

ASP.NET

后台数据库网站制作 实例经典

来宾 林盛雄 编著

This book is good for people who have minimal experience with VB.NET and ASP.NET. It is very easy to read and understand and find what you need. With little background in ASP.NET, you can find what you need for your web applications. This book is good as a starting point on the subject matter. The book goes through the items in short simple to understand paragraphs.

For experienced developers, you may want to skip the first two chapters. Chapter 2 gives an overview of relational database, how to use the database to retrieve and store the data. Chapter 3 covers methods for connecting to several different data source such as ODBC, OLEDB, OLEDB, and XML. Chapter 4 and Chapter 5 have lots of examples in reading and displaying the data using Data Reader and DataGrid objects. All the examples in the book use Data Grid control for displaying the data. Later in the book, it discusses ways of using the data grid control for creating, inserting, editing, and deleting records. I tried most of the examples in the book and they worked fine. This is one thing I have come to like about the WROCK press books is their examples are very understandable and easy to learn from.

Also, the experienced developer, who is interested in learning the subject matter may find many other sections in the book, boring because they will go through material they already know. It would be very nice to see a book such as this that can help the experienced developer who are already experienced as developers.

冶金工业出版社

ASP.NET 后台数据库网站制作 实例经典

来宾 林盛雄 编著

北 京

冶金工业出版社

2004

内 容 简 介

ASP.NET 是 ASP 的升级版, 具有许多革新意义的新特性。本书主要介绍了如何建立 ASP.NET 工作平台、进入 ASP.NET 的准备工作、ASP.NET 语言基础、ASP.NET 网页的基本结构、服务器控件、数据验证控件、数据库的创建与删除及管理、数据库的控制机制、ADO.NET 与数据库访问、应用程序和 Web Service、网络安全、综合实例分析以及网站开发技巧等内容。

本书内容翔实、由浅入深、结构清晰、实例丰富, 既可作为初学者的学习手册、中级学者良好的进阶教程和高级技术人员的参考资料, 同时也可以作为培训班的教材以及高等院校师生的参考书籍。

图书在版编目 (CIP) 数据

ASP.NET 后台数据库网站制作实例经典 / 来宾等编著.

北京: 冶金工业出版社, 2004.4

ISBN 7-5024-3493-3

I. A... II. 来... III. 主页制作—程序设计
IV. TP393.092

中国版本图书馆 CIP 数据核字 (2004) 第 017482 号

出版人 曹胜利 (北京沙滩嵩祝院北巷 39 号, 邮编 100009)

责任编辑 戈兰

湛江蓝星南华印务公司印刷; 冶金工业出版社发行; 各地新华书店经销

2004 年 4 月第 1 版, 2004 年 4 月第 1 次印刷

787mm×1092mm 1/16; 27 印张; 625 千字; 422 页; 1-2500 册

48.00 元

冶金工业出版社发行部 电话: (010) 64044283 传真: (010) 64027893

冶金书店 地址: 北京东四西大街 46 号 (100711) 电话: (010) 65289081

(本社图书如有印装质量问题, 本社发行部负责退换)

前 言

一、关于 ASP.NET

Internet 的快速发展,也带动了网页设计方法的进步,从早期静态的 HTML 到动态的 CGI,再到 ASP、JSP 与 PHP 等。虽然 ASP 易学好用,深得用户的支持,但其固有的诸如安全性、可靠性等方面的缺陷仍然无法满足程序开发人员的需要。为此,微软推出了 ASP 的下一代开发技术,即 ASP.NET。

ASP.NET 由微软的 .NET 技术细化而来,它跟 ASP 技术有关系,但又有着本质的不同,在程序代码重用、伸缩性、可靠性等方面更有优势。ASP.NET 完全基于模块与组件,具有更好的可扩展性与可定制性,在数据处理方面引入了许多新技术,有效缩短了 Web 应用程序的开发周期。

二、本书内容结构

本书共分 13 章,各章内容分别如下:

第 1 章:建立 ASP.NET 工作平台。主要介绍了 ASP.NET 的基本知识、软硬件需求、软件的安装以及 ASP 到 ASP.NET 的移植等相关知识。

第 2 章:进入 ASP.NET 的准备工作。主要介绍了 ASP.NET 相关技术、HTML 的基本概念、常用的 HTML 标记以及如何使用 HTML 控件等相关知识。

第 3 章:ASP.NET 语言基础。主要介绍了 ASP.NET 程序的结构组成、数据的基本组成及其基本运算、VB.NET 的基本控制结构等内容。

第 4 章:ASP.NET 网页的基本结构。主要介绍了第一个 ASP.NET 页、<% %>结构、Server 控件与动态网页、输入表单的基础、网页间的切换与数据传递以及 HTTP 对象等内容。

第 5 章:服务器控件。主要介绍了服务器控件示例、输入表单控件、Calendar Web 控件以及 AdRotator Web 控件等内容。

第 6 章:数据验证控件。主要介绍了 RequiredFieldValidator、CompareValidator、ValidationSummary、RangeValidator、RegularExpressionValidator 以及 CustomValidator 等数据验证控件的相关内容。

第 7 章:数据库的创建、删除和管理。主要介绍了数据库的基本知识、SQL 语言、Access 数据库以及 SQL Server 数据库等内容。

第 8 章:数据库的控制机制。主要介绍了数据完整性、游标、触发器、存储过程、约束、数据库的安全性及数据库的恢复等内容。

第 9 章:ADO.NET 与数据库访问。主要介绍了 ADO.NET 基础知识、连接数据库、DataReader 对象、DataSet 对象与 XML 文件的读写以及存储过程和数据的输出格式等内容。

第 10 章:应用程序和 Web Service。主要介绍了应用程序及 Web Service 的相关内容。

第 11 章:网络安全。主要介绍了 ASP.NET 的安全机制及网络管理的相关内容。

第 12 章：综合实例分析。主要介绍了 E-mail 自动传送、主页计数器、在线投票、在线论坛、留言板、电子商店以及新闻发布系统等相关内容。

第 13 章：网站开发技巧。主要介绍了 ASP.NET 编程中的十个技巧、窗体和功能性的关系与面向对象网络技术、ASP.NET 中的状态管理、在 ASP.NET 中进行文件处理、自定义一个服务器端的控件、验证 E-mail 是否真正存在、ASP.NET 中在线用户统计、用 ASP.NET 开发 Web 服务的技巧以及使用 ASP.NET 开发移动通讯的几种方法等内容。

三、本书特点

本书内容丰富、结构清晰，全面系统地介绍了 ASP.NET 的相关知识及其应用；并配有大量针对性强的应用实例和图解说明，技术含量高、知识面广、可操作性强，可以帮助读者理解和快速应用 ASP.NET。

四、本书适用对象

本书既可作为初学者的学习手册、中级学者良好的进阶教程和高级技术人员的参考资料，同时也可以作为培训班的教材以及高等院校师生的参考书籍。

由于编写时间仓促、编者水平有限，书中不足之处在所难免，恳请广大读者批评指正。

虽然经过严格的审核、精细的编辑，本书在质量上有了一定的保障，但我们的目标是力求尽善尽美，欢迎广大读者和专家对我们的工作提出宝贵建议，联系方法如下：

电子邮件：service@cnbook.net

网址：www.cnbook.net

此外，书中所有例子的源程序及相关说明均可从该网站免费下载，同时该网站还有一些其他相关书籍的介绍，可以方便读者选购参考。

编 者

2004 年 1 月

目 录

第 1 章 建立 ASP.NET 工作平台	1	2.3 常用的 HTML 标记	20
1.1 ASP.NET 简介	1	2.3.1 设置文字外观的标记	20
1.1.1 .NET Framework 架构	1	2.3.2 设置段落样式的标记	21
1.1.2 ASP.NET 的优势	4	2.3.3 设置对齐的标记	21
1.2 ASP.NET 的软硬件需求	5	2.3.4 表格 (Table) 标记	22
1.2.1 硬件需求	5	2.4 使用 HTML 控件	27
1.2.2 软件需求	5	2.4.1 按钮控件	27
1.3 软件的安装	5	2.4.2 文本控件	28
1.3.1 安装 IIS	5	2.4.3 选择项控件	31
1.3.2 安装 .NET Framework SDK	6	2.4.4 表格控件	34
1.3.3 安装 Visual Studio.NET	7	2.4.5 其他控件	35
1.4 ASP 到 ASP.NET 的移植	8	小结	38
1.4.1 语法和语义	8	综合练习二	38
1.4.2 语言兼容性	10	一、选择题	38
1.4.3 COM 互操作性	11	二、思考题	38
小结	12	三、上机题	38
综合练习一	12	第 3 章 ASP.NET 语言基础	40
一、选择题	12	3.1 ASP.NET 程序的结构组成	40
二、思考题	12	3.2 数据的基本组成	42
三、上机题	12	3.2.1 常量与变量	42
第 2 章 进入 ASP.NET 的准备工作	13	3.2.2 数据类型	43
2.1 ASP.NET 相关技术	13	3.3 数据的基本运算	44
2.1.1 浏览 ASP.NET 网页与开启		3.3.1 数学运算	44
.aspx 文件	13	3.3.2 字符串的输出	46
2.1.2 HTML 语言	14	3.3.3 日期与时间的输出	49
2.1.3 动态网页	15	3.3.4 数值的输出	52
2.1.4 ASP.NET 网页的程序语言	15	3.4 VB.NET 的基本控制结构	54
2.2 HTML 的基本概念	15	3.4.1 If 条件判断	54
2.2.1 标记的概念	15	3.4.2 Select Case 选择判断	55
2.2.2 区域的概念	16	3.4.3 For Next 循环	56
2.2.3 标记的属性	17	3.4.4 Do Loop 循环	57
2.2.4 层次的概念	17	3.4.5 无穷循环	59
2.2.5 超文本的概念	18	小结	60
2.2.6 其他概念	19	综合练习三	60

一、选择题	60	5.2.4 选择按钮	100
二、思考题	60	5.2.5 列表控件	104
三、上机题	60	5.2.6 Panel 控件	107
第4章 ASP.NET 网页的基本结构	62	5.3 Calendar Web 控件	109
4.1 第一个 ASP.NET 页	62	5.3.1 Calendar Web 控件简介	109
4.2 <% %>结构	63	5.3.2 Calendar Web 控件的常用属性	111
4.2.1 与 HTML 标记混合使用的		5.3.3 Calendar Web 控件的样式对象	112
Response.Write	63	5.3.4 Calendar Web 控件的事件	113
4.2.2 <%=数据%>:简化版的		5.4 AdRotator Web 控件	117
Response.Write	63	5.4.1 AdRotator Web 控件的基本属性	117
4.3 Server 控件与动态网页	64	5.4.2 产生 AdRotator Web 控件的文件	118
4.3.1 Label Web 服务器控件	64	5.4.3 AdRotator Web 控件的应用	118
4.3.2 利用 Page_Load 事件程序		小结	119
动态改变变量据	65	综合练习五	120
4.4 输入表单的基础	66	一、选择题	120
4.4.1 输入表单的基本结构	66	二、思考题	120
4.4.2 表单数据的输出	67	三、上机题	120
4.5 网页间的切换与数据传递	68	第6章 数据验证控件	122
4.5.1 网页的切换	68	6.1 RequiredFieldValidator	122
4.5.2 网页间的数据传递	69	6.2 CompareValidator	123
4.6 HTTP 对象	70	6.3 ValidationSummary	126
4.6.1 Request 对象	70	6.4 RangeValidator	130
4.6.2 Response 对象	74	6.5 RegularExpressionValidator	131
4.6.3 Application 对象	78	6.6 CustomValidator	134
4.6.4 Session 对象	80	小结	136
4.6.5 Cookies 对象	82	综合练习六	136
小结	88	一、选择题	136
综合练习四	88	二、思考题	136
一、选择题	88	三、上机题	137
二、思考题	88	第7章 数据库的创建、删除和管理	138
三、上机题	89	7.1 数据库简介	138
第5章 服务器控件	90	7.1.1 数据库的概念	138
5.1 服务器控件示例	90	7.1.2 网络技术和数据库	138
5.2 输入表单控件	92	7.1.3 关系数据库模型	140
5.2.1 文本框	92	7.2 SQL 语言	140
5.2.2 按钮控件	94	7.2.1 SQL 简介	140
5.2.3 列表框	97	7.2.2 SQL 的数据定义语言	141

7.2.3 SQL 的数据操作语句	143	8.5.3 CHECK 约束	177
7.2.4 SQL 的数据控制语句	147	8.5.4 UNIQUE 约束	178
7.2.5 SQL 的数据嵌入式使用	148	8.5.5 FOREIGN KEY 约束	179
7.3 Access 数据库	149	8.6 数据库的安全性	180
7.3.1 Access 数据库简介	149	8.7 数据库的恢复	181
7.3.2 用 Access 建立数据库	149	小结	184
7.4 SQL Server 数据库	152	综合练习八	184
7.4.1 SQL Server 简介	152	一、选择题	184
7.4.2 SQL Server 2000 的安装	153	二、思考题	185
7.4.3 SQL Server 2000 的工具	156	三、上机题	185
7.4.4 数据库的创建	157	第 9 章 ADO.NET 与数据库访问	186
7.4.5 表的创建和修改	158	9.1 ADO.NET 简介	186
7.4.6 索引和视图	159	9.2 连接数据库	187
7.4.7 实际应用	162	9.2.1 ADO.NET 访问数据库的步骤	187
小结	167	9.2.2 ADO.NET 对象模型	187
综合练习七	167	9.2.3 显示 DataSet	191
一、选择题	167	9.3 DataReader 对象	193
二、思考题	167	9.3.1 基本 Read 方法	193
三、上机题	167	9.3.2 更复杂的 Read	194
第 8 章 数据库的控制机制	168	9.3.3 把 DataReader 绑定到 DataGrid	195
8.1 数据完整性	168	9.3.4 利用 DataReader 插入记录	195
8.1.1 完整性的概念	168	9.4 DataSet 对象与 XML 文件的读写	196
8.1.2 完整性基本内容	168	9.5 存储过程	201
8.2 游标	170	9.5.1 创建存储过程	202
8.3 触发器	172	9.5.2 有返回值	204
8.3.1 触发器的概念	172	9.5.3 带输入参数	205
8.3.2 触发器的结构	173	9.5.4 带输出参数	207
8.3.3 触发器的功能	173	9.6 数据的输出格式	208
8.3.4 触发器是完整性保护的 充分条件	173	9.6.1 Repeater 控件	208
8.3.5 触发器的创建	174	9.6.2 DataList 控件	211
8.4 存储过程	174	9.6.3 DataGrid 控件	215
8.4.1 存储过程的概念	174	9.6.4 Repeater, DataList 和 DataGrid 的比较	223
8.4.2 存储过程的创建	174	小结	224
8.4.3 SQL Server 的存储过程	175	综合练习九	224
8.5 约束	176	一、选择题	224
8.5.1 PRIMARY KEY 约束	176	二、思考题	225
8.5.2 DEFAULT 约束	177		

三、上机题	225	12.1.1 MailMessage 及 SmtMail 对象	268
第 10 章 应用程序和 Web Service	226	12.1.2 MailMessage 对象的其他 属性	269
10.1 应用程序	226	12.1.3 完整例子	270
10.1.1 应用程序简介	226	12.2 主页计数器	273
10.1.2 配置 Web.config	229	12.2.1 文字计数器	273
10.1.3 编写 global.asax	233	12.2.2 图形计数器	277
10.2 Web Service	236	12.2.3 全站点计数器	280
10.2.1 Web Service 简介	236	12.3 在线投票	284
10.2.2 一个 Web Service 范例	242	12.3.1 表单设计	285
10.2.3 数据交换	244	12.3.2 数据表文件结构	286
10.2.4 使用对象和内部对象	248	12.3.3 程序设计技巧	287
小结	249	12.3.4 程序完整列表	288
综合练习十	249	12.4 在线论坛	290
一、选择题	249	12.4.1 基本功能设计	291
二、思考题	250	12.4.2 实现分层显示	295
三、上机题	250	12.4.3 实现查询功能	310
第 11 章 网络安全	251	12.5 留言板	322
11.1 ASP.NET 的安全机制	251	12.5.1 设计数据库	323
11.1.1 安全性简介	251	12.5.2 发表留言	323
11.1.2 身份验证和授权	251	12.5.3 显示留言记录	329
11.1.3 基于 Windows 的身份验证	252	12.6 电子商店	335
11.1.4 基于窗体的验证	254	12.6.1 设计数据库	336
11.1.5 授权用户和角色	256	12.6.2 浏览、购物页面	336
11.1.6 安全性和 XML Web Services ...	258	12.6.3 订单生成	351
11.2 网络管理	261	12.7 新闻发布系统	357
11.2.1 网络管理的发展历史	261	12.7.1 使用 XML 的新闻管理系统	357
11.2.2 网络管理的定义	262	12.7.2 使用 WebService 发布新闻	384
11.2.3 网络管理的功能	263	小结	389
11.2.4 网络管理协议简介	265	第 13 章 网站开发技巧	390
小结	267	13.1 ASP.NET 编程中的十个技巧	390
综合练习十一	267	13.2 窗体和功能性的关系与面向对象 网络技术	395
一、选择题	267	13.3 ASP.NET 中的状态管理	396
二、思考题	267	13.3.1 客户端的状态管理	396
三、上机题	267	13.3.2 服务器端的状态管理	398
第 12 章 综合实例分析	268		
12.1 E-mail 自动传送	268		

13.4 在 ASP.NET 中进行文件处理	399	13.8.5 避免使用 ASP.NET 会话状态 ..	411
13.4.1 通过 ASP.NET 页面进行 文件的读操作	399	13.9 使用 ASP.NET 开发移动通讯的 几种方法	411
13.4.2 ASP.NET 页面进行文件 写操作	401	13.9.1 XML 和 XSLT	412
13.4.3 在 ASP.NET 页面中完成文件 的拷贝、移动和删除操作	402	13.9.2 使用 MMIT	412
13.5 自定义一个服务器端的控件	403	13.9.3 各种方法的优点和缺点	413
13.6 验证 E-mail 是否真正存在	405	小结	413
13.7 ASP.NET 中在线用户统计	407	参考答案	414
13.7.1 用户显示页面的使用	408	第 1 章	414
13.7.2 global.asax 文件实现	408	第 2 章	414
13.8 用 ASP.NET 开发 Web 服务的 技巧	409	第 3 章	415
13.8.1 禁用 HTTP POST/GET 协议	409	第 4 章	416
13.8.2 用 TCPTrace 查看 SOAP 请求/应答消息	410	第 5 章	416
13.8.3 简化接口设计	410	第 6 章	419
13.8.4 在 Web.Config 中保存应用 私有的数据	411	第 7 章	420
		第 8 章	420
		第 9 章	421
		第 10 章	421
		第 11 章	422

第 1 章 建立 ASP.NET 工作平台

ASP.NET 是微软公司新推出的一种 Internet 编程技术,它采用效率较高的、面向对象的方法来创建动态 Web 应用程序。ASP.NET 是建立在公共语言运行库上的编程框架,可用于在服务器上生成功能强大的 Web 应用程序。本章将介绍 ASP.NET 的基本知识和.NET 平台的建立。

1.1 ASP.NET 简介

ASP.NET 又叫 ASP+,但它并不仅仅是 ASP 的简单升级,而是 Microsoft 推出的新一代 Active Server Pages 脚本语言。ASP.NET 是微软发展的新型体系结构.NET 的一部分,它的全新技术架构让每一个人的网络生活都变得更简单。ASP.NET 是一个用于 Web 开发的全新框架,其中包含了许多新的特性。

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,从而提升性能和伸缩性;ASP.NET 使用 Web 表单使开发更直观,利用面向对象技术促进组件的再利用。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和对数据捆绑的改进。供 ASP.NET 使用的库以及在 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。

1.1.1 .NET Framework 架构

ASP.NET 运行于.NET 平台,首先介绍一下微软的.NET 架构。

1. Microsoft Visual Studio.NET 简介

Web 应用程序(以因特网为基础的应用程序)的优点在于可以让企业间的商业数据及交易等行为透过因特网的通讯来彼此交换讯息。这样不但可以节省数据交换的时间,而且可以简化流程。

但是在新一代的.NET 开发平台还没有出现之前,让因特网应用程序达到上述的功能是一项浩大的工程。牵涉到的技术及程序开发语言,可能包括了 HTML、ASP、VBScript、JavaScript、C++、ADO、SQL、COM、MTS 等。这样的环境对于开发人员来说,想要快速开发一个功能强大且稳定可靠的 Web 应用程序,不是一项轻松的工作。

之前的 Microsoft Visual StudioTM(微软的解决方案开发平台)及 Windows 上的一些架构及服务,已经帮助程序设计师由单机平台的程序开发转为主从式(Client/Server)的架构来开发应用程序。

但是对于现在的环境而言,因为企业的客户或供货商等所使用的系统或软件可能不尽相同,开发人员所要面临的挑战是要如何整合所有的异质性资源,所以必须把焦点转到如何发展分布式的因特网应用程序架构,以便整合这些存在于不同平台或不同软件的异质资源。

现今最受欢迎的架构则为主/从式的架构，而发展最快速的架构则是以 Web 为基础的架构。在这种架构下程序逻辑及数据的处理都是在服务端，使用者是透过网络以浏览器来存取伺服器端的数据。这种架构的开发工具，微软早在前两版的 Visual Studio 中透过 Active Server Pages (ASP) 以及 IIS 3.0 就已经提供了。为了适应 Web 架构的快速发展及广大程序开发人员的需求，微软亦举办了上百场的研讨会和开发人员沟通、交换意见，目的是为了提供给开发人员更好的解决方案开发平台。而这个新一代开发平台目前已经推出了，即为 Visual Studio.NET。

Visual Studio.NET 开发平台里面包含 Visual Basic.NET、C#、Visual C++、ASP.NET 以及 Visual FoxPro。为了让这套开发平台更容易开发以因特网为基础的应用程序，这个开发平台做了许多和以往不同的改革，这个改革就是 .NET Framework。 .NET 架构（就是 .NET Framework，.NET 读作 dot Net）就是为了让开发分布式因特网应用程序架构变得更简单容易而发展出来的。

2. .NET Framework 的特点

.NET Framework 是微软的几个开发团队一起努力发展的成果，主要用来产生一个可以用来快速开发、部署网站服务及应用程序的开发平台。这个架构是两个项目的结果：

第一个项目的目的是用来改善 Windows 作业平台上的程序开发，特别是改善 COM（Component Object Model，组件对象模块。一种微软所制定的软件技术；让对象的功能可以被其他软件所调用，可以让组件重复使用、容易更新及维护）。

第二个项目则是制作一个以发展服务（Service）软件为目标的开发平台。

这两个项目团队三年多以前就已经在一起工作，他们希望可以发展出一种可以快速开发出以因特网为基础，而且易学易用的开发平台。为了要达到这些目标，.Net Framework 在设计时加入了下列特色：

1) 透过因特网的标准做整合

以 XML (eXtensible Markup Language, 延伸标注语言) 及 SOAP (Simple Object Access Protocol, 简单对象存取协议) 等标准通讯协议，将各种由不同环境所组成的应用程序及组件整合在一起工作。

2) 松散的整合组件

大多数具延展能力（可扩充功能）的系统，现阶段是以“异步讯息”为架构而建立的。要建立这种多层的架构非常复杂，而且工具很少。.NET Framework 不需要很严谨地定义每个组件的结构即可很轻松的整合，这样可提高程序的延展性。

3) 支持多种程序语言

许多程序设计师会使用多种语言来开发其解决方案，这是因为每种语言都有它的长处。例如某些语言对于数值计算效率较好，某些语言对于数据库的操作较为方便，而某些语言又有大量的链接库可供使用；所以没有办法强迫别人只学一种程序语言。.NET Framework 把这些语言整合起来，可以让开发人员使用不同的程序语言来开发解决方案，让程序设计师可以选择他们专长的程序语言，企业则可省去重新训练员工的成本。

4) 提高程序设计师的生产力

现今程序设计师人才非常缺乏，程序设计师在人力不足的情形之下就必须提高生产

力, 因为每个项目的时程很可能很急促; 况且公司也希望赶快结案好再进行下一个项目。正因如此, .NET Framework 的开发团队希望尽可能减少写程序会发生的问题, 让程序设计师专心于撰写企业法则 (企业处理数据的规则)。

所以 .NET Framework 有些节省时间的特色, 例如容易使用的自动交易机制、自动内存管理以及丰富的控件。

5) 完善的数据保全

目前因特网最受大家注目的是它的安全性。要设计一个安全性完善的因特网应用程序, 在设计时就必须考虑所有组件的保全设计, 而不能仅做一部分而已。 .NET Framework 在设计安全模型时即考虑到这点, 将所有的数据与程序代码做完善的安全防护。

6) 可用操作系统的服务

Windows 提供了比其他作业平台更丰富的服务及资源, 例如众多的数据存取服务、使用系统所提供的整合安全模式来做身分验证及保全的工作、交互式的使用者接口、成熟的对象模块、交易程序监视以及讯息队列服务。 .NET Framework 当然也将这些操作系统所提供出来的功能包装起来, 以更简单的方式提供程序设计师使用。

3. .NET Framework 概要

1) .NET Framework 就是微软 Web Services 引擎

有许多程序设计师和使用者都非常渴望有一个完善而且透明清楚的基础架构来建立 Web Services (因特网服务)。 .NET Framework 正是为了满足这一需求而提供的基础架构。 .NET Framework 提供了应用程序模型及关键技术, 让开发人员容易以原有的技术来产生、部署, 并可以继续发展具有高安全、高稳定, 并具高延展的 Web Services。对于 .NET Framework 而言, 所有的组件都可以成为 Web Services, Web Services 只不过是另一种形态的组件罢了。

微软将 COM 的优点整合进来, 它可以不用像 COM 那么严谨地来栓锁两个对象, .NET Framework 以松散的方式来栓锁 Web Services 这种形态的组件。这样的结果让开发人员非常容易发展出强而有力的 Web 服务组件, 提高了整体的安全及可靠性, 并且大大地增加系统的延展性。

2) .NET Framework 由三个部分组成

.NET Framework 包括了三大部分: 第一个部分是 Common Language Runtime (CLR, 所有 .NET 程序语言公用的执行时期组件), 第二部分是共享对象类别库 (提供所有 .NET 程序语言所需要的基本对象), 第三个部分是重新以组件的方式写成的 ASP.NET (旧版本则是以 asp.dll 提供 ASP 网页所需要的对象)。

Common Language Runtime (CLR) 架构在操作系统的服务上, 它负责应用程序实际的执行, 满足所有的应用程序的需求, 例如内存管理、处理安全问题、整合不同的程序语言等。Runtime 提供了许多帮助程序写作的简化功能, 以及应用程序的部署并同时加强程序稳定可靠的一些服务。

不过程序设计师实际上不会被 Runtime 影响, 因为他们所面对的是架构在 CLR 上面的共享对象类别库, 这个共享对象类别库可以被任何程序语言所使用。在这个类别中包含了以建构 Web 为基础的应用程序模型, 提供以架构 Web 服务与 Web 应用程序为目标的组件

及服务，这就是要讨论的 ASP.NET。

1.1.2 ASP.NET 的优势

1. 增强的性能

ASP.NET 是在服务器上运行的编译好的公共语言运行库代码。与被解释的前辈不同，ASP.NET 可利用早期绑定、实时编译、本机优化和盒外缓存服务。这相当于在编写代码行之前便显著提高了性能。

2. 世界级的工具支持

ASP.NET 框架补充了 Visual Studio 集成开发环境中的大量工具箱和设计器。WYSIWYG 编辑、拖放服务器控件和自动部署只是这个强大的工具所提供功能中的少数几种。

3. 威力和灵活性

由于 ASP.NET 基于公共语言运行库，因此 Web 应用程序开发人员可以利用整个平台的威力和灵活性。.NET 框架类库、消息处理和数据访问解决方案都可从 Web 无缝访问。ASP.NET 也与语言无关，所以可以选择最适合应用程序的语言，或跨多种语言分割应用程序，ASP.NET 程序可以用 VB.NET、C# 和 JavaScript 语言来编写。

另外，公共语言运行库的交互性保证在迁移到 ASP.NET 时保留基于 COM 的开发中的现有投资。

4. 简易性

ASP.NET 使执行常见任务变得容易，从简单的窗体提交和客户端身份验证到部署和站点配置。

例如，ASP.NET 页框架可以生成将应用程序逻辑与表示代码清楚分开的用户界面和在类似 Visual Basic 的简单窗体处理模型中处理事件。另外，公共语言运行库利用托管代码服务（如自动引用计数和垃圾回收）简化了开发。

5. 可管理性

ASP.NET 采用基于文本的分层配置系统，简化了将设置应用于服务器环境和 Web 应用程序。由于配置信息是以纯文本形式存储的，因此可以在没有本地管理工具帮助的情况下应用新设置。因此“零本地管理”哲学也扩展到了 ASP.NET 框架应用程序的部署。只需将必要的文件复制到服务器，即可将 ASP.NET 框架应用程序部署到服务器。不需要重新启动服务器，即使是在部署或替换运行的编译代码时。

6. 可缩放性和可用性

ASP.NET 在设计时考虑了可缩放性，增加了专门用于在聚集环境和多处理器环境中提高性能的功能。

另外，进程受到 ASP.NET 运行库的密切监视和管理，以便当进程行为不正常（泄漏、死锁）时，可就地创建新进程，以帮助保持应用程序始终可用于处理请求。

7. 自定义性和扩展性

ASP.NET 随附了一个设计周到的结构，它使开发人员可以在适当的级别“插入”代码。实际上，可以用自己编写的自定义组件扩展或替换 ASP.NET 运行库的任何子组件。实现自

定义身份验证或状态服务一直没有变得更容易。

8. 安全性

借助内置的 Windows 身份验证和基于每个应用程序的配置,可以保证应用程序是安全的。

1.2 ASP.NET 的软硬件需求

为了避免安装过程中不必要的麻烦,本节先介绍一下 ASP.NET 的软硬件需求。

1.2.1 硬件需求

1. CPU

客户机:(一台不以服务器容量工作的计算机):最低 90MHz 的 Pentium 级处理器,推荐 90MHz 的 Pentium 级处理器或更快。

服务器:(一台以服务器容量工作的计算机):最低 133MHz 的 Pentium 级处理器,推荐 133MHz 的 Pentium 级处理器或更快。

2. 内存

客户机:32MB 内存,建议 96MB 或更高。

服务器:128MB 内存,建议 256MB 或更高。

3. 硬盘

系统驱动器上要求 111MB 的可用空间,安装驱动器上要求 145MB 的可用空间。

1.2.2 软件需求

1. 操作系统

下列平台支持 Windows .NET Framework:

带有 Service Pack 2.0 以上的 Microsoft Windows 2000 Professional。

带有 Service Pack 2.0 以上的 Microsoft Windows 2000 Server。

带有 Service Pack 2.0 以上的 Microsoft Windows 2000 Advanced Server。

Microsoft Windows XP Professional。

Microsoft Windows Server 2003 (.NET Framework 作为操作系统的一部分安装)。

2. 附加软件要求

如果要访问 SQL Server,必须安装有完整的 SQL Server 或是 MSDE (SQL Server Desktop Engine)。

1.3 软件的安装

在 .NET 环境下,ASP.NET 和 ASP 是互不干扰的。以前的 ASP 文件仍然保存为“.asp”,而新的 ASP.NET 则保存为“.aspx”。

1.3.1 安装 IIS

(1) 启动“开始”菜单的“控制面板”,单击“添加或删除程序”,接着会出现“添

加或删除程序”窗口。如图 1-1 所示。

(2) 选取“添加/删除 Windows 组件”，接着会出现“Windows 组件向导”对话框，选中“Internet 信息服务 (IIS)”，单击“下一步”按钮，如图 1-2 所示，接着按照提示一路安装下去，即可完成 IIS 的安装。



图 1-1

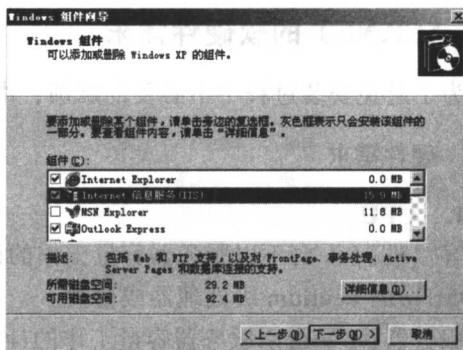


图 1-2

1.3.2 安装 .NET Framework SDK

.NET Framework SDK 中不仅包括了 .NET Framework，也包括了编写、构建、测试和部署 .NET Framework 应用程序所需的——文档、示例代码、命令行工具以及编译器。

在安装 .NET Framework SDK 前，请务必先安装 IIS，如果先安装 .NET Framework SDK，然后才安装 IIS，则 ASP.NET 网页有可能无法执行。

.NET Framework SDK 可以从微软的网站 (<http://www.microsoft.com/net>) 下载，现在 .NET Framework SDK 已经出了 1.1 版本。下载后先执行其中的 dotnetfx.exe，安装程序会自动解压，然后弹出提示安装对话框，如图 1-3 所示，选择“是”。

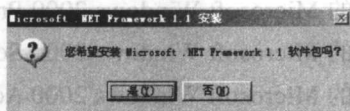


图 1-3

然后出现“许可协议”对话框，如图 1-4 所示，选择“同意”，单击“安装”。接下来出现正在安装画面，如图 1-5 所示。

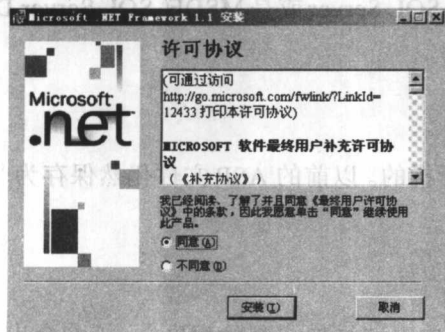


图 1-4

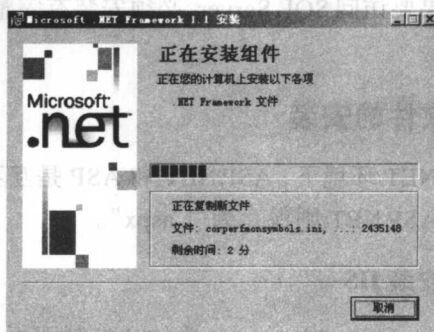


图 1-5

最后安装成功结束, 出现如图 1-6 所示的提示。

安装好 .NET Framework 后再安装 .NET Framework SDK。执行 setup.exe, 会弹出提示安装对话框, 如图 1-7 所示, 单击“是”。



图 1-6

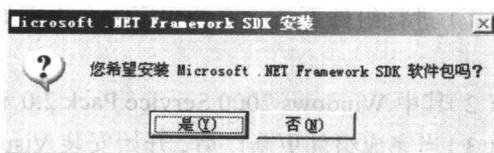


图 1-7

出现欢迎对话框, 如图 1-8 所示, 直接单击“下一步”。

接着出现许可协议对话框, 如图 1-9 所示, 选择“同意”, 单击“下一步”继续。

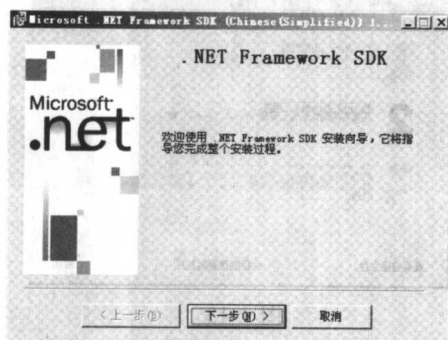


图 1-8

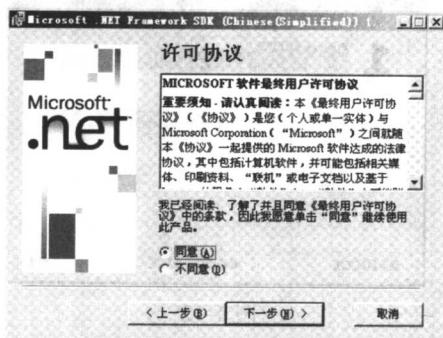


图 1-9

然后会看到如图 1-10 所示的画面。在这里选中软件开发包和 SDK 例子, 这样就可以通过这些例子来学习和使用 .NET。

选择安装目标文件夹, 如图 1-11 所示, 单击“浏览”进行更改, 选好目标文件夹后单击“下一步”开始安装。

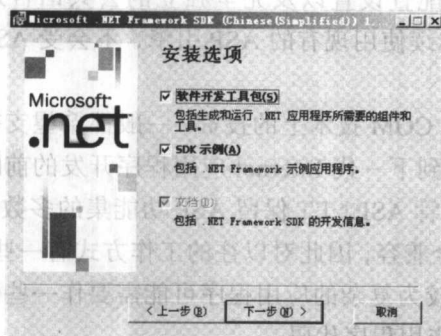


图 1-10



图 1-11

安装好 .NET Framework SDK 后, 在 .NET Framework SDK 的安装目录就可以找到 .NET Framework 的快速入门、教程和大量的例子。

1.3.3 安装 Visual Studio.NET

毫无疑问, Visual Studio.NET 是迄今为止开发 .NET 程序最优秀的开发工具。Visual