



高等学校精品规划教材

Web 程序设计及应用

刘 兵 史瑞芳 等编著



中国水利水电出版社
www.waterpub.com.cn

21 世纪高等学校精品规划教材

Web 程序设计及应用

刘 兵 史瑞芳 等编著



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书以 Web 程序设计开发的实际应用为目的,以开发所需要的关键技术为主线,全面系统地介绍了 Web 程序设计的基本理论和开发方法。全书共分为 10 章,主要内容包括:Web 程序设计的基本思想和开发环境的建立、Web 程序设计的客户端语言(HTML、CSS、JavaScript)、Web 服务器控件的使用方法、ASP.NET 内建组件对象、ASP.NET 数据访问及显示、Web 程序设计的一些典型应用(如组件、Web 服务、Web 引用)等。

本书结构合理,逻辑性强,写作特色鲜明。每个章节、每个知识点都有精心设计的典型案例程序说明其用法,各章节之间的联系紧凑、自然。为了方便教学,本书每章都配有大量的习题、案例程序和电子教案(可以从中国水利水电出版社网站及万水书苑免费下载,网址为: <http://www.wsbookshow.com> 和 <http://www.waterpub.com.cn/softdown/>)。

本书可作为高等学校计算机专业及电子信息类等相关专业的教材,也可供没有接触过 Web 程序设计但自身有一定程序设计能力的读者自学使用,还可以作为 ASP.NET 程序设计的培训教材。

图书在版编目(CIP)数据

Web 程序设计及应用 / 刘兵等编著. — 北京: 中国水利水电出版社, 2014. 6
21 世纪高等学校精品规划教材
ISBN 978-7-5170-2112-4

I. ①W… II. ①刘… III. ①网页制作工具—程序设计—高等学校—教材 IV. ①TP393.092

中国版本图书馆 CIP 数据核字(2014)第 123421 号

策划编辑: 雷顺加 责任编辑: 张玉玲 封面设计: 李 佳

书 名	21 世纪高等学校精品规划教材 Web 程序设计及应用
作 者	刘 兵 史瑞芳 等编著
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路 1 号 D 座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn
经 售	电话: (010) 68367658 (发行部)、82562819 (万水) 北京科水图书销售中心(零售) 电话: (010) 88383994、63202643、68545874 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	三河市铭浩彩色印装有限公司
规 格	184mm×260mm 16 开本 17.25 印张 438 千字
版 次	2014 年 6 月第 1 版 2014 年 6 月第 1 次印刷
印 数	0001—3000 册
定 价	35.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

前 言

没有哪一项技术像今天的 Internet 一样对人们的工作、生活的影响之大、程度之深，使得人们不能不重视它，而这一切起着巨大推动作用的就是 WWW 的出现，因此 Web 程序设计开发就成为各类应用程序中最重要的组成部分。网站开发平台和动态网页开发所使用的语言有很多种，本书着重介绍微软.NET 框架中的 ASP.NET 和客户端的相关语言。

本书系统讲解了 Web 程序设计的方法，注重 Web 基础知识，并以实用性为指导原则，力求使读者学习完本书后能够成为一名合格的 Web 程序员。本书由浅入深，每个知识点都结合实例讲解，让读者对 Web 实际应用开发中的常用知识做到心中有数，培养动手能力。

全书共分为 9 章，主要内容包括：Web 程序设计的基本思想、开发环境的建立、Web 程序设计的客户端语言（HTML、CSS、JavaScript）、Web 服务器控件的使用方法、ASP.NET 内建组件对象、ASP.NET 数据访问及显示、Web 程序设计中的典型应用（如组件、Web 服务、Web 引用）等。

本书统稿定稿工作由刘兵完成，史瑞芳编写第 1 章和第 2 章，刘冬编写第 3 章、第 8 章和第 9 章，刘兵编写第 4~7 章。感谢武汉轻工大学电气信息学院的谢兆鸿教授认真审阅了本书；感谢在程序代码编写、文字校对、图表编辑上做了大量工作的江小丽、蒋丽华、张琳、左爱群、张柱华、贾瑜、易逵、刘欣、管庶安、李禹生、丰洪才等老师。

在本书编写过程中，作者参考了网站上发表的相关技术文章，在此对这些技术文章的作者表示感谢，同时也欢迎广大读者对书中的疏漏及不妥之处进行指正，作者邮箱：lbliubing@sina.com。

作 者

2014 年 4 月

目 录

前言

第1章 Web 程序设计概述	1	2.5.2 多媒体标记	48
1.1 Web 的基本概念	1	习题二	50
1.1.1 Web 概述	1	第3章 层叠样式表 CSS	52
1.1.2 Web 浏览器的工作原理	2	3.1 层叠样式表概述	52
1.1.3 统一资源定位器 URL	3	3.1.1 层叠样式表的引出	52
1.1.4 超文本与超媒体	3	3.1.2 定义 CSS	53
1.1.5 超文本标记语言 HTML	5	3.1.3 选择器种类	54
1.1.6 超文本传输协议	5	3.1.4 外部 CSS 样式表	58
1.1.7 主页	6	3.1.5 CSS 样式规则的优先级及单位	59
1.2 Web 程序设计的方式	6	3.2 CSS 样式的属性分类	60
1.2.1 网页基础知识	6	3.2.1 背景	60
1.2.2 .NET Framework 的概念	8	3.2.2 CSS 文本	63
1.2.3 ASP.NET 应用程序	10	3.2.3 字体	64
1.3 ASP.NET 软件开发环境	12	3.2.4 鼠标属性	66
1.3.1 Internet 信息服务 (IIS)	12	3.3 CSS 框模型	67
1.3.2 Visual Studio 2010 开发环境	17	3.3.1 CSS 框模型概述	67
1.3.3 ASP.NET 第一个程序	24	3.3.2 CSS 内边距	68
习题一	26	3.3.3 CSS 边框	69
第2章 HTML 语言	28	3.4 CSS 定位	71
2.1 HTML 语言概述	28	3.4.1 定位概述	71
2.1.1 HTML 语言的结构	28	3.4.2 CSS 相对定位和绝对定位	72
2.1.2 构成网页的基本元素	29	3.4.3 浮动	75
2.2 超文本链接指针	32	习题三	80
2.2.1 链接到其他站点	32	第4章 JavaScript 语言	83
2.2.2 同一个文件中的链接	33	4.1 JavaScript 的基础知识	83
2.2.3 图像超链接	33	4.1.1 JavaScript 概述	83
2.3 框架与表单	35	4.1.2 JavaScript 工作原理	85
2.3.1 框架结构的使用	35	4.2 JavaScript 语言基本结构	87
2.3.2 表单的应用	36	4.2.1 JavaScript 的数据类型	87
2.4 HTML 中的表格	42	4.2.2 JavaScript 的变量	88
2.4.1 表格的标记	42	4.2.3 JavaScript 的运算符和表达式	89
2.4.2 表格使用实例	44	4.2.4 JavaScript 程序流程控制	92
2.5 其他标记	45	4.2.5 JavaScript 中的函数	96
2.5.1 列表	45	4.2.6 JavaScript 中的事件	99

4.3 JavaScript 中的对象	102	6.3.2 Application 对象的属性	173
4.3.1 对象的基本概念	102	6.3.3 Application 对象的方法	175
4.3.2 内置对象	104	6.3.4 Application 对象的事件	176
4.3.3 浏览器和 HTML 对象	108	6.4 Session 对象	178
习题四	115	6.5 Server 对象	180
第 5 章 Web 窗体和常用服务器控件	117	6.5.1 Server 对象的属性	180
5.1 ASP.NET 网站的逻辑结构	117	6.5.2 Server 对象的方法	181
5.1.1 概述	117	习题六	183
5.1.2 ASP.NET 网站的组成	118	第 7 章 ADO.NET 技术	186
5.1.3 ASP.NET 页面	120	7.1 SQL Server 数据开发介绍	186
5.1.4 ASP.NET 页面处理过程	120	7.1.1 SQL Server 数据库简介	186
5.1.5 服务器控件种类	124	7.1.2 创建与删除数据库	187
5.2 Web 常用服务器控件	125	7.1.3 创建与删除数据表	188
5.2.1 文本类控件	125	7.1.4 数据记录的添加与删除	189
5.2.2 按钮类控件	130	7.2 ADO.NET 基础	190
5.2.3 选择类控件	134	7.2.1 ADO.NET 概述	190
5.2.4 链接控件	140	7.2.2 ADO.NET 连接数据库的方法	191
5.3 验证控件	141	7.2.3 数据库操作的基本对象	195
5.3.1 概述	141	7.2.4 数据库基本操作通用类	198
5.3.2 RequiredFieldValidator 控件	143	7.2.5 数据库的基本操作	200
5.3.3 CompareValidator 控件	144	7.3 数据显示 GridView 控件	207
5.3.4 RangeValidator 控件	149	7.3.1 GridView 控件简介	207
5.3.5 RegularExpressionValidator 控件	151	7.3.2 GridView 控件实现数据库记录的 分页显示	211
5.3.6 CustomValidator 控件	157	7.3.3 GridView 控件对记录排序	214
5.3.7 ValidationSummary 控件	159	7.3.4 修改 GridView Web 服务器控件中 的数据	216
习题五	160	习题七	217
第 6 章 ASP.NET 内建组件对象	162	第 8 章 网站导航与风格	220
6.1 Response 对象	162	8.1 TreeView 控件	220
6.1.1 利用 Response 对象显示信息	162	8.1.1 TreeView 控件概述	220
6.1.2 利用 Response 对象重定向浏览器	164	8.1.2 TreeView 控件的主要属性和事件	221
6.1.3 Response 对象 Cookies 属性的应用	166	8.1.3 在 TreeView 控件中显示关系数据	224
6.2 Request 对象	168	8.2 网站主题	226
6.2.1 利用 Request 对象获取服务器 变量值	168	8.2.1 创建主题的方法	227
6.2.2 利用 Request 对象获取 Cookies 值	169	8.2.2 同一控件的多种定义方法	229
6.2.3 利用 Request 对象获取客户端 浏览器的信息	170	8.3 母版页和内容页	231
6.3 Application 对象	172	8.3.1 概述	231
6.3.1 Application 对象存储数据	172	8.3.2 母版页和内容页事件顺序	238

8.3.3 嵌套母版页	239	9.2 Web Service	255
习题八	242	9.2.1 Web Service 概念	255
第 9 章 ASP.NET 网络技术开发	243	9.2.2 使用 Web Service 实现天气预报	257
9.1 文件上传	243	9.3 组件应用	263
9.1.1 文件夹的操作	243	大作业	267
9.1.2 文件的操作	247	参考文献	270
9.1.3 文件上传控件	249		

第1章 Web 程序设计概述



学习目标

本章介绍 Web 程序设计的基本概念及其工作环境,同时说明 Web 程序设计的方式。通过对本章的学习,读者应该掌握:

- Web 的基本概念
- Web 程序设计的方式
- ASP.NET 软件开发环境

1.1 Web 的基本概念

Internet 现在已经成为世界上最大的信息宝库,包含从教育、科技、政策、法规到艺术、娱乐、商业等各个方面的信息。然而,在 Internet 上的信息资源既没有统一的目录,也没有统一的组织和系统,这些信息分布在 Internet 位于世界各地的计算机系统中,以文件、数据库、公告板、目录文档和超文本文档等形式存储,而且每天还有许多人在利用 Internet 对外发布信息,以便让别人有偿或无偿地使用这些信息。因此,Internet 上的信息资源几乎每时每刻都在增加和更新过程中。

这样带来的问题是在 Internet 这个世界上最大的网络化信息资源宝库中进行信息检索时常常感到无从下手。人们为了充分利用 Internet 上的信息资源,迫切需要一种更加方便、快捷的信息浏览和查询工具,在这种情况下,万维网(World Wide Web,缩写为 WWW,有人称它为 Web)诞生了。WWW 是一种“网”状的结构,形如“蜘蛛网”,通过 Internet 将位于世界各地的相关信息资源有机地整合在一起,采用“超文本”(Hypertext)方式为用户提供世界范围的多媒体(Multimedia)信息服务。人们可以通过 Internet 从世界任何地方查看希望得到的文本、影视和音像等信息。WWW 的出现被认为是 Internet 发展史上一个重要的里程碑,对 Internet 的发展起到了巨大的推动作用,也为 Internet 的进一步普及铺平了道路。

1.1.1 Web 概述

WWW 服务的基础是 Web 页面,每个服务站点都包括若干个相互关联的页面,每个 Web 页面既可显示文本、图形、图像和声音等多媒体信息,又可提供一种特殊的链接点。这种链接点可以指向一种资源,也可以是另一个 Web 页面、文件或 Web 站点,这样使全球范围内的 WWW 服务连成一体,这就是所谓的超文本和超链接技术。用户只要用鼠标在 Web 页面上单击这些超链接,就可以获得相应的多媒体信息服务。

在每个 WWW 站点上都有一个主页,是进入该站点的起始页,也就是第一页,相当于这个站点的窗口。一般是通过主页来查找该站点的主要信息服务资源,因此 Web 站点的主页都

设计得很精美，很有特色。

WWW 提供的信息形象、丰富，支持多媒体信息服务，还支持最新的虚拟现实技术，仿真三维场景。用 WWW 服务还可集成电子邮件、文件传输等许多 Internet 服务形式，大大方便了用户的使用，同时也使上网变成了一件十分轻松的事情，而且也正在改变着人们现在的生活方式。WWW 的核心是 Web 服务器，它提供各种形式的信息或服务，用户采用 Web 浏览器软件来浏览和使用这些信息或服务。

由于 WWW 的流行，许多上网的新用户接触的都是 WWW 服务，因而把 WWW 服务与 Internet 混为一谈，甚至产生 WWW 就是 Internet 的误解。其实 WWW 只是 Internet 的一部分，Internet 还拥有许多种类的服务资源。

1.1.2 Web 浏览器的工作原理

WWW 基于客户机/服务器模式，Web 浏览器将请求发送到 Web 服务器，服务器响应这种请求，将其所请求的页面或文档传送给 Web 浏览器，浏览器再将获得的信息显现在浏览器窗口，这就是所谓的下载页面过程，Web 浏览就是一个从服务器下载页面的过程。图 1-1 所示是 Web 浏览器从 Web 服务器获得 Web 页面的过程。

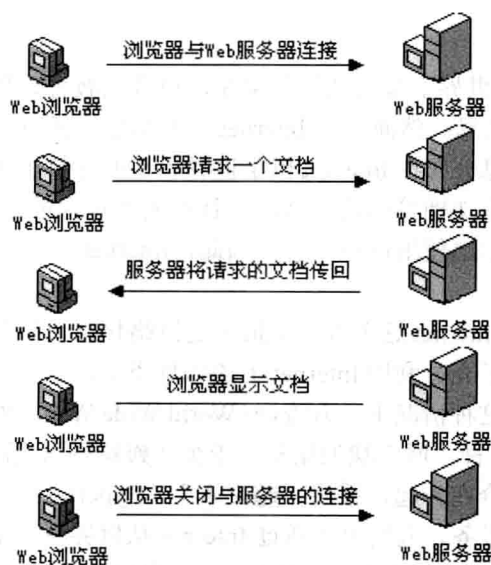


图 1-1 Web 浏览器从 Web 服务器获得 Web 页面的过程

用户输入不同的 URL，可以打开特定的 Web 服务器的相应文档，下载到浏览器上，浏览器解释 HTML 所描述的动画、声音、文本、图形、图像以及需要进一步链接的 URL，展现给用户的是极其丰富的超文本信息。

Web 浏览器最基本的功能是解释 HTML 文档，而不是能处理各种类型的文件，当遇到不能处理的某种类型文件时，就检查是否有打开这类文件的程序，常见的打开程序有 JPEG 查看器、MPEG 播放器、声音播放器、动画和图像查看器等，这样无论在 Web 站点上浏览什么类型的文件，浏览器几乎都能呈现出来。

1.1.3 统一资源定位器 URL

统一资源定位器 (Uniform Resource Locator) 是文件名的扩展。在单机系统中, 如果要找某一个文件, 需要知道该文件所在的路径和文件名; 而在 Internet 上要找一一个文件, 除了要知道以上内容之外, 还需要知道该文件存放在哪个网络的哪台主机中才行。与单机系统不一样的是, 在单机系统中所有的文件都由统一的操作系统来管理, 因而不必给出访问该文件的方法; 而在 Internet 上, 主机的操作系统都不一样, 因此必须指定访问该文件的方法。一个 URL 包括了以上所有的信息, 格式为:

protocol:// machine.name[:port]/directory/filename

- protocol: 是访问该资源所采用的协议, 即访问该资源的方法, 主要有:
 - HTTP: 超文本传输协议, 该资源是 HTML 文件。
 - FTP: 文件传输协议, 用 FTP 访问该资源。
 - MAILTO: 采用简单邮件传输协议 SMTP 提供电子邮件服务。
- machine.name: 是存放资源主机的 IP 地址, 通常以域名形式出现, 如 www.whpu.edu.cn。
- port: 是服务器在其主机上所使用的端口号。一般情况下端口号不需要指定, 因为通常这些端口号都有一个默认值 80, 只有当服务器所使用的端口号不是默认的端口号时才需要指定。
- directory 和 filename: 是指该资源的路径和文件名。

例如一个典型的 URL 为: <http://www.whpu.edu.cn/>, 从这个网址中可以看出, 采用的是超文本传输协议 (HTTP), 主机域名是 www.whpu.edu.cn。但这个网址并没有指出该主机上哪个目录的哪个文件, 以及端口号是多少。其实, 在 HTTP 协议中, 如果不在 URL 中写出端口号, 则端口号使用默认值, 而目录为 WWW 服务器根目录, 文件为根目录上的默认主页文件, 这个默认的主页文件可以在 Web 服务器上设置。

与单机系统绝对路径、相对路径的概念类似, 统一资源定位器也有绝对 URL 和相对 URL 之分。绝对 URL 和相对 URL 是相对于最近访问的 URL 而言的。例如一个浏览器打开 <http://www.whpu.edu.cn/default.asp> 的文件, 如果想看同一个目录下的另一个文件 [introduce.html](#), 可以直接使用 [introduce.html](#), 这时 [introduce.html](#) 就是一个相对 URL, 它的绝对 URL 为 <http://www.whpu.edu.cn/introduce.html>。当绝对 URL 中的部分内容省略时, 其对应的相对 URL 所表示的含义如下:

- 当协议 (例如 [http://](#)) 被省略时, 就认为与当前页面的协议相同。
- 当主机域名被省略时, 就认为是当前主机域名。
- 当目录路径被省略时, 就认为是当前目录。
- 当文件名被省略时, 就认为是默认文件。

1.1.4 超文本与超媒体

超文本的概念是特德·尼尔逊于 1969 年左右提出的。以后每两年国际上举行一次有关超文本的学术会议, 每次会议都有上百篇有关超文本的学术论文发表, 但是谁也没有想到要把超文本技术应用于计算机网络。物理学家蒂姆则机敏地抓住了这个概念, 提出了一种超文本的数据结构, 并把这种技术应用于描述和检索信息, 实现了高效率的存取, 从而发明了 WWW 这种信息浏览服务方式。

WWW 中的超文本实际上是一种解决菜单与信息分离的机制,把可选菜单项嵌入文本中的概念称为“超文本”。在超文本系统中,用户既可以阅读显示的信息,也可以在信息中选择某个超级链接条目,即用鼠标在超级链接条目上单击一下就能连接到与之相关的文件,并在浏览器上呈现这些文件的页面。超级链接可以是一些单词、短语(一般用下划线或不同颜色标明)和图标。这样,用户在查阅资料时就不必像传统方式那样完全根据菜单从头查到尾,而是在操作过程中随机地跳转,最终找到所需的资料。例如,当进入“IIS 5.1 文档”Web 页以后,便显示如图 1-2 所示的界面,用鼠标单击“安装 IIS”栏目,便会显示如图 1-3 所示的界面。

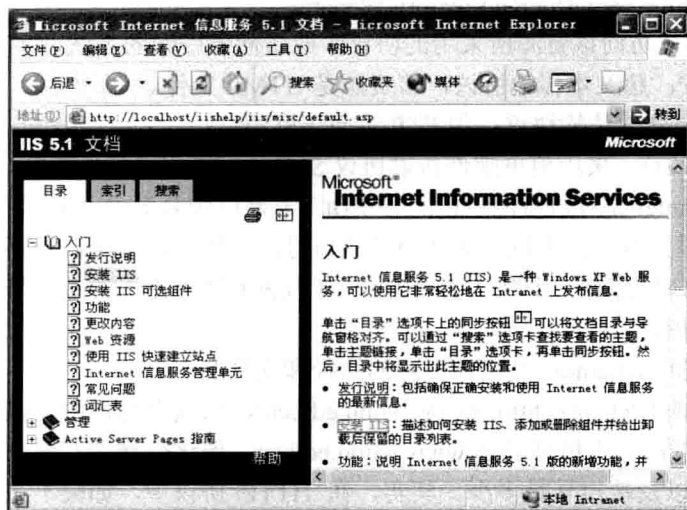


图 1-2 IIS 5.1 文档

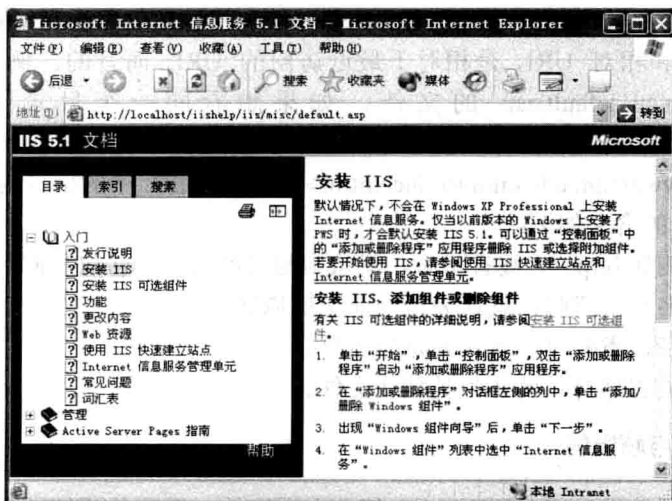


图 1-3 安装 IIS

超文本技术采用指针连接的网状交叉索引方式,对不同来源的信息加以链接。也就是说,一个超文本文件含有多个指针,而指针可以指向任何形式的文件。正是这些指针指向的“纵横交错”,使得分布在本地和远程服务器上的文本文件能够连接在一起。

超媒体是超文本的自然扩展,是超文本与多媒体的组合。在超媒体中,链接的除了文本文件以外,还有音像和动画等。Internet 的浏览服务不仅仅采用了超文本机制,而且还把声音和图像也作为浏览系统的一部分。这样一来,用户利用浏览器访问 Internet 各个站点的信息时,除了可以阅读文本资料之外,还能看到生动的画面,听到悦耳的声音,这正是超媒体机制实现的结晶。当一个超文本系统包含了针对非文本信息的索引项(也称为“信息链接”)时,该系统便称为“超媒体系统”,不过人们并不严格区分超文本和超媒体。事实上在 Internet 上通过 WWW 进行的浏览基本上都是超媒体的。但是人们通常还是把它称为“超文本系统”。

1.1.5 超文本标记语言 HTML

超媒体方式的关键除了超文本和超媒体思想的形成之外,还在于这种思想实现机制的提出,富于想象力的物理学家蒂姆在计算机网络中找到超文本超媒体思想的实现机制。他开发了一种全新的文档语言——超文本标记语言(HTML, HyperText Markup Language),使用户能够将文档中的词和图像与其他文档链接起来,不论这些文档存放在何处,只需用鼠标单击超链接,就可以将 Internet 上与之相关联的文档查找出来并显示在浏览器上。HTML 是一种专用的编程语言,用于编制要通过 WWW 显示的超文本文件页面。HTML 对文件显示的具体格式进行了详细的规定和描述。例如,规定了文件的标题、副标题、段落等如何显示,如何链接某个超文本文件,如何在超文本文件中嵌入图像、声音和动画等。

HTML 采用标准的 7 位 ASCII 码文件形式,通过一系列格式化方法表示各种超链接和信息,用 HTML 编写的文档全部都采用.htm 或.html 作为后缀。

当 WWW 浏览器读取到 HTML 文件时,就以超文本方式显示给用户。例如,下面的 HTML 语句建立了一个链接“专题分类查询信息”,并将该链接与计算机 www.whpu.edu.cn 中的文件 index.html 相连:

```
<a href=http://www.whpu.edu.cn/index.html>专题分类查询信息</a>
```

由于 HTML 是一种简单、易学的语言,并且支持多国语言,所以用户很容易掌握并建立 WWW 网页,这也是 WWW 能迅速普及的一个重要原因。

1.1.6 超文本传输协议

超文本传输协议 HTTP (HyperText Transfer Protocol) 从 1990 年开始应用于 WWW,它可以简单地被看成是客户端浏览器和 Web 服务器之间的会话。由于通过该协议在网络上查询的信息中包含了用户可以实现进一步查询的链接,因此用户可以只关心要检索的信息,而无需考虑这些信息存储在什么地方。

为了从服务器上把用户需要的信息发送回来,HTTP 定义了简单事务处理程序,由以下 4 个步骤组成:

- (1) 客户机与服务器建立连接。
- (2) 客户机向服务器递交请求,在请求中指明所要求的特定文件。
- (3) 如果请求被接受,那么服务器便发回一个应答。在应答中至少应当包括状态编号和该文件内容。
- (4) 客户机与服务器断开连接。

HTTP 协议提供了一种简单算法,使得服务器能迅速为客户机做出应答。为此 HTTP 协议应当是一个无状态协议,即从一个请求到另一个请求不保留任何有关连接的信息。另外,每

次连接 HTTP 只完成一个请求，在一次请求完成以后，服务器与客户机之间的连接便断开。

1.1.7 主页

主页 (Home Page) 就是用户在访问 Internet 上的某个站点时第一个显示在浏览器中的页面，也称为 WWW 的“初始页”。在 Internet 上，用户经常需要了解一个机构或一个企业的基本情况，有时需要了解全部情况，有时只想查询某个部门的情况。这样，一些单位为了便于用户查询，树立形象，往往在网上建立站点，发布主页，在主页上显示本单位的各种信息和图像，列出一些常用的信息链接。

从信息查询的角度来看，主页就是用户通过 WWW 连接访问超文本等各类信息资源的根；从信息提供的角度来看，由于各个开发 WWW 服务器的机构在组织 WWW 信息时是以信息页为单位的，这些信息页被组织成树状结构以便检索，那个代表“树根”信息页的超文本就是该 WWW 服务器的初始页（主页）。

1.2 Web 程序设计的方式

1.2.1 网页基础知识

要开发一个网站，首先要了解组成网站的最基本元素——网页。本节就来了解一下网页的基础知识，包括网页和服务器的交互过程、静态和动态网页以及脚本语言。

1. 网页和服务器的交互

通过互联网浏览网页时，用户会自动与网页服务器建立连接。用户提交信息资源的过程称为向服务器发出请求。通过服务器解释信息资源来定位对应的页面，并传送回代码来创建页面，这个过程称为对浏览器的响应。浏览器接受来自于网页服务器的代码，并将它编译成可视页面。在这样的交互过程中，浏览器称为“客户机”或“客户端”，整个交互的过程称为“客户机/服务器”的通信过程。

“客户机/服务器”概括了任务的分布来描述网页的工作方式。服务器 (Web 服务器) 存储、解释和分布数据，客户机 (浏览器) 访问服务器以得到这些数据。客户机和服务器使用 HTTP 协议通过 Internet 进行交互。HTTP 协议又叫超文本传输协议，是一个客户机和服务器端请求和应答的标准。浏览网页时，浏览器通过 HTTP 协议与服务器交换信息。

2. 静态页面

早期的网站发布的是静态网页，主要由 HTML 语言组成，没有其他可以执行的程序代码。静态页面一经制成，内容就不会再改变，不管何时何人访问，显示的都是一样的页面内容，如果要修改有关内容，就必须修改源代码，然后重新上传到服务器上。静态页面虽然包含文字和图片，但这些内容却需要在服务器端以手工的方式来变换，因此很难把它们描述为 Web 程序。程序代码 w1-1.htm 是使用 HTML 语言编写的一个简单静态网页代码。

程序代码 w1-1.htm

```
<html>
<head>
  <title>静态页面测试</title>
</head>
```

```
<body>
  <h1>静态页面</h1>
  <p>静态页面一般用 HTML 语言编写</p>
</body>
</html>
```

代码说明：该程序包含一个标题和一句文字。其中标题包含在标记<h1>和</h1>之间，文字包含在标记<p>和</p>之间。图 1-4 所示为该静态网页文件被浏览器解析后的结果。



图 1-4 静态网页

HTML 是互联网的描述语言，基本的 HTML 语言包含由 HTML 标记格式化的文本和图像内容。文本是 HTML 要显示的内容，标记则告诉浏览器如何显示这些内容，它定义了不同层次的标题、段落、链接、斜体格式化、横向线等。HTML 文件的后缀可以是.htm，也可以是.html。

3. 动态页面

动态页面不仅含有 HTML 标记，而且含有可以执行的程序代码，动态页面能够根据不同的输入和请求动态生成返回的页面，例如常见的 BBS、留言板、聊天室等就是用动态页面来实现的。动态页面的使用非常灵活，功能强大。

真正意义上包含动态页面的 Web 程序直到 HTML 2.0 中 HTML 表单引入时才开始出现。在一个 HTML 表单中，所有的控制都放置在<form>和</form>标记中。当用户在客户端单击“提交”按钮后，网页上的所有内容就以字符串的形式发送到服务器端，服务器端的处理程序根据事先设置好的标准来响应客户的请求。w1-2.htm 是一个由 HTML 表单构成的动态页面的代码。

程序代码 w1-2.htm

```
<html>
  <head>
    <title>动态页面</title>
  </head>
  <body>
    <form>
      用户名: <input id="Text1" type="text"/><br/><br/>
      密码: <input id="Password1" type="password"/><br/><br/>
      <input type="submit" value="登 录"/>&nbsp;
      <input id="Reset1" type="reset" value="重新输入"/>&nbsp;
      <input id="Button1" type="button" value="注 册"/>
    </form>
  </body>
</html>
```

代码说明：该程序由 HTML 表单组成，包括文本框、密码框、提交按钮、复位按钮等，这些内容和标记均被包含在表单标记之间。该网页运行效果如图 1-5 所示。



图 1-5 动态页面

现在, 尽管动态 ASP.NET 页面已经比较流行, 但 HTML 表单仍然是这些页面的基本组成元素, 所不同的是构成 ASP.NET 页面的 HTML 表单控件运行在服务器端, 所以读者必须要掌握最基本的 HTML 表单, 以便能够更好地使用 ASP.NET 平台进行程序开发。

4. 脚本语言

在网页的发展过程中出现了很多优秀的脚本语言, 如 ASP、JSP、PHP 等。脚本语言确实简化了 Web 程序的开发, 但其使用起来也有很大的缺点。首先, 它的代码和 HTML 标记杂乱地堆砌在一起, 显得很混乱, 非常不方便开发和维护, 所以当 ASP.NET 的代码和 HTML 标记分离后, 使用以往一些脚本语言的 Web 开发人员都有一种耳目一新的感觉; 其次, 脚本语言的编程思想不符合当前流行的面向对象编程思想。基于以上原因, 脚本语言必将会被其他更高级的语言 (如 ASP.NET 和 Java 等) 所代替。

1.2.2 .NET Framework 的概念

随着 Internet 的发展, 基于 B/S 架构的 Web 数据库应用程序日趋普及。基于 ASP.NET 的 Web 数据库开发平台是目前最流行的 Web 开发技术之一。ASP.NET 是微软 .NET Framework 的重要组成部分。ASP.NET 为开发动态 Web 应用程序提供了基础结构。ASP.NET 作为 Microsoft Active Server Page (ASP) 的后继产品, 是开发 Web 应用系统的理想平台。

.NET Framework 是一个开发和运行环境, 它使得不同的编程语言 (如 C# 和 VB.NET 等) 和运行库能够无缝地协同工作, 简化开发和部署各种网络集成应用程序或独立应用程序, 如 Windows 窗体应用程序、ASP.NET Web 应用程序、WPF 应用程序、移动应用程序、Office 应用程序。 .NET Framework 基本结构如图 1-6 所示。

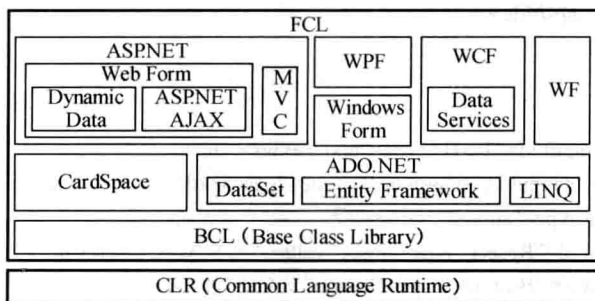


图 1-6 .NET Framework 基本结构

1. 公共语言运行库

公共语言运行库 (Common Language Runtime, CLR) 又称为公共语言运行环境, 是 .NET

Framework 的基础。运行库作为运行时管理代码的代理,提供了内存管理、线程管理和远程处理等核心服务,并且还强制实施严格的类型安全检查,以提高代码准确性。

在运行库的控制下执行的代码称为托管代码。托管代码使用基于公共语言运行库的语言编译器开发生成,具有许多优点:跨语言集成、跨语言异常处理、增强的安全性、版本控制和部署支持、简化的组件交互模型、调试和分析服务等。

在运行库之外运行的代码称为非托管代码。COM 组件、ActiveX 接口和 Win32 API 函数都是非托管代码的示例。使用非托管代码方式可以提供最大限度的编程灵活性,但不具备托管代码方式所提供的管理功能。

2. .NET Framework 类库

.NET Framework 类库(.NET Framework Class Library, FCL)是一个与公共语言运行库紧密集成、综合性的、面向对象的类型集合。使用该类库,可以高效开发各种应用程序,包括控制台应用程序、Windows GUI 应用程序(Windows 窗体)、ASP.NET Web 应用程序、XML Web Services、Windows 服务等。

.NET Framework 类库包括类、接口和值类型。类库提供对系统功能的访问,以加速和优化开发过程。.NET Framework 类型符合公共语言规范(Common Language Specification, CLS),因而可在任何符合 CLS 的编程语言中使用,实现各语言之间的交互操作。

.NET Framework 类库由基础类库(Base Class Library, BCL)和各种应用程序框架类库组成。基础类库主要提供以下功能:

- 表示基础数据类型和异常。
- 封装数据结构。
- 执行 I/O。
- 访问关于加载类型的信息。
- 调用 .NET Framework 安全检查。

各种应用程序框架类库提供构建相应应用程序的功能:

- 数据访问(ADO.NET)。
- Windows 窗体(Windows Form)。
- Web 窗体(ASP.NET)。

3. .NET Framework 的功能特点

.NET Framework 提供了基于 Windows 的应用程序所需的基本架构,开发人员可以基于 .NET Framework 快速建立各种应用程序解决方案。.NET Framework 具有以下功能特点:

(1) 支持各种标准互联网协议和规范。

.NET Framework 使用标准的 Internet 协议和规范(如 TCP/IP、SOAP、XML 和 HTTP 等)支持实现信息、人员、系统和设备互连的应用程序解决方案。

(2) 支持不同的编程语言。

.NET Framework 支持多种不同的编程语言,因此开发人员可以选择他们所需的语言。公共语言运行库提供内置的语言互操作性支持,公共语言运行库通过指定和强制公共类型系统以及提供元数据为语言互操作性提供必要的基础。

(3) 支持用不同语言开发的编程库。

.NET Framework 提供了一致的编程模型,可使用预打包的功能单元(库),从而能够更快、更方便、更低成本地开发应用程序。

(4) 支持不同的平台。

.NET Framework 可用于各种 Windows 平台, 从而允许使用不同平台的人员、系统和设备联网, 例如使用 Windows XP/Vista/Windows 7 等台式机平台或 Windows CE 之类的设备平台的人员可以连接到使用 Windows Server 2003/2008 的服务器系统。

4. .NET Framework 环境

操作系统/硬件、公共语言运行库、类库以及应用程序(托管应用程序、托管 Web 应用程序、非托管应用程序)之间的关系如图 1-7 所示。

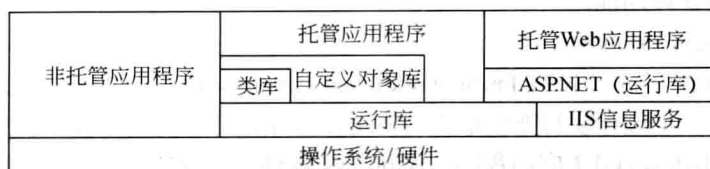


图 1-7 .NET Framework 环境

5. .NET Framework 的主要版本

目前, .NET Framework 主要包含以下版本: 1.0、1.1、2.0、3.0、3.5、4.0, 支持带最新 Service Pack 的桌面 Windows 操作系统。与之相对应, .NET Compact Framework 可用作所有 Microsoft 智能设备(包括 Pocket PC 设备、Pocket PC Phone Edition、Smartphone 设备以及其他安装有 Windows Embedded CE 的设备)中的操作系统组件。

其中, 1.0、1.1、2.0 和 4.0 版是彼此完全独立的, 即对于其中任何一个版本都可以独立存在于某台计算机上, 无论计算机上是否存在其他版本。当 1.0、1.1 和 2.0 版位于同一台计算机上时, 每个版本都有自己的公共语言运行库、类库和编译器等。应用程序开发人员可以选择面向特定的版本开发和部署应用程序。各版本之间的关系如图 1-8 所示。

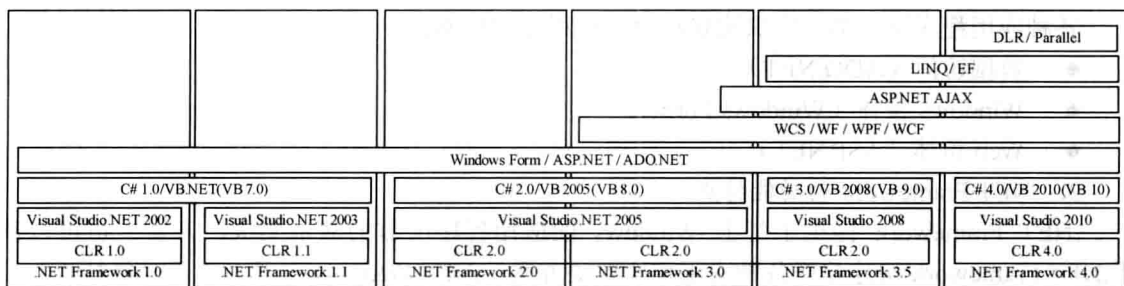


图 1-8 .NET Framework 各版本之间的关系

1.2.3 ASP.NET 应用程序

ASP.NET 应用程序是一系列资源和配置的组合, 这些资源和配置只在同一个应用程序内共享, 而其他应用程序则不能享用这些资源和配置。就技术而言, 每个 ASP.NET 应用程序都运行在一个单独的应用程序域, 应用程序域是内存中的独立区域, 这样可以确保在同一台服务器上的应用程序不会相互干扰, 不至于因为其中一个应用程序发生错误就影响到其他应用程序的正常进行。同样, 应用程序域限制一个应用程序中的 Web 页面访问其他应用程序的存储信息。每个应用程序单独运行, 具有自己的存储、应用和会话数据。