

入门与提高

杨 浩 编著

Java Script



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



软件入门与提高丛书

JavaScript 入门与提高

杨 浩 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

JavaScript 是一种功能强大的脚本设计语言, 本书全面详细地介绍了有关使用 JavaScript 来进行开发、维护、管理和设置 Web 应用程序的内容。

由于 JavaScript 的各种浏览器实现方案会存在一定的差异, 使得在 Netscape Navigator 上能正常运行的 JavaScript 脚本不一定在 Microsoft Internet Explorer 中会有同样的结果, 所以, 为适应广大读者的实际需要, 本书在以 Netscape Navigator 的 JavaScript 规范为主要讲述对象的同时, 还特地讲述了 Microsoft Internet Explorer 的 JavaScript 使用方法。

本书从 JavaScript 的基础开始讲起, 一直讲到 JavaScript 的中、高级应用以及和其他 Web 开发管理工具的协调工作, 内容新颖, 实例翔实。本书既可满足普通用户希望了解和熟悉 JavaScript 的入门的需要, 也可作为 Web 站点的设计、开发和维护人员必备的技术参考书。

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: JavaScript 入门与提高

作 者: 杨 浩

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: 北京市清华园胶印厂

发行者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印 张: 20.5 字 数: 507 千字

版 次: 1999 年 10 月第 1 版 2001 年 2 月第 7 次印刷

书 号: ISBN 7-302-03744-2/TP·2094

印 数: 25001~31000

定 价: 26.80 元

《软件入门与提高丛书》特色提示

- ☑ 精选国内外著名软件公司的流行产品,以丰富的选题满足读者学用软件的广泛需求
- ☑ 以中文版软件为介绍的重中之重,为中国读者度身定制,从而便捷地掌握国际先进的软件技术
- ☑ 紧跟软件版本的更新,连续推出配套图书,使读者轻松自如地与世界软件潮流同步
- ☑ 明确定位面向初、中级读者,由“入门”起步,侧重“提高”,愿新手老手都能成为行家里手
- ☑ 围绕用户实际使用之需取材谋篇,着重技术精华的剖析和操作技巧的指点,使读者深入理解软件的奥秘,举一反三
- ☑ 追求明晰精练的风格,用醒目的步骤提示和生动的屏幕画面使读者如临操作现场,轻轻松松地把软件用起来

丛书编委会

主 编 李振格

编 委	李幼哲	黄娟娟	周海鹏
	许振伍	吕建忠	王 冬

《软件入门与提高丛书》序

使普通用户用电脑最关键也最头疼的恐怕就是学用软件了。软件范围之广，版本更新之快，功能选项之多，体系膨胀之大，往往令人目不暇接，无从下手；而每每看到专业人士在电脑前如鱼得水，把软件玩得活灵活现，一定又是惊美不已。

“临渊羡鱼，不如退而结网”。道路只有一条：动手去用！选择您想用的软件 and 一本配套的好书，然后坐在电脑前面，开机、安装，按照书中的指示去用、去试，很快就会发现您的电脑也有灵气了，您也是个出色的舵手，能自如地在软件之海中航行了。

《软件入门与提高丛书》的推出就是为了给您一套畅游软件之海的导航器。它是一套包含了现今主要流行软件的易学易用的使用指导书。既可循序学习，亦可随查随用，使您学有所依，用有所循，快速便捷地掌握软件的操作方法和编程技术，得心应手地解决实际问题。

让我们来看一下本丛书的特色吧！

■ 软件领域

本丛书所精选的软件皆为国内外著名软件公司的知名产品，也是时下国内应用面最广的软件，同时也是各领域令人瞩目的佼佼者。目前本丛书所涉及的软件领域主要有操作平台、办公软件、编程工具、数据库软件、网络和 Internet 软件、多媒体和图形图像软件等。本丛书还将密切注视新软件的面世，及时推出新软件以及虽然应用面稍窄但技术重要的软件产品的配套书。

■ 版本选择

本丛书对于软件版本的选择原则是：紧跟软件更新步伐，以最近半年推出和未来半年即将推出的最新版本为重点，充分保证图书的技术先进性；兼顾经典主流软件，给广受青睐、深入人心的产品以一席之地；对于兼有中西文版本的软件，尽量取中文版而舍西文版，以全力满足中国用户的需要。

■ 读者定位

本丛书明确定位于初、中级用户。不管您以前是否使用过本丛书所述的软件，这套书对您都非常合适。

本丛书名中“入门”的含义是指，对于每个软件的讲解都从必备的基础知识和基本操作讲起，新用户无需参照其他书即可轻松入门；老用户亦可从中快速了解新版本的新特性和新功能，自如地踏上新的台阶。至于书名中的“提高”，则蕴涵了图书内容的重点所在。以我们的经验，当前软件的功能日趋复杂，不学到一定的深度和广度是难以在实际工作中应付自如的。因此本丛书在让读者快速入门之后，就以大量明晰的操作步骤和典型的应用实例，教会读者更丰富全面地使用软件技术和应用技巧，使读者真正对所学软件融会贯

通、熟练在手。

📖 内容设计

本丛书内容设计的策略,是在仔细分析用户使用软件的困惑所在,并结合目前电脑图书市场现状的基础上确定的。简而言之,就是实用、明确和透彻。既不是面面俱到的“用户手册”,也并非详解原理的“功能指南”,而是独具实效的操作和编程指导书。一切围绕用户的实际使用需要选择内容,使读者在每个复杂的软件体系面前能“避虚就实”,直指目标;对于每个功能的讲解,则力求以明确的步骤指导和丰富的应用实例准确地指明如何做,读者只要按书中的指示和方法做成、做会、做熟,再举一反三,就能扎扎实实地轻松过关。

📖 风格特色

本丛书在风格上力求文字精练、图表丰富、脉络清晰、版式明快。另外,在策划写作时还特别设计了一些非常有特色的段落,以在正文之外为读者指点迷津。这些段落包括:

💡 **注意**——提醒可能出现的问题和容易犯的错误,以及如何避免,让您少一些傻眼的时刻和求救的烦恼。

📖 **提示**——提示可以进一步参见的章节,以及有关某个内容的详细信息,使您可深可浅,收放自如。

🔗 **技巧**——指点一些捷径,透露一些高招,让您事半功倍,技高一筹。

🖱️ **试一试**——精心设计各种操作练习,只要照猫画虎,试上一试,就不仅能在您的电脑上展现出书中出现的美妙画面,还能了解书中未详述的其他实现方法和可能出现的其他操作结果。随处可见的“试一试”,让您边学边用,时有所得,常有所悟。

经过紧张的策划、设计和创作,本套丛书已陆续面市,市场反应良好。许多书在两个月内迅速重印,自面世以来,本丛书已累计售出近百万册。在大量的读者反馈卡和来信中给我们提出了很多好的意见和建议,使我们受益匪浅。严谨、求实、高品味、高质量,一直是清华版图书的传统品质,也是我们在策划和创作中孜孜以求的目标。尽管倾心相注,精心而为,但错误和不足在所难免,恳请读者不吝赐教和指正,我们定会全力改进,在后续工作中提高。

本丛书在创作过程中得到了微软中国公司产品部的大力支持,对于他们在软件和技术资料的提供及有关目录的审定方面所给予的协助,表示衷心的感谢。

《软件入门与提高丛书》编委会
1999年5月

目 录

前言	1
第 1 章 Internet 及网络开发入门	3
1.1 什么是 Internet	4
1.2 JavaScript 简介	8
1.3 其他常用 Web 开发语言简介	18
1.3.1 VBScript 简介	18
1.3.2 Java 简介	19
1.3.3 CGI 简介	20
第 2 章 初识 HTML	21
2.1 HTML 基本标注	22
2.1.1 版面设计标注	26
2.1.2 具有字符控制功能的标注	30
2.1.3 颜色属性	34
2.1.4 在页面中加入注释	35
2.2 HTML 的图像、表格和链接	35
2.2.1 插入图像	35
2.2.2 引用表格	37
2.2.3 列表标注	43
2.2.4 使用链接	46
2.2.5 使用编程标注	49
2.3 HTML 中的表单	50
2.4 在表单中使用界面对象	51
2.5 在 HTML 中使用 JavaScript	62
第 3 章 JavaScript 的开发环境	65
3.1 JavaScript 开发环境总览	66
3.2 利用 Microsoft FrontPage 开发	66
3.3 利用 ActiveX Control Pad 开发	69
3.4 利用 JavaScript Editor 开发	73

第 4 章 JavaScript 的变量、表达式和操作符	79
4.1 JavaScript 的类型和变量	80
4.2 JavaScript 的常量	84
4.3 JavaScript 的表达式	85
4.4 JavaScript 的操作符	87
第 5 章 JavaScript 的基本语法	93
5.1 JavaScript 语句总览	94
5.2 变量声明和赋值语句	94
5.3 函数定义语句	95
5.4 条件和分支语句	100
5.5 循环语句	103
5.6 对象操作语句	107
5.7 注释语句	108
第 6 章 JavaScript 的对象模型	109
6.1 JavaScript 中对象的概念	110
6.2 对象的属性和方法	112
6.3 创建新对象	116
第 7 章 内置对象和函数	121
7.1 基本内置对象和函数总览	122
7.2 几个基本的内置对象	122
7.2.1 String 对象	123
7.2.2 Math 对象	127
7.2.3 Date 对象	137
7.2.4 Array 对象	142
7.2.5 Event 对象	145
7.2.6 RegExp 对象	146
7.3 JavaScript 的内置函数	151
7.4 JScript 中的内置对象和函数	153
第 8 章 浏览器对象详解	159
8.1 使用浏览器对象	160
8.2 使用 Netscape Navigator 的对象	165
8.2.1 浏览器对象 navigator	165
8.2.2 浏览器对象 Window	166
8.2.3 浏览器对象 document	173
8.2.4 其他浏览器对象	174

8.3 使用 Internet Explorer 的对象	179
8.4 HTML 网页的外观设计	181
第 9 章 在 JavaScript 中处理事件	185
9.1 事件处理的基本概念	186
9.2 事件处理句柄总览	188
9.3 常用事件及处理	191
9.3.1 浏览器事件	191
9.3.2 鼠标事件	194
9.3.3 键盘事件	204
9.3.4 其他事件	205
9.4 高级事件处理方法	207
第 10 章 JavaScript 高级使用技巧	209
10.1 使用帧对象	210
10.2 使用 Cookie 和 URL 存放用户信息	215
10.3 使用图形	222
10.4 使用声音	229
第 11 章 使用 JavaScript 创作	235
11.1 程序示例：控制浏览器的外观和属性	236
11.2 程序示例：网页管理器	247
11.3 程序示例：桌面日历	259
11.4 JavaScript 程序排错	270
第 12 章 与其他 Web 工具协同工作	275
12.1 JavaScript 的可扩展性	276
12.2 在 JavaScript 中使用插件	276
12.3 在 JavaScript 中使用 Java	284
12.4 JavaScript 的局限性以及未来展望	288
附录	290
附录 1 JavaScript 保留字一览表	290
附录 2 JavaScript 常用对象一览表	290
附录 3 JavaScript 颜色对照表	309
附录 4 Internet 上有关 JavaScript 的网站、讨论组以及其他	311

前言

JavaScript 是一种通用的、基于原型的、面向对象的脚本语言，它当初的设计目标是在不占用很多系统和网络资源的情况下提供一种可以嵌入不同的应用程序的通用代码。JavaScript 主要从 Java 借用了不少的语法形式，但也受到别的语言或开发工具的影响，如 Perl 和 Awk。

JavaScript 是一种高级的脚本描述性语言，并不需要依赖于特定的机器和操作系统，所以说它是独立于操作平台的。JavaScript 最初是在 Netscape Navigator 2.0 以及 Netscape LiveWire 1.0 上实现的。典型的 Netscape Navigator 3.0 实现了 JavaScript 的 1.1 版本。本书将试图在讲解语法概念时尽量不涉及某种具体的解决方案，而是直接描述这种语言的标准对象和函数。这样，尽管 JavaScript 最初是一种在 Netscape Navigator 和 Netscape LiveWire 上实现的脚本语言，但本书所讲述的内容则同样适用于 Microsoft Internet Explorer 浏览器的环境以及 UNIX、Linux 下的各种浏览器。

值得一提的是，尽管 JavaScript 最终是要朝统一的规范化方向发展的，但从目前来说，由于各软件公司之间存在着不同的商业利益，所以各种浏览器软件所能支持的 JavaScript 语言集合也不完全一致。这就使本书难于全面兼顾各种浏览器的各种特例情况，并使有些个别的示例程序不适用于某种浏览器。为了保持叙述的连贯性，本书在示例程序中将主要讨论 Netscape Communicator 的环境，当然也会兼顾到 Microsoft Internet Explorer 的需要。在作者认为可能会导致发生脚本不兼容的情况时，将会专门注明浏览器的种类，务请读者留意，不要产生误解。

上面已说过，JavaScript 的设计目标是要尽可能少地占用系统和网络资源。那么，它在技术上又是采用了什么方案来实现的呢？简单地说，在传统的网络方式中，当一个用户要向存储和运行在远端主机上的程序发送信息的时候，信息在客户端是不经过任何处理就直接提交到远端主机上的。如果信息量很大，就会严重地影响网络的性能。如果能有一种机制，使之可以在本地客户机上对信息进行初步加工，而只传送加工好的信息，就可以保持网络传送的速度和效率，提高网络的运行性能。JavaScript 就是这样一种机制。你也可以想象有一个 JavaScript 的可执行程序在你的机器上运行。对于一个典型的含有 JavaScript 的 HTML 文件，浏览器可以识别 JavaScript 语句，并由此响应诸如鼠标单击、表格输入和浏览页面等方面的操作。例如，可以使用 JavaScript 编写一个小程序，来验证用户输入密码的合法性。如果验证不通过，就可以让 JavaScript 操纵浏览器弹出一个对话框，也可以通过 JavaScript 来播放一段声音作为警告，或执行一段 Java 的程序等。这就隐含着必须要有一个解释程序来处理嵌在 HTML 文件中的 JavaScript 所描述的运行动作。总之，正像你在本书中将会学到的，你可以使用 JavaScript 来充分利用客户端的资源而使得 Web 页面更加富于交互性，更加有趣。

正如从 JavaScript 和 Java 的名字就可以看出的那样，它们是很相似的。但是它们到底还是不大一样。Java 是一种比 JavaScript 更复杂得多的语言，常用于更快和更安全的执行；而 JavaScript 则是一种相当容易理解的语言。由于许多 Java 的特性在 JavaScript 中并不支持，所以 JavaScript 的程序员可以不那么注重编程技巧。当然，JavaScript 也有自己独特的一些特性。例如，JavaScript 是一种动态类型化的脚本语言。也就是说，在程序里并不需要声明变量类型，变量的类型是不定的，可以在运行时动态地改动。这虽然大大增加了编程的灵活性，但是也降低了安全性。JavaScript 支持函数功能，同样不需要任何的声明步骤。

下表列出了 JavaScript 与 Java 的区别，可以供读者比较：

JavaScript	Java
不编译，由客户端解释执行	在客户端执行前由服务器编译
面向对象，没有类和继承，内置可扩展的对象	面向对象，程序由继承的类组成
嵌入 HTML 文件	组成 Applet，从 HTML 中调用
无需定义变量类型	必须定义变量类型
动态绑定：对象在运行时被检查	静态绑定：对象在编译时被检查

总之，同 Java 相比，JavaScript 由于它简单的语法结构，特殊的内置功能以及最小的系统开销，可以被广泛得多的不同网页创作者所接受和使用。

最后要指出的是，JavaScript 从诞生至今，一直处于不断发展和变化之中，不仅离功能的终极目标还有很大的距离，就是已经实现的部分功能和特性以后也可能会发生改变和调整。因此，从这个意义上来说，本书所讲述的内容将永远是不完整的，读者也永远不要期待把本书的内容作为 JavaScript 的最终规范来学习。JavaScript 永远是一个发展的世界，我们希望和您在这个世界里分享它的无穷潜力。

作者
1999 年 6 月

第 1 章

Internet 及网络开发入门

本章要点:

本章将带领读者进入一个飞速发展的新天地——Internet 世界。尽管本书讲述的是有关 JavaScript 从入门到应用的内容，但如果您对于 Internet 的世界一无所知，后面所讲的内容就可能比较难于理解。本章的目的就是要对 Internet 世界中的一些常用的应用和开发概念加以解释和阐述，以让读者尽快对 Internet 的过去、现在和将来有一个大致清晰的认识。

本章将介绍如下内容:

- Internet 的发展和现状
- JavaScript 的特点
- 其他几种常用的 Web 开发语言

1.1 什么是 Internet

为了充分理解一项新技术并预测它的发展方向，通常要了解该技术的历史和发展过程。虽然近两三年来中国的 Internet 已有长足的发展和提高，国内的中文网站层出不穷，内容也在以几何级数的规模不断递增，大家在日常生活中也经常听到 Internet、浏览器和 E-mail 等词语；但是在五六年前，即使从世界范围来说，Internet 也还只是一个规模相对较小的网络。本节主要讨论 Internet 的发展过程。

在 20 世纪 80 年代早期，现在所谓的 Internet 当时还是一个很小的网络——ARPANET (美国国防部高级计划研究署网络)。在其后大约 15 年时间里，这个网络主要用于科研和工程开发。Internet 的概念被提出来以后，许多大学和美国的政府机构都连到 Internet 上参与发布和交换信息。尽管 Internet 早期专门用于教育和科研，但许多商业机构都意识到在 Internet 上蕴藏着无穷无尽的商业潜力，因此也做了许多开创性的工作，并在 Internet 上迅速建立了一批最初的商业网站。

在现在很流行的 WWW 实现之前，几乎所有的信息发布和交换都是通过 E-mail(电子邮件)、FTP(文件传输协议)、Telnet(远程登录协议)、NewsGroup(新闻组)、Archie 以及 Gopher 来实现的。

E-mail(电子邮件)是 Internet 上使用最广泛和最受欢迎的服务。它可以使得在广大地理区域中的个人、团体和机构方便快捷地交换信息，构成了网络用户之间进行快速、简便通信的重要手段。网络用户能够利用电子邮件发送和接收文字、图像和语音等多种形式的信息。使用电子邮件的前提是拥有自己的电子信箱，即 E-mail 地址。这个地址实际上是在一个专用的服务器上建立的用于存储邮件的磁盘空间。电子邮件地址的典型格式为：username@mailserver.com，其中 username 代表用户名，符号@读作“at”，意思是“在”，mailserver.com 部分代表邮件服务器的域名，它唯一地标识了一个邮件服务器。

 **试一试：** 值得一提的是，利用电子邮件也可以获得其他各种服务(如 FTP、Gopher、Archie、WAIS 等)。当用户想从这些信息中心查询资料时，只需要向其指定的电子信箱发一封含有一系列信息查询命令的电子邮件，该邮件服务器程序将自动读取、分析该邮件中的命令，若无错误则将检索结果通过邮件方式发给用户。

FTP(文件传输协议)可以把数据和信息从一台计算机通过互联网络传到另一台计算机上。无论两台计算机相距多远，只要它们都加入 Internet 并且都支持 FTP 协议，则这两台计算机之间就可以进行文件的传送。FTP 实质上是一种实时的联机服务，在进行工作时，用户首先要登录到目的服务器上，在此之后用户可以在服务器目录中寻找所需文件。FTP 服务理论上可以传送几乎任何类型的文件，如文本文件、二进制文件、图像文件和声音文件等。

 **试一试：** 很多 FTP 服务器都支持匿名(anonymous)登录，用户在登录到这些服务器时无须事先在这个服务器上拥有自己的用户名和口令，而只要以

anonymous 为用户名和自己的 E-mail 地址作为口令就可以访问该 FTP 服务器, 以获得该服务器提供的服务。

远程登录也是 Internet 提供的最基本的信息服务之一, Internet 用户的远程登录是在网络通信的支持下使自己的计算机暂时成为远程计算机仿真终端的过程。要在远程计算机上登录, 首先应给出在远程计算机中的合法帐号和密码, 经过身份确认后就可以建立连接, 这时的用户可以像使用自己的计算机一样使用远程计算机了。采用这种服务, 无论地理上相隔多远, 一个普通的个人也可以坐在家中通过网络使用公司里的中央主机来处理日常的工作事务, 甚至可以让最先进的超级计算机来执行自己的程序, 程序的结果也可以通过网络传送回正确的地方。

由于 Internet 是一个松散型的网络, 甚至没有一个统一的管理机构, 所以大量的信息都分散在 Internet 的汪洋大海中。如果不知道信息的确切位置, 就根本无法在其中找到自己所要的信息。Archie 和 Gopher 提供一种文本的界面, 用于提供早期的 Internet 信息服务。由密西根大学开发的 Gopher 使用了具有菜单界面的信息系统, 通过不同的菜单层次使得用户可以选择查询的主题, 从而缩小信息的搜索范围。为了扩充单一的 Gopher 的索引信息量, Gopher 服务器之间也是互相连通的, 通过定期地经常与其他 Gopher 服务器联系, Gopher 初步实现了广域的信息服务, 可以被认为是现代 WWW 发展演变的原型。但是, Gopher 也存在着缺点和不足, 主要是 Gopher 传递的信息实际上仅限于文本, 这就大大降低了 Internet 对于一般用户的吸引力; 另外, Gopher 对处于不同地理位置的信息访问机制也不是很完善。之所以有这样的种种不足, 是由于当时的技术条件有限:

- 缺乏统一的规范和标准, 不同厂商、不同平台的主机所用规范不同。
- 仅具有字符文本的操作界面。
- 缺乏跨平台使用的、可扩展的新技术。

在 WWW 出现之前, 通常进入 Internet 的方式是使用 UNIX 主机。用户一般使用 UNIX 提供的各种基于文本的 UNIX 应用程序, 但由于其工作方式以及不太友好的界面, 致使许多可能乐于尝试新技术的人们在 UNIX 面前望而生畏, 从而丧失了进一步了解和接触的机会。

80 年代开始的个人电脑革命使得大多数人可以利用桌面上的 PC 从事日常的工作和学习活动。但是, 单独的计算机环境使得人们在交换这些电子形式的文档时倍感不便——人们必须把存储在单独计算机中的文字文档打印下来, 把声音文档录制下来, 然后使用传统的邮寄方式把这样的材料从一个地区传送到另一个地区, 这不仅花费了人力和物力, 还带来了时间的延迟和安全上的各种问题。为了解决这个问题, 人们提出了这样的观点, 即把分散在世界各处的计算机互相联结起来, 通过某种网络的形式来传递各种电子形式的文档, 而不管这些文档是文本、声音, 还是别的什么形式。人们普遍认为, 这样的传递形式会省去许多不必要的中间环节, 从而节省时间并提高效率。

开发出一套规范的、跨平台使用的和可扩展的标准成为推动 Internet 向前发展的关键所在, 于是 HTML(超文本标记语言)就应运而生了。由瑞士日内瓦的欧洲粒子物理实验室 CERN 开发的这一套语言规范是一种可以在 Internet 上传递多媒体信息, 易于学习和扩展的超文本标记语言。为了在 Internet 上传递 HTML 文档, 原有的 TCP/IP 协议也进一步演化成 HTTP 协议(超文本传输协议)。WWW (World Wide Web)是随着 HTTP 协议和 HTML

一起出现的一种全新的服务方式,这种服务也简称为 Web 服务。它的出现,有效地克服了早期信息传递的种种局限,把文本、声音、动画以及各种媒体的表现形式有机地融合在一起。提供 WWW 服务的 Web 服务器利用 HTTP 协议来传递 HTML 文件到用户计算机的浏览器上,以静态和交互的方式(如文本、图像、动画以及 Java applet 等)显示各种对象。一旦必要,浏览器会通过网络向 Web 服务器发出请求,查询的结果也由网络传回给用户。

但是,标准毕竟只是一种标准,要想把 Internet 的构想真正变成现实,实现工作和生活的进一步网络化就必须有一个前提:就是在目前比较通用的平台(如 Microsoft Windows 以及 Macintosh)上要有真正实用的用户端应用程序,使用户觉得易于理解和接受,从而乐于使用,尤其是对于普通的非计算机专业的人员来说就更是如此。1993 年由 NCSA(National Center for Supercomputing Applications, 即美国国家超级计算应用中心)发布的 Mosaic 浏览器以及以后由 Netscape 公司推出的 Netscape Navigator 浏览器推动了 WWW 的普及和发展。目前在基于 Intel 芯片的计算机上运行的主要浏览器有 Microsoft 公司的 Internet Explorer 系列版本以及 Netscape 公司的 Netscape Navigator 系列版本。

随着文本、图像、动画、声音和交互程序的统一,与传统的 Gopher 相比,WWW 已经成为信息交换和检索的有效工具。WWW 之所以取得如此的成功,其根本原因在于它克服了以前许多网络规范的固有缺点。在过去,Internet 上几乎所有的信息都是字符文本格式,这样的信息不能按照多种形式表达,从而导致搜索和理解上的困难;而 WWW 上的信息是可以有多种格式的,易于浏览和理解。例如,在讨论商业计划时,利用图表、影像甚至交互程序,可以使参与讨论的人更容易理解计划的重点和精华;而在学校教育领域,利用 WWW 从学校中央的主服务器上得到各种媒体形式的教学材料和背景,无疑是教育学的一次意义重大的革命。总之,WWW 集成了多媒体的强大表现方式,从而被从商业、教育、服务到制造业等大多数领域的用户所接受和使用。

刚才提到的 Netscape Navigator 以及 Internet Explorer 浏览器都支持 JavaScript 语言规范。两种浏览器的界面可以参见图 1.1 和图 1.2。另外还有一些比较小的浏览器,可以用于配置比较低的、有特殊用途的计算机或各种电子设备。这里就不详细介绍了。

正是由于 WWW 的出现和发展,才使 Internet 的使用迅速由传统的科研机构走向千家万户的生活。我们当中的每一个普通人,只要有一台联入 Internet 的计算机,就都可以感受汹涌澎湃的信息浪潮,并且可以很容易地通过各种“超级链接”从一种信息源转到另一种信息源而不必理会其中的传输细节。所谓的超级链接将指向 Web 页面上的一个具有 URL(统一资源定位器)的对象。当用户单击一个超级链接时,浏览器会自动地把超级链接的目标页面装入浏览器,供用户进行浏览。从这个意义上来说,URL 可以被用来定位 Internet 网上的各种资源。

无论 Internet 上的信息是如何的丰富,但如果没有一种机制,来帮助用户找到所需要的信息,那么 Internet 就永远也不会有任何的科研和商业价值。取代传统的 Gopher 以及 Archie 的是诸如 Yahoo! (<http://www.yahoo.com>) 这样的新一代搜索引擎。这些数据库搜索引擎所提供的搜索功能可以使网络用户只需要使用一些关键字就可以比较容易地从信息中找到所需要的内容。值得一提的是,在本书的创作过程中,作者也曾利用 Yahoo! 等搜索引擎找到了很多最新的 JavaScript 资源。

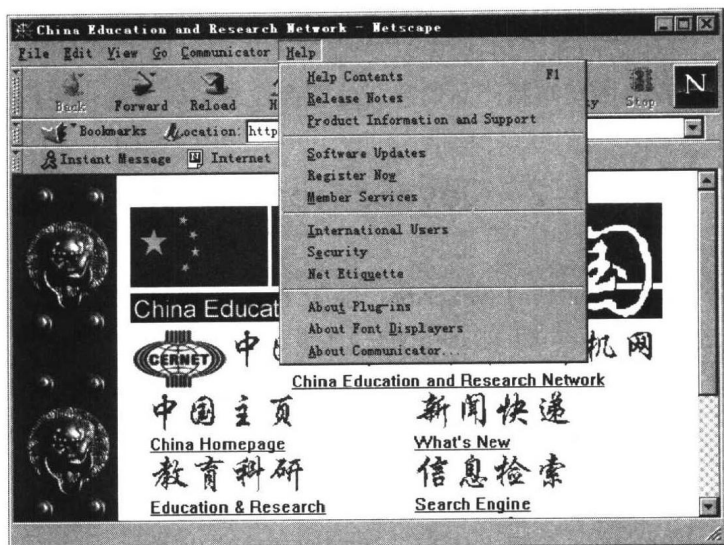


图 1.1 典型的 Netscape Communicator 界面



图 1.2 典型的 Microsoft Internet Explorer 界面

试一试： 在 Internet Explorer 或是 Netscape Communicator 的 URL 栏中键入 <http://www.yahoo.com>, 就可以进入 Yahoo! 的搜索引擎。

一些商业公司也利用 WWW 服务获得了很多的商业机会。Microsoft 公司在它的 Windows 操作系统中集成了 Internet Explorer 浏览器, 促进了网络化的发展进程, 力图在浏览器软件市场上占有更大的市场份额, 以便获取更多的商业利润。Yahoo 公司专门从事 Internet 信息的分类和整理, 在短短的几年时间里, 迅速发展成为信息服务领域中的主流公司。实际上, Internet 所能提供的服务以及发展机会还远远不止这些。随着 Internet 的飞速发展, 每天都会诞生新的服务, 如今像网络电话(IP Phone)、网络会议(Netmeeting)和网上传呼(ICQ)等都得到极大的应用。

总之, Internet 可能是自从电视发明以来最令人振奋和最具有影响力的信息媒体。随着越来越多的人开始访问和使用 WWW 的服务, 许多机构正逐步把它们的经营范围扩大到 Internet 上, 以在上面发布新产品信息, 处理商业事务以及提供用户服务。

1.2 JavaScript 简介

JavaScript 语言的前身叫作 LiveScript。自从 Sun 公司推出著名的 Java 语言之后, Netscape 公司引进了 Sun 公司有关 Java 的程序设计概念, 将自己原有的 LiveScript 重新进行设计, 并更名为 JavaScript。之所以取名为 JavaScript, 原因就在于 JavaScript 作为一种嵌入 HTML 文档的、基于对象的脚本设计语言, 其中很大一部分的语法同 Java 语言很相似, 而且 JavaScript 的设计可以使它很容易地同 Java 语言一同工作, 它可以充分支持 Java 的 applet 小应用程序, Java 的 applet 小应用程序也可以很方便地访问已有的 JavaScript 代码。所以, 也可以把 JavaScript 看成是 Java 语言的某种简化版本。

JavaScript 的正式官方定义是 Netscape 公司和 Sun 公司在推出 JavaScript 时的联合声明: “JavaScript 是一种易于使用的对象描述语言, 它是为了创建真正的联机应用程序而设计的, 无论是在用户端还是在服务器端, 这种应用程序都可以将对象和资源连接在一起。HTML 主页设计者和企业应用程序开发人员都可以使用 JavaScript 动态地描述在用户机上以及服务器上运行的对象的行为。”

嵌入 HTML 文档中的 JavaScript 源代码实际上是作为 HTML 网页的一部分存在的。在用户使用任何一种支持 JavaScript 的浏览器来浏览具有 JavaScript 源代码的 HTML 网页时, 由浏览器本身对该 HTML 网页进行分析、识别、解释并执行用 JavaScript 编写的源代码。

为什么要使用 JavaScript 而不是 CGI(公共网关接口)或别的编程形式呢? 要回答这个问题就要根据用户项目对于 Web 网站性能的要求。如果您的网页已经包含有实现一定功能的 CGI 脚本, 那么 JavaScript 就可以使用一些在用户端的小应用程序来代替原来放在服务器上的 CGI 脚本; 而且, 用户端的 JavaScript 可以使您在把信息提交到服务器之前就可以验证用户的输入信息。随着网络标准的不断演化, 会有更多富有表现力的标记被引入到 HTML 中。将这样的标记同 JavaScript 程序相结合, 就可实现各种崭新的文档组织和表现形式。

JavaScript 应用的优点包括:

首先, 在 JavaScript 这样的用户端脚本语言出现之前, 传统的数据提交和验证工作均由用户端浏览器通过网络传输到服务器上进行。如果数据量很大, 这对于网络和服务器的资源来说实在是一种无形的浪费。一个比较好的解决方案是在提交之前在用户端由 JavaScript 实现自动的验证;

其次, JavaScript 可以方便地操纵各种浏览器的对象, 可以使用 JavaScript 来控制浏览器的外观、状态甚至运行方式, 可以根据用户的需要“定制”浏览器, 从而使网页更加友好;

再之, JavaScript 可以使多种任务仅在用户端就可以完成而不需要网络和服务器的参