



ASP.NET 2.0 MVP Hacks and Tips

ASP.NET 2.0 编程珠玑

——来自 MVP 的权威开发指南

(美) David Yack 等著
Joe Mayo
杨浩 全吉成 译



清华大学出版社

ASP.NET 2.0 编程珠玑

——来自 MVP 的权威开发指南

(美) David Yack 等著
Joe Mayo
杨浩 全吉成 译

清华大学出版社

北 京

David Yack, Joe Mayo, Scott Hanselman, Fredrik Normén, Dan Wahlin, J. Ambrose Little, Jonathan D. Goodyear
ASP.NET 2.0 MVP Hacks and Tips

EISBN: 0-7645-9766-3

Copyright © 2006 by Wiley Publishing, Inc., Indianapolis, Indiana

All Rights Reserved. Authorized translation from the English language edition published by Wiley Publishing, Inc.

本书中文简体字版由 Wiley Publishing, Inc. 授权清华大学出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2006-1583

版权所有, 翻印必究。举报电话: 010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

本书防伪标签采用特殊防伪技术, 用户可通过在图案表面涂抹清水, 图案消失, 水干后图案复现; 或将表面膜揭下, 放在白纸上用彩笔涂抹, 图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

ASP.NET 2.0 编程珠玑——来自 MVP 的权威开发指南/(美)杰克(Yack, D.), (美)梅尔(Mayo, J.)等著; 杨浩, 全吉成译. —北京: 清华大学出版社, 2006. 12

书名原文: ASP.NET 2.0 MVP Hacks and Tips

ISBN 7-302-14116-9

I. A… II. ①杰… ②梅… ③杨… ④全… 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 134541 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社 总 机: 010-62770175 客户服务: 010-62776969

组稿编辑: 王 军

文稿编辑: 郑雪梅

封面设计: 康 博

版式设计: 康 博

印 刷 者: 清华大学印刷厂

装 订 者: 三河市新茂装订有限公司

发 行 者: 新华书店总店北京发行所

开 本: 185 × 260 印张: 23.25 字数: 566 千字

版 次: 2006 年 12 月第 1 版 2006 年 12 月第 1 次印刷

书 号: ISBN 7-302-14116-9/TP · 8479

印 数: 1 ~ 4000

定 价: 48.00 元

前言

无论您是从书店中看到本书，从在线站点上订购它，还是幸运地获得赠书，在第一次看到本书时，都想知道它讲了什么。我们与出版社和其他人讨论过这个概念，有趣的是，听到了人们的各种理解。实际上，本书并不是讲授如何从安全的角度“攻击”ASP.NET 应用程序，也不是指导读者如何攻击劣质的应用程序，而是介绍如何使用 ASP.NET 建立和发布应用程序的各种技术。

澄清了本书的目的后，就要进一步介绍其内容，说明为什么这是您应购买和推荐给他人的一本好书。本书的每位作者都是资深的专业人员，在 .NET 平台方面有丰富的经验。他们都是 Microsoft MVP (Most Valuable Professional, 最有价值专家)，在各种在线和脱机环境下回答过成千上万个 ASP.NET 开发人员提出的问题。他们将通过本书与您共享他们回答过的许多问题的解决方案。您将从他们建立应用程序的丰富经验中获益匪浅。另外，由于 ASP.NET 2.0 的发布，本书还将探讨 ASP.NET 2.0 的所有新特性。

这不是 ASP.NET 的入门级图书，市面上已有许多图书介绍 ASP.NET 应用程序的开发。另外，本书也不是解释 ASP.NET 中每个特性和选项的参考指南。在许多提供参考资源和改进 MSDN 文档说明的图书中，有非常丰富的参考信息源。

“窍门程序”的概念

我们用“窍门程序(hack)”这个术语，表示鲜为人知的解决方案、未正式发布的特性、提示和技巧。一些人把它称为窍门程序，其他人称为“富有创造性的解决方案”。您可能对此有自己的命名，但它们基本上是相同的东西。任何规模的应用程序都或多或少地提高了 ASP.NET 的能力，把某种形式的窍门程序作为整个解决方案的一部分。

以前的窍门程序有页面模板、多窗体、URL 重写特性和 SQL 高速缓存依赖性。这些流行的窍门程序在无数产品应用程序中使用过。除了这些窍门程序之外，还有上百个其他窍门程序未得到广泛的认可，因此 ASP.NET 开发人员没有机会使用它们。本书就介绍一些不为人知、但很有用的窍门程序，开发人员在产品环境下可以使用它们。

使用窍门程序的原因

首先，要明白一个概念：所有的应用程序都应该是完美无缺的，也就是说，它们应该很好地构建，在开发过程中实现所选方法论和哲学体系的每个方面。在此过程中，它们还不能超出预算，当然还要按时完成。于是，在产品阶段，应用程序的维护就是免费的。而且，您可能还听过这样的观点：不应该使用未正式发布的特性，以及避免使用非主流的提示、技巧和技术。但我们并不是生活在这样一个童话式的世界中，现实是：我们必须建立健壮的 ASP.NET 应用程序，来获得有形资产价值。

在完美的世界中, ASP.NET Framework 应满足所有应用程序的需要, 开发是一个没有异常的代码生成过程。但是, MVP 总是要处理开发小组中产生的实际应用问题, 并用 ASP.NET 发布他们的解决方案。

我们坚信, 不存在这样的完美世界, 实际存在的应用程序都不是完美无暇的。另外, 我们几乎可以肯定, 本书中至少有一个窍门程序在大多数实际的 ASP.NET 应用程序中使用过。

无限期地等待在基本产品中支持某个特性, 或者没完没了地研究和试用各种方法来解决一个问题, 常常是不可行的。找出有创造性的解决方案, 同时考虑其优缺点和潜在的问题, 才能适时地开发出有效的应用程序。

窍门程序将最终提升产品

“Hack 是一个不太好的名称, 但它是创建下一代开发工具的一个重要部分。”

—— Jonathan Goodyear, Microsoft MVP, asp.netPRO, 2003 年 11 月

要衡量应用程序是否成功, 应考虑其商务价值和利用技术获得持续不断的竞争优势的能力。

第 1 章“窍门程序回顾”讨论了 ASP.NET 2.0 中包含的一些旧窍门程序。这一章还将讨论 ASP.NET 2.0 中可用作窍门程序的许多改进, 这些窍门程序现在已经是 ASP.NET 的组成部分。这是重要的一章, 介绍了 ASP.NET 2.0 的许多新特性, 如果读者是 ASP.NET 2.0 的初学者, 一定要认真阅读。

在某个窍门程序演变, 并最终在产品中实现时, 应尽可能学习和利用其内在的方法。

虽然 ASP.NET 2.0 听起来很不错, 但您坚持使用 ASP.NET 1.x, 本书仍对您有所帮助。首先, 本书讨论的许多内容都可以应用于 ASP.NET 1.x 应用程序, 或者在 ASP.NET 1.x 应用程序中做轻微的修改。另外, 第 1 章很适合于找出 1.x 解决方案, 并对迁移至 2.x 做出规划。

在 ASP.NET 2.0 的测试版中, 大多数产品反馈都标记上了“未来”。其原因很简单: 如果它们继续给产品添加其他内容, 该产品将不会附带这些内容。

显然, 在 ASP 和 .NET 继续走向成熟的过程中, 能发明的主要技巧将越来越少, 但开发团体的创造力将继续提供更快完成和发布应用程序的崭新方式。

窍门程序帮助我们学习

即使找不出可应用于某个问题的窍门程序, 看看它是如何工作的也是很有价值的。深入探讨目前的解决方案, 可以学到 ASP.NET 内部工作原理的更多知识。例如, Microsoft MVP Paul Wilson 公布了在 ASP.NET 1.x 中可以把多个窗体放在一个页面上的技术, 详细解释了页面请求的过程。这些详尽的论述不仅适用于他的窍门程序, 还有助于理解页面的生存期。在把这个知识用于应用程序的特定问题时, 对我们肯定是有帮助的。

如何使用本书

使用本书并没有指定的方式。本书并不是 ASP.NET 的入门级图书, 但在读者学习

ASP.NET 时可以提供很多帮助。在建立 ASP.NET 应用程序时,本书将是一个很有价值的项目指南。我们相信,每个团体都应把本书作为其团体库中的一个资源。

使本书中的窍门程序适合于自己

本书中的窍门程序将为解决实际的应用程序问题提供思路,并引导读者进行深入的思考。这决不是说明性的建议,因为每个人的应用程序都是不同的。在使用本书时,最重要的是确定如何把本书中的窍门程序应用于自己的应用程序和业务环境。并不是所有的解决方案都适合于所有的应用程序。

例如,在看到一段代码时,您可能突然认识到,这是一个非常好的思路,很适合于自己完成某个工作,或实现应用程序中的某个功能。

本书的读者对象

MVP 窍门程序并不是用于 MVP 的,正相反,本书中的窍门程序可由要学习这些技巧的开发人员使用,建立更好的应用程序,或建立他们自己的专业技巧库。

本书的内容

- 第 1 章回顾了 ASP.NET 以前版本中的窍门程序,描述了它们如何影响今天的技术,提升开发人员目前使用的技术和工具。一些窍门程序经得起时间的考验,例如 ASP.NET v1.1 URL 重写窍门程序。本章介绍了读者可能使用过的、目前已成为 ASP.NET Framework 的一部分的技术。例如,在 ASP.NET 中,模板页面现在称为 Master Page。
- 第 2 章是开始新项目,主要讲述了在扎实的基础上组织、规划和开始新项目的重要性。本章包含许多资源,在开始新项目时,最好利用其中的大多数资源。
- 第 3 章介绍了提供程序。我们可以扩展内置的提供程序,也可以建立自己的提供程序。即使坚持使用 ASP.NET 1.1,也可以在本章中找到使用提供程序的提示。
- 第 4 章介绍了智能客户机(比较聪明的客户机)。从客户回调到有一系列 ATLAS 的 AJAX,探讨了智能客户机如何应用于 ASP.NET 2.0。
- 第 5 章描述了更便于调试应用程序的一些新颖而强大的方式,演示了查看复杂对象的内容,减少调试窗口的混乱,以及使业务对象自动验证等技术。
- 第 6 章解释了如何通过定制控件创建窍门程序。在创建 RSS 控件时,将学习如何用几个控件窍门程序解决一些困难的问题。还将介绍如何实现一些新的 ASP.NET 2.0 控件特性,如动作列表。
- 第 7 章阐述了动态的行扩展技术,更新所有的行,以及 GridView 控件的页面模板。这一章提供了为 GridView 创建和定义自己的模板的知识。

- 第 8 章研究了扩展带有数据绑定特性的各种控件的方式。本章主要关注提高重用性和减少冗余代码的方法。还提供了扩展 `DataSource`、`Parameters` 和 `BoundFields` 的例子。
- 第 9 章解密了比较神秘的 `ViewState`，提供了解密它的工具和按照自己的意愿操纵它的窍门程序。
- 第 10 章陈述了处理应用程序高速缓存和 Web 窗体刷新的强类型化、面向对象的方式。还讨论了它与传统的 ASP.NET 高速缓存相比的优缺点。另外，本章论述了 ASP.NET 高速缓存查看器和管理器拖放插件，它们可以添加到任一 ASP.NET 应用程序中。
- 第 11 章展示了如何操作 ASP.NET 2.0 转换向导。这一章建立了一个代码示例，其功能可以在 ASP.NET v1.1 站点上查看。我们可以看到转换过程对这段代码进行了什么处理，说不定其中的某些内容您会感兴趣。
- 第 12 章探讨了前面创建、定义和操作所有这些新选项的部署方式，包括新 Web Deployment Project 的内容和如何为自己的项目定制它们。
- 第 13 章讨论了 Visual Studio 2005 IDE 及其新特性。提高效率的各种支持程序，例如片断、项模板和项目模板，都与窍门程序密切相关，能显著提高开发人员的效率。
- 第 14 章描述了安全方面的窍门程序，有助于使站点更坚固。不讨论安全性的 .NET 图书是不可想象的。本章讨论了 SQL 注射攻击和各种安全窍门程序，以帮助避免相关的攻击，最大限度地减少损失，例如参数化 SQL IN 子句。本章还将讨论如何定制 Login 控件，获得更好的输入验证功能。
- 第 15 章将学习如何创建自己的窍门程序。了解打包窍门程序，与团队共享它们的几种不同方式。也许您的窍门程序将成为 .NET、Visual Studio 或另一个工具的未来版本的一部分，每天将有上百万开发人员使用这些工具。
- 第 16 章详细解释了 Master Page 在 ASP.NET 2.0 应用程序中使用的不同方式。本章讨论的主题包括动态修改 Master Page，在 IIS 应用程序之间共享 Master Page，以及编程处理 Master Page 控件。
- 第 17 章深入研究了 ASP.NET HttpHandler 和 HttpModule 体系结构，包括样板文件、处理程序和模块的用法。

使用本书的要求

大多数作者在编写本书的过程中都使用了 Visual Studio 2005 标准版或更高版本。本书的大多数示例都可以运行在 Visual Studio 2005 的所有版本上。如果您遇到错误，在撞南墙之前，最好查看一下勘误表，也许其中已有一个更新的例子。

用户支持

我们总是想知道您对本书的看法，您喜欢哪些内容，您不喜欢哪些内容，这些信息将有助于我们下一次做得更好。如果您有什么意见和建议，请向 fwkbook@tup.tsinghua.edu.cn 发邮件。但是，请您一定要在您的信中注明本书的书名。

如何下载本书的示例代码

在您登录到 Wrox 站点 <http://www.wrox.com/> 时，只需使用 Search 工具或使用书名列表就可以找到本书。接着在 Code 栏中单击 Download 链接，或单击本书信息页面上的 Download Code 链接，就可以获得本书所有的源代码。另外，您也可以从本书的合作站点 www.tupwk.com.cn/downpage 上下载本书的所有源代码。

从以上站点上下载的文件使用 WinZip 进行了压缩。在把文件保存到硬盘的一个文件夹中时，需要使用解压缩软件(如 WinZip 或 PKUnzip)对该文件解压缩。在解压缩时，代码常常放在各自的章节文件夹中。在开始解压缩过程时，一定要将解压缩软件 WinZip 或 PKUnzip 设置为使用文件夹名。

勘误表

尽管我们已经尽了各种努力来保证文章或代码中不出现错误，但是错误总是难免的，如果您在本书中找到了错误，例如拼写错误或代码错误，请告诉我们，我们将非常感激。通过勘误表，可以让其他读者避免受挫，当然，这还有助于提供更高质量的信息。请给 fwkbook@tup.tsinghua.edu.cn 发电子邮件，我们就会检查您的信息，如果是正确的，就把它发送到该书的勘误表页面上，或在本书的后续版本中采用。

要在网站上找到勘误表，可以登录 <http://www.wrox.com>，通过 Advanced Search 工具或书名列表查找本书，然后在本书的信息页面上，单击 Book Errata 链接。

E-Mail 支持

如果您希望直接就本书的问题向对本书知之甚多的专家咨询，那么，就向 support@wrox.com 发电子邮件，在电子邮件的“主题”(Subject)栏中，加上本书的名称和 ISBN 的最后 4 位号码。典型的电子邮件应该包括下列内容：

- 在“主题”栏加上书的名称、ISBN 的最后 4 位数字和问题所在的页码。
- 在邮件的正文中加上您的姓名、联系信息和问题。

我们不会发给您垃圾邮件。我们只需要详细的情况以节省您的宝贵时间和我们的时间。当您发送电子邮件时，它会直接链接到以下支持链：

- 客户支持——您的消息会传送到我们的客户支持人员，他们是阅读信息的第一人。他们有常见问题的文件，会迅速回答一般性的问题。他们回答关于本书和网站的一般性问题。
- 编辑支持——更深的问题会转发到负责本书的技术编辑处。他(或)她具有编程或特殊产品的经验，能够回答某个主题的详细技术问题。

- 作者支持——最后，在编辑都不能回答问题的情况下(这种情况很少出现)，这些问题将转发到作者。我们试图保护作者不要从写作中分心，但是，我们也很愿意将特殊的问题转发给他们。所有的 Wrox 作者帮助支持他们的书籍。他们向客户和编辑回复电子邮件，所有的读者都会从中受益。

Wrox 支持过程只能提供直接与已出版的图书相关的问题。对于超出此范围的问题可以通过 <http://p2p.wrox.com>/论坛的团体列表来提供支持。

p2p.wrox.com

P2P 邮件列表是为作者和同行的讨论而设立的。我们在邮件列表、论坛和新闻组中提供“程序员到程序员的支持”(programmer to programmer support)，还包括一对一的电子邮件支持系统。如果把问题发送给 P2P，就可以相信，您的问题不仅是由支持专家解答，而且还要提供给我们邮件列表中的许多 Wrox 作者和其他业界专家。在 p2p.wrox.com 上，可以从许多不同的列表中获得帮助，不仅在阅读本书时获得帮助，还可以在开发应用程序时获得帮助。在网站的.NET 类别中，最适合本书的是 `beginning_vb` 和 `vb_dotnet` 列表。

要订阅一个邮件列表，可以遵循下面的步骤：

- (1) 进入 <http://p2p.wrox.com>。
- (2) 从左侧的菜单栏中选择合适的列表。
- (3) 单击想加入的邮件列表。
- (4) 按照指示订阅和填写电子邮件地址和密码。
- (5) 回复接收到的确认电子邮件。
- (6) 使用订阅管理器加入更多的列表，设置自己的邮件设置。

为什么这个系统提供最好的支持

您可加入该邮件列表中，也可以每周分类接收它们。如果您没有时间或设备接收该邮件列表，可以搜索我们的在线文档。垃圾邮件和广告邮件会被删除，您自己的电子邮件地址会被独特的 Lyris 系统保护起来。任何加入或退出列表的查询，或者与列表相关的一般问题，都应发送到 listsupport@p2p.wrox.com。

目 录

第 1 章 窍门程序回顾	1
1.1 由 ASP.NET 2.0 替代的 向导窍门程序	1
1.1.1 ASP.NET 向导先驱	1
1.1.2 ASP.NET v2.0 中的向导	2
1.1.3 实现 ASP.NET 2.0 向导	2
1.2 Master Page: 以前和现在	9
1.2.1 ASP.NET v1.1 中的 Master Page 模板	10
1.2.2 ASP.NET v2.0 中的 Master Page	10
1.2.3 实现 Master Page	10
1.3 URL 重写功能	13
1.3.1 为什么要重写 URL	14
1.3.2 ASP.NET v1.1 的窍门程序	14
1.3.3 ASP.NET v2.0 的替代品	14
1.3.4 实现 URL 映射功能	14
1.4 小结	24
第 2 章 开始新项目	25
2.1 组织项目和解决方案	26
2.1.1 创建解决方案	27
2.1.2 创建 Web 项目	28
2.1.3 从 Web 文件夹中删除文件	28
2.1.4 ASP.NET 小组的 一个窍门程序	29
2.1.5 创建公共类库项目	29
2.1.6 选择命名空间	30
2.1.7 使用数据库项目	31
2.1.8 使用 SQL Server 项目	33
2.1.9 使用 Web 安装和部署项目	34
2.2 创建项目引用	34
2.3 使用源代码控制	36

2.4 使用 Page 基类	36
2.5 使用 Master Page	38
2.6 建立第一个内容页面	40
2.7 建立主题	41
2.7.1 使用 Skin 还是 CSS	41
2.7.2 建立 Skin 文件	41
2.7.3 给 Skin 文件命名	41
2.8 Web 配置和团队	42
2.8.1 简单方式	42
2.8.2 外部文件方式	42
2.9 导出模板	43
2.10 小结	43
第 3 章 提供程序的功能	45
3.1 什么是提供程序模型	45
3.2 ASP.NET 2.0 附带的提供程序	46
3.3 扩展内置的提供程序	48
3.3.1 继承 SqlMembershipProvider	48
3.3.2 配置新的提供程序	49
3.3.3 检查提供程序的配置	49
3.3.4 重写 ValidateUser	50
3.4 建立自己的成员提供程序	51
3.5 建立自己的角色提供程序	55
3.6 建立角色提供程序代理	57
3.6.1 实现角色代理提供程序	58
3.6.2 为角色代理提供 程序修改 Web.Config	59
3.7 建立定制提供程序	60
3.7.1 定制提供程序的示例	60
3.7.2 定制提供程序的基础	61
3.8 使用 ASP.NET 1.1	66
3.9 小结	67
3.10 提供程序的其他资源	67

第4章 更智能的 Web 客户程序	69
4.1 更智能的 Web 客户程序.....	69
4.2 AJAX 体验.....	70
4.3 之后就有了 ASP.NET 2.0.....	70
4.4 客户回调的工作原理.....	71
4.5 Atlas.....	82
4.5.1 ASP.NET 小组的一个 窍门程序.....	82
4.5.2 Atlas 不仅仅是一个 客户端脚本库.....	83
4.5.3 现在要做的工作.....	93
4.6 小结.....	94
第5章 调试已创建好的代码	97
5.1 一步到位的显示功能.....	97
5.1.1 可视化器的概念.....	97
5.1.2 建立可视化器.....	97
5.1.3 测试可视化器.....	102
5.1.4 部署可视化器.....	103
5.2 调整对象以便于调试.....	104
5.2.1 设置对象成员的可见性.....	105
5.2.2 用类型代理简化对象视图.....	107
5.2.3 给业务对象添加调试逻辑.....	108
5.2.4 总结.....	110
5.3 小结.....	113
第6章 控件的窍门程序	115
6.1 用 XML 控件发布 RSS.....	115
6.1.1 内容管理.....	115
6.1.2 定义控件.....	116
6.1.3 提取内容.....	116
6.1.4 高速缓存以提高性能.....	117
6.1.5 管理 XSLT 源代码.....	118
6.2 快速定制类型转换器.....	120
6.2.1 创建定制的类型转换器.....	120
6.2.2 建立定制的动作列表 和设计器.....	125
6.2.3 管理设计期间的支持.....	129
6.2.4 进入 Visual Studio 2005 自动对象模型.....	131
6.2.5 实现定制的类型编辑器.....	132
6.2.6 在设计期间调试.....	139
6.3 小结.....	139
第7章 GridView 窍门程序	141
7.1 行的动态扩展.....	141
7.2 为 GridView 控件创建 PagerTemplate.....	146
7.3 删除 GridView 的选中行.....	152
7.4 更新 GridView 的所有行.....	157
7.5 添加选中的行.....	165
7.6 使用标题中的上下箭头 进行排序.....	170
7.7 给 GridView 控件添加 客户端脚本.....	173
7.8 用 IPostBackEventHandler 扩展 GridView.....	175
7.9 小结.....	180
第8章 数据绑定	181
8.1 继承自己的 DataSource 控件.....	181
8.2 使用数据源配置向导.....	183
8.2.1 添加控件设计器.....	188
8.2.2 生成与应用程序 相关的 DataSource 控件.....	190
8.3 扩展 DataSource 控件.....	190
8.3.1 在插入过程中处理默认值.....	191
8.3.2 扩展 XMLDataSource 来调用对象.....	194
8.4 使用定制参数.....	198
8.5 扩展 QueryStringParameter.....	198
8.5.1 建立定制参数来获取数据.....	201
8.5.2 使用反射建立定制参数.....	203
8.5.3 从另一个 DataSource 中获得值.....	205
8.6 使用定制字段.....	209
8.7 小结.....	214

第 9 章 ViewState	215	第 12 章 部署窍门程序	261
9.1 ViewState: 是邪恶之神 还是聪明的狐狸	215	12.1 Copy Web Site 向导	261
9.2 LosFormatter: 遗漏的 串行化器	216	12.2 Publish Web Site 向导	263
9.3 处理 ViewState	219	12.3 使用 Windows 安装程序	264
9.3.1 用 ASP.NET 1.1 分解 ViewState	219	12.4 产品团队的一个窍门程序	264
9.3.2 压缩 ViewState	222	12.4.1 安装插件	265
9.3.3 ViewState 的其他存储位置	225	12.4.2 添加到解决方案中	265
9.3.4 把 ViewState 移到页面 的底部	227	12.4.3 添加项目之后	266
9.4 小结	230	12.4.4 修改默认选项	266
第 10 章 高速缓存窍门程序	231	12.4.5 加快开发人员的 构建速度	266
10.1 静态高速缓存	231	12.4.6 构建服务器的特殊配置	267
10.1.1 Cache 示例类	231	12.4.7 设置编译选项	268
10.1.2 示例解决方案剖析	234	12.4.8 输出程序集选项	268
10.1.3 初始化高速缓存	234	12.4.9 与其他项目共享 App_Code	269
10.1.4 读取高速缓存	236	12.4.10 签署选项	269
10.1.5 修改高速缓存	238	12.4.11 部署选项	269
10.1.6 高速缓存的改变通知	239	12.4.12 进一步定制部署项目	270
10.1.7 刷新高速缓存	241	12.5 准备服务器	274
10.2 查看 ASP.NET 高速缓存	242	12.6 小结	275
10.3 小结	247	12.7 其他部署资源	276
第 11 章 从 1.x 迁移到 ASP.NET 2.0	249	第 13 章 使用 Visual Studio	277
11.1 创建 v1.x 演示应用程序	249	13.1 ASP.NET 代码片断	277
11.1.1 用户控件	250	13.1.1 使用片断	277
11.1.2 类	251	13.1.2 使用预生成的片断	278
11.1.3 页面	252	13.1.3 创建定制的片断	278
11.2 理解迁移	255	13.1.4 管理片断	280
11.2.1 执行迁移	255	13.1.5 使用片断函数	281
11.2.2 旧网站移到什么地方	257	13.2 简化开发工作的设置	283
11.2.3 重要的文件变化	258	13.2.1 Source 视图和 Design 视图	283
11.2.4 理解用户控件引用 的变化	259	13.2.2 使用另一个 Web 服务器	284
11.2.5 转换问题	259	13.2.3 导入导出团队设置	285
11.3 小结	260	13.3 模板	285
		13.3.1 定制的项模板	285
		13.3.2 定制的项目模板	286
		13.4 小结	287

第 14 章 安全窍门程序	289
14.1 避免 SQL 注射.....	289
14.2 IN 表达式的参数化.....	292
14.3 防止规范化攻击.....	295
14.4 使用新文件上传控件.....	297
14.5 安全地使用动态模拟功能.....	297
14.6 在 Login 控件中验证强密码.....	299
14.7 小结.....	300
第 15 章 建立自己的窍门程序	301
15.1 确定何时有关窍门程序.....	301
15.1.1 窍门程序是否超出了 技术文档.....	302
15.1.2 是否找出已有功能 的新用法.....	302
15.1.3 它是否不为人知.....	303
15.1.4 它是否可以重用.....	303
15.1.5 是否可以与其他人共享.....	304
15.1.6 可以拥有一个窍门程序.....	304
15.2 打磨窍门程序以发布它.....	304
15.2.1 注释代码.....	304
15.2.2 使代码含义明确.....	305
15.2.3 使用公共编码约定.....	305
15.2.4 编写泛型例程.....	305
15.2.5 把窍门程序实现为 组件或控件.....	308
15.2.6 编写文档说明.....	309
15.3 通过团队测试窍门程序 并获得反馈.....	309
15.4 小结.....	310
第 16 章 Master Page	311
16.1 Master Page 基础.....	311
16.1.1 创建 Master Page.....	312
16.1.2 创建内容页面.....	313
16.2 编程修改 Master Page.....	316
16.2.1 修改 Master Page 的标题部分.....	316
16.2.2 查找 Master Page 中的控件.....	316
16.2.3 定义 Master Page 的属性.....	317
16.2.4 使用 MasterType 指令.....	318
16.3 嵌套 Master Page.....	319
16.3.1 创建嵌套的 Master Page.....	319
16.3.2 在 Visual Studio .NET 2005 中使用嵌套的 Master Page.....	321
16.4 在 IIS 应用程序之间共享 Master Page.....	324
16.4.1 把 Master Page 发布 为程序集.....	325
16.4.2 使用 Web Deployment Projects 插件.....	328
16.5 动态修改内容页面中的 Master Page.....	330
16.6 小结.....	334
第 17 章 处理程序和模块	335
17.1 HttpModule.....	335
17.1.1 IP Blacklisting HttpModule.....	335
17.1.2 用 HttpModule 重写 或重定向 URL.....	340
17.2 HttpHandler.....	344
17.2.1 样板 HttpHandler.....	344
17.2.2 不鼓励依赖专用于 图像的 HttpHandler.....	347
17.2.3 用 HttpHandler 合成图像.....	351
17.2.4 用 HttpHandler 生成闪电.....	353
17.3 联合使用处理程序和模块.....	357
17.4 小结.....	358

第 1 章

窍门程序回顾

在变化越来越快的世界中，总是存在窍门程序。现在，它们是开发人员的一种聪明、前沿的创造。将来的某一天，它们有可能成为主流代码或最佳实践方式，并集成到产品或过程中。窍门程序的成功标志是，有一天它变成了一种标准。

本章主要介绍已成功的窍门程序。我们将回顾 ASP.NET v1.1 中已得到广泛认可的一组窍门程序。它们非常有效，Microsoft 已在 ASP.NET v2.0 中添加了对它们的支持。本章并不是重新编写已发明的窍门程序，而主要探讨先驱和他们的工作，正是他们出色的工作使 Microsoft 在 ASP.NET v2.0 中添加了对窍门程序的支持。对于已转而使用 ASP.NET 2.0 的开发人员来说，要学习的是如何将这组窍门程序集成到 .NET Framework 中和相关的 Visual Studio 2005 支持。了解以前的窍门程序如何影响 ASP.NET 的当前版本，将使我们更加欣赏窍门程序，并激励我们开发出能正面影响 ASP.NET 未来版本的窍门程序。

1.1 由 ASP.NET 2.0 替代的向导窍门程序

向导是引导用户完成某个过程的用户界面工具。这里讨论的是添加到应用程序中的向导。开发人员已使用它们多年，例如安装程序，把复杂的操作简化为一系列步骤的工具等。它们传统上是客户端桌面应用程序的一个标准部分，但在 Web 应用程序中以窍门程序的面目出现。

1.1.1 ASP.NET 向导先驱

在早期的 ASP.NET v1.1 中，John Peterson 编写的向导窍门程序是作为一个范例在 ASP101.com(asp101.com/samples/wizard_aspx.asp)上发布的，称为“向导的 ASP.NET 版本(多页面窗体)”。这是从以前经典的 ASP 3.0 的实现更新而来的。在 ASP.NET 2.0 向导宣布支持它后，Tom Blanchard 在 123aspx.com/redir.aspx?res=32798 上撰写了文章“CodeSnip: Simulating the ASP.NET 2.0 Wizard Control with ASP.NET 1.x”。一些人还使用另一个已有的解决方案：User Interface Process(UIP) Application Block 上的 Wizard Navigator，它是由 Microsoft

Patterns and Practices Group 在 <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnpag2/html/cabctp.asp> 上创建的。

提示：

Microsoft 的在线链接停了许多，但搜索 MSDN Architecture Center 的 Patterns and Practices 资源页面，仍能找到 UI Application Block。

1.1.2 ASP.NET v2.0 中的向导

向导在 ASP.NET 2.0 中不再是一个窍门程序，它们很容易使用，在 Visual Studio 2005 上得到了很好的支持。下一节将介绍如何实现向导。

1.1.3 实现 ASP.NET 2.0 向导

ASP.NET 2.0 向导实现为控件。要使用它们，只需把一个新的向导控件添加到可视化设计器中，设置其属性即可。本节描述如何通过一个范例菜单选择应用程序来实现 ASP.NET 2.0 向导。在这个过程中要进行许多选择，向导将根据输入参数管理导航。下面的步骤说明了如何创建范例应用程序：

(1) 创建一个新的 Web 项目。为此，选择 File | New | Web Site，再选择 ASP.NET Web Site 模板，把目录位置改为 WizardDemo，如图 1-1 所示。

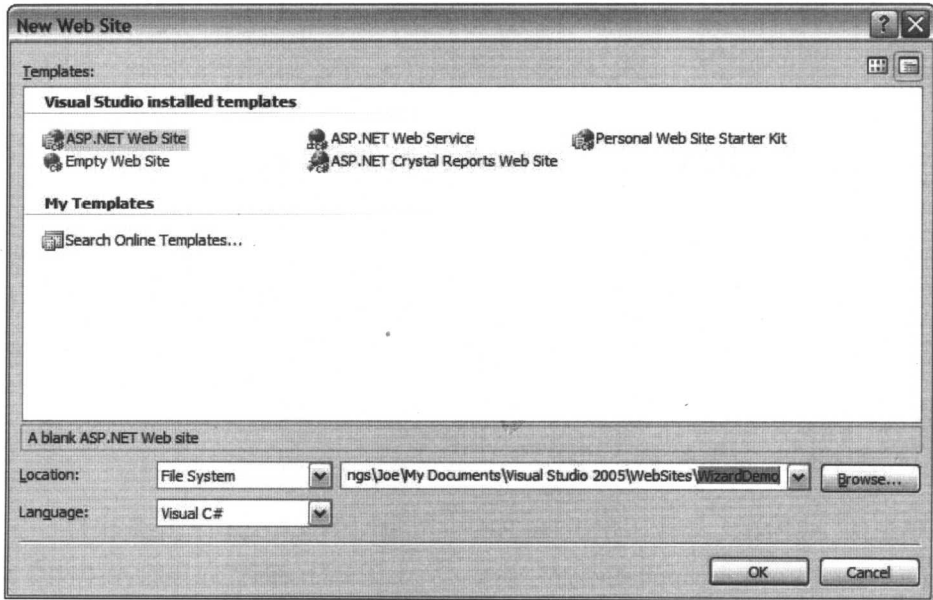


图 1-1

(2) 单击 OK 按钮，Visual Studio 2005 就会创建一个新项目，如图 1-2 所示。

(3) 如果当前在 HTML 视图上，如图 1-2 所示，就单击编辑器左下角的 Design 链接，进入设计视图。

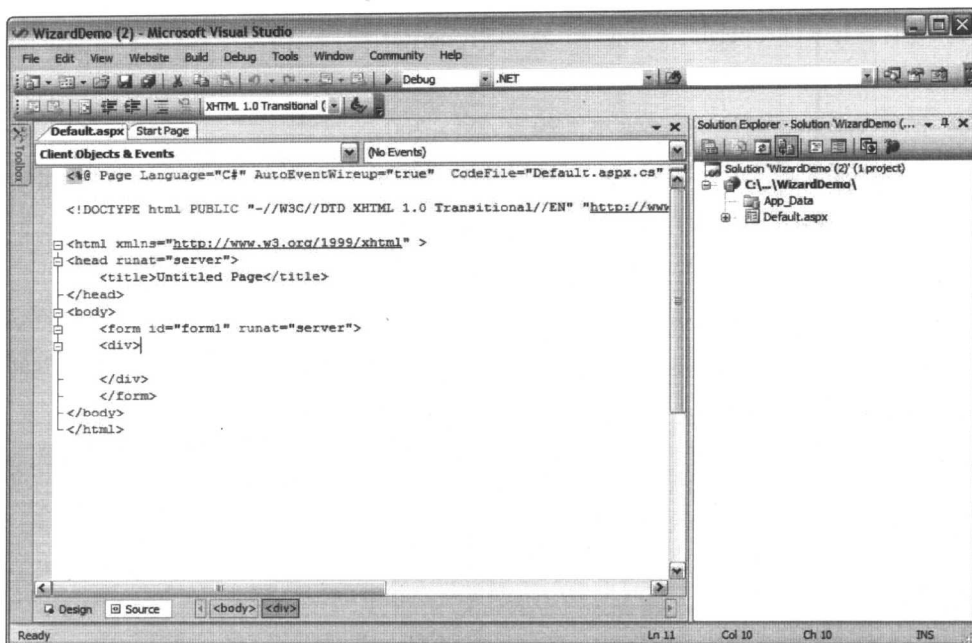


图 1-2

(4) 在设计视图中，找到工具箱，它位于 IDE 的左边。把鼠标指针停放在 Toolbox 标签上，就可以打开它。单击工具箱的 Standard 区域，选择 Wizard 控件。把一个 Wizard 控件拖放到窗体上，如图 1-3 所示。

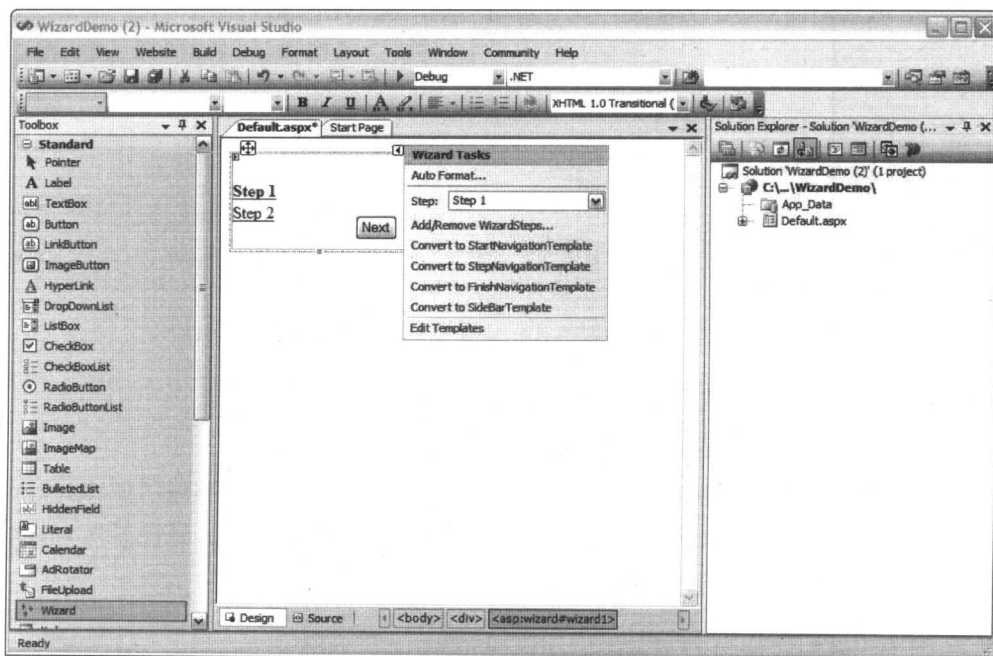


图 1-3

(5) 选择 View | Properties Window, 显示 Properties 窗口。在这个窗口中, 把 HeaderText 属性设置为 Menu Selector。

(6) 扩展 Wizard 控件, 使其大小大约为 400×200。

(7) Wizard 控件在第一次添加到 Web 窗体上时, 显示了两个链接, 分别称为 Step 1 和 Step 2。选择 Wizard 控件中的 Step 1。

(8) 在 Wizard 控件中选择编辑区域, 输入 “What is your Eating Preference?”。

(9) 从工具箱中把一个 RadioButtonList 控件拖放到 Wizard 控件的编辑区域, 在 Action List 上选择 Edit Items。在 ListItemCollection 编辑器上给两个新项的 Text 属性添加 Meat Eater 和 Vegetarian 项。图 1-4 显示了第 5~9 步的结果。

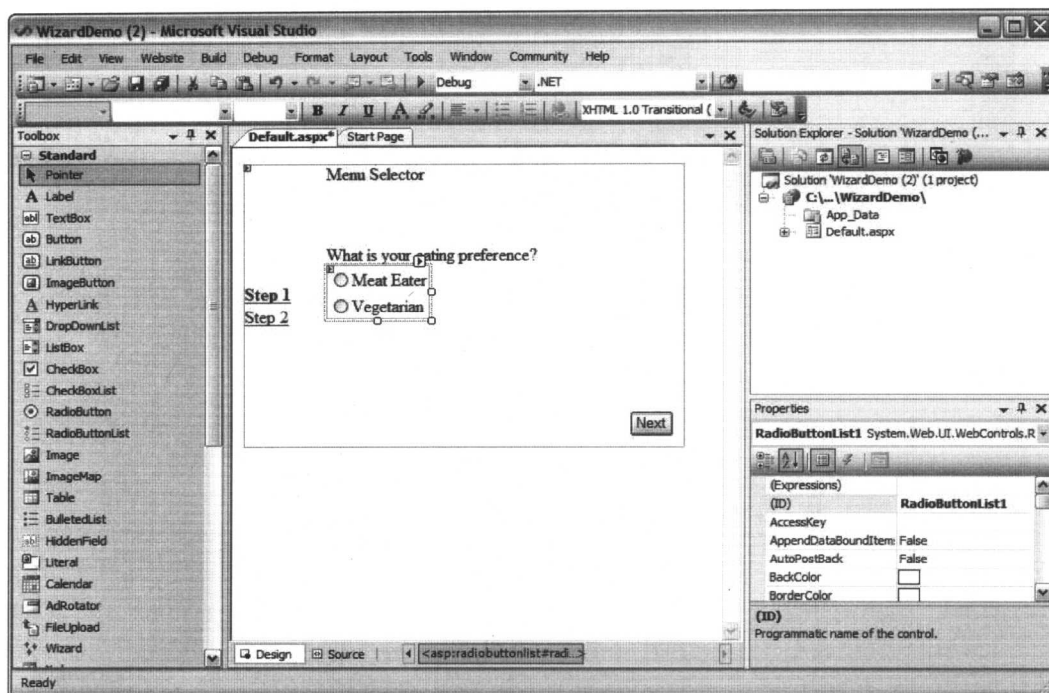


图 1-4

(10) 在 Wizard 控件中选择 Step 2。

(11) 在编辑区域输入 “Please Select Main Course:”。

(12) 把一个 RadioButtonList 控件从工具箱拖放到 Wizard 控件的编辑区域。选择 Action List 中的 Edit Items, 在 ListItemCollection 编辑器中为三个新项的 Text 属性添加 Chicken、Fish 和 Steak 项。图 1-5 显示了第 10~12 步的结果。