情况下, 图形化的浏览器(大多数的 Windows、苹果以及 X Windows 浏览器)在显示此网页的窗口的标题栏上显示“主题”。



网页的真正主题并不一定出现在网页内部, 而是作为显示此网页的那个浏览器窗口的标题。

然而, 大多数网页有另一种标题, 或者说是屏幕上的文本或图像, 它们的作用和在书本或杂志中的典型作用一样——位于页中显眼且重要的顶部附近, 从而给网页赋予了名字。

- **标题** 浏览器通常以较大字体、粗体或者其它的显示类型来突出显示。一个网页可以有许多的标题, 而且标题还可以有多达 6 层的嵌套深度; 也就是说, 可以有子标题、子标题的子标题等等。
- **普通文本** 是组成网页的基本的、多用途的文本。典型情况下, 网页的作者将普通文本中的线和块当作“段落”。但是按照 Netscape 作者的说法, 网页上任何离散的文字块都被称为段落——不管它是一个标题、普通文本或者是其它什么东西, 它们的类型由赋予此段落的“属性”所确定。
- **签名** 通常显示在网页的底部。签名标识了网页的作者, 并且常常还包含作者(或者网络管理员)的电子邮件地址, 从而使访问者可以发送有关此网页的评论或者问题。而这里的电子邮件地址格式有时是一个 mailto 的链接, 这样访问者就能够通过单击它而打开电子邮件应用程序, 其中, 邮件消息的目的地址是预先设置好的, 即此网页的作者。
- **水平线** 是用来装饰网页的, 它将网页分割成逻辑上的几个部分。
- **内嵌图像** 是那些嵌入网页布局中的图像, 它们让网页变得生动活泼, 或者使得网页的信息量更大。
- **背景色或样式** 可以是一种单一的颜色, 或者是一个内嵌图像, 但是和普通的图像不同的是, 它们覆盖了整个网页的背景, 文本和其它图像就显示在其上面。
- **动画** 可以是网页中出现的文本或者图片, 不同的是它们以某种方式运动。图片可以忽隐忽现或者按照简单的动作进行旋转, 而文本既可以忽隐忽现, 也可以从屏幕上滚动而过。
- **超级链接** (或者是简单的链接)可以链接到很多不同的对象上去: 如其它的网页、多媒体文件(外部的图像、动画、声音、视频)、文档文件、电子邮件地址以及在其它类型的服务器(如 Telnet、FTP 和 Gopher)上的文件或者应用程序。链接还可以引导至当前网页的某个特定的位置。
- **图像地图** 是一些内嵌图像, 此图像的不同区域下面包含有不同的链接。
- **列表** 可以是黑点(就如本行中所示的圆点), 或按数字进行的编号, 或者其它什么。
- **表单** 是一些区域, 访问者可以在其中的空白处填写, 以回复在线的问卷、预订货物或者服务等。

2. 看不到的部分

除了看到的那些东西以外, 网页(或者说那些组成此网页的文件)还有其它一些元素也可以被包括在内。通常, 访问者并不能看到这些元素, 不过它们有如下的效果:

- **鉴定** 网页文件可以包含多种鉴定信息, 包括作者的名字(和 / 或电子邮件地址), 以及一些特殊的编码, 这些编码可以帮助搜索引擎确定网页的主题与内容。

- 注释 注释是作者希望在直接阅读网页的HTML代码时能够看到的文本，在浏览器中是看不到的。注释通常包括HTML文件的结构或者组织的注意事项。

新术语 HTML(Hyper Text Markup Language的缩写) 是网页存储时所使用的计算机文件格式。一个HTML文件事实上就是一个具有一些特殊代码的文本文件，它告诉浏览器如何来显示此文件——文本的每个块所要用的尺寸、在哪里放置图片等等。

- JavaScript代码 在HTML文件内，JavaScript的语句行可以给网页添加一些特殊的动态性能，就像一个对时间敏感的讯息。
- Java applet 以单独的文件存在，这种Java程序模块可以提高访问者、浏览器和服务器之间的交互能力。比如说，Java就非常适合于编写在Web上玩的交互游戏。
- 图像地图和表单处理代码 用来处理图像地图和交互表单的程序代码。

动手实践：区分网页的不同部分

1) 打开Web浏览器，连接到因特网上并到想要去的任何网页上去(可以用本书附带的因特网探索者(Internet Explorer)的拷贝，或者用所拥有的任何其它的浏览器)，见图1-2。

2) 查看浏览器所显现的窗口标题栏(在窗口最上面的一栏，通常能看到正在使用的应用程序的名字)。你可能看到正在访问的网页的主题，以及正在使用的浏览器程序的名字，见图1-3。

图1-2 第1步：打开浏览器并访问所喜欢的任何网页



图1-3 第2步：在浏览器的标题栏处找到网页的主题



3) 在此网页(和其它的网页)中继续寻找，看看能否找到在本学时前面所谈及的一些部件(如果需要，可以参考前面的图1-1)。



在大多数的浏览器中, 其窗口最底部都会有一个状态栏。每当指向一个链接(但不单击)时, 将可以在状态栏中看到这个链接所指向的目的地。

除了发现链接所指向的地方外, 还可以了解在网页中出现的图片文件。将鼠标指向一个图片, 右击, 然后选择弹出菜单中的“属性”项。那么就会弹出一个对话框, 它告诉你此文件的名称、大小以及所指的图片的类型。

使用这些技术, 就可以通过学习所访问的站点的设计, 来逐步推进自己编写网页的技巧。

1.2 网页如何工作

在编写网页的时候, 不管是如何进行的, 最后所得到的都是一个 HTML 文件, 它可以在 Web 服务器上发布。

新术语 一个 HTML 文件 (见图 1-4) 包含了出现在网页上的所有文字, 以及一些 HTML 标记。

这些标记是 HTML 文件中的一些代码, 它告诉浏览器应该做什么事情。比如说, 如果在一个字的前面有一个 **** 标记, 而在其后有一个 **** 标记, 这些就告诉浏览器, 在显示网页的时候这个字需要以粗体显现。

除了控制网页的格式外, HTML 文件中的标记将每一个大块文本当作网页的一个特定元件。比如说, HTML 标记区分出作为网页主题的一行文本、作为段落的成块文本、作为链接的特定的行或者字等等。其它的 HTML 标记指定网页在显示时, 浏览器需要在网页内嵌入的内嵌图像的文件名。

图1-4 图1-1中显示的网页所对应的HTML源文件

```
File Edit Search Help
</table>
</div></td>
<td valign="bottom" width="73%"><h1>Getting the Most From <em>The Piano
Education <a name="Page">Page</a></em></h1>
<hr>
</td>
</tr>
<tr>
<td width="17%" background="trebturc.gif" align="left"><!--webbot
bot="Include" U-Include="pepnav.html" TAG="BODY" startspan -->
<div align="center"><center><table border="2" cellspacing="0"
cellpadding="0" bordercolorlight="FFFFFF" bordercolordark="000000"
bordercolor="0000FF" width="126" background="trebtanc.gif">
<tr>
<td width="124" valign="middle" align="center"><strong><em><font
face="Arial" size="2"><a href="pnoedm.html">Home</a></font></em></strong></td>
</tr>
<tr>
<td width="124" valign="middle" align="center"><strong><em><font
face="Arial" size="2"><a href="pnofaq.html">FAQ
(Help)</a></font></em></strong></td>
</tr>
<tr>
<td width="124" valign="middle" align="center"><strong><em><font
face="Arial" size="2"><a href="pnoedpre.html">Premiering</a>
</font></em></strong></td>
```

Web 浏览器是这样的一种应用程序, 它至少知道如下的两件事情:

- 访问远程 Web 服务器上面的 HTML 文档(这是通过一种叫做 HTTP 的通信协议来实现的)。
- 翻译文档中的 HTML 标记, 从而使标题显示为标题, 链接就成为链接等等, 如此这般来处理。

一个很重要并且要记住的便是：HTML标记并不提供在字处理器中所能够得到的，那种对页面的精确的格式控制。大多数情况下，HTML只是区分什么是什么。每一个浏览器在决定如何在屏幕上显示这些元件的时候都是不尽相同的。

新术语 扩展 是对标准HTML语言的额外补充，通常是浏览器的编制者所创建的，其目的在于使得浏览器可以做一些没有包括在HTML标准中的事情。参见本学时后面的“扩展：爱他们！恨他们”。

写到这里，可以介绍一下目前最流行的两种浏览器——各种各样版本的 Netscape Navigator和Internet Explorer——它们以压倒一切的优势占据了绝大多数的浏览器市场。虽然这两者支持的HTML标记只有细微的差别，但是，这两者之间的永久竞争的结果将会导致这两个浏览器完全一样地显示大多数的网页。这就意味着，对于大多数在Web上潜在的访问者来说，你的网页外观将和在你的浏览器中所看的效果大体相同。

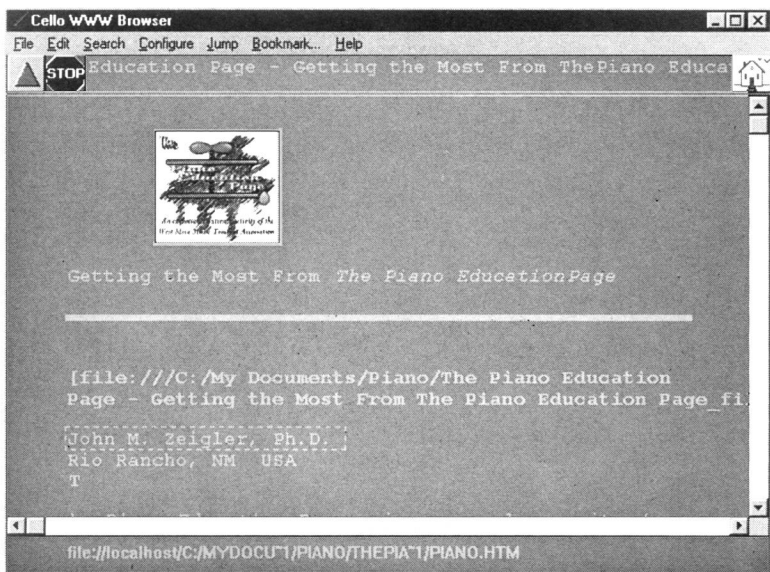
对于那些使用这两种主流的浏览器之外的人来说，网页将显示相同的文本内容和总体的组织，但是对于它的图像内容和其它美学的部分来说，则可能因不同的浏览器而呈现巨大的差异。事实上，在某些情况下，图片和任何其它的图像装饰根本就不会被显示出来。



你可以下载在图1-5和图1-6中显示的浏览器，并在其中显示网页，看看你的作品在不同的浏览器中是如何显现的。获得这些浏览器的网址在第23学时“测试并更新你的网页”中给出。

为了举例说明这种“浏览器与浏览器”之间的区别，图1-5和图1-6中显示了和前面的图1-1中完全相同的网页。然而，图1-1是用Netscape Navigator来显示此网页，而图1-5和图1-6则是用另外两种浏览器来显示：分别是Cello和DosLynx。比较这两幅图，并观察它们在显示时有哪些区别。

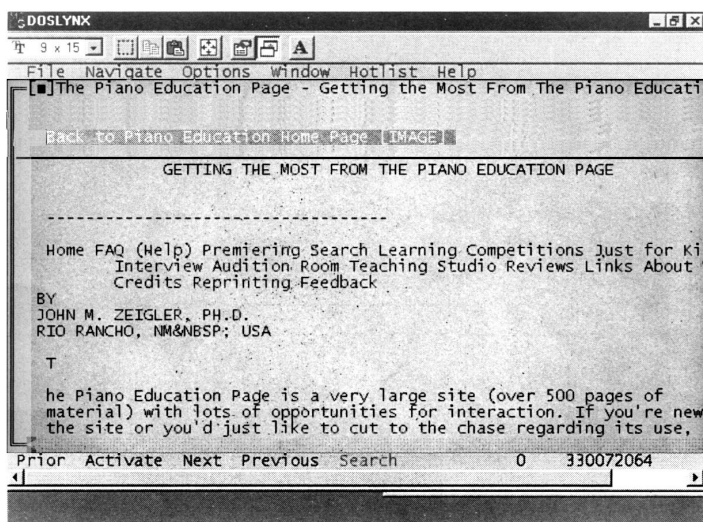
图1-5 在Cello显示的与图1-1中完全相同的网页



在图1-6中显示的浏览器，即DosLynx，是一个基于DOS(还记得DOS，是吧)的文本浏览器。这些浏览器正在迅速地从Web上消失(但是仍然会有)，它们不能够显示内嵌图像，并且以

相同的大小和外观来显示所有的文本，尽管像标题这样的一些重要部件，可以用粗体和下划线来突出显示。一些用户使用纯文本的浏览器完全是出于某种选择，但是大多数人这样做则是因为他们缺少合适的因特网帐号类型，或者缺少合适的用于图像浏览器的硬件。

图1-6 在DosLynx显示的与图1-1中完全相同的网页



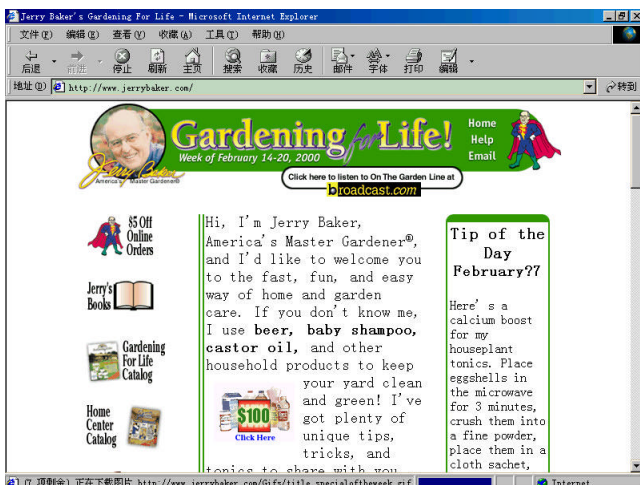
除了使用这些纯文本的浏览器外，对于其它一些不能在线地看到图像的人来说，则是因为他们在浏览器中使用了定制属性，关闭了图像的显示(这样可以加速网页的显现)。

尽管纯文本浏览器几乎已经消失了，作为一个 Web 作者的新手，记住有些人有可能看不到网页中的图片是很重要的。对他们来说，必须确保不管图片出现与否，网页中的文本都可以完成所有的任务。

动手实践：检查网页的 HTML 源代码

1) 在 Web 浏览器中查看任何网页，见图 1-7。

图1-7 第1步：查看任何网页



2) 改变浏览器的视图，从而就可以看到此网页的源代码，见图 1-8。

在Internet Explorer 5中(本书附带有此软件)，选择“查看”菜单中的“源文件”。

在Netscape Navigator中，选择“查看”菜单中的“源代码”。

在其它浏览器中，在菜单项中寻找“源文件”或者“HTML”这样的字眼。

图1-8 第2步：选择一个菜单项，它显示正在查看的网页的源代码



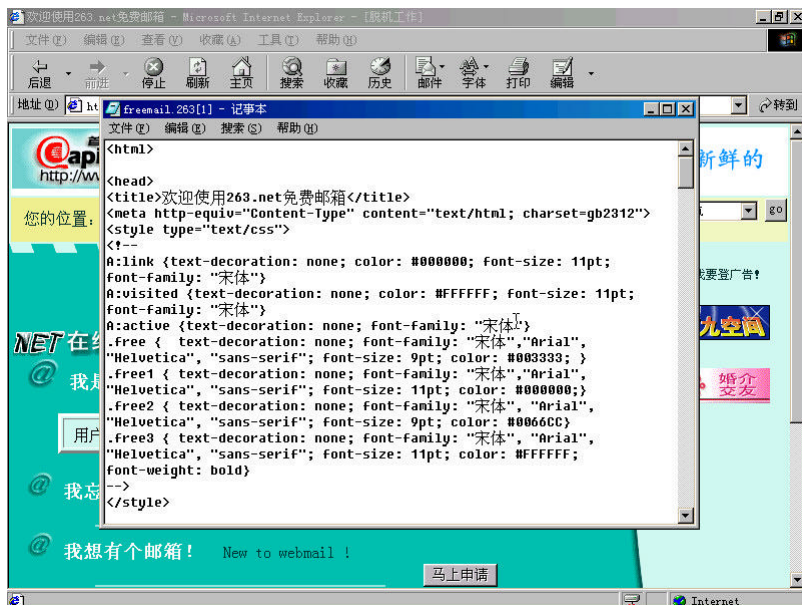
3) 查看HTML代码，见图 1-9。如果看起来有些混乱，千万别担心——你无需真正知道如何用眼睛来“解读”HTML文件。但是如果仔细观察，将会在不同的代码中看到如下的内容：

在网页中实际出现的文本

网页中的图片的文件名

链接所指向的网址

图1-9 第3步：查看代码，对网页到底是怎么组成的建立一些感觉



4) 当检查完HTML源代码之后，关闭显示代码的窗口，然后就回到正常的浏览器视图。

1.3 图片、声音和其它的媒体

因为HTML文件仅仅包含文本，所以在网页中看到的图像，以及可以访问的多媒体，都不是HTML源代码本身的一部分。事实上，它们是以如下任意一种方式链接到此网页上的：

- 内嵌图像 是一幅图像文件，其文件名和地址在 HTML 文件本身中被引用，并且用标记区分为图像。内嵌图像是嵌入网页布局的图像——是访问网页的时候在浏览器中所看到

的所有图像。

- 外部媒体 是一些图像、声音或者视频文件，它们的名字和地址以链接方式出现在HTML文件中。这些文件并不像网页中的一部分那样，可以自动地出现或者播放。相反，网页显示其链接，当它们被激活时，就将这些文件下载，然后再播放或者显示。

不管是内嵌还是外部的，在网页中使用媒体文件，对用于查看网页的浏览器来说的确是一种挑战。此浏览器必须能够显示图像，从而显示这些内嵌图像。外部媒体文件可以由浏览器本身播放，或者更通常的情况是，由浏览器打开辅助应用程序(或者插件)来播放。

当选择将媒体嵌入网页时，必须仔细考虑所要使用的文件类型。HTML文件的纯文本规则就是让各种各样类型的计算机用户可以访问此网页。图像文件就不太可能被众多系统所读取，而声音和视频文件就更少了。即便是限定在PC和苹果机的范围内，也需要考虑此媒体是否将被大多数的浏览器和/或者辅助应用程序所支持。关于此，将在第13学时“为网页找到图片”内做进一步的讲述。

1.4 扩展：爱他们，恨他们

HTML是标准化的，所以任何Web浏览器都可以读取任意的Web文档——当然是某些特定的类型的。

这就是问题所在：大多数现代的浏览器支持HTML 2，它们是一组构建得非常完善的标记，它们是由对因特网标准有影响力的委员会所制定的。标准化是好的，因为它给网页的作者提供了一种方式，用它可以确保大多数浏览器能够读取他们所发布的内容。因为任何浏览器都能够理解和翻译所有的HTML 2标记，作者只需要坚持限定在这些标记内，就可以确保他们的网页能够最大限度地被在线观众所访问。

然而，标准的问题在于其自身的演化是非常缓慢的。而在互联网上，唯一可以允许慢的只有下载；革命是要求迅速的。考虑这个问题：第一个图像浏览器在四年前就出现了，而现在我们讨论的是实时视频。作为一个图像的交互的环境，Web的整个诞生和成熟是在一个单独的高层行政部门里面完成的。不可思议！

HTML 2包括了对所有基本对象——如标题、普通文本、水平线、列表、链接、内嵌图像等等的支持。但是现在已经有了HTML 3甚至HTML 4，它们包含了所有HTML 2的标记，而且添加了和如下这些类似的功能：

- 居中和靠右排列的文本
- 表格
- 帧
- 数学函数
- 将文本放置在内嵌图像周围的方法(而不是仅仅图像的上面或者下面)。



当为企业的内部网创建网页时，由于那里的所有用户使用的都可能是同一种浏览器，就可以无需考虑扩展问题——你可以运用此浏览器支持的所有的标记。

处于领先地位的浏览器，包括Navigator和Internet Explorer，都支持HTML 3，甚至也支持大多数的HTML 4标记。但是，网页改善的速度是如此之迅速，以致于网景和微软都不断将

新的标记和其它功能嵌入到他们的浏览器中，而这些标记和功能都不是来自于任何已获批准的HTML标准。这些标记被称作为扩展。

新术语 扩展 就是一个HTML标记，它使得在网页上某些新的功能成为可能，但是它目前还不是正式的HTML标准中的一部分。

当在Web文档中使用时，这些扩展的效果通常只可以在那些特定的，支持它们的浏览器中看到。当然，Navigator支持所有的网景的扩展，而Internet Explorer支持所有的微软的扩展。然而，不是所有的浏览器都支持每一个扩展。这就是为什么须要小心处理它们的原因。

总的来说，当用一个不兼容的浏览器来访问一个使用这些标记的网页时，并不会发生什么可怕的事情。虽然那些基于扩展的奇特格式不会显现，但是网页的“肉”——它的文本和图像——仍然是可以读取的。



那些想利用扩展的好处的作者就必须要知道，有些访问者是无法看到网页的真实全貌的。这就是为什么，在网页上面越来越多地看到这样的信息：“推荐使用网景的Navigator以获得最佳效果”或者“针对Internet Explorer作了改善”。这是作者在告诉你，他或她使用了扩展——如果你想利用网页的所有特性，就最好选择一个兼容的浏览器。

1.5 组织一个网站的方法

最后，在投入到创建网页的工作之前，必须考虑一下如下的几个问题：

- 讯息如何才能够被分解成一系列有组织主题？
- 一个网页要多长，或者要有多少个网页(连接在一起形成一个网站)，才能完成必须要讲述的内容？



在研究并优化了主题的分解，以及讯息的大纲之后，你也许就会发现已经写出网页的标题了。

粗略记录下文档将会覆盖的主题或者子主题的一个列表。有多少个这样的主题？每一个主题又需要多少的内容？在这个简单操作之后，将能够对文档的规模和范畴有一个比较恰如其分的理解。

现在看看这些主题。它们是否按照从开始到结束的逻辑顺序进行，其中每一个新的部分取决于前面部分的内容？或者，这些材料看上去是否很自然地分解成了一些子主题(以及更低级别的主题)？如何调整主题的次序，从而可以使得它们之间的过度更符合逻辑，或者将相关主题组织在一起？

在展开分解工作时(不是那种崩溃(breakdown)——而是主题的分解)，一个简单的框架就出现了。在编写文档之前，对框架做越多的优化，就会越紧扣主题，而且编码的效率也会越高。更重要的是，最后所产生的Web文档将以一种清晰而明朗的方式来展现讯息。



在规划一个包含三个或者更多网页的文档时，先在一张纸上粗略地描述出每一页，从而决定哪些信息归属于哪一页。将这些纸片粘贴在墙上，并且通过画线或者连线，来规划网页中的链接。

在建构框架时，考虑展示的逻辑组织以及它的内容，它们如何才能够和网页上能够看到的一些通常的组织结构相匹配：

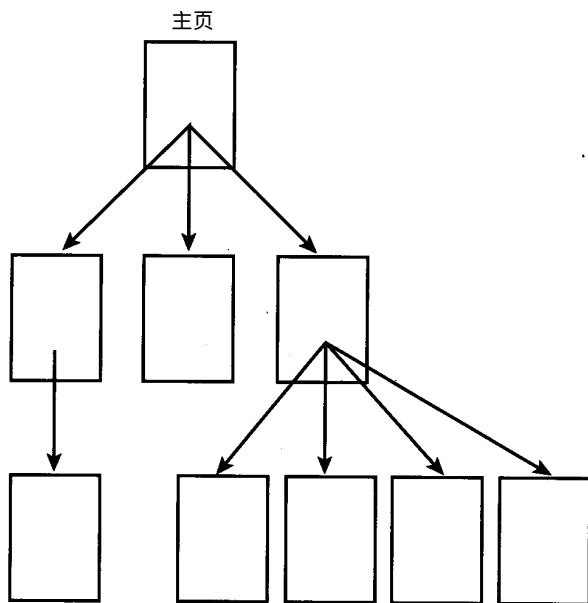
布告板——一个单独的、简单的网页，它通常描述一个人、小的业务或者简单的产品。大多数的个人网站都是这种类型。它们通常包含一些链接，这些链接是指向网络上的相关(或者最喜欢)资源的，但是不指向相同文档内的任何其它的网页(网景的网页向导就建造这种类型的网页)。

单页线性——一个网页，或长或短，它被设计成从头到尾地进行阅读。通常使用一些规则将这样一个网页分解成虚拟的“页”。读者可以翻阅整个网页，但是也可以使用一个内容和目标的表格来快速跳至任何部分。这种类型最适合于比较短的文档(少于 10个满屏)，而且这个文档中所有的信息很自然地从头到尾地过度。

多页线性——和单页线性同样的基本思想，但是它被分解成多个逻辑上连贯的、一个接一个的网页，从开头到结束，就像一个故事一样。可以通过在每一页的底部放置一个指向下一页的链接来引导读者遍历整个系列的网页。

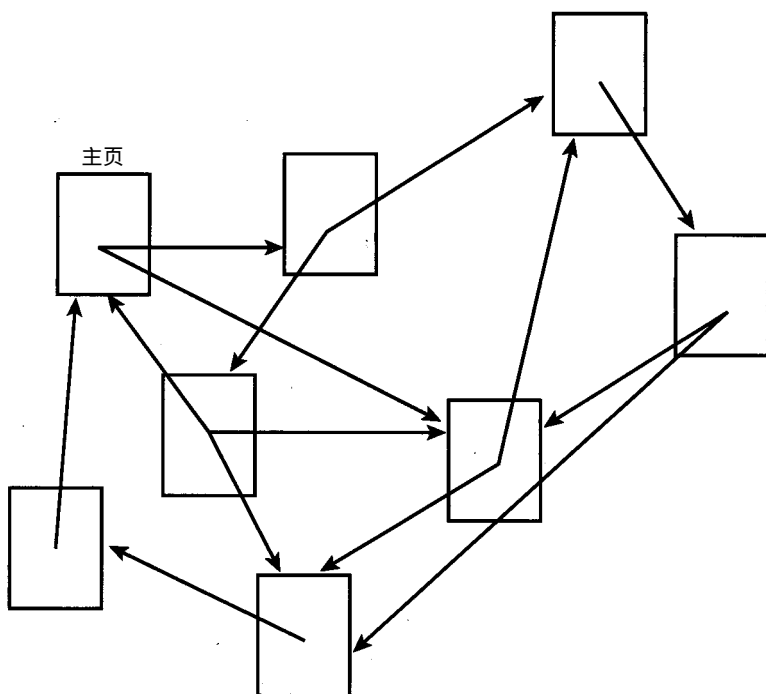
分层——典型的网站结构。一个首页(有时候会与主页混淆)包含到其它网页的链接，每一页包含一个主要的主题区。每一个这样的网页又可以包含指向更多的网页的多个链接，进一步将主题分解，从而到达更特定的信息。这样的结果便是一个树型的结构，就像在图 1-10中所显示的那样。

图1-10 一个分层的结构



网状——一个网状的结构(见图 1-11)是一个没有层次的分级的结构。这个文档中有多个网页，而在其中的任意一个网页中又都包含有连接到其它网页的链接。可能会有一个首页，但是从那里进入之后，读者就可以在此网中逛来逛去，且不须沿一个特定的路径。网状的结构是松散并且自由游走的，最适宜于娱乐、休闲的主题，或者那些难于进行顺序或层次分解的主题(提示：在选择一个网站结构之前，确信要传达的讯息的确需要这种结构——也许你只是不知道重点在哪里)。

图1-11 一个网状的结构



还有其它一些方法可以组织信息，它们是上面所展示的结构变体。但是这些结构中必有一个和要传达的讯息以及文档的大致外形是相类似的。换一种说法就是，如果还没有确定哪一个结构是最适合要创建的网页的话，就需要对讯息进行更多的处理，并以不同的方法将它分解，直至有一种结构出现。

1.6 课时小结

如果还有时间，就可以搞一些好的咖啡，坐下来，开始拿一个向导(参见第3学时“快速完成一个个人网页”)或者任何网页编辑器来随便试一试。最后就会得到一个适合发布(如果不是用于阅读)的HTML文件。

但是，这是一个能够实现你当初想要网页时的目标的文件吗？如果你正在寻找一个网友的话，你的网页会吸引他们吗？如果你正在寻找客户或者顾客，你的网页能够帮助你胜过竞争对手一等吗？如果你正在提供有用的信息，采用的方式是否能够让网页的访问者找到直觉并且易于操纵？

要创建一个好的网页，必须首先打好基础，知道一个网络文档是怎么工作的、它能够做什么、不能做什么。这是在本学时中已经掌握的。我并不要控制你的文档应该是什么样、什么感觉或者怎么操作——这必须依靠你的灵感。但是我尝试给你灌输一些想法，从而在下一步，不管是什么任务，可以作出合理而全面的选择。

1.7 专家答疑

问题：基本上，你是在告诉我，在我的网页中，我可以做所有那些很“酷”的事情，但是我必须坚持做那些无聊的事情，因为这些很“酷”的东西是基于扩展的，从而有些人不能

够看到图像。这样是不是有点像耍无赖？

解答：无赖？不是。实际上，这并不是一个非常严重的问题。首先要记住，绝大多数浏览网络的人们都可以看到图像，而且大多数人所使用的浏览器是可以处理大多数扩展的。所以即便遇到一个极其困难的图像问题，而且是基于扩展的格式，你的网页在大多数人面前也会非常棒。通过使用一些在以后我会教你的技术，就可以容许削弱图像或者削弱扩展，从而你的网页可以像对于其它任何人一样，对他们来说也是很有用的。

但是，很重要的一点是仍然要记住这样一个事实，那就是仅仅因为是在一个微软的网页编辑器中编写，这并不意味着只是在为 Internet Explorer 的用户而编写。一个聪明的作者能够让他或她的文档信息丰富，并且看上去很“酷”，而且同时还能够被所有的用户所访问。