

- 
1. 本书所有范例原始程序代码
 2. ASP.NET 2.0论坛与教学课程视频文件
(ASP.NET 2.0新功能操作步骤与应用技巧等实务范例)
 3. 附赠AJAX版TextBox超强控件及原始程序代码

ASP.NET 与 AJAX

董大伟 编著
微软最有价值专家



深度剖析范例集

帮助您迅速熟悉 **ASP.NET 2.0** 的新功能与核心理念, 彻底掌握实务应用技巧

- 带您一同探索ASP.NET 2.0令人叫绝的新功能, 让您轻松遨游在ASP.NET 2.0的新技术中
- 深入探讨ASP.NET 2.0中的**Call Back**技术, 以众多的范例带您进入**AJAX**网页设计的领域
- 提供大量的专业范例: 电子商务、在线付款、购物车、博客(BLOG)、电子相册、Web统计图表、EIP、讨论区、个人通讯录(WebPart)、AJAX版WebMail、划位订票系统、即时在线会议、即时筛选后端数据库的ComboBox、密码强度检查机制等

微软公司
开发工具产品销售经理
胡德民
推荐

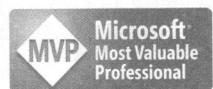


中国青年出版社
中国青年电子出版社

<http://www.21books.com> <http://www.cgchina.com>

ASP.NET 与 AJAX

董大伟 编著
微软最有价值专家



深度剖析范例集



中国青年出版社
中国青年电子出版社

<http://www.21books.com> <http://www.cgchina.com>

律师声明

北京市邦信阳律师事务所谢青律师代表中国青年出版社郑重声明：未经版权所有人和中国青年出版社书面许可，任何组织机构、个人不得以任何形式擅自复制、改编或传播本书全部或部分内容。凡有侵权行为，必须承担法律责任。中国青年出版社将配合版权执法机关大力打击盗印、盗版等任何形式的侵权行为。敬请广大读者协助举报，对经查实的侵权案件给予举报人重奖。

本书中文繁体版本版权由台湾博硕文化股份有限公司 (DrMaster Press Co.,Ltd) 获作者董大伟授权独家出版发行，中文简体版本版权由博硕文化股份有限公司 (DrMaster Press Co.,Ltd) 获作者同意授权中国青年出版社独家出版，中青新世纪图书发行中心独家发行。

侵权举报电话：

全国“扫黄打非”工作小组办公室

010-65233456 65212870

<http://www.shdf.gov.cn>

中国青年出版社

010-64069359 84015588转8002

E-mail: law@21books.com MSN: chen_wenshi@hotmail.com

版权登记号：01-2006-6007

图书在版编目 (CIP) 数据

ASP.NET与AJAX深度剖析范例集 / 董大伟编著. — 北京：中国青年出版社，2007.8

ISBN 978-7-5006-7708-6

Ⅰ.A... Ⅱ.董... Ⅲ.主页制作—程序设计 Ⅳ.TP393.092

中国版本图书馆CIP数据核字 (2007) 第 106645号

ASP.NET与AJAX深度剖析范例集

董大伟 编著

出版发行： 中国青年出版社

地 址：北京市东四十二条21号

邮政编码：100708

电 话：(010) 84015588

传 真：(010) 64053266

责任编辑：肖 辉 张海玲

封面设计：钟 夏

印 刷：北京新丰印刷厂

开 本：787×1092 1/16

印 张：38

版 次：2007年8月北京第1版

印 次：2007年8月第1次印刷

书 号：ISBN 978-7-5006-7708-6

定 价：69.90元 (附赠1CD)

本书如有印装质量问题，请与本社联系 电话：(010) 84015588

读者来信：reader@21books.com

如有其他问题请访问我们的网站：www.21books.com

前言

“惊喜”是笔者看到ASP.NET 2.0时的第一感觉！为什么从ASP.NET 1.1到ASP.NET 2.0会有这么多的改变和革新呢？如果您参加过ASP.NET 2.0微软研讨会，发现很多新的控件，很多新的功能，很多新技术让您不用再写代码了，相信您一定会立刻欣喜万分，在脑海里对未来充满了无限期盼，那笔者要恭喜您，但是只恭喜您一半！

为何只恭喜一半？因为ASP.NET 2.0的确是值得您从ASP.NET 1.1转换升级的好产品，但是原因绝对不只是新增的“控件”而已。在ASP.NET 2.0的核心当中，有很多的重要功能和机制完全改观，以Ajax为概念的CallBack机制，会让您在设计一个网页时，脑袋里的思考逻辑与以前完全不同……让您可以从Client端完全不换页（PostBack）就可以访问和更新后台的数据库内容，或获取后台数据动态显示到页面上！

写作契机与本书特色

笔者脑袋中第一个浮现的问题是，我们要怎么写一本书，才能让读者对这些新功能融会贯通？因为有太多的新东西了，这些新东西要怎么应用到您的程序里才是最贴切的？如果我们写一本书，只是将新功能介绍从头写到尾，是不是就真的能符合您的需要了？

举一个例子，“CallBack机制”，这个笔者认为ASP.NET 2.0最重要的新功能，它又是一个新的概念，要如何才能跟读者介绍清楚？对笔者来说是一个很大的挑战，若元素太多，意义太少，只会让读者见树不见林，岂不万分可惜？

笔者基本确定，根本不可能在一本书里面把ASP.NET 2.0从头到尾介绍完，因此决定以实际范例的角度来介绍，希望带给广大读者ASP.NET 2.0新的体验。

本书主要内容

本书共有29章，从ASP.NET 2.0的新功能与新改变开始，陆续讨论到高级应用的范例。大体上可以分为两个部分。

第1~6章：围绕ASP.NET 2.0带来的新功能及其应用，展开详细的讲解。包括：构建网站、Visual Studio 2005的重要改变、ASP.NET 2.0新功能剖析、数据访问机制的重大变革、深入探讨DataSource及其应用、ASP.NET 2.0的登录与身份验证机制。

第7~29章：本书的第二部分全部都是运用ASP.NET 2.0技术来完成的中大型项目，从数据库处理、页面上的控件、JavaScript的互动到CallBack机制等，这些项目完全利用ASP.NET 2.0新技术来开发。因此即使您以前曾经开发过类似项目，也建议您重新阅读一次，看看利用ASP.NET 2.0来开发时，有什么差别，以及在代码和控件使用上的改变。

建议有经验的ASP.NET 1.1开发人员，在阅读本书时，依照章节的顺序从头开始读起。您可以在本书前半部看到VS 2005以及ASP.NET 2.0所有重要的新功能以及相关机制和背景技术。千万不要错过DataSource和所有与数据绑定有关的章节，它将会大幅改变您对数据库应用程序开发的编写方式和思考逻辑，而这样的改变将会减少相当多的代码，并且提高开发与维护效率。

另外笔者要特别推荐，CallBack机制介绍的部分，也是有经验的ASP.NET 1.1开发人员千万不可错过的章节，在这个章节之中，您会看到如何利用ASP.NET 2.0来构建不需要PostBack就可以访问后台数据库的技术，可以看到Ajax机制在ASP.NET 2.0中的集成。而这一部分，是未来Web应用程序必然发展的方向。

本书使用的数据库

过去笔者的书籍一向使用SQL Server作为数据源，但在本书中，将同时使用Access和SQL Server 2005 Express为主要的范例数据库。笔者多次收到读者来信表示，在实际要架设网站的环境中，无法创建SQL Server连接（可能由于ISP不支持或软件版权与预算问题）。也有读者表示，部署到正式环境后，因为数据库连接问题而大费周折（很多的ISP在SQL Server上的管理接口不是很理想），当我们看到这些问题，进而联想到ASP.NET 2.0的设计目标之一，又是希望能够让开发人员可以减少代码及减少处理和主要目的不相关的工作（例如数据库的创建），如果因为数据库的问题增加您在学习上的困扰，实在得不偿失，因此本书在范例的部分，同时采用Access与SQL Server 2005 Express作为数据库连接方式，在ASP.NET 2.0数据库处理的章节，也同时介绍AccessDataSource和SqlDataSource，希望提供给读者最方便的学习环境。

此外，即使您过去习惯使用Access作为主要的数据库，到了ASP.NET 2.0，笔者要特别提醒您不要忽略了SQL Server 2005 Express，在安装Visual Web Developer（VWD）时，有一个精简版本的SQL Server Express，您会发现它非常好用。

您可以在VWD中创建一个SQL Server数据库（.mdf），然后像使用Access一样地使用它，本书会在讨论DataSource的章节一并介绍，它可以用来取代您过去以Access作为数据库的原因，因为不需要额外的部署或处理连接问题，只需要和项目一起复制到Server上即可（当然，Server端需要安装SQL Server 2005或Express）。

您可能又会想到，Membership和Role以及Login控件等ASP.NET组件，内置是采用SQL Server Express，那该怎么办呢？别担心，在本书介绍ASP.NET 2.0登录机制的章节中，将会教您如何将现有的ASP.NET 2.0登录控件与您自己的数据库连接（无论是Access还是SQL Server Express），可以解决非常多的问题。

因此，您采用本书中的范例，只需要直接Copy到安装有.NET Framework 2.0的网站上，几乎不需要修改，就可以正确执行，希望对你的学习有所帮助。

顺便说一句，如果您采用Access数据库别忘了要安装MDAC 2.6……

翻开本书，让本书带着您深入ASP.NET 2.0，当您对于ASP.NET有更深的体会和认识时，就会知道，ASP.NET 2.0带给您我的，是何等的“惊喜”！

作者

2007年6月

第 1 章 构建自己的网站

1-1 Web 应用程序的革新和演变	2
1-1-1 Web 应用程序的普及	2
1-1-2 Web 应用程序主要存在的问题	2
1-2 ASP.NET 2.0 的应用	3
1-2-1 ASP 与 ASP.NET 2.0 之间的选择	3
1-2-2 从 ASP.NET 1.1 到 ASP.NET 2.0 的重大演变	3
1-2-3 ASP.NET 2.0 与其他解决方案的比较	3
1-2-4 选择 ASP.NET 2.0 的理由	4
1-3 ASP.NET 2.0 的重要变革	5

第 2 章 Visual Studio 2005 的重要改变

2-1 Visual Studio 2005	8
2-2 版本区分	9
2-3 安装与设置	9
2-3-1 安装 Visual Studio 2005 Professional	9
2-3-2 Visual Web Developer (VWD)	11
2-4 创建网页与项目	11
2-4-1 创建第一个 Web 项目	12
2-4-2 项目的保存类型	12
2-5 VS 2005 的新功能与改变	13
2-5-1 拖动或单击绘制一个控件	13
2-5-2 编写控件事件	14
2-5-3 虚拟 IIS 服务器	16
2-5-4 支持 MsgBox	17
2-5-5 SmartTag	17
2-5-6 IntelliSense	18
2-5-7 解决方案资源管理器的改变	19
2-5-8 插入表	20
2-5-9 插入层	21
2-5-10 设置绝对位置	22
2-5-11 选取 HTML 标记	22
2-6 代码段的使用与创建	23
2-6-1 使用代码段	23
2-6-2 创建自己的代码段	24
2-6-3 使用导入的代码段	25
2-7 测试与执行项目	26

2-7-1 Web.Config	26
2-7-2 调试与逐语句执行	26
2-7-3 异常支持——错误处理建议方案	27
2-7-4 Edit & Continue	27
2-7-5 运用 MsgBox 和 InputBox	28
2-8 网站的部署工具	28
2-8-1 如何部署自己的网站	28
2-8-2 数据库的部署问题	30
2-8-3 部署后的权限设置	30
2-8-4 编译模式的影响	31
2-9 网站的编译模式	31
2-9-1 ASP.NET 的编译模式	31
2-9-2 如何预编译网站	31
2-9-3 针对网站部署预编译	32
2-9-4 部署预编译后的文件	33
2-10 IIS 虚拟目录管理属性页面	34
2-10-1 “常规”选项卡	34
2-10-2 “自定义错误”选项卡	35
2-10-3 “授权”选项卡	35
2-10-4 “身份验证”选项卡	36
2-10-5 “应用程序”选项卡	36
2-11 同时使用不同的语言开发	37
2-12 创建与使用 SQL Server 2005 Express 数据库	37
2-12-1 如何创建数据库	37
2-12-2 如何创建表	38
2-12-3 如何使用数据库	40
2-12-4 关于 ASPNETDB.MDF	41
2-12-5 使用 SQL Server 2005 Express 的优缺点	41

第 3 章 ASP.NET 2.0 新功能剖析

3-1 新增的 Callback 机制与 Ajax 概念	44
3-1-1 Callback 机制的定义	44
3-1-2 不需 PostBack 获取 Server 端时间	45
3-1-3 范例——不换页的 Server 端电子时钟	52
3-1-4 范例——如何不换页直接更新客户端网页内容	54
3-1-5 Callback 机制运行流程与应用	57
3-1-6 更多范例	58
3-2 关于 PostBack 机制的改变	58

3-2-1 利用 PostBackURL 属性 Submit 到其他网页	59	3-15 Request 命令的使用	90
3-2-2 PreviousPage 属性	59	第 4 章 数据库访问机制的重大变革	
3-2-3 IsCrossPagePostBack 属性	59	4-1 DataSource 的出现	94
3-2-4 小结	60	4-1-1 DataSource 的基础范例	94
3-3 ASP.NET 1.1 Web 控件的改变	61	4-1-2 DataSource 与 DataControls 之间的框架	97
3-3-1 TextBox 控件	61	4-1-3 条件互动式 DataSource 范例——零代码供应商数据浏览	98
3-3-2 Tabel 控件	61	4-1-4 数据是否允许编辑	101
3-3-3 OnClientClick 属性	62	4-1-5 编辑 FormView 模板并添加按钮	102
3-3-4 PostBackURL 属性	62	4-1-6 范例——多层次选项列表	104
3-3-5 Label 控件	63	4-1-7 DataSource 程序设计架构小结	108
3-3-6 Panel 控件	64	4-2 GridView	108
3-3-7 支持 DataSource 的控件	64	4-2-1 取代 DataGrid 的 GridView	109
3-4 ASP.NET 2.0 中 WebForm 的变革	65	4-2-2 按条件浏览列表	111
3-4-1 利用 SetFocus 控制焦点	65	4-2-3 编辑、新建、删除按钮的图形化设置	112
3-4-2 Layout 模式的改变	66	4-2-4 GridView 字段设置	113
3-4-3 利用“插入层”命令创建 Layer	66	4-2-5 CallBack 功能介绍	114
3-4-4 自动定位选项	66	4-2-6 以转换成 TemplateField 的形式添加 DropDownList	115
3-5 Code-Behind 技术的改变	67	4-2-7 删除前的确认信息	118
3-6 Generic Handler 机制	71	4-2-8 添加 HyperLink 与其他网页互动	120
3-6-1 Generic Handler 机制的原理	71	4-2-9 获取选择的项目	124
3-6-2 创建 Generic Handler (.ashx) 程序	71	4-2-10 GridView 的重要事件	124
3-6-3 利用 .ashx 下载文件	73	4-3 范例——制作简单的聊天室	125
3-6-4 利用 .ashx 创建动态图形生成器	74	4-4 FormView	129
3-6-5 动态图形生成的原理	74	4-4-1 FormView 的特性	129
3-6-6 动态图形生成的实例操作	75	4-4-2 FormView 中的模板	129
3-6-7 运行流程说明	77	4-4-3 FormView 中的控件	130
3-6-8 drawInit 函数	78	4-4-4 控件的数据绑定方式	131
3-6-9 ImageServices 的应用	78	4-4-5 双向绑定 (Bind) 与单向绑定 (Eval)	132
3-7 与 JavaScript 的集成	79	4-4-6 数据绑定语法	133
3-8 项目框架与特殊目录	79	4-4-7 设计版面	133
3-9 Resource 文件的使用	81	4-4-8 FormView 中的控制按钮	134
3-10 使用新增的 My Class	82	4-4-9 FormView 的重要事件	135
3-10-1 My Class 在 WebForm 中的支持	82	4-4-10 利用 FormView 设计窗体时的 Bug	136
3-10-2 使用与创建自己的 My Class	82	4-5 DetailsView	140
3-11 Partial Type	85		
3-12 DataSource 与数据绑定控件	86		
3-13 ASP.NET 2.0 的强大功能	86		
3-14 WebServices 的改变与异步机制	87		

4-5-1 FieldHeaderStyle 属性	140
4-5-2 显示图形字段	141
4-6 关于控件的模板	143
4-7 控件的 TemplateField 与数据 绑定	144
4-7-1 FormView 中的字段数据绑定	145
4-7-2 将字段转换为 TemplateField	147
4-7-3 添加新列	149
4-8 数据控件的 Rows、Cells 应用	151
4-9 通过 ItemUpdating 事件设置字段的默认值	153
4-10 范例——零代码 Master-Detail	154
4-10-1 框架与页面	154
4-10-2 AccessDataSource	154
4-10-3 预览	156
4-10-4 页面调整	156
4-10-5 自动编号问题	157
4-11 范例——数据库高级功能	158
4-11-1 弹出式字段选择对话框	158
4-11-2 FormView 中的控件	160
4-11-3 DataSource 的应用	163
4-11-4 删除数据前的 alert() 提示对话框	166
4-11-5 Detail 的新建页面	166
第 5 章 深入探讨 DataSource 的应用	
5-1 DataSource 控件的本质	170
5-1-1 DataSource 控件与页面 (.aspx) 之间的关系	170
5-1-2 以代码动态创建并使用 DataSource 来访问数据库	172
5-1-3 ADO.NET 与 DataSource	173
5-1-4 AccessDataSource 与 SqlDataSource	173
5-2 SelectCommand	174
5-2-1 AccessDataSource 的 SelectCommand	174
5-2-2 SelectCommand 的基本用法	175
5-2-3 在 SelectCommand 中使用参数	175
5-2-4 使用 QueryStringParameter	177
5-2-5 使用 ControlParameter	178
5-2-6 SqlDataSource 参数字段的 不同	179
5-3 DataSource 与页面 .aspx 之间的关系	180
5-4 DataSource 的高级 Command	184
5-4-1 利用 UpdateCommand 修改数据	184
5-4-2 利用 InsertCommand 插入新数据	185
5-4-3 利用 DeleteCommand 删除数据	186
5-5 DataSource 的各种事件	186
5-5-1 声明具有事件的对象	186
5-5-2 DataSource 事件列举	187
5-6 DataSource 的重要属性	187
5-7 DataSource 的同步问题	188
5-8 DataSource 的设置向导	188
5-9 SelectCommand 的属性生成器	189
5-10 如何在 Web.Config 中设置数据库位置或连接	191
5-10-1 将数据库连接保存在 Web.Config 中	191
5-10-2 如何动态调整数据库位置	192
5-11 2-tier 或 3-tier 数据访问	193
5-12 App_Data 文件夹	194
5-13 创建 SysDB 类——继承 SqlDataSource	194
5-14 SQL Express 数据库	195
5-14-1 创建 SQL Express 数据库	195
5-14-2 数据库连接字符串	196
第 6 章 ASP.NET 2.0 登录与身份验证机制	
6-1 ASP.NET 2.0 登录机制的基本概念	198
6-1-1 用户身份验证方式	198
6-1-2 ASP.NET 2.0 的登录控件	199
6-1-3 ASP.NET 2.0 内置的账号管理机制	199
6-1-4 后台管理界面	200
6-2 登录类型控件的使用	201
6-2-1 第一个登录机制	201
6-2-2 登录控件的身份验证模式	202

6-2-3 登录控件的自动导入页面属性	203
6-2-4 LoginView 控件	203
6-2-5 LoginStatus 控件	204
6-2-6 LoginName 控件	206
6-3 管理类型控件的使用	206
6-3-1 CreateUserWizard 控件	206
6-3-2 ChangePassword 控件	208
6-3-3 PasswordRecovery 控件	209
6-4 会员角色与网页访问规则	209
6-4-1 创建角色	209
6-4-2 创建访问规则	210
6-5 登录机制的程序化处理	211
6-5-1 环境与设置	212
6-5-2 Login 控件	212
6-5-3 不使用 Login 控件的表单 身份验证	213
6-5-4 FormsAuthentication 类	213
6-6 Membership 类	214
6-6-1 通过 ValidateUser 验证用户 账号密码	215
6-6-2 通过 CreateUser 创建用户	215
6-6-3 删除用户	216
6-6-4 查找用户	217
6-6-5 用户数据获取	217
6-7 Roles 类	217
6-7-1 创建一个角色	218
6-7-2 Roles 的功能	218
6-7-3 AddUserToRole	219
6-7-4 RemoveUserFromRole	219
6-7-5 GetUsersInRole	219
6-7-6 GetRolesForUser	219
6-7-7 IsUserInRole	219
6-7-8 在网页中的运用	219

第 7 章 ProFile 范例——制作用户 配置文件

7-1 概念	222
7-1-1 ProFile 的用途	222
7-1-2 ProFile 的存放位置	222
7-2 ProFile 的使用	222
7-2-1 在 Web.Config 中配置	223
7-2-2 在程序中使用	223
7-2-3 匿名用户	224

7-2-4 参数类型	225
7-3 ProFile 配置文件的程序化控制	225
7-3-1 获取变动状态	225
7-3-2 判断用户	225
7-3-3 参数列表	225
7-4 本章小结	226

第 8 章 MasterPage 范例——使用 母版页

8-1 MasterPage 的基本概念	228
8-2 创建 MasterPage	229
8-3 创建内容页面 (ContentPage)	231
8-4 ContentPlaceHolder 控件与 Content 控件	233
8-5 通过程序控制 MasterPage 上的 对象——Master 类	237
8-6 通过程序动态设置 MasterPage	239
8-6-1 MasterPageFile 属性	239
8-6-2 通过 Session 在 PreInit 中设置 MasterPageFile 属性	240

第 9 章 Theme 与 Skin 范例

9-1 Theme 与 Skin	244
9-1-1 什么是“面板 (Skin)”	244
9-1-2 什么是“主题 (Theme)”	244
9-1-3 如何使用主题	245
9-1-4 利用 SkinID 区分多组 Skin	246
9-1-5 在主题中加入图形	247
9-1-6 主题的有效范围	248
9-2 利用代码控制 Theme	249
9-2-1 动态决定页面的主题与控件的 面板	249
9-2-2 按照用户或权限决定不同的 主题	249
9-2-3 按照字段不同的功能使用不同 的面板	251

第 10 章 WebParts 范例

10-1 如何创建 WebParts	254
10-1-1 WebParts 的原理	256
10-1-2 创建 WebPartZone	256

10-1-3 在 WebPartZone 中设置 WebPart.....	257
10-1-4 WebPartZone 的重要属性.....	260
10-2 如何动态拖动调整 WebPart.....	261
10-2-1 WebPartManager 的 Display Mode.....	261
10-2-2 CatalogZone.....	263
10-2-3 在 CatalogZone 中放置 PageCatalog.....	264
10-2-4 在 CatalogZone 中放置 DeclareCatalog.....	265
10-2-5 WebPart 的导出与导入.....	267
10-3 WebPart 的动态编辑.....	268
10-3-1 EditorZone.....	268
10-3-2 AppearanceEditorPart.....	269
10-3-3 LayoutEditorPart.....	269
10-4 WebPart 的个性化与 DisplayMode 设置.....	269
10-4-1 进入特定模式前的检查.....	270
10-4-2 声明性目录的使用.....	271
10-4-3 禁用个性化机制.....	271
10-4-4 重设 WebPart 个性化设置.....	271
10-4-5 利用 PersonalizableAttribute 创建个性化属性字段.....	272
10-4-6 个性化机制的存放位置.....	273
10-5 代码控制.....	273
10-5-1 控制 WebPart 的行为与特征.....	273
10-5-2 动态添加、删除与改变 WebPart 位置.....	274
10-5-3 WebPartManager 的重要事件与属性.....	275
10-5-4 关闭 WebPart 时的信息.....	276
10-5-5 WebPart 的事件触发顺序.....	276
10-5-6 WebPart 之间的数据交换.....	277
10-5-7 在 WebPart 中控制 WebPartManager.....	278

第 11 章 MultiView 范例

11-1 MultiView 与 View 控件.....	282
11-1-1 MultiView 的概念.....	282
11-1-2 MultiView 的用法.....	282
11-1-3 使用说明.....	283
11-2 MultiView 与 View 控件的内部机制.....	283

11-2-1 通过 ActiveViewIndex 在不同的 View 之间切换.....	283
11-2-2 与 Button 控件的配合.....	284
11-2-3 CommandName 与 CommandArgument.....	284
11-3 View 中的控件.....	285
11-4 实现选项卡功能.....	286
11-5 高级选项卡.....	287
11-5-1 添加滚动条和边框.....	287
11-5-2 与 MasterPage 配合.....	288

第 12 章 Wizard 范例

12-1 使用 Wizard 控件.....	292
12-1-1 Wizard 页面设计.....	292
12-1-2 添加 / 移除步骤页面.....	293
12-1-3 页面的 HTML 结构.....	294
12-1-4 关于页面的设置.....	295
12-2 Wizard 的重要事件.....	295
12-2-1 FinishButtonClick.....	295
12-2-2 NextButtonClick 与 PreviousButtonClick.....	296
12-2-3 SideBarButtonClick.....	296
12-3 Wizard 的重要属性.....	296
12-3-1 ActiveStepIndex.....	296
12-3-2 DisplaySideBar.....	296
12-3-3 SideBarStyle.....	296
12-3-4 StepStyle.....	297
12-3-5 HeaderText.....	297
12-4 代码控制.....	298
12-4-1 关于 PostBack 的影响.....	298
12-4-2 利用代码切换页面与控制流程.....	298

第 13 章 导航控件与 ImageMap 控件

13-1 ASP.NET 2.0 中的导航控件.....	302
13-1-1 站点地图 (Web.SiteMap).....	302
13-1-2 SiteMapDataSource.....	304
13-2 SiteMapPath 控件.....	304
13-2-1 创建与使用 SiteMapPath 控件.....	304
13-2-2 导航控件与 SiteMapDataSource 之间的关系.....	306

13-2-3 SiteMap 与 Menu 的配合	306	15-3-1 数据源	345
13-3 Menu 控件	308	15-3-2 过滤条件	346
13-3-1 创建 Menu 控件	308	15-3-3 DataList 版型	348
13-3-2 在代码中创建 Menu 控件	310	15-3-4 如何绑定数据库中的 图形文件	349
13-4 TreeView 控件	310	15-4 后台管理站点	351
13-4-1 创建与使用 TreeView 控件	310	15-4-1 删除产品	352
13-4-2 TreeView 控件应用范例	311	15-4-2 修改产品	354
13-5 ImageMap 控件	312	15-4-3 插入产品	357
13-5-1 HotSpots 属性	313	15-4-4 总结与提高	358
13-5-2 HotSpotMode	315		
13-5-3 范例——制作世界地图	315		
第 14 章 WebPart 设计范例——使用 UserControl		第 16 章 高级范例——会员与订单 处理机制	
14-1 制作 UserControl	320	16-1 结构说明	362
14-1-1 创建 UserControl	321	16-1-1 网站文件结构	362
14-1-2 数据结构	322	16-1-2 相关数据结构	362
14-1-3 设计 UserControl 的功能	323	16-2 网页设计	363
14-1-4 将 GridView 的字段转换为 TemplateField	325	16-2-1 后台会员管理	363
14-1-5 FormView 的 DefaultMode	326	16-2-2 查询订单情况	366
14-1-6 设置 FormView 字段的 默认值	328	16-2-3 登录、添加新会员与更改 密码	373
14-1-7 GridView 字段的显示与隐藏	329	16-3 总结与提高	376
14-2 使用 UserControl	331	第 17 章 高级范例——购物车与在线 付款机制	
14-2-1 在常规页面中使用 UserControl	331	17-1 功能结构	378
14-2-2 在 WebPartZone 中使用 UserControl	332	17-1-1 网页文件结构	378
14-2-3 创建 UserControl 的个性化 属性	334	17-1-2 相关的表结构	378
14-3 本章小结	337	17-1-3 运作流程	379
		17-2 页面操作	380
第 15 章 高级范例——电子商城		17-2-1 产品浏览与购物	381
15-1 结构说明	340	17-2-2 购物车与结账网页	385
15-1-1 网站版面结构	340	17-2-3 MasterPage 上的处理机制	388
15-1-2 网站文件结构	341	17-2-4 定向到在线付款网页	390
15-1-3 网站数据库结构	341	17-2-5 接收在线付款结果	393
15-2 创建网站 MasterPage 页面	342	17-3 总结与提高	396
15-2-1 使用 MasterPage 创建版面	342	17-3-1 订单浏览与通知功能	396
15-2-2 创建 SiteMapPath	343	17-3-2 超市代收机制	396
15-2-3 TreeView	344		
15-3 产品展示功能	345	第 18 章 高级范例——电子签名窗体	
		18-1 原理与结构	398
		18-1-1 运作流程图	400
		18-1-2 数据结构	400

第 20 章 高级范例——TreeView

讨论区

第 20 章 高级范例——TreeView

讨论区

20-1 讨论区结构 444

第 21 章 高级范例——电子相册

第 22 章 高级范例——Ajax 版 WebMail

第 23 章 高级范例——Web 统计图表制作

23-1 综合范例——WebChart 图表	
制作	494
23-1-1 数据源	494
23-1-2 创建 WebChart 类	494

23-1-3 使用 BarChart 对象绘制图表	495
23-1-4 BarChart 对象说明	497
23-2 PieChart 与 LineChart 的加入	499
23-2-1 使用 Partial Class	499
23-2-2 LineChart 的设计	500
23-2-3 PieChart 的设计	505
第 24 章 高级范例——全新互动式 EIP 系统	
24-1 系统规划	510
24-1-1 功能规划	510
24-1-2 页面规划	510
24-1-3 数据库规划	511
24-1-4 账号与权限管理机制	513
24-2 主页面设计	513
24-2-1 WebPartZone 设计	514
24-2-2 登录机制	516
24-2-3 账号创建机制	518
24-2-4 进入 WebPart 调整模式	519
24-2-5 页面表格规划	519
24-2-6 主菜单设计	520
24-2-7 利用 ProFile 机制设计个性化系统环境设置	521
24-3 子程序规划	524
24-3-1 数据库访问	524
24-3-2 EIP 功能	524
24-3-3 StudyHost20	525
24-4 附属子功能设计	526
24-4-1 日历	526
24-4-2 布告栏	530
24-4-3 待办事项	534
24-5 EIP 系统总结	538
24-5-1 互动机制	538
24-5-2 DataSource 控件	539
第 25 章 Ajax 范例——具有即时筛选功能的 ComboBox	
25-1 页面上的设置	542
25-2 Callback 代码	542
25-2-1 创建 Callback 触发机制	542
25-2-2 创建 Server 端的 Callback 处理函数	543
25-3 Callback 流程回顾	545
25-4 页面上的其他机制	546
第 26 章 Ajax 范例——Google Mail 的密码强度检查机制	
26-1 结构说明	550
26-1-1 需求	550
26-1-2 目的	550
26-2 实例操作	550
26-2-1 结构	550
26-2-2 程序编写	551
第 27 章 Ajax 范例——即时划位订票系统	
27-1 结构说明	556
27-1-1 系统需求	556
27-1-2 网站页面结构	558
27-1-3 网站文件结构	558
27-1-4 网站数据库结构	558
27-1-5 Callback 机制的应用	559
27-2 创建维护页面	560
27-2-1 用户界面代码	560
27-2-2 后台数据库访问代码	561
27-2-3 页面上的数据处理	563
27-3 创建订票页面	564
27-3-1 登录机制	564
27-3-2 显示座位表	565
27-3-3 Callback 即时订票功能的创建	567
27-3-4 Callback 订票机制的处理程序	570
27-3-5 订票机制 CommLib.Order	571
27-4 其他页面	573
27-4-1 创建结账页面	573
27-4-2 超时页面	575
27-4-3 注销与登录页面	575
27-5 总结与提高	576
27-5-1 给座位表添加图形	576
27-5-2 座位规划	576
27-5-3 在线付款	577
27-5-4 保留座位	577
第 28 章 Ajax 范例——网页即时信息	
28-1 系统规划	580

28-1-1 发送信息	580	29-1-1 页面规划	590
28-1-2 接收信息	580	29-1-2 数据库设计	590
28-1-3 信息的存储机制	581	29-2 页面设计	591
28-2 页面设计	581	29-2-1 主页面	591
28-2-1 发送信息页面	581	29-2-2 创建 CallBack 触发机制	591
28-2-2 主页面上的 CallBack 功能	583	29-2-3 在 Server 端处理 CallBack 的 操作	592
28-2-3 总结与提高	587	29-2-4 CallBack 返回值的接收与 处理	593
第 29 章 Ajax 范例——即时在线 会议室		29-2-5 数据库访问	594
29-1 系统规划	590	29-2-6 邀请用户	595

Chapter 01

构建自己的网站

欢迎您进入 ASP.NET 2.0 的程序设计领域。在 2006 年初，微软终于让“Visual Studio 2005 中文版”正式推出，同时随之而来的，当然就是 ASP.NET 2.0 的隆重登场。

ASP.NET 2.0 与 ASP.NET 1.1 相比，有着相当大幅度的改版与更新，许多过去 Web 应用程序开发人员所梦寐以求的功能，都在这个版本中体现。除此之外，更增加了相当多的 Web 控件以及集成性的机制，让 ASP.NET 开发人员的数量及能力大增。过去必须千辛万苦自行构建的机制，在 ASP.NET 2.0 中将变成默认的基本功能。

在这一章中，我们将为您介绍 Web 应用程序的演变以及建议您更换到 ASP.NET 2.0 的理由。从这个章节里，您将可以了解到为什么我们会向您强烈推荐运用 ASP.NET 2.0 来开发 Web 应用程序，以及 ASP.NET 2.0 相对于其他开发技术的优势等内容。

1-1 Web应用程序的革新和演变

1-1-1 Web应用程序的普及

Web 应用程序发展至今,已经有好几个年头了,从最初的网页、CGI,到现在我们用的 ASP、ASP.NET、PHP、JSP,现在有越来越多的工具帮助我们构建一个可以在因特网上执行的应用程序。

不过,我们究竟是在开发应用程序,还是网页?这依旧是一个很耐人寻味的话题。从一开始,因特网上要呈现的是文档,而不是程序,因为只呈现文档,所以是没有设计任何程序化的功能在里面的。

每一个显示的页面都是单纯的 HTML,页面和页面之间通过 HyperLink 彼此链接,这是 WWW 上面最原始的假设。

但是显然很多人觉得不够用,因为这样的网页不能互动,所以逐渐地有了 ASP 和 ASP.NET。ASP 的出现,已经给我们带来了相当大的帮助,通过 IIS 和在 IIS 上面执行的 ASP 程序,我们可以和用户互动,网页慢慢地不只是网页,开始可以以程序的面貌出现,而 ASP 大幅地降低了开发一个可以在 Internet 上面执行的程序的难度,从这时候开始,Web 应用程序开始雨后春笋般地出现。

早期的 Windows 应用程序现在好像只剩下 Web 应用程序还在使用,特别是在一些基本的应用上(不需要大量的运算,或是复杂的后台逻辑处理等),Web Solutions 好像变成了惟一的选择。

Web 应用程序有着几个突出的优点,例如:

- 不需要在客户端安装,用户只要能上网就能执行;
- 程序更新的时候客户端也不需要更新;
- 客户端只需要浏览器就可以操作(因此可以跨平台,不管 Client 端是什么机器,都可以执行)等。

1-1-2 Web应用程序主要存在的问题

发展至今,Web 应用程序依旧有着自身的问题,如同所有事情一样,缺点多半是与优点同时出现的,例如:

- 无法控制 Client 端的资源(因为安全性的理由,没有在客户端安装任何 ActiveX 或 JavaApplet 的 Web 应用程序,无法访问到客户端的资源,诸如打印机、读卡机、硬盘等);
- 互动时必须换页(PostBack),导致使用 ASP.NET 构建的大型 Web 应用程序,在页面上的控件很多的时候(表示网页本身的 HTML 代码就很大),换页非常困难;
- 受制于带宽,如果带宽不足,对客户来说换页更是难上加难(单击某个按钮,要等一两秒后画面才会出现,有时候还会超时);
- Client 端的事件和错误处理无法轻易捕捉(因为 Client 端只有浏览器,没有任何其他的程序在处理)。

正因为如此,除非我们写了很多的 Client 端 JavaScript(当然,也可以写 VBScript),否则 Client 端的这些问题始终是 Web 应用程序开发人员心中永远的痛。

针对这些问题,ASP.NET 2.0 提供了很多实用的解决方案。

1-2 ASP.NET 2.0的应用

1-2-1 ASP与ASP.NET 2.0之间的选择

如果您是 ASP 程序设计师, 至今依旧在 ASP 环境中执着开发, 笔者要语重心长地建议您尽快转换到 ASP.NET 2.0, 因为您会发现 ASP.NET 2.0 的开发人员, 不需要特别努力, 单单在页面上拉拉控件, 就可以完成您在 ASP 环境中努力三天三夜开发出的功能。

所以, 您可以上网下载我们的教学影片 (<http://book.StudyHost.com>), 当您看到 ASP.NET 2.0 的超强功能时, 相信您会发现, 从 ASP 到 ASP.NET 就算有再遥远的距离, 您也该勇往直前。更何况, 从 ASP 转换到 ASP.NET 2.0 也没有想像的那么困难, 有 ASP 基础的程序设计师, 转换到 ASP.NET 可说是易如反掌。

1-2-2 从ASP.NET 1.1到ASP.NET2.0的重大演变

简单地说, 仅一个新增的 CallBack 机制, 就值得您从 ASP.NET 1.1 升级到 ASP.NET 2.0。

您曾经听过 Ajax 或看过 Ajax 范例吗? 在 ASP.NET 中, 有没有可能不换页, 就可以让用户更新 Server 端数据库中的内容?

如果您看了我们的购物车范例, 您会发现, 我们的购物车增减购买的产品数量时, 根本不需要换页 (但是已经立刻反馈到后台的数据库中了)。

您曾经想要从 Client 端 Select 一些 Server 端的数据, 但是不想让画面闪烁换页吗? 在 ASP.NET 2.0 中的 CallBack 机制, 让您不需要把数据库中的数据一次读出, 就可以让 Client 端通过 CallBack 机制, 在不换页的情况下, 读取 Server 端的数据来更新页面。

ASP.NET 2.0 中加入了众多新控件, MasterPage 和 Theme 机制等众多的新功能, 让您过去在开发 ASP.NET 1.1 时, 需要写大量代码的情况大幅改变, 相信您在看本书的时候, 常会不由得赞叹: “这部分不需要编写一行代码……”。

1-2-3 ASP.NET 2.0与其他解决方案的比较

事实上, ASP.NET 1.0 的确与 JSP 有很多相似之处。

对 ASP 的开发人员来说, ASP 到 ASP.NET 1.0/1.1 可能有点不适应, 因为从 JSP 在网页里面的 Script 代码, 变成以 Web 控件与 WebForm 为主体的面向对象程序设计, 变化实在不算小, 反而从 JSP 的角度看 ASP.NET 1.1, 则有点理所当然的感觉。

但是到现在 ASP.NET 2.0 的出现, 我们可以说, 微软开始逐渐拉开 ASP.NET 与其他 Web 开发方案之间的距离。让 ASP.NET 2.0 的开发人员, 与使用其他工具的程序设计师相比, 具有相当大的优势。包括那些可以自由在页面上拖动的 WebPart, 可以套用版型的 MasterPage, 可以修饰页面的 Theme 和 skin 等。不仅如此, 大量节省代码的登录控件组 (包含完整的 Forms 认证机制、Membership 和 Role 类) 以及把过去 UI (用户界面) 和数据库访问机制彻底简化的数据访问控件 (DataSource 和 xxxxView 字尾的超强控件), 让页面上的窗体维护 (Master-Detail 或电子窗体) 机制变得几乎不需要编写程序。