

 .NET 开发专家

ASP.NET 2.0

技术详解与应用实例

/HOPE/.NET/programming

更多图书请访问 www.bhp.com.cn  

HOP[®] programming

数据访问

成员角色

站点导航

Provider模型

案例剖析

...

快速掌握和应用
ASP.NET 2.0
新技术

更多的技术支持请访问

<http://bbs.aspcool.com>

<http://mqingqing123.cnblogs.com>

<http://www.aspx-net.com>

李万宝 编著

兵器工业出版社



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

TP393.092

581D

2007

.NET 开发专家

ASP.NET 2.0

技术详解与应用实例

HOPE/.NET/programming

更多图书请访问 www.bhp.com.cn V 9

HOP^erogramming

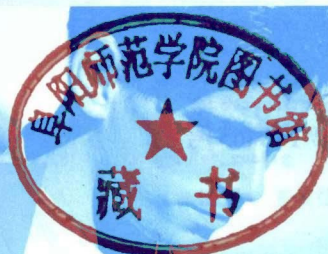
数据访问

成员角色

站点导航

Provider模型

案例剖析



快速掌握和应用
ASP.NET 2.0
新技术

更多的技术支持请访问

<http://bbs.aspcool.com>

<http://mqingqing123.cnblogs.com>

<http://www.aspx-net.com>

李万宝 编著 著 编著 李万宝 编著

兵器工业出版社



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

内 容 简 介

本书详细地介绍了 ASP.NET 2.0 新技术在 Web 开发方面的具体实现。通过自定义 Provider 模型说明了 ASP.NET 2.0 Provider 实现的原理；通过自定义数据源控件介绍了数据源控件的运行本质。除此以外，本书还详细介绍了母版、主题、站点导航、WebParts 和 ASP.NET 2.0 应用程序。本书的后四章介绍了 ASP.NET 2.0 Person Starter Kits, ASP.NET 2.0 Club Starter Kit, ASP.NET Classfieds Starter Kit 和 ASP.NET Time Tracker Start kit 的实现要点，进一步说明了 ASP.NET2.0 新技术的实际使用。

本书适合具有一定 ASP.NET 1.1 基础而想进一步了解 ASP.NET 2.0 的读者。

书中所附光盘的 CODE 文件夹中提供了部分示例对应的源代码。其中，“14”文件夹中提供了 Access 版本的 Provider 示例模型，beerhouse 文件夹中是一个多项目解决方案示例代码，供大家参考。

图书在版编目（CIP）数据

ASP.NET 2.0 技术详解与应用实例 / 李万宝编著.

北京：兵器工业出版社；北京希望电子出版社，2007.2

ISBN 978-7-80172-794-7

I. A... II. 李... III. 主页制作—程序设计

IV. TP393.092

中国版本图书馆 CIP 数据核字（2006）第 146465 号

出版发行：兵器工业出版社 北京希望电子出版社

邮编社址：100089 北京市海淀区车道沟 10 号

100085 北京市海淀区上地 3 街 9 号

金隅嘉华大厦 C 座 611

电 话：(010) 82702660（发行） (010) 62541992（门市）

经 销：各地新华书店 软件连锁店

印 刷：北京双青印刷厂

版 次：2007 年 2 月第 1 版第 1 次印刷

封面设计：刘孝琼

责任编辑：郭春临 宋丽华 李兴旺

责任校对：但明天

开 本：787×1092 1/16

印 张：47

印 数：1-4000

字 数：1351 千字

定 价：68.00 元（配 1 张光盘）

（版权所有 翻印必究 印装有误 负责调换）

前言

ASP.NET 2.0 是微软公司推出的 ASP.NET 1.1 的后续版本。虽然 ASP.NET 2.0 保持了和 ASP.NET 1.1 的兼容性，但是它在设计与体系结构上和 ASP.NET 1.1 已经有了很大的差别。鉴于此，笔者在参考同类书籍和网站资料的基础上编写了本书。

在撰写本书时，笔者试图表达以下两个观点。

观点一：如何理解 ASP.NET 2.0 的数据处理。

利用 ASP.NET 2.0 提供的数据源控件，开发一个关于数据库的具有编辑、分页和排序功能的页面可能用不了十分钟。利用 ASP.NET 2.0 内置数据源控件的支持，使得 Web 开发变得越来越简单、方便。这种开发方式带来的另外一个好处是，利用内置的数据访问功能，ASP.NET 2.0 能够使用大家公认的最为优化的代码处理数据的逻辑处理。但是，另外一个缺点是，数据源控件屏蔽了太多内置实现的细节，使得想要扩展数据的逻辑处理变得复杂。本书就来揭开 ASP.NET 2.0 数据源运行的本质，使得开发人员很容易扩展数据源控件的功能。在本书里还可以学会如何建立简单的自定义数据源控件，来进一步了解数据源控件的本质。

观点二：如何理解 ASP.NET 2.0 的 Provider 模型。

Provider 模型在 ASP.NET 2.0 里太重要了。从成员角色到站点导航，从系统监视到 Session 会话，ASP.NET 2.0 都使用了 Provider 模型。但是 Provider 模型同样给我们屏蔽了许多内置实现的细节。本书通过介绍自定义 Provider 模型来详细向你展示 ASP.NET 2.0 每一个 Provider 模型实现的具体细节。本书介绍的每一个自定义 Provider 模型都是完整的示例，既可以作为练习又可以直接在自己的项目里使用。

在本书的后半部分通过对 ASP.NET 2.0 Starter Kits 的介绍与扩展详细说明了如何使用 ASP.NET 2.0 的新技术。

本书的附录部分介绍了如何建立多项目应用程序解决方案以及 Web Site 和 Web Project 的区别与联系。

由于时间仓促，加之水平有限，书中难免存在错误和不足之处，敬请广大读者批评指正。如果有问题、建议或者相关讨论可以发送邮件到 mqingqing123@hotmail.com 和编者交流。读者还可以到 <http://www.aspx-net.com> 获取帮助。

读者对象

本书主要是为具有一定 ASP.NET 1.1 基础而想进一步了解 ASP.NET 2.0 的读者而编写的。本书也适合熟悉 HTML 语言和 ASP.NET 基本技能（包括 Web 页的基本概念，窗体、脚本和数据访问），而不具有 ASP.NET 方面的经验的读者。

对于以前没有开发过 Web 页或者还不熟悉 ASP.NET 的基本交互功能的读者，学习本书可能有些困难。建议这类读者先学习 ASP.NET 基本知识和技能之后，再学习本书。

光盘使用说明

这里介绍一下书中所附光盘代码的使用。

书中所附光盘的 CODE 文件夹中提供了部分示例对应的源代码。其中：

“14”文件夹中提供了 Access 版本的 Provider 示例模型，beerhouse 文件夹中是一个多项目解决方案示例代码，供大家参考。

下面介绍一下如何使用书附光盘中附录对应的代码。

附录代码里提供了附录 A 和附录 B 使用的源代码。要使用附录 A 的源代码，请直接使用 VS.NET 2005 打开 web.sln。

附录 A 在开发时文件存放路径是 E:\Code\fulu，所以如果直接打开该项目，且不是同样使用该目录则可能会发生错误。

该错误修复过程如下：

(1) 用记事本打开 web.sln，其内容如下：

```
Microsoft Visual Studio Solution File, Format Version 9.00
# Visual Studio 2005
Project("{E24C65DC-7377-472B-9ABA-BC803B73C61A}") = "E:\...\Web\","E:\Code\fulu\Web\","{A9AC1E66-234B-4473-8801-B758AC35C818}"

ProjectSection(WebsiteProperties) = preProject
    Debug.AspNetCompiler.VirtualPath = "/Web"
    Debug.AspNetCompiler.PhysicalPath = "E:\Code\fulu\Web\"
    Debug.AspNetCompiler.TargetPath = "PrecompiledWeb\Web\"
    Debug.AspNetCompiler.Updateable = "true"
    Debug.AspNetCompiler.ForceOverwrite = "true"
    Debug.AspNetCompiler.FixedNames = "false"
    Debug.AspNetCompiler.Debug = "True"
    Release.AspNetCompiler.VirtualPath = "/Web"
    Release.AspNetCompiler.PhysicalPath = "E:\Code\fulu\Web\"
    Release.AspNetCompiler.TargetPath = "PrecompiledWeb\Web\"
    Release.AspNetCompiler.Updateable = "true"
    Release.AspNetCompiler.ForceOverwrite = "true"
    Release.AspNetCompiler.FixedNames = "false"
    Release.AspNetCompiler.Debug = "False"
    VWDPort = "1082"
    DefaultWebSiteLanguage = "Visual C#"
EndProjectSection
```



```
ProjectSection(ProjectDependencies) = postProject
    {393B6F1D-F793-4C4A-B782-376898B337E0} =
{393B6F1D-F793-4C4A-B782-376898B337E0}
EndProjectSection
EndProject

Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "DAL_BLL",
"E:\Code\fulu\DAL_BLL\DAL_BLL.csproj",
"{393B6F1D-F793-4C4A-B782-376898B337E0}"
EndProject

Global
GlobalSection(SolutionConfigurationPlatforms) = preSolution
    Debug|.NET = Debug|.NET
    Debug|Any CPU = Debug|Any CPU
    Debug|Mixed Platforms = Debug|Mixed Platforms
    Release|.NET = Release|.NET
    Release|Any CPU = Release|Any CPU
    Release|Mixed Platforms = Release|Mixed Platforms
EndGlobalSection

GlobalSection(ProjectConfigurationPlatforms) = postSolution
    {A9AC1E66-234B-4473-8801-B758AC35C818}.Debug|.NET.ActiveCfg =
Debug|.NET
    {A9AC1E66-234B-4473-8801-B758AC35C818}.Debug|.NET.Build.0 =
Debug|.NET
    {A9AC1E66-234B-4473-8801-B758AC35C818}.Debug|Any CPU.ActiveCfg =
Debug|.NET
    {A9AC1E66-234B-4473-8801-B758AC35C818}.Debug|Mixed
Platforms.ActiveCfg = Debug|.NET
    {A9AC1E66-234B-4473-8801-B758AC35C818}.Debug|Mixed Platforms.Build.0
= Debug|.NET
    {A9AC1E66-234B-4473-8801-B758AC35C818}.Release|.NET.ActiveCfg =
Debug|.NET
    {A9AC1E66-234B-4473-8801-B758AC35C818}.Release|.NET.Build.0 =
Debug|.NET
    {A9AC1E66-234B-4473-8801-B758AC35C818}.Release|Any CPU.ActiveCfg =
Debug|.NET
```

```

    {A9AC1E66-234B-4473-8801-B758AC35C818}.Release|Mixed
Platforms.ActiveCfg = Debug|.NET
    {A9AC1E66-234B-4473-8801-B758AC35C818}.Release|Mixed
Platforms.Build.0 = Debug|.NET
    {393B6F1D-F793-4C4A-B782-376898B337E0}.Debug|.NET.ActiveCfg =
Debug|Any CPU
    {393B6F1D-F793-4C4A-B782-376898B337E0}.Debug|Any CPU.ActiveCfg =
Debug|Any CPU
    {393B6F1D-F793-4C4A-B782-376898B337E0}.Debug|Any CPU.Build.0 =
Debug|Any CPU
    {393B6F1D-F793-4C4A-B782-376898B337E0}.Debug|Mixed
Platforms.ActiveCfg = Debug|Any CPU
    {393B6F1D-F793-4C4A-B782-376898B337E0}.Debug|Mixed Platforms.Build.0
= Debug|Any CPU
    {393B6F1D-F793-4C4A-B782-376898B337E0}.Release|.NET.ActiveCfg =
Release|Any CPU
    {393B6F1D-F793-4C4A-B782-376898B337E0}.Release|Any CPU.ActiveCfg =
Release|Any CPU
    {393B6F1D-F793-4C4A-B782-376898B337E0}.Release|Any CPU.Build.0 =
Release|Any CPU
    {393B6F1D-F793-4C4A-B782-376898B337E0}.Release|Mixed
Platforms.ActiveCfg = Release|Any CPU
    {393B6F1D-F793-4C4A-B782-376898B337E0}.Release|Mixed
Platforms.Build.0 = Release|Any CPU
EndGlobalSection

GlobalSection(SolutionProperties) = preSolution
    HideSolutionNode = FALSE
EndGlobalSection

EndGlobal

```

(2) 请将粗体内容更改为你要存放项目路径的地址。

如果使用附录 B 的源代码，直接打开 WebApplication1 目录下的 WebApplication1.sln。

编 者

目 录

第1章 ASP.NET 2.0 安装、使用

与新特性说明.....	1
1.1 ASP.NET 2.0 的安装与基本使用	1
1.1.1 下载、安装.NET Framework 2.0.....	1
1.1.2 Visual Studio.NET 2005 的安装.....	2
1.1.3 Microsoft SQL Server 2005 Express Edition 的下载和安装	4
1.1.4 建立第一个应用程序 MyFirstPage 及 VS.NET 2005 的介绍	6
1.1.5 定制开发环境	6
1.1.6 添加一个新的页面	8
1.1.7 建立页面和代码分离文件	12
1.1.8 使用内置的 Web 服务器 (built-in Web Servers)	12
1.2 SQL Server 2005 数据库的基本使用	13
1.2.1 启动 SQL Server 2005 服务	13
1.2.2 建立数据库	14
1.2.3 使用 ASP.NET 2.0 读取数据库	16
1.2.4 App_Data 目录与数据库的 动态加载	17
1.2.5 数据库连接字符串的 快速配置与读取	19
1.3 C# 2.0 语法新特性简介	21
1.3.1 泛型	21
1.3.2 匿名方法	25
1.3.3 迭代器	26
1.3.4 不完全类型	29
1.4 IIS、应用程序和 ASP.NET 2.0	30
1.4.1 IIS 快速浏览	30
1.4.2 ASP.NET 2.0 和应用程序	31
1.4.3 使用 Visual Studio.NET 2005 新建和打开 Web 项目	33
1.4.4 利用 VS.NET 2005 和 IIS 管理 Web 项目的基本准则	35
1.4.5 IIS 和 ASP.NET 2.0 版本问题	36
1.4.6 在 VS.NET 2005 内置的 Web 服务器 和 IIS 之间切换	37

1.5 ASP.NET 2.0 代码组织与应用程序

编译方式.....	37
1.5.1 内置代码与隐藏文件	37
1.5.2 ASP.NET 2.0 提供的解决方案	40
1.5.3 页面的编译	42
1.5.4 Inherits 属性	46
1.5.5 页面部署的实现	47

1.6 App_Code 目录与类的动态编译以及 多语言使用

1.6.1 App_Code 目录	52
1.6.2 Bin 目录与全局程序集缓存	54
1.6.3 在 App_Code 里使用多种语言	54

第2章 数据访问

2.1 数据源控件概述

2.1.1 DataSource 数据源控件总体概述	58
2.1.2 数据源控件类型概述	58

2.2 SqlDataSource 控件

2.2.1 SqlDataSource 的基本使用	59
2.2.2 使用存储过程	65
2.2.3 返回值 DataReader/DataSet 的选择 ..	67
2.2.4 数据编辑	67
2.2.5 数据筛选	71
2.2.6 SqlDataSource 参数来源	73
2.2.7 默认参数的选择与更改	76
2.2.8 数据缓存	79
2.2.9 SqlDataSource 事件	81
2.2.10 利用 VS.NET 2005 的可视化 支持	83

2.3 ObjectDataSource 控件

2.3.1 SqlDataSource 和 ObjectDataSource 控件的比较	83
2.3.2 ObjectDataSource 的概述	84
2.3.3 绑定到数据访问层	85
2.3.4 绑定到业务逻辑	89
2.3.5 DataKeyNames 和 OldValuesParameterFormatString	91
2.3.6 冲突检测 ConflictDetection	96

2.3.7 数据分页排序	101
2.3.8 ObjectDataSource 事件与设计器的支持	107
2.4 AccessDataSource 控件	108
2.5 数据集与数据适配器	108
2.5.1 强类型化数据集简介	108
2.5.2 建立数据集与数据适配器	109
2.5.3 为数据适配器增加参数	117
2.5.4 返回聚合值	119
2.5.5 进一步理解数据集和数据适配器	120
2.5.6 模式的编译	122
2.5.7 自动生成类文件	128
2.5.8 DataMember	128
2.6 工厂模式与在 ADO.NET 2.0 里的实际应用	129
2.6.1 简单工厂模式	129
2.6.2 抽象工厂模式	131
2.6.3 工厂模式在 ADO.NET 2.0 中的实际应用	135
2.7 GridView 控件	138
2.7.1 ImageField	138
2.7.2 CheckBoxField 列	142
2.7.3 模板列	143
2.7.4 按钮列	147
2.7.5 选择分页与排序	151
2.7.6 格式化数据	157
2.7.7 GridView 事件	160
2.8 DetailsView 控件	166
2.8.1 DetailsView 控件概述	166
2.8.2 数据分页	168
2.8.3 数据编辑	170
2.9 FormView 控件	173
2.10 数据源控件与数据绑定控件的进一步简单讨论	176
2.10.1 ASP.NET 2.0 为什么提供新的数据源模型	176
2.10.2 ASP.NET 2.0 中的数据绑定控件	176
2.10.3 数据绑定机制概述	177
2.10.4 数据绑定控件具体实现细节	179

2.10.5 自定义 CustomerDataSource 数据源控件	188
2.10.6 自定义 CustomerDataSourceDesigner 设计器	193
2.11 自定义数据源参数	199
第 3 章 样式、主题和外观	201
3.1 CSS 技术	201
3.1.1 什么是 CSS	201
3.1.2 进一步了解 CSS 样式单	201
3.1.3 样式单的嵌套使用	204
3.1.4 样式中属性值的单位	205
3.1.5 绝对路径和相对路径	206
3.1.6 CSS 属性	209
3.1.7 CSS 中的滤镜	223
3.2 向 HTML 控件应用样式	226
Style 属性	228
3.3 向 Web 服务器控件应用样式	229
3.4 使用主题定义页面样式	233
3.4.1 建立一个主题	233
3.4.2 全局主题和应用程序主题	235
3.4.3 为页指定主题	235
3.4.4 在 web.config 里指定主题	236
3.4.5 对控件禁用主题	236
3.4.6 默认外观与命名外观	236
3.5 主题和外观的内容	237
3.6 使用主题的服务器端样式	242
声明服务器端样式	242
3.7 Theme, StyleSheetTheme 和控件属性优先级	243
3.8 Visual Web Developer 支持	243
3.9 主题和配置文件	244
第 4 章 母版	246
4.1 母版页的概述	246
4.1.1 母版页	246
4.1.2 替换占位符基本机制	252
4.1.3 母版页中的 URL 重置	255
4.1.4 母版页和配置	256
4.2 通过程序访问母版	256
4.2.1 母版和内容页的运行级别机制	256
4.2.2 处理 PreInit 事件	258

4.3 与母版交互.....	260
4.3.1 使用 FindControl 查找控件	260
4.3.2 使用 MasterType 指令	262
4.4 深入母版页和内容页交互.....	264
4.4.1 处理母版控件里的事件	264
4.4.2 处理页面回发	266
4.4.3 母版页面事件顺序	267
4.5 Headers,Scripts 和 Meta Tags 元素	268
4.6 访问 Page 指令	270
4.7 在母版里使用 FindControl	271
4.8 名称重写	274
4.9 母版与主题	275
4.10 嵌套母版	275
第 5 章 成员角色与登录控件	278
5.1 身份验证	278
5.1.1 身份验证回顾	278
5.1.2 ASP.NET 2.0 的成员角色 管理概述	279
5.1.3 登录控件概述	280
5.2 典型应用程序身份验证和授权设计	280
5.3 成员关系提供模型	296
5.3.1 ASPNETDB.MDF 数据库	296
5.3.2 自定义 SqlmembershipProvider	296
5.3.3 成员关系类	298
5.4 角色提供模型	303
5.4.1 角色管理模型	303
5.4.2 管理角色	305
5.4.3 基于角色的授权	305
5.4.4 角色管理类	307
5.5 综合使用成员角色关系	311
5.6 成员角色数据库 ASPNETDB.MDF 的 探讨	312
5.6.1 应用成员/角色模式到已有的 数据库	313
5.6.2 使用现有 SQL Server 2000 数据库 存放成员/角色关系	314
5.6.3 成员/角色关系对应的 SQL 脚本	315
5.6.4 asp_regsql.exe 命令	317
5.6.5 始终设置成员, 角色关系里的 applicationName 属性	318
5.7 使用 IIS 进行 ASP.NET 2.0 成员/角色 管理	320
5.7.1 使用 IIS 进行 ASP.NET 2.0 成员/角色 管理的原因	320
5.7.2 采用的技术	320
5.7.3 应用程序和项目	321
5.8 自定义 Provider 角色模型	337
5.8.1 Provider 模型整体概述	337
5.8.2 membership Provider 的基础	338
5.8.3 为什么开发自定义 Membership 和 Role 的 Provider	338
5.9 扩展登录控件	347
5.9.1 可自定义的登录控件	347
5.9.2 登录错误更多提示信息 与写入日志	349
5.9.3 使用带有验证码的注册页面	353
5.10 ASP.NET 2.0 Provider 介绍	356
5.10.1 Provider 概述	356
5.10.2 Provider 模型	357
5.10.3 ProviderBase 类	358
5.10.4 Provider 的注册与配置	360
5.10.5 自定义 Provider 的注意事项	361
5.11 自定义 Provider 模型 之 Membership Providers	362
5.11.1 MembershipProvider 类	362
5.11.2 Membership 数据应用范围	363
5.11.3 ReadOnlyXmlMembership- Provider	364
5.12 自定义 Provider 之 Role Provider	369
5.12.1 RoleProvider 类	369
5.12.2 ReadOnlyXmlRoleProvider	369
5.13 登录控件与成员 API 函数	374
5.13.1 使用默认的向导控件功能	374
5.13.2 使用默认 ID	374
5.13.3 自定义 ID	374
第 6 章 站点导航系统	376
6.1 站点导航的实现	376
6.1.1 站点导航回顾	376
6.1.2 定义站点地图	377
6.1.3 使用导航控件显示导航	379

6.2 SiteMap 编程	381
6.2.1 SiteMap 类简介	381
6.2.2 SiteMap 的基本使用	381
6.3 基于角色的导航	383
6.3.1 配置 ASP.NET 2.0 的成员 和角色关系	383
6.3.2 配置导航系统使用安全裁减	384
6.3.3 使用基于角色的安全裁减	385
6.4 自定义站点地图 Provider	388
6.4.1 自定义 Provider	389
6.4.2 注册自定义的 Provider	393
6.5 扩展 Provider 提供模型	393
6.5.1 为什么需要扩展 Provider 模型	393
6.5.2 建立扩展的 Provider 类	394
6.5.3 配置 web.config	398
6.6 修改提供程序返回的站点导航数据	399
6.7 SitemapPath 控件	402
6.7.1 SitemapPath 控件概述	402
6.7.2 SitemapPath 的使用	403
6.7.3 SitemapPath 控件常见属性 的意义	404
6.8 Menu 控件	406
6.8.1 Menu 控件概述	406
6.8.2 Menu 控件的基本使用	407
6.8.3 使用 Menu 控件的外观	408
6.8.4 事件	413
6.9 TreeView 控件	413
6.9.1 TreeView 控件概述	413
6.9.2 TreeView 控件的基本使用	414
6.9.3 使用样式	415
6.9.4 利用数据库存放 TreeView 数据	431
6.10 MultiView 和 View Web 服务器 控件概述	433
6.11 Wizard 控件	436
6.11.1 Wizard 的组成	436
6.11.2 向导提交模型	438
6.11.3 向导控件的综合使用	447
6.12 自定义 Provider 之 SiteMap Provider	455
6.12.1 自定义 SiteMap Provider 要点	455
6.12.2 自定义 SqlSiteMapProvider	457

第 7 章 使用 Web 部件进行个性化设置	461
7.1 个性化设置	461
7.1.1 WebPart 控件	461
7.1.2 Web 部件控件集概述	462
7.2 WebPart 的使用	463
7.2.1 使用 WebPartManager 控件 管理 Web 部件	463
7.2.2 使用 WebPartZones 存放 Web 部件数据	463
7.2.3 使用用户控件	465
7.2.4 测试运行效果	465
7.2.5 更改模式	467
7.2.6 启用更改布局的功能	469
7.3 EditorZone 与 EditorPart 控件	470
7.3.1 AppearanceEditorPart 控件	472
7.3.2 LayoutEditorPart 控件	472
7.3.3 PropertyGridEditorPart 控件	472
7.3.4 BehaviorEditorPart 控件	476
7.3.5 自定义 EditorWebPart	478
7.3.6 自定义谓词和编辑器	479
7.4 Web 部件之间的通信	481
7.5 自定义 WebPart Provider 之 Web Parts Personalization Providers	486
7.5.1 自定义 Web Parts Personalization 概述	486
7.5.2 TextFilePersonalizationProvider	488
第 8 章 存储用户配置文档	491
8.1 用户配置文件概述	491
8.1.1 配置文件的必要性 以及 ASP.NET 2.0 的新特性	491
8.1.2 定义配置文件架构	491
8.1.3 匿名配置文件	492
8.1.4 经过身份验证的配置文件	493
8.1.5 将匿名配置文件迁移到经过身份验证 的配置文件	496
8.1.6 用自定义基类定义配置文件属性	498
8.1.7 控制配置文件的自定义保存行为	498
8.1.8 删除配置文件	500
8.2 进一步理解 Profile 的功能	501
8.2.1 一个简单 Profile 的定义和使用	501

8.2.2	Profile Provider 模型	502	9.3.4	状态管理	528
8.2.3	用户惟一标识 (Unidentified User)	503	9.3.5	控件和页适配器: 处理设备、浏览器 或标记的行为	528
8.2.4	Profile 的编译与执行	504	9.3.6	数据源和数据绑定控件	529
8.2.5	Profile 的配置	505	9.3.7	Web 资源	530
8.2.6	序列化	506	9.3.8	客户端脚本管理	531
8.2.7	Profile 的继承	508	9.3.9	设备筛选 (设备标记或浏览器)	531
8.2.8	Profile_MigrateAnonymous 事件	509	9.3.10	回调	531
8.2.9	ProfileManager 类	509	9.3.11	页状态	532
8.3	CustomerProvider 之 Profile Provider	510	9.3.12	服务器控件设计时	532
8.3.1	ProfileProvider 和 SettingsProvider 类	510	9.3.13	应用程序服务提供程序	533
8.3.2	Profile Data 数据应用	511	9.3.14	本地化	534
8.3.3	GetPropertyValues	511	9.3.15	编译和预编译	535
8.3.4	SetPropertyValues	513	9.3.16	生成提供程序	535
8.3.5	TextFileprofileProvider	513	9.3.17	虚拟路径提供程序	536
8.3.6	使用 TextFileProfileProvider	520	9.4	配置文件的使用与检索	537
第 9 章	ASP.NET 2.0 应用程序的应用与扩展	521	9.4.1	配置文件格式	537
9.1	页框架	521	9.4.2	配置节和节组	538
9.1.1	页框架生命周期	521	9.4.3	添加配置节	538
9.1.2	生命周期事件	522	9.4.4	集合设置	539
9.1.3	其他的页生命周期注意事项	522	9.4.5	使用位置和路径	539
9.2	ASP.NET 应用程序	523	9.4.6	锁定配置设置	540
9.2.1	用户从 Web 服务器请求 应用程序资源	523	9.4.7	标准 ASP.NET 配置节	540
9.2.2	ASP.NET 接收对应用程序的 第一个请求	523	9.4.8	检索配置	541
9.2.3	为每个请求创建 ASP.NET 核心对象	524	9.4.9	使用应用程序设置	541
9.2.4	将 HttpApplication 对象 分配给请求	524	9.4.10	使用连接字符串	542
9.2.5	由 HttpApplication 管理处理请求	524	9.5	使用管理工具	543
9.2.6	生命周期事件 和 Global.asax 文件	525	9.5.1	ASP.NET MMC 管理单元	543
9.2.7	编译生命周期	526	9.5.2	Web 应用程序管理工具	544
9.2.8	HTTP 模块	527	9.5.3	ASPNET_REGSQL 工具	545
9.3	服务器控件与状态	527	9.5.4	ASPNET_REGIIS 工具	545
9.3.1	服务器控件运行库	527	9.6	使用管理 API	545
9.3.2	控件类	527	9.6.1	检查配置设置	546
9.3.3	控件类和属性元数据	528	9.6.2	更新配置设置	548
			9.6.3	对配置节进行加密	549
			9.7	配置进程模型	551
			9.7.1	进程模型配置	551
			9.7.2	被动进程回收	552
			9.7.3	主动进程回收	552
			9.7.4	记录进程模型事件	552
			9.8	对缓存的改进	553

9.8.1 基于轮询的失效	553	10.5 主题与服务器端样式	592
9.8.2 SQL Server 2005 基于通知的 缓存失效	554	10.6 外观文件——Default.skin	593
9.9 Web 事件	554	10.7 简历页——Resume.aspx	594
9.9.1 事件和事件提供程序	555	10.8 链接收藏页——Links.aspx	594
9.9.2 配置事件	556	10.9 数据库设计	594
9.9.3 创建自定义事件	556	10.10 App_Code 目录	596
9.9.4 检测应用程序	557	10.11 使用 HttpHandler 处理照片	598
9.9.5 事件限流	557	10.12 管理相册	600
9.10 编码与本地化	557	10.13 用户注册	602
9.10.1 编码	557	10.14 扩展 Personal Web Site Starter Kit	602
9.10.2 本地化	558	10.14.1 扩展管理员页	602
9.11 URL 映射	558	10.14.2 添加 Manage.aspx 页	604
9.12 ASP.NET 的提示与技巧	560	10.14.3 添加 Users.aspx 页	605
9.12.1 跨页回发	560	10.14.4 建立 Content.aspx 页	607
9.12.2 验证组	561	10.15 Personal Web Site 总结	614
9.12.3 焦点 API 和 DefaultButton	562	第 11 章 Club Web Site Starter Kit	
9.12.4 编译生成提供程序	564	——俱乐部站点	615
9.12.5 不编译页	564	11.1 Club Web Site Starter Kit 的 安装和配置	615
9.12.6 确保非 ASP.NET 文件的安全	565	建立管理员账户	616
9.12.7 客户端脚本功能	566	11.2 母版页——default.master	617
9.13 自定义 Provider 之 Session Provider	569	11.3 注册页——member_register.aspx	619
9.13.1 自定义 Session Provider 概述	569	11.4 用户详细页 ——Member_Details.aspx	623
9.13.2 自定义 TextFileSessionStateProvider	571	11.5 查看用户列表页 ——member_list.aspx	626
9.14 自定义 WebEvent Provider	577	11.6 添加相册	629
9.14.1 自定义 WebEvent Provider 概述	577	11.7 照片列表页 ——PhotoAlbum_List.aspx	631
9.14.2 自定义 TextFileWebEventProvider	578	11.8 相册内容页 ——PhotoAlbum_Contents.aspx	637
第 10 章 Personal Web Site Starter Kit		11.9 日历事件页 ——Events_Calendar.aspx	641
——个人站点	581	11.10 事件的编辑、添加和删除	643
10.1 Personal Web Site 的安装	581	11.11 新闻模块	647
10.2 建立成员与角色	582	11.12 首页 default.aspx	648
10.2.1 角色的管理	582	11.13 链接和联系页	651
10.2.2 建立管理员账户	582	11.14 CWS 中主要技术的实现	651
10.3 相册管理	583	11.14.1 日期和时间控件	651
10.3.1 相册页——Albums.aspx	583	11.14.2 RolloverButtons.cs 文件	657
10.3.2 照片集页——Photos.aspx	586		
10.3.3 照片的详细页——Details.aspx	588		
10.4 母版页——Default.master	591		

11.14.3 SharedRoutines.cs 和 MemberDetails.cs 文件.....	658
11.15 数据库处理.....	659
11.15.1 数据库和数据适配器.....	659
11.15.2 Club Web Site 语言 Bug 修复.....	659
11.16 扩展 Club Web Site Starter Kit.....	663
11.16.1 使用 SQL Server 2000 数据库.....	663
11.16.2 增加文件系统.....	666
11.17 重构 CWS 的 Provider 模型.....	671
11.18 扩展 CWS 以便使用 IIS 远程工具.....	673
第 12 章 Classfieds Web Site Starter Kit ——产品交易系统.....	675
12.1 Classfieds Web Site Starter Kit 介绍.....	675
12.2 用户注册页——Register.aspx.....	681
12.3 登录页——Login.aspx.....	685
12.4 发布广告页——PostAd.aspx.....	686
12.5 查看发布的信息页 ——MyAds.aspx.....	692
12.6 进入管理员页.....	696
12.7 搜索页.....	700
12.8 数据处理要点.....	706
第 13 章 TimeTracker Starter Kit ——时间跟踪.....	711
13.1 简介.....	711
13.2 创建应用程序角色.....	713
13.3 创建用户.....	714
13.4 更改注册详细信息.....	715
13.5 母版页——MasterPage.master.....	716
13.6 创建项目——Project_Details.aspx.....	717
13.7 列出项目页——Project_List.aspx.....	719
13.8 输入时间页——TimeEntry.aspx.....	720
13.9 查看项目报告页 ——Report_Project.aspx.....	721
13.10 查看项目资源详细信息页 ——Report_Resources.aspx.....	721
13.11 Time Trackr 数据处理要点.....	722
附录 A 构建多项目解决方案.....	729
A.1 构建 Web 框架.....	729
A.2 添加 DAL.BLL 层.....	730
A.3 构建 Web 站点.....	733
A.4 部署与优化.....	735
附录 B Web Site 与 Web Project.....	736
B.1 Web Project.....	736
B.2 Web Site 的局限性.....	737

第 1 章 ASP.NET 2.0 安装、使用与新特性说明

2005 年 11 月 7 日, 微软正式推出了 ASP.NET 2.0, 和 ASP.NET 1.X 版本相比, 新版的 ASP.NET 2.0 在 Web 开发方面提供了更大的技术支持和更友好的开发环境, 使得基于浏览器的 Web 开发更加简单与易用。本章主要介绍 ASP.NET 2.0 的安装和新增加的主要特性, 另外还介绍了相关的 Visual Studio.NET 2005、SQL Express 2005 软件的安装和使用, 以及 ASP.NET 2.0 新增加的语法等。

1.1 ASP.NET 2.0 的安装与基本使用

本节介绍了 .NET Framework 2.0 和相关软件的安装与使用, 以帮助读者快速配置并使用 ASP.NET 2.0 来开发自己的 Web 页面。

1.1.1 下载、安装 .NET Framework 2.0

1. 免费下载 ASP.NET Framework 2.0

ASP.NET 2.0 是一个免费的软件包, 读者可以到 <http://www.asp.net> 下载, 图 1-1 显示的是 ASP.NET 站点提供的下载链接图。

注意: 由于站点的变更性, 你可能看到实际访问时的位置略有不同。

在该图里单击 Download the .NET Framework 2.0 链接, 页面自动转入 <http://msdn.microsoft.com/netframework/downloads/updates/default.aspx> 页面让用户进行下载, 参考图 1-2。

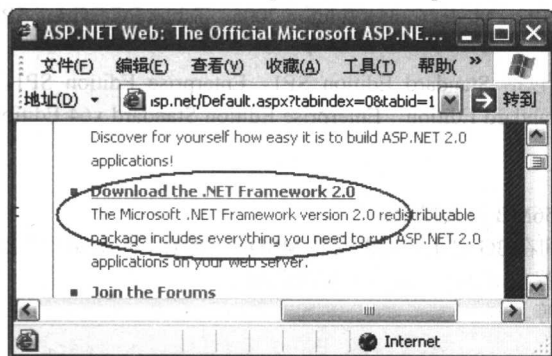


图 1-1 <http://www.asp.net> 页面

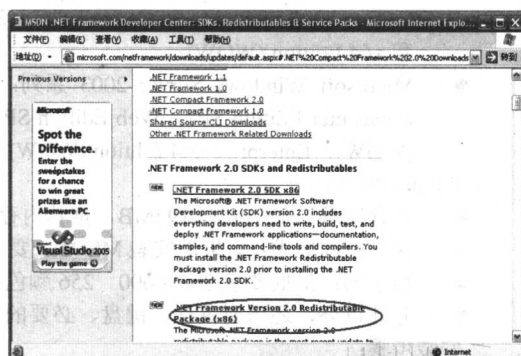


图 1-2 .NET 框架下载页面

在图 1-2 里, 你可能看到许多 .NET Framework 2.0 的软件包和文档。这里需要区分两个概念: 一个是 .NET 框架发行包 (.NET Framework Version 2.0 Redistributable Package (x86)), 一个是开发环境 (.NET Framework 2.0 SDK x86)。

所谓 .NET 框架发行包 (Redistributable Package) 就是你开发 ASP.NET 所必须安装的软件包, 如果没有安装该软件包, 你将无法开发 ASP.NET。

另外一个就是软件开发包 (SDK, Software Development Kit), 软件开发包通常是為了便于软件开发, 由一些公司开发的集成的开发环境, 例如 PowerBuilder、Visual Studio.NET 2003 等。在这种软件开发包里面为了便于用户的使用, 除了包含必要的核心软件包外, 它还提供诸如帮助文档、示例代码、调试工具、编译器、必要的接口等内容。所以从这种意义上说软件开发包通常包含发行包。

不过在微软提供的开发包里并没有包含发行包, 这可能是软件包重复包含 .NET 发行包会造成内容过多而使得通过网络下载时间长, 而用户在使用时可能并不需要软件包里的内容的缘故。所以在微软的下载页面里, 已经将这两个包进行分开提供。你在实际使用时应该先下载 .NET Framework 发行包并安装, 然后再根据个人的需要决定是否下载安装和使用 .NET 开发包。

2. .NET Framework 2.0 安装需求

ASP.NET 2.0 Framework 支持的操作系统有: Windows 2000 Service Pack 3, Windows 98, Windows 98 Second Edition, Windows ME, Windows Server 2003, Windows XP Service Pack 2。

但是在安装.NET 2.0 前, 这些操作系统必须安装如下软件:

- Windows Installer 3.0, Windows Installer 3.1 或者其后的版本 (对于 Windows 98/ME, 需要安装 Windows Installer 2.0 或其后的版本)。
- IE 5.01 或其后的版本。

提示: Windows Installer 和 IE 5.0.1 等软件可以免费到微软官方网站下载来进行安装。

另外, 磁盘空间需求: 280 MB。

注意: 如果你以前安装过 ASP.NET Framework 2.0 的测试版, 例如 beta1, beta2 或者 Community Technical Preview(CTP), 在使用正式版以前, 必须先卸载原来的测试版。

1.1.2 Visual Studio.NET 2005 的安装

Visual Studio.NET 2005 (简称 VS.NET 2005, 下同) 是基于 .NET Framework 2.0 框架的新的软件开发环境。在软件开发方面, VS.NET 2005 提供了更好的开发环境支持。下面介绍安装该软件的软硬件需求。

Visual Studio.NET 2005 已经包括了 Windows Installer3.X, .NET Framework 1.1/2.0 和下一节要介绍和安装的 SQL Server 2005。这里之所以单独提出来是因为如果没有 Visual Studio.NET 2005, 可以单独安装这些组件。

安装 VS.NET 2005 的软件需求如下:

支持的操作系统:

- Microsoft Windows 2000 系列: 版本包括 Professional SP4, Server SP4, Advanced Server SP4;
- Microsoft Windows XP Professional 系列: 版本包括 Professional SP2, Home Edition SP2, Media Center Edition 2002 SP2, Media Center Edition 2004 SP2, Media Center Edition 2005, Tablet PC Edition SP2 和 x64 Edition (WOW);
- Microsoft Windows Server 2003 系列: 版本包括 Standard Edition SP1, Enterprise Edition SP1, Datacenter Edition SP1, Web Edition SP1, Standard Edition, Enterprise Edition Standard x64 Edition (WOW), Enterprise x64 Edition (WOW)等。

硬件基本要求:

- 内存要求: 最小内存 192MB, 推荐内存存在 256MB 及其以上;
- 磁盘空间要求: 如果不安装 MSDN, 要求空间在 3G 以上, 如果安装 MSDN, 要求空间在 4G 以上;
- 显示器: 最低要求 800×600 256 颜色; 推荐在 1024×768 16 位颜色以上;
- 除此以外, 还需要鼠标、键盘、必要的光驱等。

安装步骤:

当把安装盘放到光驱后, 一般会自动运行, 如果你禁用了自动运行, 也可以双击 autorun.exe 来进行手动运行, 此时会出现 Visual Studio.NET 2005 安装对话框, 如图 1-3 所示。

这里的安装包括三个内容:

- Install Visual Studio 2005: 这是必须的, 用于安装 Visual Studio 2005;
- Install Product Document: 这是可选的, 用于安装 MSDN;
- Check for Service Release: 这是可选的, 用于检测最新版本软件发行包。

这里选择 Install Visual Studio 2005, 系统将打开 Visual Studio.NET 2005 安装向导, 单击下一步, 出现如图 1-4 所示的系统加载安装程序向导。

接下来选中 I accept the terms of the License Agreement 复选框接受协议, 并输入你的产品序列号, 如图 1-5 所示。

在下一步里, 选择安装类型, 这里我们选择默认的 default 安装类型, 如图 1-6 所示, 它将安装 C++、C#、VB、J#语言版本的默认功能组件。这里的组件主要包括如下几个部分。

- Microsoft Windows Installer 3.1
- Microsoft .NET Framework 2.0
- MSXML 6.0 Parser
- Microsoft Document Explorer 2005

- Microsoft Visual Studio 2005
- .NET Compact Framework 1.0 SP3
- .NET Compact Framework 2.0
- Microsoft Visual J# 2.0 Redistributable Package
- SQL Server 2005 Mobile Edition
- Microsoft Device Emulator version 1.0



图 1-3 安装 Visual Studio.NET 2005 对话框

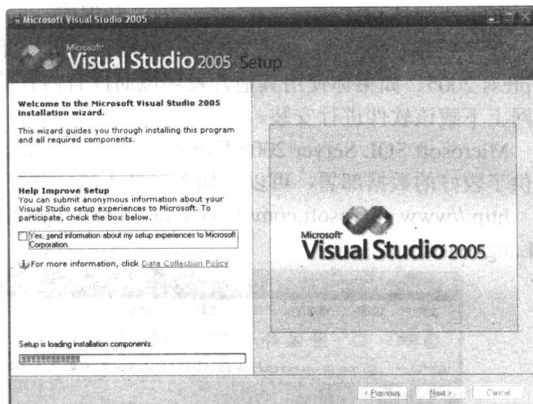


图 1-4 Visual Studio.NET 2005 安装向导

单击下一步系统开始安装，直到完成，如图 1-7 所示。

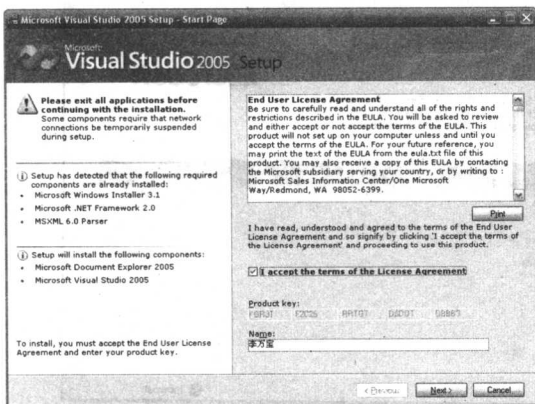


图 1-5 接收协议并输入产品序列号

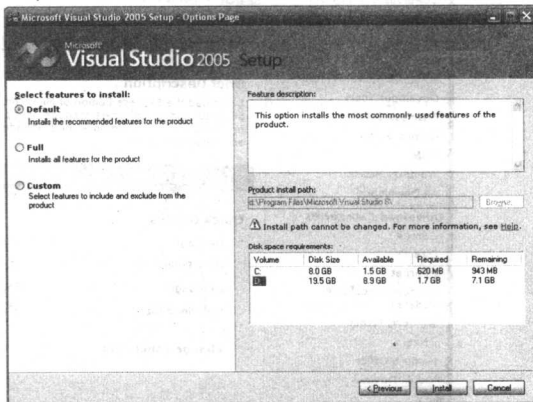


图 1-6 选择安装类型

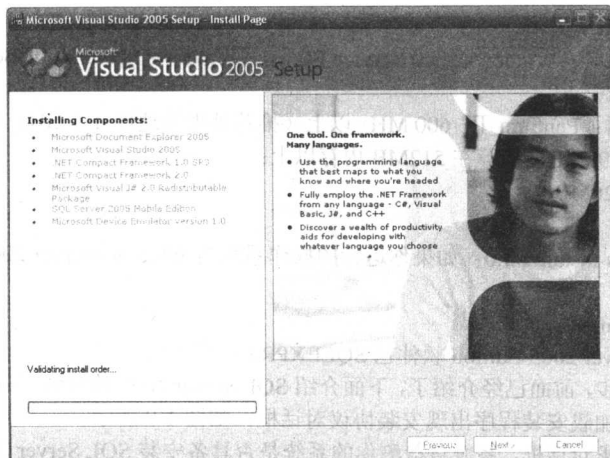


图 1-7 Visual Studio.NET 2005 安装默认组件