

实例通系列丛书



ASP.NET网站设计实例通

王易 编著



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



实例通系列丛书

ASP.NET 网站设计实例通

王 易 编著

清 华 大 学 出 版 社

(京)新登字 158 号

内 容 提 要

如今互联网已是一个全世界热门的主题,而 ASP.NET 是 Microsoft 推出的用于设计和架构新一代 Internet 网站的编程环境,是建设电子商务网站的利器。本书旨在引导读者深入 ASP.NET 的精髓,使读者迅速掌握用 ASP.NET 构建一个完整站点的技能。

本书现场构建了一个完整的 Internet 网站,全书共分为 10 章,第 1 章讲解基本概念,内容包括 ASP.NET 的设计理念和运行模型。第 2 章讲解 ASP.NET 网页设计的基本内容。第 3 章讲解 ASP.NET 的文件操作,实现文本模式和图形模式两种计数器。第 4 章讲解用户注册系统,包含 ASP.NET 的数据验证技术和数据库链接技术(ADO.NET)。第 5 章深入讲解 ASP.NET 技术,使用 XML 打造留言板。第 6 章和第 7 章是网页设计的经典之作,是 ASP.NET 技术的综合应用,本章现场构建了在线论坛和在线聊天室。第 8 章构建了一个基于 XML 的新闻系统,并使用 WebService 发布新闻。第 9 章使用 ASP.NET 设计并实现一个电子商店。第 10 章是 ASP.NET 的进阶内容,讲解文件上传和 E-mail 的自动传送功能。

本书可以作为 ASP.NET 程序设计人员的参考资料,也可以作为网站建设及维护人员的培训班教材。

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: ASP.NET 网站设计实例通

作 者: 王 易

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: 北京市密云县胶印厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 22.75 字数: 553 千字

版 次: 2002 年 1 月第 1 版 2002 年 1 月第 1 次印刷

书 号: ISBN 7-302-04984-X/TP·2807

印 数: 0001~5000

定 价: 32.00 元

前 言

微软公司推出的.NET 战略引起了 IT 业的广泛关注，特别是 ASP.NET 的推出，更掀起了一股强劲的“.NET”旋风。

ASP.NET 完全不同于 ASP，它是一个用于 Web 开发的全新框架。ASP.NET 提供了更易于编写、结构更清晰的代码，这些代码很容易进行再利用和共享；ASP.NET 使用编译后的语言，从而提高性能和伸缩性；ASP.NET 使用 Web 表单使开发更直观，利用面向对象技术提高组件的再利用率。另外，ASP.NET 中还包括有页面事件、Web 控件、缓冲技术以及服务器控件。对数据捆绑的改进、供 ASP.NET 使用的类库以及在 Microsoft.NET 框架中允许通过 Web 使用客户商用函数，为程序员提供了更多新的开发机会。和 ASP 相比，ASP.NET 具有很多的优势：

(1) ASP.NET 使代码更加清晰、易读。现有的 ASP 应用程序中总是充满了长短的代码，结构复杂、不易维护。而 ASP.NET 的代码不仅更容易编写，而且结构更加清晰，比 ASP 代码更易于阅读。同时，ASP.NET 代码的结构方式提高了其再利用性和共享性。

(2) ASP.NET 改进了配置、伸缩性、安全性和可靠性。对于简单的 ASP 应用程序，配置其实并不算什么问题，但是如果想将这类程序移植到一个利用组件的 N 层结构中就会遇到 DLL 陷阱问题（组件注册、版本、锁定的 DLL 等）。ASP.NET 取消了组件注册以及 DLL 锁定，全面使用了 XML 配置文件，从而解决了这个问题。这样只需要执行复制文件的工作就能配置一个 Web 应用程序。

(3) ASP.NET 对各种不同的浏览器提供了更好的支持。对于 ASP 开发人员来说，浏览器兼容问题好像是一个永恒的问题。为了解决此问题，或者为低一级的浏览器编写代码，例如用 HTML 3.2，或者限制页面的浏览范围。而无线应用程序协议 WAP 设备的引入还会深化这个问题。在 ASP.NET 中，首先由服务器探测客户端的浏览器，然后再根据不同的客户端生成不同的客户端代码，这个工作由系统自动完成，无需程序员来考虑。

(4) ASP.NET 造就了一类新的 Web 应用程序。目前的 Web 应用程序通常都是同一种模式：一套线性应用程序，然后将逻辑植入其中之一。ASP.NET 允许开发人员打破这种单一模式，创造出动态而且伸缩性更强的应用程序，这种应用程序能够更好地满足公司的商业需求，并提供一个更加丰富的开发环境。

为了使读者迅速掌握使用 ASP.NET 来创建 Web 应用程序的技能，本书以实例为主，每章都有一个独立的主题，全书共分 10 章，涵盖了实现一个完整站点的所有技术。

- 第 1 章 ASP.NET 入门

什么是 ASP.NET？它的运行模式和以往的 ASP 有什么不同？它的优势在哪里？本章将引导读者进入 ASP.NET 打造的下一代 Internet。

- 第 2 章 页面设计基础

设计网页需要从HTML开始,本章首先对HTML语言的语法进行简单的介绍。ASP.NET引入了大量的Web控件,使得Web应用程序的编写更类似于传统的Window程序,大大降低了Web应用程序设计的难度。接下来介绍各种Web控件并利用Web控件制作一个广告轮播器。

● 第3章 访问计数器

访问计数器是网站设计中一个永恒的主题,在设计计数器时最重要的是文件的读写操作。在ASP.NET中使用System.IO命名空间来进行文件操作,本章首先讲解System.IO的基本使用方法,然后使用System.IO来实现文本模式的计数器和图形模式的计数器。

● 第4章 会员注册系统

实现会员注册系统最关键的是数据验证技术和数据库链接技术。在ASP.NET中加入了大量的数据验证控件,大大简化了数据验证工作。在ASP中我们已经领略到了ADO的强大功能,而在ASP.NET中使用的是比ADO强大得多的ADO.NET来操作数据库。本章将从这两方面着手来实现会员注册系统。

● 第5章 访问留言板

XML已经成为互联网上数据存储的默认标准,在编写ASP.NET的留言板时势必要采用XML来存储留言数据。本章将讲解如何操作XML文件,并使用XML来打造一个完美的留言板。

● 第6章 在线论坛

本章是对ASP.NET技术的综合演示,包含有数据验证技术、数据库链接技术等。本章中还通过ASP.NET技术来实现浏览树和分页功能,是网页设计中的经典之作。

● 第7章 在线聊天室

在线聊天室也是网站设计中一个经久不衰的主题。本章首先讲解Web应用程序的基本概念,然后逐步实现一个功能完美的聊天室。其中包含的技术有:实现私聊,实现无刷新更新数据,处理用户断线等。

● 第8章 新闻发布系统

本章的新闻管理系统是构建于XML文件之上的,通过本章的学习能对XML文件的操作有更深刻的了解。WebService提供了跨互联网存取数据的功能,所以本章的新闻发布系统通过WebService来实现。

● 第9章 电子商店

在电子商务如日中天的今天,电子商店是网站设计的一个必不可少的主题。实现电子商店最重要的技术是实现购物篮、货架和收银台。目前网站最流行的电子商务展示技术莫过于Cult3D,所以本章最后采用Cult3D来实现一个货架。

● 第10章 文件上传和E-mail自动传送

ASP.NET中内嵌了文件上传和E-mail自动传送功能,这是ASP.NET独具一格的地方,在这一章中将通过这两个功能来构建网站的信息传送系统。本章最具特色的是通过文件上传和System.IO实现网络硬盘的功能。

通过这10章的学习,读者能够迅速掌握ASP.NET,并且能够开发出完整高效的站点。为了便于读者学习,本书附带的光盘中含有书中用到的所有例程和使用方法。

目 录

第 1 章 ASP.NET 入门	1
1.1 ASP.NET 的运行模型	1
1.1.1 为什么要使用 ASP.NET	1
1.1.2 ASP.NET 的运行模式	4
1.2 使用 ASP.NET 前的准备工作	6
1.2.1 建立 ASP.NET 的操作平台	6
1.2.2 Web.Config	8
1.2.3 移植本书范例	10
第 2 章 页面设计基础	12
2.1 速成 HTML	12
2.1.1 HTML 文档结构	12
2.1.2 表格 Table	15
2.1.3 表单 Form	18
2.1.4 窗框 Frame	22
2.1.5 A 元素	25
2.2 Server 控件	26
2.2.1 HTML 控件	26
2.2.2 Web 控件	36
2.2.3 显示日历	44
2.3 制作广告轮播器	46
2.3.1 准备 XML 文件	47
2.3.2 布置广告轮播器	48
2.4 设计梦幻驿站首页	48
第 3 章 访问计数器	52
3.1 进入 System.IO 名空间	52
3.1.1 File 对象	52
3.1.2 FileStream 对象	55
3.1.3 StreamReader 对象	55
3.1.4 StreamWriter 对象	57

3.2	文字计数器	58
3.2.1	文字模式的计数器	58
3.2.2	包装文字计数器	61
3.3	图形计数器	63
3.3.1	图形模式的计数器	64
3.3.2	包装图形计数器	66
3.4	全站点计数器	68
3.4.1	Application 对象和 Session 对象	68
3.4.2	实现全站点计数器	70
3.4.3	包装全站点计数器	73
第 4 章	会员注册系统	76
4.1	ASP.NET 中的数据验证技术	76
4.1.1	为什么要使用数据验证控件	76
4.1.2	使用 RangeValidator 控件	79
4.1.3	使用 CompareValidator 控件	82
4.1.4	使用 RegularExpressionValidator 控件	84
4.1.5	使用 CustomValidator 控件	88
4.1.6	使用 ValidationSummary 控件	90
4.2	新一代的数据库存取模型——ADO.NET	93
4.2.1	何为 ADO.NET?	93
4.2.2	建立数据库链接	97
4.2.3	读取数据库中的数据	99
4.2.4	写入、更新数据	108
4.3	建立会员注册系统	110
4.3.1	设计注册数据表	110
4.3.2	设计注册表单	111
第 5 章	访问留言板	133
5.1	文本留言板	133
5.1.1	设计前的思考	133
5.1.2	存储留言资料	134
5.1.3	显示留言信息	136
5.2	XML 留言板	141
5.2.1	为什么要使用 XML	141
5.2.2	建立 XML 文件	142
5.2.3	显示 XML 记录	143
5.2.4	建立留言页面	147

5.3 使用 Repeater 的留言板	150
5.3.1 XML 和 DataSet 对象	150
5.3.2 使用 Repeater 控件	154
5.3.3 用 Repeater 控件显示留言	158
5.4 使用 DataList 的留言板	160
5.4.1 DataList 和 Repeater 的区别	160
5.4.2 设置单行数据笔数	162
5.4.3 设置 SelectItemTemplate 模板	163
5.4.4 实现 DataList 留言板	165
5.5 设计梦幻驿站留言板	167
5.5.1 设计前的准备	168
5.5.2 建立留言页面	168
5.5.3 显示留言记录	173
第 6 章 在线论坛	178
6.1 设计前的思考	178
6.1.1 建立数据表	178
6.1.2 建立张贴页面	179
6.2 实现分层显示	183
6.2.1 实现机制	183
6.2.2 显示文章内容	192
6.3 实现分页功能	198
6.3.1 初始化记录范围	198
6.3.2 实现分页	199
6.4 查询论坛内容	203
6.4.1 布置查询表单	203
6.4.2 执行查询	205
6.4.3 退出查询	211
第 7 章 在线聊天室	215
7.1 Application 和 Session 对象	215
7.1.1 Web 应用程序基础	215
7.1.2 Application 对象	216
7.1.3 Session 对象	219
7.2 设计聊天室的界面	221
7.2.1 设计聊天室的登录界面	222
7.2.2 设计聊天室的主界面	225
7.3 实现私聊	231

7.3.1	修改 Global.asax	231
7.3.2	修改 Send.aspx	232
7.3.3	显示聊天内容	235
7.4	实现无刷新聊天室	238
7.4.1	在两个 Frame 间传送数据	239
7.4.2	刷新用户列表	242
7.4.3	刷新聊天内容	246
7.5	聊天室的其他技术	250
7.5.1	建立信息中心	250
7.5.2	处理用户断线	254
第 8 章	新闻发布系统	258
8.1	使用 XML 的新闻管理系统	258
8.1.1	构建 XML 文件	258
8.1.2	发布新闻	260
8.1.3	浏览新闻	267
8.1.4	更新新闻标题列表	273
8.1.5	更新新闻内容	290
8.2	使用 WebService 发布新闻	296
8.2.1	建立 WebService	296
8.2.2	建立 WebService 代理程序	299
8.2.3	使用代理程序	302
第 9 章	电子商店	304
9.1	设计前的准备	304
9.2	实现货架	305
9.2.1	显示同类商品	305
9.2.2	显示商品信息	310
9.3	实现购物篮	311
9.3.1	设计购物篮外观	311
9.3.2	显示购物篮数据	315
9.4	实现收银台	321
9.4.1	填写用户信息	321
9.4.2	生成订单	323
9.5	使用 Cult3D 展示商品	326
9.5.1	Cult3D 技术简介	326
9.5.2	使用 Cult3D 的货架	327

第 10 章 文件上传和 E-mail 的自动传送	332
10.1 文件上传	332
10.1.1 上传单个文件	332
10.1.2 上传多个文件	334
10.2 网络硬盘	337
10.2.1 网络硬盘功能预览	337
10.2.2 实现浏览功能	339
10.2.3 添加新目录	344
10.2.4 上传文件	345
10.2.5 删除文件和目录	346
10.3 E-mail 的自动传送	348
10.3.1 MailMessage 及 SmtpMail 对象	348
10.3.2 MailMessage 对象的其他属性	350

第 1 章 ASP.NET 入门

下一代的 Internet 将会是“Programmable Web”大放异彩的年代，各个网站与各个应用系统之间能够更加紧密地结合在一起运行。ASP.NET 推出以后，要开发 Information Agent 这类程序，将不再是困难的事。

为了实现“Programmable Web”的目标，各网站必须定义一致的数据格式和通信协议。目前最流行、应用最广泛的数据交换格式是 XML，它使用 SOAP (Simple Object Access Protocol) 作为通信协议。XML 与 SOAP 都是 Internet 公开的标准。

Microsoft 在 2000 年 7 月推出新一代的 Internet 开发平台 Microsoft.NET，希望能够协助企业快速建立 Internet、Intranet 应用程序。Microsoft.NET 要实现两个目标：一是提供广义的 Internet 功能，使应用程序可以通过任何信息设备接入 Internet，包括手机和信息家电；另一个是 Programmable Web，它将使网站成为一个可编程的组件，用户可以通过 Webservice 对它进行存取。

1.1 ASP.NET 的运行模型

随着 ASP 的使用者越来越多，ASP 的局限性也越来越突出。为了推广下一代网络操作平台“.Net FrameWork”，微软开发出了新一代的网络应用程序操作平台——ASP.NET。

1.1.1 为什么要使用 ASP.NET

在选择使用 ASP.NET 开发网页应用程序之前，读者肯定会问为什么要使用 ASP.NET？ASP 以其简单易用的优点迅速风靡了全世界，过去有将近一百万网页程序设计师选择使用 ASP 来开发网页应用程序，原因无非也就是“简单”。但是 ASP 的程序代码不够结构化，其中混合了显示界面的 Tag 标记、Client 端的脚本、Server 端的程序代码模块以及设计期间 (Design-Time) 的标注，不仅程序很难维护，排错方面也是一个很大的问题，使用时经常捉襟见肘。

1. ASP 的缺陷

ASP 除了结构性差以外，在其他方面也存在许多不足。

(1) 缺乏良好的开发模型和程序语言

ASP 只支持直译式语言 (Interpreted Script Language)，而且缺乏以组件为基础、以事件驱动模型的开发模式。很多编写过 VB 程序的人刚使用 ASP 进行网页制作时，都有一个

共同的感觉——为什么 ASP 不采用 VB 作为程序语言,而要采用功能较少的 VB Script?一般的解释是 VB Script 较小,容易解释,执行效率高于 VB,但是正因为这样,VB Script 和 VB 相比便要大打折扣。除了 VB Script 以外,ASP 也接受 JavaScript 为程序语言,但其他程序语言完全被排除在外。

(2) 组件部署与组态困难

若 ASP 网页使用 COM 组件,则应用程序部署就不是“Copy 程序”一句话可以解决的。COM 在使用之前,需要先进行注册,将组件的一些信息,如组件文件所在的位置、线程模型和一些识别码等写到系统注册表中。COM 组件虽说可以通过接口 (Interface) 来解决版本更新问题,但是事实上,DLL 经常会被不同版本的同一个 DLL 覆盖,导致原有的程序不能正常运行。

(3) 页面逻辑和业务逻辑混合

正是因为 ASP 程序代码很简单,所以在网页中许多简单操作都需要通过程序来实现。比如说,若要在网页中保留一些状态信息、进行输入验证或将数据填入下拉式菜单等,都需要将脚本代码安插在适当的 HTML 标记中。由于使用 ASP 制作的网页是从上到下逐行执行的,因此,有时为了程序的执行顺序,还需要手工更改程序所在的位置才能得到正确的结果。这样页面逻辑和业务逻辑混合在一起,代码难于读懂,页面的维护难度大大增加。

(4) Session 状态信息只限于单机上使用

若服务器只是在单机上执行,这个限制不会带来太大的困扰。但是当用户数量较大,一台服务器并不能服务所有的用户时,Server Farm 便应运而生。Server Farm 是将许多主机组织在一起,给用户提供一个一致的用户映像。因此,处理用户发出要求的主机很可能每次都不同,当 Session 状态信息不能够跨电脑时,系统的延展性就面临极大的挑战。

通常为了解决系统延展性问题必须自行编写程序,将 Session 状态数据存放在数据库中,如 SQL Server,或者自行开发 ASP State Server,以保留状态信息。

2. ASP.NET 的设计理念

ASP.NET 不能说是 ASP 的升级版本,因为它是为了建立动态 Web 应用程序面重新打造的全新技术。ASP.NET 的特色包括:

(1) 简化开发操作

前文提及,虽然 ASP 很容易使用,但是网页程序过大时会显得相当复杂。ASP.NET 在代码编写方面最大的特色是将页面逻辑和业务逻辑分开,它分离程序代码与显示内容,使丰富多彩的网页更容易编写,同时程序代码看起来更洁净、更简单。

(2) 语言独立性

在使用 ASP 制作网页时可以使用多种语言来编写程序代码,不过只限于脚本语言,如 VB Script、JScript。ASP.NET 则允许使用编译式的语言,提供较好的执行效率和跨语言的兼容性,如 Visual Basic.NET、C#、C++ With Managed Extension、JScript.NET。另外有一些合作厂商也提供开发 .NET 应用程序的支持,如 Cobol、Pascal、Perl、Smalltalk 等。

(3) 分离程序代码和网页内容

有时可能会觉得使用 ASP 制作网页过于复杂,这是因为程序代码太长,而且又将网页

显示的 HTML 标记和程序代码混杂在一起的缘故,以致于会有杂乱无章的感觉。ASP.NET 使用事件驱动 (Event-Driven) 与数据绑定 (Data Binding) 的开发方式,将程序代码与用户界面接口彻底分开。另外,可以使用 Code-Behind 的方式将程序代码和页面显示标记分离在不同的文件中,使程序的可读性更强。当然,如果仍然要使用 ASP 编写方式,仍可以在 ASP.NET 网页中使用 `<%=%>` 在 HTML 代码区输出变量,两者的执行效率并不会相差太多。

(4) 提高执行效率

由于 ASP.NET 的程序代码是编译过的,所以执行时会比 ASP 的直译方式快很多。另外,ASP.NET 也提供快取 (cache) 的能力,有效地缩短服务器的应答时间。

(5) 简化部署与组态的操作

若在 ASP 网页中调用组件,程序的部署过程会变得非常复杂。目前部署组件时需要复杂的注册操作,同时组件在使用中经常会被锁定而无法更新版本。应用程序部署到其他计算机上时还需要进行某些组态设置操作。因为这些因素,应用程序在部署时经常会遇到很多的问题。

在 ASP.NET 中不需要考虑组件注册的问题,直接将相关文件复制到目的计算机相应的目录下就可以了。

在组态 (Configuration) 方面,ASP.NET 采用容易阅读的 XML 组态文件,称为 Web.Config,只要将此文件放在 Web 应用程序的相关目录中就可以完成组态的操作。因此,可以使用文件复制或 FTP 的方式来设置 Web 应用程序的组态。

(6) 支持多种客户端类型

ASP.NET 提供许多 Server Control,在网页执行期间,能自动检测浏览器所支持的功能,并自动产生客户端浏览器可用的网页。所以对不同的客户端浏览器,可以只编写一份代码。

(7) 支持下一代的 Web Service

简单地说,Web Service 是指可以跨 Internet 调用的应用程序,提供应用程序重复使用的功能,它能使两个不同的系统拥有一个沟通的管道。

(8) 增进适用性

ASP.NET 能解决应用程序故障。对于内存泄漏 (Memory Leak) 的情况,能自动重新启动进程以增进适用性,从来不死机。任何会造成内存泄漏的程序代码或产生无穷循环或没有关闭使用的资源的程序代码将只会影响到一条线程。一旦 ASP.NET 检测到某条线程运行失常,它将会把新的请求 (Request) 导向新的线程,不再和这条有问题的线程进行交互。这条受损的线程将会被自动回收,而最后一个使用到此线程的请求 (Request) 将会被毁灭,这表示 Server 永远不需要重新启动来配置线程。

(9) 增进延展性

除了使用编译过的语言和快取能力增进延展性之外,ASP.NET 使用全新的 Session-State 管理方式,使应用程序可以很容易地在 Web Farm、Web Garden (一台 Server 上有多个处理进程) 中使用 Session。

(10) 更多的支持工具

提供更丰富的、以组件为基础的程序化模型。使开发 Web 应用程序的过程就像开发传

统的应用程序一样。

(11) 更佳的安全机制

在 ASP 中惟一能使用的验证方式是 Windows Authentication; 而 ASP.NET 则提供三种不同的登录验证方式: Windows、Passport 和 Cookie。也可以利用 Impersonation 功能, 使用登录者的权限执行一些程序代码或存取资源。

(12) Session 可以跨进程、跨机器

使用 ASP.NET, Session 的状态可以在不同计算机的不同进程 (Process) 中维护, 以解决 Web Farm 的 Session 维护问题。

ASP 和 ASP.NET 的比较如表 1-1 所示。

表 1-1 ASP 和 ASP.NET 比较

	ASP	ASP.NET
扩展名	.asp	.aspx
使用语言	VB Script、JScript 等脚本语言	任何支持 Common Language Sepecification 的语言, 如 Visual Basic.NET、C# 等
程序和网页设计	程序代码与网页混杂, 难以重复使用	程序代码可与显示界面分离, 独立成一个类 (Class)
是否支持 Web Service	不支持	支持
服务器程序	ASP.dll	xspisapi.dll
组态设置方式	使用 IIS 的 MMC 嵌入式管理单元	使用 XML 类型的组态文件: Web.Config
数据存取方式	使用 ADO (Active Data Object)	使用 ADO.NET

1.1.2 ASP.NET 的运行模式

在 ASP.NET 中可以使用哪些语言呢? 微软提供了三种程序语言:

- VB 功能比 VB Script 强, 注意这里的 VB 指的是 Visual Basic.NET, 和 Visual Basic 6.0 相比两者有着很大的差异。
- C# 改良后的 C++ 程序设计语言, 和 Java 非常类似。(注: 读作 C Sharp)
- Java Script 和 ASP 中使用的 JavaScript 一致。

其实, ASP.NET 不只支持以上 3 种语言, 凡是可以编译成 MSIL (Microsoft Intermediate Language) 的程序语言都可成为制作 ASP.NET 网页的语言。

1. MSIL 与 ASP.NET 的运作模式

MSIL 是另一种新的程序语言吗? 制作 ASP.NET 网页需要重新学习这种语言吗? 回答是否定的。MSIL 只是一种中介语言, 介于高级程序语言 (如 VB) 及机器码之间, 在 ASP.NET 中, 我们编写的程序语言先编译成 MSIL, 然后 MSIL 再被编译成机器码加以执行, 过程如图 1-1 所示。

为什么不直接将程序编译成机器码然后执行呢? 首先必须说明的是 MSIL 十分接近机器码, 从 MSIL 编译成机器码后执行的速度很快, 而介于 MSIL 与机器码之间的编译器称

为 JIT (Just In Time) 编译器, 通过不同计算机的 JIT 编译器, MSIL 可以即时地被编译成机器码, 然后在不同计算机中执行, 如图 1-2 所示。



图 1-1 ASP.NET 的编译过程

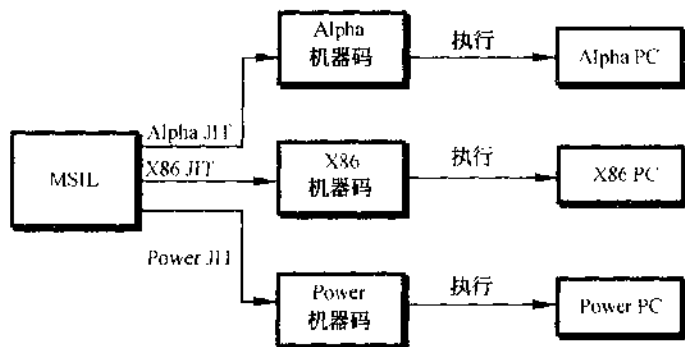


图 1-2 ASP.NET 在不同计算机上的编译方法

这样 MSIL 便具有跨平台的特性。但是对于 ASP.NET 开发者来说, 最关心的是以上运作模式的执行效率, 因为相对于 ASP 的执行模式, ASP.NET 的运作模式多一次编译工作。关于两者的执行效率, 可参看图 1-3。ASP.NET 第二次编译速度优于 ASP, 但是 ASP.NET 的第一次编译速度却慢于 ASP, 即 ASP.NET 两次编译执行所需的时间要大于 ASP 编译执行的时间。

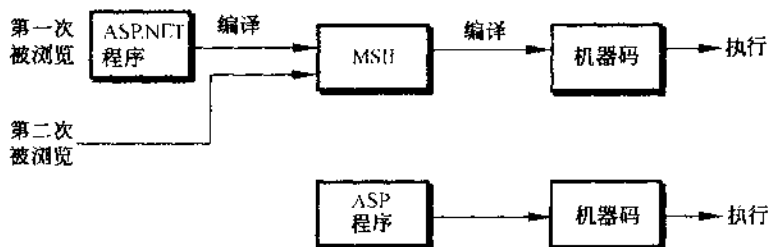


图 1-3 ASP 和 ASP.NET 编译方式的比较

ASP.NET 网页第一次被浏览时, 会经过两次编译, 所以速度较慢, 但是第一次被浏览后, MSIL 会被存储下来, 所以当同一网页第二次被浏览时, 只需要花费从 MSIL 编译到机器码然后再执行的时间, 结果比 ASP 网页快。

2. 不同程序语言的互通

前面已经说过凡是可以编译成 MSIL 的程序语言, 都可以成为 ASP.NET 的程序语言, 这意味着我们可以采用任何一种可以使用 MSIL 编译器的语言来编写 ASP.NET。该特性对于大型工程的开发特别有用, 因为每个人的专长可能不一样, 有些人熟悉 VB, 有些人熟悉

Java, 在不限定程序语言的情况下, 要把参与工程的人员组织起来将更容易一些。

当以.NET 程序语言, 如 C#、Visual Basic.NET 编写好程序代码后, 便可以使用.NET 提供的编译器来编译程序, 以便产生 EXE 或 DLL 文件。这时所编译出来的程序并不是 CPU 可以执行的机器码, 而是中间语言 (Intermediate Language, IL)。在执行时, CLR (Common Language Runtime) 的 Class Loader 会将 IL 程序代码载入内存, 然后通过及时 (Just-In-Time) 的方式将其编译成此平台可以执行的程序。正是由于这种特性, 在工程开发中, 也许有些人使用 C# 开发组件, 有些人使用 VB 开发组件, 有些人使用其他语言开发组件, 但是由于所有的组件都会先编译成 MSIL, 因此无论网页制作者使用哪一种程序语言, 网页都可以直接使用这些组件 (即将组件的源代码和使用组件的网页的源代码一起编译), 从而达到分享资源、节省成本的目的。运作过程如图 1-4 所示。

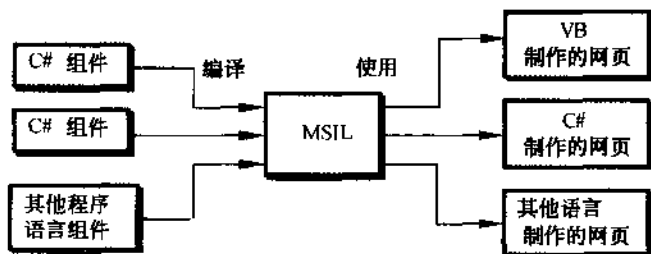


图 1-4 不同语言的互通

1.2 使用 ASP.NET 前的准备工作

在开始使用 ASP.NET 之前, 首先需要建立 ASP.NET 的操作平台, 然后需要对 ASP.NET 站点进行配置, 最后在这一节中将介绍如何移植本书的程序。

1.2.1 建立 ASP.NET 的操作平台

1. 建立 ASP.NET 操作平台之前

建立 ASP.NET 操作平台之前, 需要声明的一点是: 将 ASP 操作平台升级到 ASP.NET 操作平台之后, ASP 应用程序照样能够运行, 只是 ASP 网页不能和 ASP.NET 网页共享公共信息, 比如 Session 对象, Application 对象等, 即使两者在同一个目录下也不可以。

另外一个问题是需要将现有的 ASP 网页转换成 ASP.NET 网页吗? 下面是一些有用的建议:

- 希望效率更高时 当 ASP.NET 网页第一次被浏览时, Server 会先将其编译成 MSIL, 并且储存下来, 而该网页再度被浏览时, 不再重新编译 (除非.aspx 文件的内容有所改变), 因此可以提高效率。此外, ASP.NET 还具备有网页及数据 Cache 功能, 这也可以提高网页的响应速度。

- 需要经常维护或修改的网页 ASP.NET 能够采用 Visual Basic.NET 或者 C# 作为程序语言, 具备有完整的面向对象功能, 有助于维护网页。
- 新开发的网页 如果是在原来的 ASP 站点上新增的网页, 最好仍然采用 ASP 网页, 因为 ASP.NET 网页不能和其他 ASP 网页进行信息交换; 如果是开发全新的站点, 当然采用 ASP.NET。

2. 建立 ASP.NET 操作平台

目前 ASP.NET 所支持的系统有 Windows NT 4.0 版 (须安装 Service Pack 5 以上版本)、Windows 2000 (Server 或 Professional 版) 及 Windows 9x。建议最好使用 Windows 2000 来进行 ASP.NET 的开发。安装 ASP.NET 的必要元件如下:

- IIS 5.0 (Windows 2000) 或 IIS 4.0 (Windows NT Server 4.0 版)
- Internet Explorer 5.5 或者以上版本
- .NET Framework SDK

(1) IIS 5.0 的安装

如果使用的是 Windows 2000 Server, 在安装 Windows 2000 之后, IIS 5.0 会自动安装在系统中, 但是如果使用的是 Windows 2000 Professional, 则需要安装 IIS 5.0, 安装过程如下:

首先启动“控制面板”中的“添加/删除程序”, 在弹出的对话框中选择“添加/删除 Windows 组件”, 接着会出现“Windows 组件向导”对话框, 在该对话框中, 单击“下一步”按钮。接下来会出现如图 1-5 所示的画面。

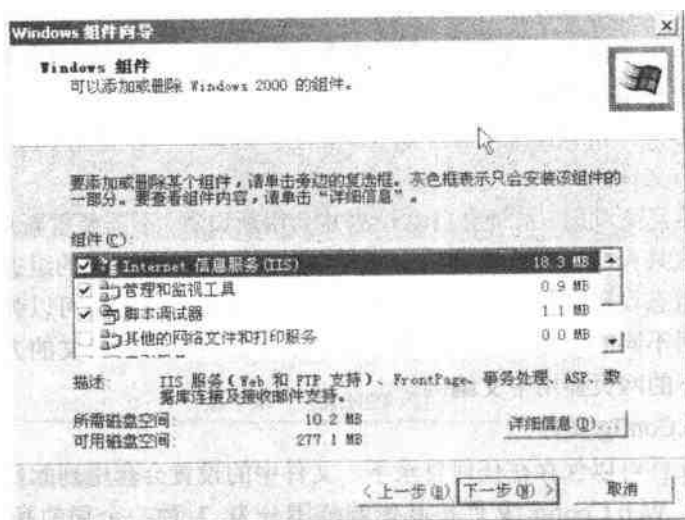


图 1-5 安装 IIS 5.0

在图 1-5 中选中“Internet 信息服务 (IIS)”复选框, 然后单击“下一步”按钮, 按照提示操作即可安装好 IIS 5.0。

(2) 安装 Internet Explorer

Windows 2000 默认的 Internet Explorer 版本是 5.0 版, 在这个版本中对 XML 支持尚有