



Microsoft
Dynamic HTML

Microsoft Dynamic HTML

使用详解

James Meade
(美) David Crowder 著 潇湘工作室 译
Rhonda Crowder

- 全面完整地介绍DHTML
- 每个概念都伴有实际操作项目
- 代码实例丰富



机械工业出版社



CMP

本书完整而全面地介绍了微软公司的DHTML (动态HTML), 全书共分为四大部分: 第一部分解释了DHTML及其强大的能力; 第二部分, 介绍了如何在Web页面上创建多媒体效果; 第三部分介绍专业的模型站点, 并创建自己的动态解决方案; 第四部分是附录, 其中有关键的DHTML元素列表、样式属性列表和颜色表等。本书将概念的介绍与实际应用结合起来, 每个新的概念都伴有一个实际项目来进行操作。本书的代码实例丰富, 读者可以借鉴这些代码立即制作出精美的网页。

James Meade, David Crowder, Rhonda Crowder: Microsoft Dynamic HTML.

Authorized translation from the English language edition published by The Coriolis Group, Inc.

Copyright 1998 by The Coriolis Group, Inc.

All rights reserved.

本书中文简体字版由机械工业出版社出版, 未经出版者书面许可。本书的任何部分不得以任何方式复制或抄袭。

版权所有, 翻印必究。

本书版权登记号: 图字: 01-98-1418

图书在版编目(CIP)数据

Microsoft Dynamic HTML 使用详解/(美)麦德(Meade,j.)等著; 潇湘工作室译.
-北京: 机械工业出版社, 1998.10

书名原文: Microsoft Dynamic HTML

ISBN 7-111-06758-4

I.M… II. ①麦… ②潇… III.超文本标记语言, HTML IV.TP393

中国版本图书馆CIP数据核字(98)第24654号

出 版 人: 马九荣 (北京市百万庄大街22号 邮政编码 100037)

责任编辑: 温莉芳 于 静

北京第二外国语学院印刷厂印刷 • 新华书店北京发行所发行

1998年10月第1版第1次印刷

787 × 1092mm 1/16 • 19.75印张

定价: 52.00元 (附光盘)

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

303

前言

我已经看到过许多次了。由于被以前的升级弄得疲惫不堪，人们对HTML另一个新版本的第一反应就是“打哈欠”。“几个新的标记而已”，他们这样认为。然后他们看到了这个新版本：Microsoft Dynamic HTML（也称为DHTML），跟着便倍加关注。“哦”，他们说，“看一看那些可能性”。“全新的生活方式”，有些人说，“HTML可能会成为应用程序的通用语言。”

即使是在软件革新的电子化世界中，Microsoft Dynamic HTML也不是每天都可以产生的那种突破性技术。由于DHTML，现在可以在Web页上做各种各样的事情，这些事情以前必须在服务器上才能完成，当然，如果你能找到一种方式来完成它们。动画物体、产生特殊效果、与用户交互、提供数据库信息，它就是一个全新的世界。但是，不要认为就仅仅只有这里所说的这些。自己去看吧，让本书带着你。

当然，使用这样一种突破性产品时需要掌握一些技巧。Microsoft DHTML把多种能力组合到了单个强大的软件包中。除了要求你使用期望的HTML代码之外，它还要求应用一些脚本编写的知识（这意味着需要使用某种形式的基本编程语言进行编程）。

并且，仿佛HTML和脚本编写还不够，DHTML还要求理解DHTML作用于Web环境中的对象结构（object structure）。你可以使用样式和样式表。了解一些有关Web页面的设计并没有坏处。如果想要真正认真地考虑商业应用程序，则最终很有可能还需要使用数据库。

当然，DHTML中的丰富可能性是本书真正的目的。我们的任务是解释所有这些丰富性，并使它们便于使用，你只需做很小的努力，就可以享受DHTML的强大功能。

为什么写本书

Microsoft DHTML是一种真正的突破性技术。程序员对它感到非常兴奋。现在可以在Web页面上做一些以前不涉及服务器就不可能完成的事情。你可以几乎即时地改变页面上文本或图像的大小、颜色和其他特性，可以把文本或对象从一个位置即时地移动到另一个位置，或者让它们连续地移动（这意味着产生动画）。

使用DHTML可以不必回到服务器更新文本或图像就改变页面上的内容（从而消除了长时间等待的必要，这是网络冲浪者所深恶痛绝的）。

使用DHTML，可以将Web页面与一个数据库相连接，从而使所有种类的商业应用程序都可以工作得非常快。例如，人们可以连接到其上，查看他们有多少渡假时间，或他们在选股计划中积累了多少资金。

所有这些功能都可以发生，其中最重要的部分是：让它们真正成为交互性的。用户单击，然后图像移动。用户输入名字，然后他们个人的股票信息就出现了。用户把股票数据按公司排序，然后就可以看到哪个公司为自己带来了最多的收入。

除了交互性之外，便是DHTML对象和事件模型。由于有了对象模型，就可以对页面上的任何内容进行脚本编写：任何词、任何段落、任何链接以及想要的任何东西。其可能性是无限的。

由于有了事件模型，你可以俘获任何用户行为（鼠标动作、键盘动作、窗口状态的改变等等），之后便可以使用俘获的行为来触发页面上的某些动作。用户一旦连接到并使用了DHTML后，便可以只使用他们的本地计算机来玩Web页面上的交互式游戏。

因此，只需要很低的价格便能够为Web作者和用户带来强大的功能。现在很流行声称所有的计算机程序都是“容易”而“简单”的，并且任何人都可以编写它们。我不想违背这种流行趋势，说DHTML有什么不同。因为它当然是容易理解的。

但是，对于它有很多事情要做，并且不同的事情来自于不同的目的。本书的目的就是为了让任何具有基本计算机知识的人来实现这种动态新媒介中的高级内容。

例如，程序员对于脚本编写通常做得非常好。但是，他们可能不知道HTML，并且可能不了解数据库（程序员和数据库人员属于两种不同类别的人）。HTML编写人员现在可能很多，很可能不了解脚本编写（或数据库，或样式表，或……你知道了吧）。而如果没有脚本编写，就不能享受DHTML的许多强大功能。

数据库人员对于HTML和脚本编写可能都感到不太适应，但是，很可能人们将对他们说，“我们需要在DHTML中运行我们的效益数据库。完成这件事情需要多长时间？”

样式表有点儿特殊，没有理由认为程序页、数据库人员或我所提到的任何人非常熟悉它。但是，如果在使用DHTML时不采用样式表，则会丧失很好的机会。

最后，几乎没有人真正熟悉对象和事件模型，因为它们在DHTML中是全新的内容。对象模型最初可能具有一点儿挑战性，但是，你将会需要用它来制作DHTML中的许多“动态”内容。

本书是很有必要的，学习本书，你就可以很容易地过渡到DHTML中，并利用其丰富的功能。其中有许多功能人们会长期忽略它们。甚至最顽固地憎恨微软公司的人在看到你可以按照所想的方式来编写页面中的任何内容，而且完全不涉及到服务器时，他们板着的面孔也会和缓起来的。

DHTML是一种突破。没准儿它会形成一种文化冲击。你需要本书来防止这种冲击，并引导你享受其强大的功能。

本书读者

本书假定你已经知道关于计算机的一些知识，并且可能已经使用过一些应用程序。你甚至可能创建过应用程序。在本书中没有太多的“当心，它将不会咬你”之类的介绍性内容。

你可能是一个程序员，但也不必一定是程序员。你可能从DHTML之前的HTML版本中了解到一些关于HTML的知识，但是，如果你只是刚刚才找到“标记”是什么东西等也没问题。你可以是使用计算机的设计人员；可能你只是很聪明；或者你能够坚持并且愿意做一些事情实现自己的愿望。

要点就是你至少知道一点点东西，你当然可以知道许多。不管你是从哪里开始（除了真正的从零开始），本书都将会对你有所帮助。

在本书最初出版时，实际上没有DHTML专家，因为它是一门新技术（以及一些现有技术的全新集成）。因而，本书定位的读者不是HTML的中高级用户，而是定位于那些聪明的计算机用户，他们很可能不了解DHTML，或至少不熟悉其一些组成部分。

本书的目的就是让你真正了解DHTML中的所有功能。然后，在后续章节中，你将有机会把一些综合的理解应用到高级应用程序中。这些后续章节为更高级的用户（以及读过本书前

面部分的用户)提供了一些内容。

DHTML"Explorer"指什么

当你探索一个领域时,将会熟悉其内部的每一项重要内容。当你探索一门技术时,也将会为了实用的目的来控制它。本书不仅仅是“介绍”DHTML,本书将深入探索它。

随着对本书的学习,你将不仅仅是看到概念,而且还要亲自来测试它们。在本书中先学习概念,然后在实际情况下尝试它们。

本书中的项目也可以从配套光盘上得到,其中包含你可以很容易地用于自己的Web页面的动态应用代码。本书最后的实例研究将把你带入到真实应用程序的世界中,并介绍如何创建自己的类似应用程序。

本书的目标不仅仅是介绍一门技术,而是赋予你使用其强大功能的技巧。

本书的内容

本书的主题为DHTML,这将要求介绍另外几种技术。没有一定程度的复杂性,你将不能继续下去。

而且,本书开始的几章介绍了DHTML中的独立主题(但不一定是“简单的”主题)。

在第一部分,你将发现DHTML中“Dynamic(动态)”的真正意义。在入门章节解释了DHTML及其突破性的能力之后,你将接触到这些能力,并尝试其每一种能力:动态样式、定位(非动态的)、动态内容、动态定位。你还将学习到使用JScript进行脚本编写的基本知识。在本书中发现一个概念之后,便可以立即在一个项目中使用它。

在第二部分,你将尝试一些实用工具,这些工具使DHTML享有“动态”的名称。在这部分中,将看到如何在Web页面上创建多媒体效果,如何从样式表中受益,以及如何理解和应用对象模型。在这一部分中,还将尝试DHTML的商业使用。在其中将看到如何获得数据库连接的好处,同时在本地计算机上独占式工作。你还将看到在页面上创建应用程序最有用的方法,即VBScript。并且,你将看到如何集成小应用程序与小脚本程序(它们在内部包含有完全的应用程序)。

第三部分面向那些真正想要采用DHTML创建Web站点的人。你将看到出于专家手中的模型站点,查看这些站点的内部工作原理,并了解如何应用在本书中所学到的内容来创建自己的动态解决方案。

最后,第四部分是一系列附录,它们不遵照通常的规则,它们包罗万象,有些内容必须在某些地方出现,但是从来没有人使用过。你应该使用这些附录。有一个附录的内容是关键DHTML元素列表,并提供了充足的解释和句法,让你可以在多数应用程序中正确使用它们。样式属性列表和颜色表将帮助你找到在把一个页面放置在背景上所需要的东西,或找到设置字体大小的方式。并且,尽管本书面向页面制作人员,而不是页面设计人员,但还是有一个附录总结了关于正确设计Web页面的当前观点。

最后还有一个词汇表,当你初次进入一个以前不擅长的领域时,它是非常有用的。

其他一些内容

“动态性”是你需要做的事情,而不是要你说的话。我们很高兴在这里介绍本书,但是,

使用DHTML得到动态效果的最好方式是自己亲自动手。

你在第1章中就马上动手做，这一章介绍了DHTML中的几乎每一个概念，你可以使用其中一些概念来创建Web页面。

然后，第2章显示了不使用脚本编写如何产生动态效果。之后，第3章介绍了使用脚本编写产生动态效果的事实真相。

Microsoft DHTML是一个全新的世界，我们愿意在本书中帮助你掌握它。如果你对它有新的发现，并且愿意让我们分享一些见解，我们很高兴能收到你的来信。我们的电子邮件地址是words@lisco.com和dcrowder@linkfinder.com。

目 录

前言

第一部分 立即获得动态效果

第1章 关于动态HTML的动态	1
1.1 创建“静态”HTML文档	1
1.2 查看DHTML页面	4
1.2.1 动态样式	4
1.2.2 动态内容	5
1.2.3 脚本编写和对象结构	7
1.3 查看另一个动态可能性	10
1.3.1 定位和动态定位	10
1.3.2 多媒体	10
1.3.3 数据绑定	11
1.3.4 减轻服务器负载	12
1.4 小结	12
第2章 创建动态样式和定位 (不用脚本编写)	14
2.1 使用事件获得动态	14
2.2 使用内嵌样式获得动态	19
2.3 定位(不使用脚本编写)	25
2.3.1 绝对定位和相对定位	26
2.3.2 在第三维空间中工作: 重叠	29
2.3.3 使用定位的元素	31
2.4 小结	37
第3章 开始使用脚本编写	39
3.1 脚本编写基础	39
3.1.1 脚本编写作为Internet编程	39
3.1.2 把脚本放入代码	40
3.1.3 创建事件处理程序	40
3.1.4 掌握脚本编写术语	42
3.2 在脚本中创建函数	48
3.2.1 使用函数取代或改变页面项	48
3.2.2 使事情有条件发生	50
3.2.3 使用循环控制程序的执行	51
3.3 什么地方出错	53

3.4 在Web上得到帮助	54
3.5 小结	54
第4章 添加动态内容和动态定位	56
4.1 使用文本获得动态	56
4.1.1 命名脚本中使用的元素: ID属性	56
4.1.2 使用innerText和outerText属性 替换文本	56
4.1.3 使用insert AdjacentText方法 插入文本	60
4.2 使用整个元素获得动态效果	61
4.2.1 把链接放置到页面上	62
4.2.2 替换一个元素、标记和全部	62
4.2.3 插入新元素	71
4.3 动态定位内容	72
4.4 小结	83

第二部分 揭开动态的秘密

第5章 多媒体效果基础	85
5.1 在元素上产生漂亮的效果: 滤镜	85
5.1.1 添加多媒体效果	85
5.1.2 使用样式来插入一种效果	109
5.2 用持续时间产生过渡效果	111
5.3 小结	114
第6章 创建和使用级联样式表	115
6.1 什么是样式和样式表	115
6.2 不使用样式表来创建动态样式	116
6.3 创建和链接样式表	118
6.3.1 在HTML文档中创建样式	118
6.3.2 在单独的样式表中创建样式	118
6.3.3 理解选择符	119
6.3.4 理解样式如何级联	131
6.3.5 读者的选择	132
6.4 管理和操作样式表	135
6.4.1 使用样式表内嵌附加字体	135
6.4.2 决定是否在样式表	135

6.4.3	对样式表添加规则	136
6.4.4	使用样式表来设置分页	137
6.5	使用样式表创建动态样式	138
6.6	CSS的Web资源	140
6.7	小结	140
第7章	通过对象模型获得动态效果	142
7.1	到底什么是对象模型	142
7.2	了解all集合	142
7.3	使用window对象	155
7.4	使用事件模型触发动态	162
7.4.1	使用event对象	162
7.4.2	使用事件冒泡方法高效地创建特殊效果	162
7.5	小结	165
第8章	数据库连接揭秘	167
8.1	使用数据	167
8.1.1	服务器端连接的问题	167
8.1.2	客户端端解决方案: 数据绑定	167
8.2	在客户机上绑定和显示数据	168
8.2.1	创建或访问数据源	168
8.2.2	绑定数据到页面	170
8.3	在客户机上操作数据	173
8.4	小结	176
第9章	建立高级动态数据库	177
9.1	完善的表策略	177
9.1.1	控制表显示的大小	177
9.1.2	筛选数据	178
9.1.3	设置显示哪一条记录	182
9.2	绑定到非表元素	183
9.2.1	为文本使用数据的元素	183
9.2.2	DATAFORMATAS属性	197
9.2.3	为URL使用数据的元素	198
9.3	小结	207
第10章	用VBScript 进行脚本编程	208
10.1	关键的VBScript概念	208
10.1.1	使用变量	208
10.1.2	加入运算符	208
10.1.3	使用消息框	209
10.1.4	创建子过程和函数	209

10.2 控制程序执行	212
10.2.1 if语句	212
10.2.2 Select Case语句	212
10.3 小结	215
第11章 应用控件和组件	217
11.1 用内置控件使生活更轻松	217
11.2 小应用程序、ACTIVEX控件和 小脚本程序快速入门	219
11.2.1 回顾小应用程序	219
11.2.2 浏览ActiveX控件	220
11.2.3 加入小脚本程序	220
11.3 小结	225
第12章 使用Microsoft Script Debugger	226
12.1 安装Script Debugger	226
12.2 运行Script Debugger	227
12.2.1 启动Script Debugger	227
12.2.2 查看并调试脚本源代码	229
12.2.3 修正错误	231
12.2.4 使用Command窗口	232
12.2.5 使用调用栈	233
12.3 有错误的调试器	233
12.4 小结	234

第三部分 实例研究

第13章 动态商业主页	235
13.1 查看Project Cool的样例Web页	235
13.1.1 Project Cool的建议	235
13.1.2 自己尝试一个简单的Web页	237
13.2 评价所显示的Project Cool页面	238
13.2.1 保持简单	238
13.2.2 组织清楚	238
13.2.3 确信人们可与之联系	239
13.2.4 加入帮助文本	239
13.2.5 让它更富有魅力	240
13.3 查看主页代码	240
13.3.1 加入颜色	243
13.3.2 加入公司名称	243
13.3.3 组织链接	244

第四部分 附 录

13.3.4 加入Email地址244

13.3.5 为运动的土星图像进行
脚本编程244

13.4 小结245

第14章 实例研究：样式表246

14.1 定位和显示样式表246

14.2 “利用设计者”和其他基本的Nielsen
建议248

14.3 “限制字体”和实际样式表的
其他建议249

14.3.1 确保页面在禁用样式表时仍然
工作正常249

14.3.2 不要使用两种以上字体250

14.3.3 不要用绝对字体大小250

14.3.4 不要用!important属性覆盖用户
的设置250

14.3.5 为链接使用标准颜色250

14.4 查看其他模型样式表251

14.5 小结251

第15章 实例研究：使用小应用程序252

15.1 查看使用小应用程序的页面252

15.1.1 查看旋转徽标253

15.1.2 查看蜡烛小应用程序255

15.2 回顾Java源代码256

15.3 小结272

附录A 词汇表273

附录B DHTML页面的设计思想280

B.1 开始：成为动态的……但要
有节制280

B.2 考虑的关键：内容，内容，内容280

B.3 创建设计来增强内容281

B.4 设计Web页面，而不是书本页面283

B.5 有用的参考283

附录C 关键DHTML元素284

附录D DHTML的CSS样式特性292

D.1 World Wide Web Consortium (W3C)
和CSS1292

D.2 使用属性292

附录E 颜色表295

E.1 使用color特性295

E.2 按名称使用颜色295

E.3 使用十六进制记法296

附录F Web制作工具300

F.1 样式表编辑器300

F.2 CSS转换器301

F.3 支持CSS/DHTML的HTML编辑器302

F.4 CSS样式检查程序303

第一部分 立即获得动态效果

第1章 关于动态HTML的动态

本书的全部内容都以动态为中心，但是，在这一章中，你将先体验一下DHTML中可能发生的事情：动态样式、动态内容、动态定位，甚至是（简而言之）多媒体效果。

对于每个人都用以制作Web页面的标记语言，微软公司赋予它一个全新的名字，这可能有点冒昧。也就是说，在每个人都知道并且使用老而平淡的HTML（微软公司有时也称之为“静态HTML”）的世界里，微软公司新推出了DHTML。

在这一章中，你将发现有关DHTML（Dynamic HTML）的许多内容，可以自己确定该名称是否恰当。首先，你将试一试老的HTML，它是DHTML的基础。你将发现超文本标记语言的基本构件：标记（tag）和元素（element）。

然后，你就开始了热身过程。你将发现使HTML成为DHTML的关键所在：样式（style）、属性（attribute）、脚本编写（scripting）以及对象结构（object structure）。利用DHTML中的样式，可以改变项目的外观，而不必编写脚本。在使用样式工作时，实际上是在使用HTML元素的属性进行工作。脚本编写，尽管我们不想说它，实际上就是编程，即使用JavaScript或VBScript编程。利用对象结构，可以给页面上的每一个元素分配一个地址，以便在脚本中引用它们。

看到了DHTML的本质之后，你将了解该语言的几个更深层的内容：定位（positioning）、多媒体（multimedia）和数据绑定（data binding）。多媒体涉及到动态样式，具有新奇的输出结果。定位是指定一个元素出现在页面上什么位置的能力。数据绑定则是新的扩展功能，利用它可以从一个外部数据库绑定数据到DHTML文档。

最后，简要介绍从这种称为动态的语言中所能得到的最好结果，即减少了服务器的负载。

1.1 创建“静态”HTML文档

你已经知道的关于如何创建HTML文档的所有知识同样适用于编写在Internet Explorer 4（IE4）中显示的DHTML。DHTML并不是一种完全不同的语言，而是使用HTML的一种新方法。在DHTML中，仍然使用以前用过的熟悉的HTML元素，但是，你将能更好地控制它们。使用DHTML，不是简单地创建一个静态Web页面，这个页面只有一种而且是唯一一种外观，你可以创建一个能够响应鼠标单击甚至随时间而改变的动态页面。为了查看HTML与DHTML的共同之处，最好的方式就是编写一个简单的文档，如在本章后面的“练习：编写简单的文档”一节中所做。

提示 IE4是唯一能完全支持Microsoft DHTML的并能显示用DHTML代码编写的所有内容的浏览器。如果试图在IE4中显示用普通HTML编写的页面，没有问题。但是，如果反过来，在IE4之前的浏览器中显示DHTML，就会出现问题了。据微软公司说，页面

质量会“适度降低”，但仍然可以显示。但是，DHTML的“动态特性”将会丢失。例如，在IE4中可以缩放的轮廓在其他浏览器中将只简单地显示为一个完全展开的框。

创建HTML文档

DHTML不仅仅是对HTML的一系列扩展，也就是说，不是一些按以前使用HTML的方式来使用的新元素。由于其对象结构，页面上的每一个元素都是一个单独的对象，具有可以控制的属性，在某些方面，DHTML与传统的HTML彻底脱离开来（在本章后面将更详细地介绍对象结构）。

但是，你已经知道的关于HTML的任何知识都适用于DHTML。HTML由标记、元素、属性和其他内容（像文本和图像）等组成。动态DHTML也是一样！除此之外，由于其对象结构，使用DHTML，你可以使所有标记和元素成为动态对象。以前这是不可能的。

当然，微软公司也无法抵制添加几个新扩展功能的诱惑。当在第三部分中介绍数据绑定时，你将会看到其中一些。

考虑标记和元素

在下一节中，你编写的代码包含所有HTML文档的基本组成元素。下面是简短的介绍。

位于小于号和大于号之间的文本部分是标记，这是HTML的基本构件。它们是一些指令，告诉浏览器要做什么事情。每一个元素都具有一个开始标记，告知该元素从哪儿开始。

用一个斜杠开始的标记是结束标记，例如</HTML>，它标志一个特定元素的结尾。任何元素的结束标记，除了开头的斜杠之外，与该元素的开始标记是完全相同的。不是所有的元素都具有结束标记，有些元素的结束标记可有可无，不是非有不可。

标记是不区分大小写的。不管它们是大写、小写，还是混合大小写，浏览器都可以读取它们。对于Web浏览器，<HEAD>与<head>或<HeAd>是完全相同的。但是，在实际编程中，你会发现，把它们全部大写会很有用，因为可以很容易地从其他文本中认出它们。

全部字符串（开始标记，标记内的任何内容，以及结束标记）是一个元素。例如，下面是一个元素：

```
<P>Click the "dynamic" text below, and it gets bigger.</P>
```

属性告诉你元素的特殊特征。它们就像是程序中或数学中的变量一样。当代码执行时，浏览器给它们赋值（在标准HTML中载入页面时，或在使用DHTML指定的情况下）。例如，<H1>这个标题元素包含有一个ALIGN属性：

```
<H1 ALIGN="left">This is a document heading.</H1>
```

当浏览器执行这一行时，它将这个标题左对齐。与所有标准HTML一样，其对齐的确切位置取决于正在使用的浏览器。

项目 编写简单的文档

一开始，我们要创建一个简单的HTML文档，不包含有任何动态元素。在本章后面，将让这个文档成为动态的。

下面是这个范例：

1) 启动一个文本编辑器，用于创建HTML文档。许多Web制作人员使用Windows的“记事本”。

2) 在“记事本”中，输入如下范例文本，如图1-1中所示：

```
<HTML>
<HEAD><TITLE>Text that Grows</TITLE></HEAD>
<BODY>
<P>Click the "dynamic" text below, and it gets bigger.</P>
</BODY>
</HTML>
```

3) 把记事本页面保存为HTML文档。我们使用名字Basic.txt.htm(原文误为BasicText.html,译者注)。使用你的操作系统的过程把这个文档保存在想要的目录中。

提示 要确保使用了扩展名.HTM或.HTML。否则,记事本将把该文件保存为文本文件,并且,尽管你可以从浏览器中打开它,但是,如果不具有.HTM或.HTML扩展名,浏览器将不会把它识别为HTML页面。所有的浏览器将要做的事情就是把它作为文本文件显式,除非扩展名告诉它该文件是一个Web页面。

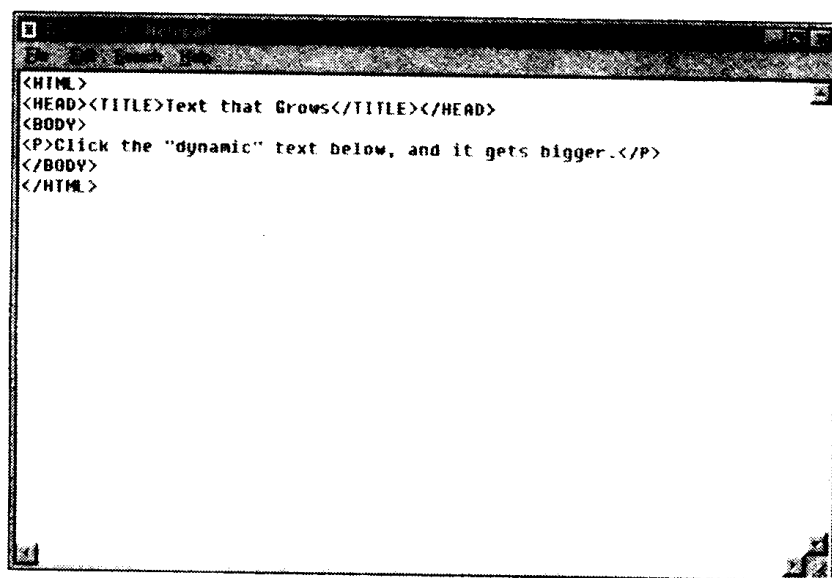


图1-1 这里是记事本中的范例文本

4) 关闭记事本。

5) 在IE4中本地显示该文件。只需要在File(文件)、Open(打开)上单击,然后单击Browse(浏览)按钮,找到包含该文件的文件夹,并指定刚才所创建的文件。在OK(确定)按钮上单击。这个文档将会显示出来,如图1-2中所示。

提示 查看代码

使用记事本显示当前文本代码的最快方式是,从IE4主菜单上单击View(查看)、Source(源文件)。许多HTML作者都有在记事本和浏览器之间切换的习惯。在记事本中编写代码,保存该文档,然后在IE4中打开它查看其效果。然后再回到记事本中修改。不断地进行尝试,像任何普通的HTML作者一样。

在浏览Web时,使用记事本查看其代码并没有什么问题,但是,当使用记事本打开本地文件时,可能会遇到一个小问题。如果从一台Unix计算机下载压缩的HTML页面,并在PC上对它解压缩,则将没有从Unix换行符到PC换行符之间的转换。在这种情况下,即使该网页可以在IE4中很好地显示,但是,使用记事本查看其源代码将会显示许多小黑框,它们与HTML代码混合在一起。

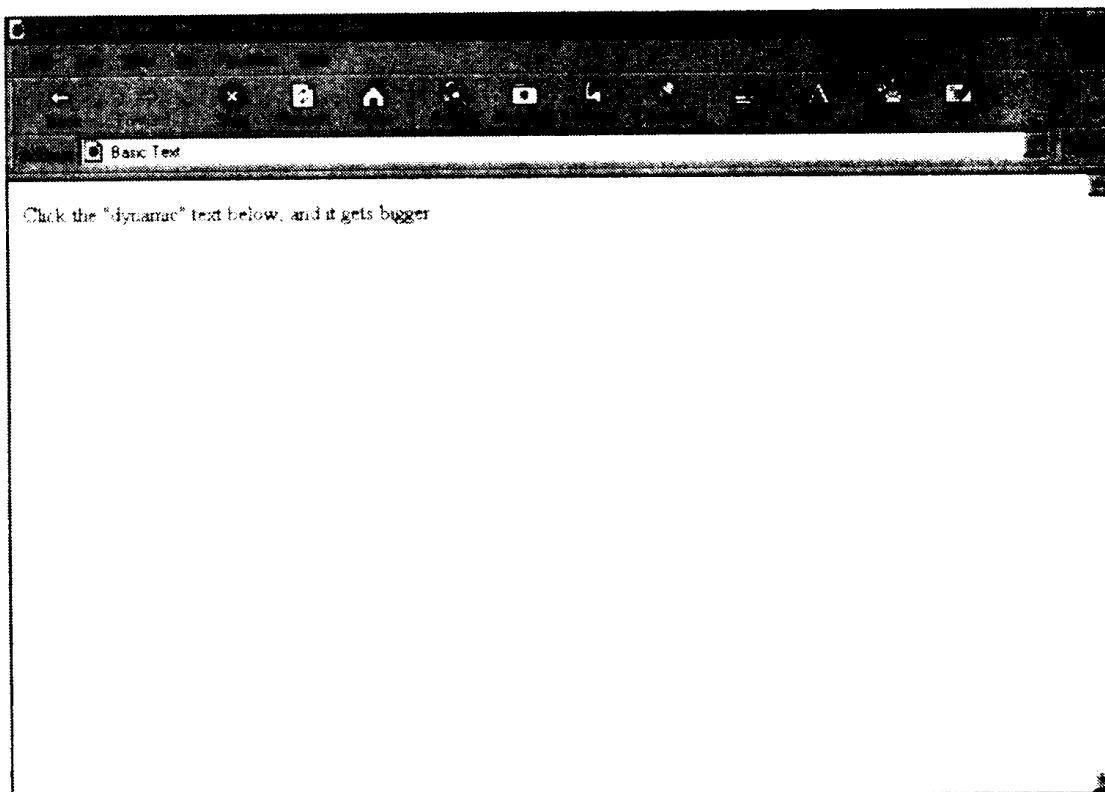


图1-2 刚刚创建的简单文档

1.2 查看DHTML页面

如前一节中所示，在浏览器中，HTML文档的标记代码显示为一个页面。在本章的第一个范例中，显示的页面就静止在那儿。你不能与之交互，它也不会改变。但是，使用了DHTML之后，显示将成为动态的，这意味着可以动态改变其样式和内容，而不必返回到服务器下载一个新Web页面，以替换掉现有的页面。

1.2.1 动态样式

在这种新的DHTML中，样式变得比以前更具有交互性了。在DHTML中，样式就是指内容如何显示在页面上，而不是指其具体的内容。样式包括颜色、字样、间距、缩进、页面上的位置，以及文本的可见性。

在普通HTML中，如果想要改变一个样式，就必须更新整个页面。实质上，必须返回到服务器，并产生一个新的页面，而不是仅仅改变想要改变的元素而不涉及到服务器。尽管从本质上来说，更新页面的过程对于用户和HTML作者都是透明的，但是，它消耗了不必要的时间和资源。如果更新的是复杂的页面，则将会需要大量时间。在这段时间，你可以来一份三明治，喝一杯咖啡，甚至去接听电话，并且，每次在有任何元素发生改变时，都必须经历这个过程。

当页面需要更新时，将发生与第一次访问它时相同的过程。而Web浏览器遵照超文本传输协议（HTTP）标准，发送一个信号给存储该页面的服务器。如果成功地建立了联系，它随后给服务器发送一条命令，其中包含对该文档的请求（GET命令）、文档的名称以及浏览器正在使用的HTTP的版本。浏览器还要指出它自身以及它运行的平台，然后，声明它将从Web服务

器接收的文件类型，以响应该请求。

Web服务器然后告诉浏览器它正在使用的HTTP版本，以及浏览器的请求是否成功。在此之后，它发送声明日期和时间的信息，指出服务器软件，并包括几行关于请求文档的数据。最后才发送实际的Web页面。整个过程从页面上的第一幅图像开始，然后对下一幅图像又重复这个过程，依此类推。

哪怕只是想要改变标题的颜色，也要如此折腾一番。当然，使用老的方法，不可能仅改变一项。实际上是需要重新创建整个新的Web页面，除了一点细微的差别之外，与以前的完全相同。并且对每一种可能的改变，都要这样做。

避免这种（使用预DHTML方法）情况的唯一办法就是创建一个公共网关接口（Common Gateway Interface, CGI）程序，它可以为你创建新的Web页面。但是，这不能算作是一种解决方法。CGI编程比使用脚本编程和DHTML要困难得多。使用CGI，仍然需要经历HTTP请求过程，但是，Web服务器上的负载可能会更高，并且处理时间甚至会更长，因为Web服务器在可以发送响应给浏览器之前需要运行CGI程序，并生成其输出。

CGI程序通常是使用Perl语言编写的，这是一种非常难的编程语言。但是，它也可以使用C语言甚至一种Unix脚本来编写。当然，这意味着你必须已经是一个熟练的程序员，至少是在开始使用CGI之前愿意去掌握Unix。你还必须在Web服务器上具有高级安全权限，以享有cgi-bin的访问权（cgi-bin通常是存储CGI脚本的目录），而大多数的Internet服务提供者又不情愿给它们公司之外的人授予这种权限。

许多ISP提供的折衷方案是让他们内部的程序员检查并批准你的程序，然后，他们自己安装该程序。这不仅延长了安装时间，而且，通常你每月会花费更多的机时费用。它还有效地防止了你修补或润色程序，因为每一次想要修改它时，都将需要再次通过审查过程。在这种情况下，要想精通CGI编程艺术，几乎是不可能的。

DHTML为这些问题提出了一个完美的解决方案。它把控制你的Web站点的权力交回到你的手中，这个权利本来就属于你。它使用了一种编程语言，这种语言使用和学习起来都要比CGI标准容易得多，并且功能强大，足以让你做所能想像到的任何事情。最后，它不必从远方的Web服务器请求并下载新的文档，就可以工作得很好。

在DHTML中，可以改变标题的颜色和大小，而不必返回到服务器并重新更新页面。例如，看看图1-3中简单的Web页面（我们将在本章后面给出一个类似范例的代码，这里只是看一下其动态样式）。如果在IE4中单击该文本，将会得到如图1-4所示的页面。文本行“This text can change!”变大了，并且还改变了颜色。

当然，这里仅改变了几个词和大小，这是一个简单的范例。但是，这种改变页面样式的能力为Web页面作者提供了好几种可能性。改变页面上的内容就是在与用户交互。用户做一些事情，然后页面做一些事情，所有这些都不必返回到服务器。例如，如果想要强调推荐产品的文本，则可以把该产品名称设计成当鼠标接近它时改变大小和颜色。

1.2.2 动态内容

另一个DHTML特征就是微软公司称之为“动态内容”（dynamic content）的东西。你可以像改变样式一样改变页面上的词和图片。此外，使用DHTML，页面的内容可以响应简单的用户事件而改变，例如鼠标单击，并且，你不必像在普通HTML中一样要建立多个页面以产

生需要的结果。

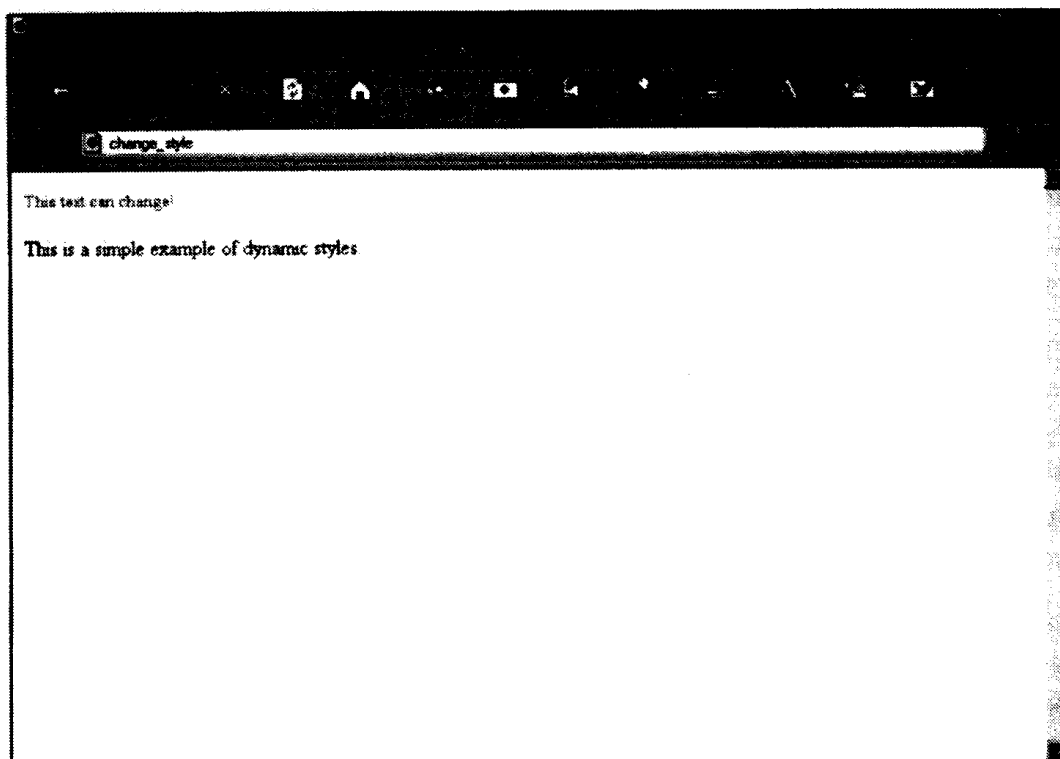


图1-3 单击之前的动态页面

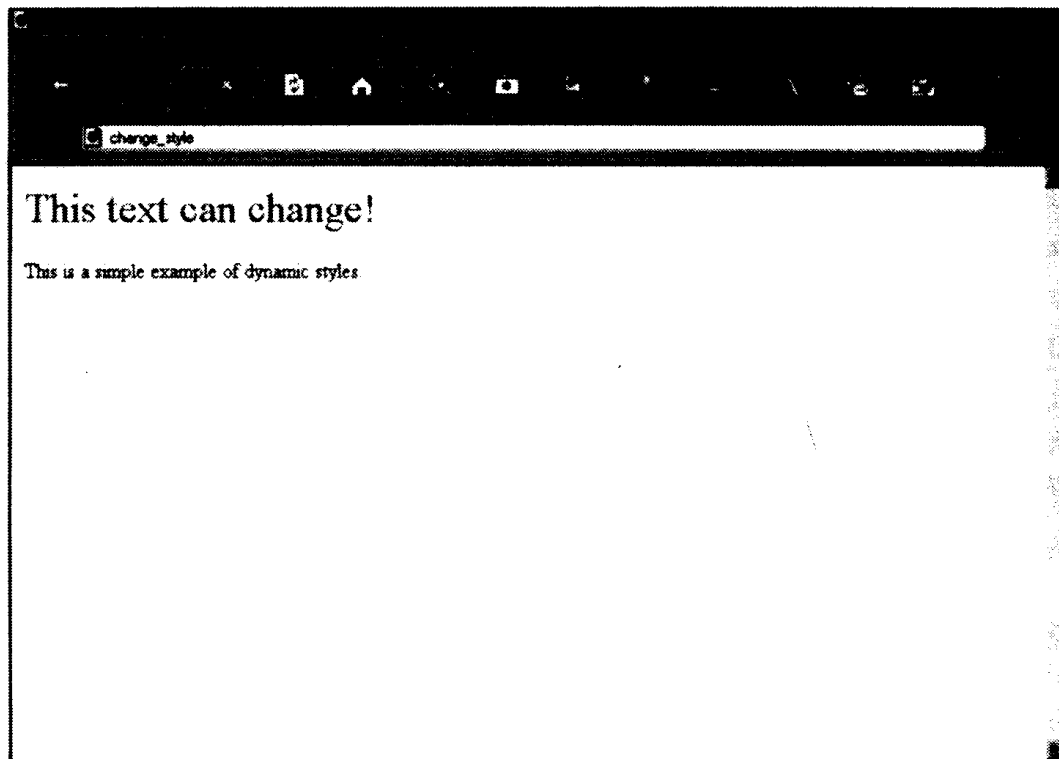


图1-4 单击之后的动态页面

动态内容有许多种可能的用途。例如，可以在用户单击某个文本时显示弹出式文本解释。图1-5在IE4的频道栏中显示了弹出式文本。

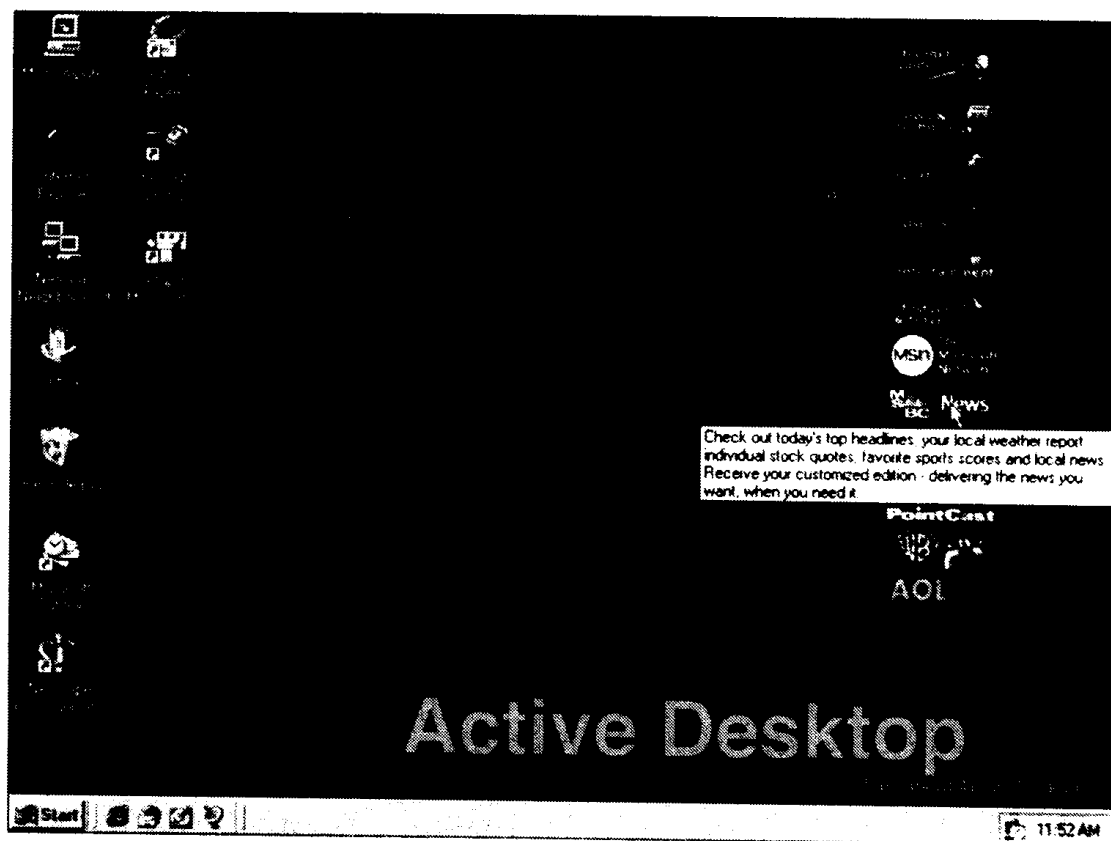


图1-5 使用动态内容可以显示解释性文本

当人们在Internet上进行一次搜索时，他们总是回到一个列有目的Web站点的总结页面。使用DHTML，你可以对搜索结果进行编程，以使用户鼠标指向一个结果时将显示一些附加信息，从而提供关于该站点的描述，这比站点的URL（Uniform Resource Locator，统一资源定位符）更有意义。

在动态内容的另一个应用中，可以使用脚本来扫描页面元素，并在页面的开头插入一个目录。如果内容发生改变，则该目录也会改变。

而且，动态内容的使用不仅仅局限于文本。页面上的任何元素都是可以操作的对象。如果想要改变所显示的图像，也可以。你可以建立一个具有数百项的编目，而只编写一个Web页面的代码来显示它们，一次显示一个。你甚至可以建立完全由可替换的元素组成的Web页面，根据用户的不同行为做出相应的改变。

1.2.3 脚本编写和对象结构

动态样式和内容是使用DHTML的两个迷人之处。但是，作为Web作者，你可能会有这样的疑问：如何实现它呢？其答案最开始没有人会喜欢，但是，当我们进一步了解它之后，将会感到它是最好的。为了在DHTML中使页面上的内容发生改变，通常需要编写脚本。

脚本就是计算机程序，通常用JavaScript或VBScript编写，然后由浏览器负责解释执行它。脚本可以是简单的，但是必须要编写它们。脚本获取页面上的某些元素，然后响应某些用户事件而改变它们。我们将在第3章中更详细地介绍脚本。它就是编程，它不是琐碎小事。幸运的是，对于初级DHTML程序员，已经有许多可用的脚本，可以根据自己的用途来研究并修改它们（在第3章中，我们将告诉你在哪里可以找到这些脚本）。

在页面上产生动态效果的另一种方式就是使用样式。在一个文档中指定样式的改变时，甚至不需要使用脚本，并且浏览器将在页面上显示出这些改变。在一个内嵌样式中使用STYLE=属性，如第2章中所述。

DHTML与普通HTML的最大差别就是，在DHTML中，你可以获取页面上的每一个元素，并对它进行编程。每一个元素在对象结构中都有一个地址。在第7章中，我们将更详细地介绍对象结构，但是，你不必完全了解它就可以使用它。对象模型的关键在于它为每一个元素提供了一个地址，即索引值（index value）。然后，你可以在脚本编写语句中引用该索引值，并对该元素做一些事情。在DHTML之前，做这些事情是根本不可能的。是对象结构和脚本编写使得DHTML具有动态特性的。

大写D与小写d的Dynamic

名称会导致混淆。微软公司和Netscape（90年代浏览器大战的两个参与者）都建立了一种称为“动态HTML”的语言。第一次出现在Netscape Communicator浏览器中的Netscape版本，在动态中使用了小写的“d”。而微软公司的版本则具有一个大写的“D”。尽管每个人都希望二者最终能合并，但是在目前，它们是两个不同的动态HTML版本。尽管在动态HTML的简单样式或CSS方面二者的工作原理大致相同，但是，在微软公司的动态HTML中提供了一些特殊的东西，例如数据绑定，它在Netscape的动态HTML中不能起作用，反之亦然。

项目 创建第一个DHTML文档

这里的动态样式示例将显示如何使用对象模型使网页成为动态的。图1-6显示了一个具有一些文本的Web页面，而图1-7则显示了在用户单击了它之后的相同页面。

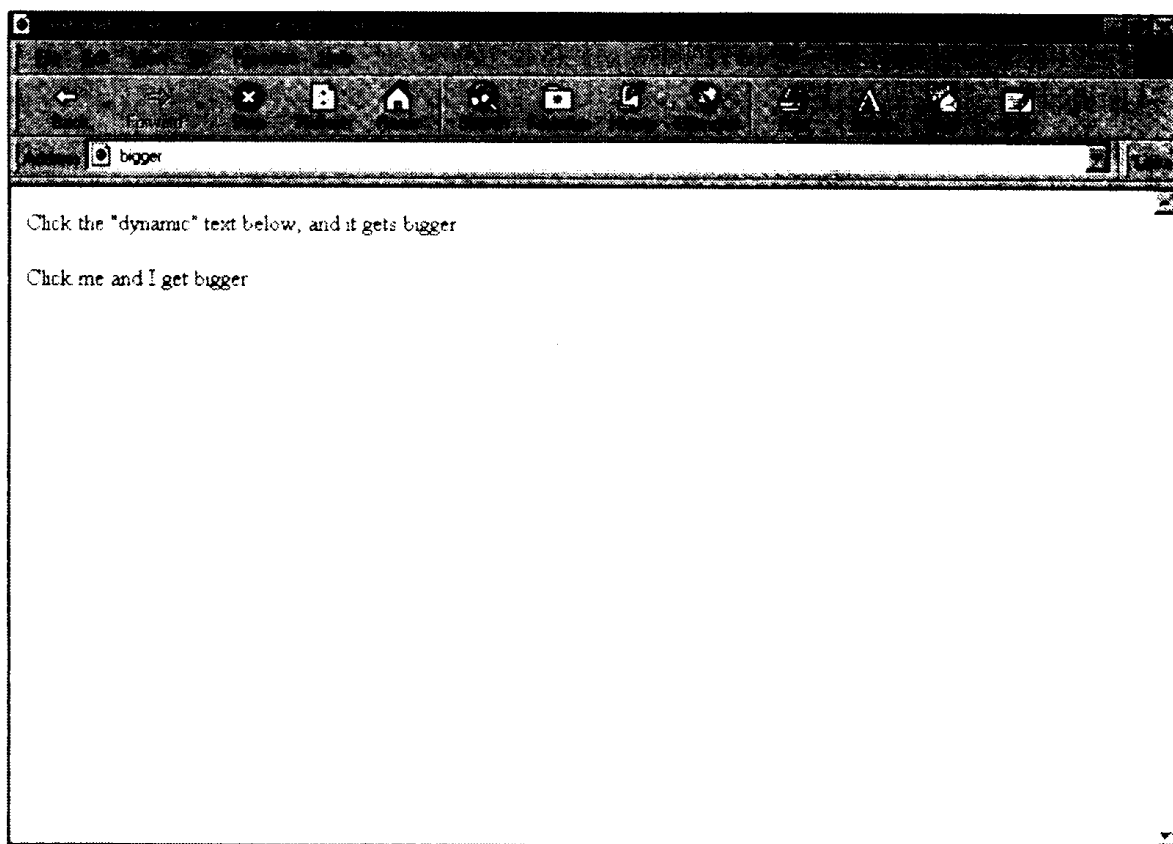


图1-6 这里是单击之前的文本