

PROGRAMMER TO PROGRAMMER™



Professional Visual Basic 2008

Visual Basic 2008

高级编程

(第5版)

(美) Bill Evjen 等著
Billy Hollis
张敏 丁峰 译



清华大学出版社

Visual Basic 2008 高级编程

(第 5 版)

(美) Bill Evjen 等著
Billy Hollis
张 敏 丁 峰 译

清华大学出版社

北 京

Bill Evjen, Billy Hollis, et al.
Professional Visual Basic 2008
EISBN: 978-0-470-19136-1

Copyright © 2008 by Wiley Publishing, Inc.
All Rights Reserved. This translation published under license.

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: 01-2009-2062

本书封面贴有 Wiley 公司防伪标签, 无标签者不得销售。
版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

Visual Basic 2008 高级编程(第5版)/(美)伊夫杰(Evjen, B.), 郝利斯(Hollis, B.)等著; 张敏, 丁峰译.
—北京: 清华大学出版社, 2009.6

书名原文: Professional Visual Basic 2008

ISBN 978-7-302-20086-4

I. V… II. ①伊…②郝…③张…④丁… III. BASIC 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 067748 号

责任编辑: 王 军 梁卫红

装帧设计: 孔祥丰

责任校对: 胡雁翎

责任印制: 李红英

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 清华大学印刷厂

装 订 者: 北京市密云县京文制本装订厂

经 销: 全国新华书店

开 本: 185×260 印 张: 74 字 数: 1990 千字

版 次: 2009 年 6 月第 1 版 印 次: 2009 年 6 月第 1 次印刷

印 数: 1~3500

定 价: 139.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770177 转 3103 产品编号: 029293-01

前言

2002 年,随着 VB.NET 的引入,Visual Basic(VB)迎来了自发布以来最大的变革。十几年来,VB 一直没有进行大的修订。但.NET 已经远远超出了这种修订,它改变了软件开发的几乎所有方面。从集成 Internet 功能到创建面向对象的架构,VB.NET 向传统的 VB 开发人员发出了挑战,要求他们学习全新的概念和技术。

2008 年,Microsoft 对 Visual Basic 语言做了进一步的改进(重命名为 VB 2008)。作为一种真正的面向对象语言,VB 2008 引入了很多新增功能,为底层数据提供了更好的访问特性。传统的 VB 6 开发人员必须从头学习这些内容,但对于熟悉 VB 以前版本的开发人员来说,VB 2008 是比较容易学习的,本书就能提供这方面的帮助。

本书介绍的是 .NET Framework 3.5,它不仅详细介绍了 VB 的最新版本(即 VB 2008),还探讨了新的架构,通过这个架构,VB 开发人员能利用 Windows Presentation Foundation(WPF)建立新的应用程序类型,创建基于 Windows Communication Foundation(WCF)、Windows Workflow Foundation(WF)和 Windows CardSpace 的库。本书还介绍了一个很重要的功能:LINQ。

这些新增功能构成了 .NET Framework 3.5,这个版本安装在 .NET Framework 2.0 和 3.0 上,并以 .NET Framework 2.0 为基础。本书的一些章节将重点讨论 .NET Framework 3.0 和 3.5 引入的新技术。

首先,需要学习 VB 2008 和旧版本之间的区别。在一些情况下,相同的功能要采用不同的方式来实现。这些改变不是任意而为的,而是有很充足的理由。我们必须抛弃一些旧习惯,培养新习惯。

其次,我们必须接受新概念。完全面向对象的概念、新的组件技术、新的用于本地和 Internet 界面的可视化工具,所有这些都是在 VB 中高效开发应用程序的必备技能。

本书将全面阐述 VB。首先讨论 .NET Framework,最后讨论部署 .NET 应用程序的最佳方式。中间部分讨论从数据库访问到与诸如 XML 等其他技术集成的一切内容,并详细地介绍新增功能。VB 2008 已经成为一种强大且易于使用的语言。在编写网络程序时,就如同编写桌面程序一样得心应手。本书将介绍 .NET Framework 2.0、.NET Framework 3.0 和新发布的 .NET Framework 3.5。如果读者使用 1.0、1.1 或 2.0 编程,本书的一些章节就可能不太合适。要特别注意 .NET Framework 3.5 的新增功能。

VB 的重要性

在 .NET 的早期应用中,Microsoft 的新语言 C#得到了广泛的关注。但随着 .NET 应用

的增多, VB 的重要性也逐渐显现出来。Microsoft 对外宣称, 当生产率是最重要的因素时, VB 是开发应用程序的最佳语言。

VB 的未来开发将强调以最高效的方式访问整个 .NET Framework 的能力, 而 C# 开发强调的是编写代码的经验, 这非常切合 VB 的传统作用: 语言开发人员在现实世界中使用它能尽快创建商务应用程序。

区别不仅仅在学术方面。 .NET Framework 的一个最重要的优点是, 它允许用非常少的代码编写应用程序。在商务应用程序中, 主要目标是编写业务逻辑, 尽可能减少例程编码任务。换言之, 其价值在新的 .NET Framework 中不是编写出许多代码, 而是用尽可能少的代码编写出更健壮、有效的应用程序。

VB 非常适合于这种类型的开发, 因此成为目前既经济又实惠的软件开发工具, 将来它还会为此目的做进一步的细化和改进。

本书读者对象

本书主要针对有经验的、希望深入学习 VB 2008 的开发人员。本书讨论了最常见的编程任务和概念, 既适合于从早期版本过渡到 VB 2008 版本的开发人员, 也适合于以前使用过 VB, 目前需要更深入理解的开发人员。

本书对 VB 概念进行了全面的介绍, 但 .NET Framework 的内容广博而丰富, 不可能在一本书里讲清楚, 本书没有把 Web 开发作为最重要的内容, 但包含了 VB 中基于浏览器的编程内容, 专业的 Web 开发人员应该参考清华大学出版社引进并出版的《ASP.NET 3.5 高级编程》一书。

使用本书需要的条件

尽管可以使用 .NET Framework SDK 中的命令行工具来创建 VB 应用程序, 但是本书的大部分内容仍将使用 Visual Studio 2008(专业版或更高, 其中包含 .NET Framework SDK)。可以使用 Visual Studio 2005、Visual Studio .NET 2002 或者 Visual Studio .NET 2003, 但一些例子将无法运行, 因为代码中的很多功能都是旧版本中没有的。

此外:

- 需要 .NET Framework 3.5(如果机器上没有 .NET Framework 2.0, .NET Framework 3.5 会自动安装 .NET Framework 2.0)。
- 某些章节使用了 SQL Server 2005 和 SQL Server 2008。也可以使用 VS 2008 自带的 SQL Express 来运行样例代码。
- 某些章节使用了 Internet Information Services(IIS)。IIS 包含在 Windows Vista、Windows Server 2008、Windows 2003 Server、Windows 2000 Server、Windows 2000 Professional 和 Windows XP 中, 但它不会默认安装。
- 第 30 章在介绍排队事务时使用了 MSMQ。MSMQ 包含在许多 Windows 版本中, 但它也不会默认安装。

本书主要内容

第1章将讨论 VB 2008 中的许多常用类型。本章的主题包括类型概述、类型转换、引用类型、数组和其他集合、参数传递和装箱，还介绍了 VB 的基本语法。

第2章论述了 VB 中面向对象编程的特性。本章将定义对象、类、实例、封装、抽象、多态性和继承性。

第3章讲述了继承的概念及在 VB 中如何利用继承。我们将解释如何创建简单和抽象的基类，如何创建可以派生其他类的基类。本章把前面几章的理论应用于实践。还将讨论4个主要的面向对象概念(抽象性、封装性、多态性、继承)，以及如何将这些概念应用到设计和开发中，以创建高效的面向对象的应用程序。

第4章探讨了.NET平台的核心——公共语言运行库(CLR)。CLR负责管理.NET平台编译的代码的执行。我们将论述版本控制和部署、内存管理、跨语言集成、元数据和 IL Disassembler。

开发人员通常用英语建立应用程序。但随着应用程序的用户在不断增长，开发人员意识到需要对应用程序进行全球化处理。当然，最理想的情况是从一开始就使建立的应用程序能正确应对全球用户。但在许多情况下，这是不太可能的，因为应用程序需要完成额外的工作。.NET Framework 为解决应用程序的全球化做了许多努力。API 的变化、给服务器控件增加的功能甚至 VS 本身都需要我们做出额外的努力，才能更方便地使应用程序全球化。第5章将介绍建立全球化应用程序时要考虑的一些重要事项，详细探讨 System.Globalization 名称空间和它为应用程序提供的所有功能。

第6章介绍了新版本最大的改进之一——泛型。泛型是一个通用的强类型集合，它可以减少出错几率，提高性能，并为集合提供 Intellisense 功能。

第7章介绍了名称空间及其层次结构。解释了名称空间的概念和某些常用的名称空间。此外，还介绍了如何创建新的名称空间，以及在项目中如何导入名称空间，如何给名称空间起别名。本章还介绍了 VB 中的 My 名称空间。

第8章通过详细讨论 CLR 异常处理程序和 Try...Catch...Finally 结构，描述了 VB 2008 中错误处理和调试工作的原理。除此之外，还描述了错误和跟踪的记录，以及如何利用这些方法获取程序工作情况的反馈。

第9章集中讨论了 ADO.NET 对象模型，以便建立灵活、快速和可伸缩的数据访问对象和应用程序。该章还讲述了 ADO 向 ADO.NET 的转化，讨论了 ADO.NET 主要对象，以便建立对 .NET 应用程序的数据访问。

第10章展示了 .NET Framework 中生成和处理 XML 的功能。我们介绍了 .NET Framework 的 XML 相关名称空间，并详细讨论了一部分名称空间中的类。本章还涉及到一些使用 XML 的技术，比如 ADO.NET 和 SQL Server。本章关注的是 LINQ 推出之前的 XML。

第11章介绍了 .NET Framework 3.5 中最酷、最令人期待的功能：LINQ。LINQ 可以方便地访问底层数据，它基本上位于 ADO.NET 的上层。Microsoft 把 LINQ 作为一个轻型版本，给底层的数据库提供了一个强类型化的接口。使用 LINQ 可以查询对象、数据集、SQL Server 数据库、XML 等。

第 12 章介绍了 .NET 提供的安全功能和相关工具。Caspol.exe 和 Permview.exe 可以帮助建立和维护各种安全策略。System.Security.Permissions 名称空间可用于管理权限。我们还通过代码演示了 System.Security.Cryptography 名称空间的功能。

第 13 章将介绍开发 .NET 应用程序的下一代 IDE: VS 2008。这个 IDE 的以前版本是 Visual Studio .NET 2003、Visual Studio .NET 2002 和 VS 2005。本章主要讨论 VS 2008 版本, 以及如何使用它更快地建立更好的应用程序。

第 14 章描述了如何联合使用新的 SQL Server 2008 和 .NET 应用程序。SQL Server 提供了与应用程序的强大连接, 本章将解释如何高效地使用这个功能强大的数据库。

第 15 章讲述了 Windows 窗体, 主要讨论了窗体和内置控件。还讨论了哪些是新的, 哪些是对 VB 以前版本的修改, 以及 System.Windows.Forms 名称空间。

第 16 章讨论了在构建 Windows 窗体应用程序时会用到的高级功能。

第 17 章将介绍 Windows Presentation Foundation(WPF), 它是 .NET 3.0 的一个新组件, 在 .NET Framework 3.5 中做了改进。WPF 提供了建立应用程序的一种新方式。该章将介绍 Microsoft 提升 GUI 显示功能的这种新方式, WPF 所提供的显示层相当流畅和丰富。该章还将介绍 WPF 基础, 以及如何建立简单的基于 XAML 的应用程序。

许多公司都在 Windows 窗体上投入了巨资, 他们也不打算将其应用程序完全转换到这个新技术上。因此, Microsoft 提供了将 WPF 集成到 Windows 窗体应用程序的重要方式, 还允许将 Windows 窗体组件引入 WPF 应用程序中。第 18 章将介绍这些功能。

第 19 章介绍了 ASP.NET 3.5 的基础知识, 包括使用 VS 2008 建立 Web 应用程序, 以及应用程序和页面框架。

第 20 章介绍了 ASP.NET 3.5 最新版本中的几个高级功能, 包括跨页面传送、master 页面、站点导航、个性化等。

第 21 章介绍 Silverlight 提供的 Web 新技术。这个新技术提供了使用 XAML 的方式, 为终端用户提供了更流畅的浏览体验。

第 22 章介绍了如何使用 VB 处理 Microsoft Office 应用程序。

第 23 章讨论了程序集的概念及其在 CLR 中的作用。我们还学习了程序集的结构、内容以及包含的信息。另外介绍了程序集清单以及它在部署中的作用。

第 24 章介绍了 Windows 窗体和 Web 窗体的部署选项, 包括 ClickOnce 部署特性和 .msi 文件的创建。

第 25 章介绍了 COM 和 .NET 组件的互操作性, 以及 Microsoft 提供了哪些工具将这两项技术链接起来。

第 26 章将学习线程, 理解 .NET Framework 中的各种对象如何让用户开发出多线程的应用程序。我们将讨论如何创建线程, 如何将它们与进程关联在一起, 以及多任务和多线程之间的差异。

第 27 章将介绍一个新功能——Windows Workflow Foundation(WF), 它便于将工作流集成到应用程序中。WF 在 .NET Framework 3.0 中引入, 是 .NET 3.5 和 VS 2008 的一个重要部分。

第 28 章探讨了如何利用 VB 2008 创建和使用 Web 服务。该章还讨论了 CLR 提供的用于建立和利用 Web 服务的抽象类, 论述了一些支持 Web 服务的技术。另外阐述了使用分

布式体系结构的缺点以及 Web 服务的未来。

第 29 章讨论了如何在传统的 3 层应用程序设计中使用远程服务。本章将论述远程服务的基本体系结构，并建立一个基本服务器和客户程序，使用一个 singleton 对象将客户的请求传递到业务层上。接着介绍如何使用串行化把服务器上更复杂的对象传递给客户，以及如何使用调用环境(call context)，在每个调用中，把额外的数据从客户端传送给服务器，但无须改变对象模型。

第 30 章探讨了 .NET 组件服务，特别讨论了事务处理和排队组件。

第 31 章介绍了开发网络应用程序时需要使用的各种协议。我们还讨论了如何把庞大的网络融合到自己的应用程序中。

第 32 章介绍建立面向服务的组件的新方式——Windows Communication Foundation (WCF)，它可以通过许多协议进行基于标准的通信。WCF 是 Microsoft 为企业内部和外部的组件相互通信提供的一种新方式。

第 33 章讨论了在 Windows 服务的生成过程中，如何使用 VB。该章将讨论 Windows 服务的创建、安装、运行和调试。

第 34 章介绍了如何从 Web 上下载资源，如何设计自己的通信协议，以及如何在应用程序中重用 Web 浏览器控件。

附录 A 介绍了 VB 编译器 vbc.exe 所提供的功能。

附录 B 介绍了 Visual Basic Power Packs 工具，这是一组计划外的软件包，主要用于帮助维护传统 Visual Basic 6.0 应用程序的开发人员迁移到 Visual Basic .NET 上。

附录 C 列出了一些可用的 VB 资源。

源代码

在读者学习本书中的示例时，可以手工输入所有的代码，也可以使用本书附带的源代码文件。本书使用的所有源代码都可以从本书合作站点 <http://www.wrox.com/> 或 www.tupwk.com.cn/download 上下载。登录到站点 <http://www.wrox.com/>，使用 Search 工具或使用书名列表就可以找到本书。接着单击本书细目页面上的 Download Code 链接，就可以获得所有的源代码。

注释：

由于许多图书的标题都很类似，因此按 ISBN 搜索是最简单的，本书英文版的 ISBN 是 978-0-470-19136-1。

在下载了代码后，只需用自己喜欢的解压缩软件对它进行解压缩即可。另外，也可以进入 <http://www.wrox.com/dynamic/books/download.aspx> 上的 Wrox 代码下载主页，查看本书和其他 Wrox 图书的所有代码。

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 wkservice@vip.163.com 发电子邮件，我们就会检查您的信息，如果是正确的，我们将在本书的后续版本中采用。

要在网站上找到本书的勘误表，可以登录 <http://www.wrox.com>，通过 Search 工具或书名列表查找本书，然后在本书的细目页面上，单击 Book Errata 链接。在这个页面上可以查看到 Wrox 编辑已提交和粘贴的所有勘误项。完整的图书列表还包括每本书的勘误表，网址是 www.wrox.com/misc-pages/booklist.shtml。

P2P.WROX.COM

要与作者和同行讨论，请加入 p2p.wrox.com 上的 P2P 论坛。这个论坛是一个基于 Web 的系统，便于您张贴与 Wrox 图书相关的消息和相关技术，与其他读者和技术用户交流心得。该论坛提供了订阅功能，当论坛上有新的消息时，它可以给您传送感兴趣的论题。Wrox 作者、编辑和其他业界专家和读者都会到这个论坛上来探讨问题。

在 <http://p2p.wrox.com> 上，有许多不同的论坛，它们不仅有助于阅读本书，还有助于开发自己的应用程序。要加入论坛，可以遵循下面的步骤：

- (1) 进入 p2p.wrox.com，单击 Register 链接。
- (2) 阅读使用协议，并单击 Agree。
- (3) 填写加入该论坛所需要的信息和自己希望提供的其他信息，单击 Submit。
- (4) 您会收到一封电子邮件，其中的信息描述了如何验证账户，完成加入过程。

注释：

不加入 P2P 也可以阅读论坛上的消息，但要张贴自己的消息，就必须加入该论坛。

加入论坛后，就可以张贴新消息，响应其他用户张贴的消息。可以随时在 Web 上阅读消息。如果要想让该网站给自己发送特定论坛中的消息，可以单击论坛列表中该论坛名旁边的 Subscribe to this Forum 图标。

关于使用 Wrox P2P 的更多信息，可阅读 P2P FAQ，了解论坛软件的工作情况以及 P2P 和 Wrox 图书的许多常见问题。要阅读 FAQ，可以在任意 P2P 页面上单击 FAQ 链接。

目 录

第 1 章 VB 2008 核心元素	1	1.11 小结	41
1.1 原来的关键字和语法	2	第 2 章 对象语法综述	42
1.2 值类型和引用类型	7	2.1 面向对象的术语	43
1.3 命令：条件语句	10	2.1.1 对象、类和实例	43
1.3.1 If Then	10	2.1.2 对象的组成	43
1.3.2 比较运算符	10	2.2 使用对象	46
1.3.3 Select Case	12	2.2.1 对象的声明和实例化	46
1.4 值类型(结构)	13	2.2.2 对象引用	47
1.4.1 布尔类型	13	2.2.3 取消对象的引用	48
1.4.2 整数类型	13	2.2.4 前期及后期绑定	48
1.4.3 无符号类型	15	2.3 创建类	52
1.4.4 小数类型	15	2.3.1 创建基类	52
1.4.5 Char 和 Byte	17	2.3.2 构造函数方法	68
1.4.6 DateTime	17	2.3.3 终止和清除	69
1.5 引用类型(类)	18	2.4 高级概念	70
1.5.1 Object 类	18	2.4.1 重载方法	70
1.5.2 String 类	19	2.4.2 重载构造函数方法	73
1.5.3 DBNull 类和 IsDBNull()函数	22	2.4.3 共享方法、变量和事件	74
1.5.4 Nullable 类型	22	2.4.4 运算符重载	79
1.5.5 数组	23	2.4.5 委托	81
1.5.6 集合	26	2.4.6 类和组件	85
1.6 命令：循环语句	27	2.5 小结	86
1.6.1 For Each 和 For Next	28	第 3 章 面向对象编程	87
1.6.2 While、Do While 和 Do Until	29	3.1 继承	87
1.7 装箱	30	3.2 多接口	129
1.8 参数传递	31	3.2.1 对象接口	129
1.9 变量的作用域	32	3.2.2 辅助接口	130
1.10 数据类型的转换	33	3.3 抽象性	136
1.10.1 隐式转换和编译器选项	34	3.4 封装性	139
1.10.2 XML 字面量	36	3.5 多态性	141
1.10.3 执行显式转换	38	3.5.1 方法签名	141
		3.5.2 实现多态性	142
		3.6 继承	151

3.6.1 使用继承的场合	151	5.5.1 使用本地资源	196
3.6.2 继承和多接口	153	5.5.2 使用全局资源	202
3.6.3 多级继承	158	5.6 Windows 窗体中的 资源文件	203
3.6.4 “脆弱”的基类问题	159	5.7 小结	207
3.7 小结	162		
第4章 公共语言运行库	163	第6章 泛型	208
4.1 .NET 应用程序的组成元素	163	6.1 泛型的使用	208
4.1.1 模块	164	6.1.1 泛型类型	209
4.1.2 程序集	165	6.1.2 泛型方法	214
4.1.3 类型	165	6.2 创建泛型	215
4.2 版本化与部署	166	6.2.1 泛型类型	215
4.2.1 对版本化更好的支持	166	6.2.2 泛型方法	223
4.2.2 更好的部署	167	6.2.3 约束	223
4.3 跨语言集成	167	6.2.4 泛型和后期绑定	227
4.3.1 通用类型系统	168	6.3 小结	227
4.3.2 元数据	169		
4.3.3 对元数据更好的支持	169	第7章 名称空间	228
4.3.4 属性	170	7.1 名称空间的概念	228
4.3.5 Reflection API	172	7.1.1 名称空间与引用	231
4.4 IL Disassembler	173	7.1.2 常用的名称空间	233
4.5 内存管理	173	7.2 导入名称空间并指定别名	235
4.5.1 传统的垃圾回收机制	174	7.2.1 导入名称空间	235
4.5.2 更快地为对象分配内存	181	7.2.2 为名称空间指定别名	236
4.5.3 垃圾回收器的优化	181	7.2.3 在 ASP.NET 中引用 名称空间	237
4.6 小结	183	7.3 创建自己的名称空间	237
第5章 本地化	184	7.4 My 关键字	240
5.1 文化和区域	184	7.4.1 My.Application 名称空间	241
5.1.1 理解文化类型	185	7.4.2 My.Computer 名称空间	244
5.1.2 线程	185	7.4.3 My.Forms 名称空间	247
5.2 在 ASP.NET 中声明 全局文化	188	7.4.4 My.Resources 名称空间	247
5.3 在 ASP.NET 中使用 文化设置	189	7.4.5 My.User 名称空间	248
5.4 转换数值和操作	190	7.4.6 My.WebServices 名称空间	248
5.4.1 理解日期之间的区别	190	7.5 扩展 My 名称空间	248
5.4.2 理解数字和货币的区别	191	7.6 小结	250
5.4.3 理解排序字符串的区别	194		
5.5 使用 ASP.NET 资源文件	196	第8章 异常处理和调试	251
		8.1 VB 6 中的错误处理	251

8.2	.NET 中的异常处理	252	9.3.6	DataAdapter 对象	290
8.2.1	异常的重要属性和方法	252	9.3.7	SQL Server .NET 数据 提供程序	294
8.2.2	异常与 VB 6 中 Err 对象的区别	253	9.3.8	OLE DB .NET 数据 提供程序	294
8.3	结构化异常处理的关键字	254	9.4	DataSet 组件	294
8.3.1	Try、Catch 和 Finally 关键字	254	9.4.1	DataTableCollection 对象	295
8.3.2	Throw 关键字	256	9.4.2	DataRelationCollection 对象	295
8.3.3	抛出新的异常	257	9.4.3	ExtendedProperties 属性	295
8.3.4	Exit Try 语句	258	9.4.4	创建和使用 DataSet 对象	296
8.3.5	嵌套的 Try 结构	259	9.4.5	ADO.NET 的 DataTable 对象	298
8.3.6	异常属性的使用	260	9.4.6	ADO.NET 对 DataSet 和 DataTable 的改进	299
8.3.7	Message 属性	261	9.5	使用通用提供程序模型	301
8.3.8	InnerException 和 TargetSite 属性	261	9.6	ADO.NET 中的连接池	303
8.4	与 VB 6 样式的错误处理 交互操作	265	9.7	创建数据访问组件	304
8.5	错误记录	266	9.7.1	构造函数	305
8.5.1	事件日志	266	9.7.2	属性	306
8.5.2	事件、方法和属性	267	9.7.3	存储过程使用的 XML 结构	307
8.5.3	写入追踪文件	269	9.7.4	方法	307
8.6	使用 Trace 类分析问题和 测试性能	270	9.7.5	使用 DataSet 对象绑定 DataGrid	319
8.7	小结	273	9.8	小结	322
第 9 章	用 ADO.NET 3.5 访问数据	274	第 10 章	在 VB 2008 中使用 XML	323
9.1	ADO.NET 的体系结构	275	10.1	XML 简介	324
9.2	ADO.NET 的基本功能	275	10.2	XML 串行化	325
9.2.1	ADO.NET 的常见任务	275	10.3	System.Xml 文档支持	331
9.2.2	ADO.NET 的基本名称 空间和类	278	10.4	XML 流样式分析程序	332
9.2.3	ADO.NET 组件	279	10.4.1	写入 XML 流	333
9.3	.NET 数据提供程序	281	10.4.2	读取 XML 流	336
9.3.1	Connection 对象	281	10.4.3	使用 MemoryStream 对象	346
9.3.2	Command 对象	281	10.4.4	文档对象模型(DOM)	349
9.3.3	通过 Command 对象使用 存储过程	282	10.5	XSLT 转换	356
9.3.4	DataReader 对象	286			
9.3.5	命令的异步执行	287			

10.5.1	使用 XSLT 转换不同的 XML 标准	360	11.3.5	创建自己的 DataContext 对象	388
10.5.2	System.Xml.Xsl 中定义的其他类和接口	362	11.4	定制对象和 O/R 设计器	389
10.6	ADO.NET	363	11.5	查询数据库	391
10.6.1	ADO.NET 和 SQL Server 2000 内置的 XML 功能	364	11.5.1	使用查询表达式	391
10.6.2	XML 与 SQL Server 2005	367	11.5.2	查询表达式详述	392
10.6.3	XML 和 SQL Server 2008	367	11.5.3	用表达式过滤	392
10.7	ASP.NET 3.5 中的 XML	367	11.5.4	联合	393
10.7.1	XmlDataSource 服务器控件	368	11.5.5	数据项的组合	394
10.7.2	XmlDataSource 控件的名称空间问题	372	11.6	存储过程	396
10.7.3	Xml 服务器控件	372	11.7	LINQ to XML	396
10.8	小结	374	11.8	LINQ to XML 和 .NET 3.5	397
第 11 章	LINQ	375	11.8.1	创建 XML 文档的新对象	397
11.1	LINQ to SQL 和 Visual Studio 2008	376	11.8.2	Visual Basic 2008 冒险走上了另一条道路	397
11.1.1	用 LINQ to SQL 调用 Products 表: 创建控制台应用程序	376	11.8.3	名称空间和前缀	397
11.1.2	添加 LINQ to SQL 类	377	11.9	.NET Framework 3.5 中的新 XML 对象	397
11.1.3	O/R 设计器	378	11.9.1	XDocument 对象	398
11.1.4	创建 Product 对象	379	11.9.2	XElement 对象	398
11.2	对象到 LINQ 对象的映射	381	11.9.3	XNamespace 对象	399
11.2.1	DataContext 对象	382	11.9.4	XComment 对象	401
11.2.2	Table(TEntity)对象	385	11.9.5	XAttribute 对象	402
11.3	在没有 O/R 设计器的情况下工作	385	11.10	Visual Basic 2008 和 XML 字面量	402
11.3.1	创建定制对象	385	11.11	使用 LINQ 查询 XML 文档	404
11.3.2	用定制对象和 LINQ 查询	386	11.11.1	查询静态的 XML 文档	405
11.3.3	用查询限制所调用的列	387	11.11.2	查询动态的 XML 文档	406
11.3.4	使用列名	388	11.12	处理 XML 文档	407
			11.12.1	读取 XML 文档	407
			11.12.2	写入 XML 文档	409
			11.13	使用 LINQ to SQL 和 LINQ to XML	410
			11.13.1	建立 LINQ to SQL 组件	410

11.13.2 查询数据库并输出 XML	411	13.2.1 在代码中设置窗体属性	482
11.14 小结	412	13.2.2 Visual Studio 2008 的 其他组件	490
第 12 章 .NET Framework 中 的安全性	413	13.3 增强范例应用程序的功能	490
12.1 安全的概念与定义	414	13.3.1 定制代码	492
12.2 System.Security.Permissions 名称空间中的权限	415	13.3.2 构建应用程序	496
12.2.1 代码访问权限	417	13.4 Visual Studio 2008 中的 特色功能	501
12.2.2 基于角色的权限	418	13.4.1 构建配置	501
12.2.3 身份权限	420	13.4.2 任务列表	504
12.3 管理代码访问权限	420	13.4.3 Command 窗口	504
12.4 管理安全策略	425	13.4.4 Server Explorer	504
12.4.1 确定应用程序的 最小权限	434	13.4.5 在 Visual Studio 2008 中记录和使用宏	505
12.4.2 使用 Visual Studio 判断 最小权限集合	436	13.4.6 类图	507
12.4.3 安全性工具	438	13.4.7 Team System	508
12.4.4 使用 SecurityException 类处理异常	439	13.4.8 Team Editions for Developers	510
12.5 加密基础	440	13.4.9 性能工具	511
12.6 小结	460	13.4.10 单元测试	512
第 13 章 Visual Studio 2008	461	13.5 小结	513
13.1 Visual Studio 2008 从 Express 到 Team Suite 的各种版本	462	第 14 章 使用 SQL Server	515
13.1.1 从项目模板上创建项目	464	14.1 SQL Server Compact Edition	515
13.1.2 Solution Explorer 窗口	468	14.1.1 连接 SQL Server Compact Edition 数据库	517
13.1.3 My Project 视图	469	14.1.2 同步数据	520
13.1.4 Assembly Information 屏幕	469	14.2 SQL Server 2005 中的 CLR 集成	526
13.1.5 编译设置	470	14.2.1 决定使用 T-SQL 还是 VB	527
13.1.6 调试属性	472	14.2.2 创建用户定义的类型	528
13.1.7 引用	474	14.2.3 创建存储过程	541
13.1.8 资源	475	14.2.4 在 SQL Server 中 使用 Web 服务	547
13.1.9 设置	476	14.3 小结	553
13.2 Visual Studio 中的 ProVB_VS 项目	479	14.4 资源	553

第 15 章 Windows 窗体	554
15.1 Windows 窗体 2.0 版的 更新内容	554
15.1.1 现有控件的改进	554
15.1.2 新增控件	556
15.1.3 对老版本 Windows 窗体控件的替换	556
15.1.4 默认的窗体实例	557
15.2 System.Windows.Forms 名称空间	558
15.3 窗体的使用	558
15.3.1 通过 Sub Main 显示窗体	558
15.3.2 设置启动窗体	559
15.3.3 启动位置	559
15.3.4 窗体边框	560
15.3.5 始终置顶—— TopMost 属性	560
15.3.6 附属窗体	560
15.3.7 改变窗体的透明度	561
15.3.8 可视化继承	563
15.3.9 滚动窗体	563
15.3.10 MDI 窗体	563
15.3.11 VB 2008 中的 MDI 样例	564
15.3.12 对话框窗体	566
15.3.13 运行时的窗体	568
15.4 控件	569
15.4.1 以 Tab 键切换 控件的顺序	569
15.4.2 动态调整控件的 大小和布局	569
15.4.3 FlowLayoutPanel 控件	571
15.4.4 TableLayoutPanel 控件	573
15.4.5 Panel 和 GroupBox 容器控件	573
15.4.6 扩展的 Provider 控件	574
15.4.7 数据输入的高级功能	577
15.4.8 验证数据输入	579
15.4.9 工具栏与新增的 ToolStrip 控件	580

15.4.10 菜单	583
15.4.11 通用对话框	584
15.4.12 拖放操作	586
15.4.13 标准 Windows 窗体 控件小结	588
15.4.14 已撤销的控件	591
15.4.15 处理相关控件组	592
15.4.16 在运行时添加控件	593
15.5 其他编程技巧	594
15.6 小结	594

第 16 章 Windows 窗体的 高级功能	595
16.1 在可视化控件中封装逻辑	595
16.2 在 Windows 窗体中开发 自定义的控件	596
16.2.1 继承现有的控件	596
16.2.2 构建复合控件	596
16.2.3 从头编写控件	597
16.3 继承现有的控件	597
16.3.1 基本步骤	597
16.3.2 给派生的控件 添加代码	597
16.3.3 其他有用的特性	600
16.3.4 为派生的控件 自定义事件	601
16.3.5 限制选中项数的 CheckedListBox	602
16.4 Control 与 UserControl 基类	605
16.4.1 Control 类	605
16.4.2 UserControl 类	606
16.5 复合控件	606
16.5.1 创建复合的 UserControl	607
16.5.2 改变控件的大小	608
16.5.3 提供子控件的属性	608
16.5.4 一个具体的例子	608
16.6 从头构建控件	612
16.7 给工具箱中的控件 添加图标	617

16.8	在控件中嵌入其他控件	618	19.2	建立 ASP.NET 3.5 页面的 IDE	686
16.9	小结	620	19.3	建立 ASP.NET 应用程序	687
第 17 章	Windows Presentation Foundation	621	19.3.1	应用程序的定位选项	687
17.1	内容、位置、原因、方式——WPF 策略	621	19.3.2	IIS	689
17.2	光栅图形和矢量图形	623	19.3.3	FTP	689
17.3	下一个 Windows 项目应使用 WPF 吗	623	19.3.4	Web 站点需要 FrontPage Extensions	690
17.4	创建 WPF 应用程序	624	19.4	ASP.NET 页面结构选项	690
17.4.1	使用 WPF 和 XAML	627	19.4.1	内联编码	692
17.4.2	定制用户界面	642	19.4.2	后台编码模型	694
17.4.3	定制按钮	651	19.5	ASP.NET 3.5 的页面指令	695
17.4.4	Expression Blend	654	19.6	ASP.NET 页面事件	696
17.5	小结	663	19.7	ASP.NET 应用程序文件夹	697
第 18 章	集成 WPF 和 Windows 窗体	664	19.7.1	\App_Code 文件夹	698
18.1	集成库	664	19.7.2	\App_Data 文件夹	701
18.2	在 Windows 窗体中包含 WPF 控件	666	19.7.3	\App_Themes 文件夹	701
18.2.1	创建 WPF 控件库	666	19.7.4	\App_GlobalResources 文件夹	701
18.2.2	Windows 窗体应用程序	669	19.7.5	\App_LocalResources 文件夹	702
18.3	在 WPF 中包含 Windows 窗体控件	676	19.7.6	\App_WebReferences 文件夹	702
18.4	集成的限制	681	19.7.7	\App_Browsers 文件夹	702
18.5	小结	681	19.8	Global.asax	702
第 19 章	使用 ASP.NET 3.5	683	19.9	ASP.NET 服务器控件	705
19.1	ASP.NET 的目标	683	19.9.1	服务器控件的类型	706
19.1.1	开发效率	683	19.9.2	用服务器控件构建页面	707
19.1.2	管理和维护	683	19.9.3	使用服务器控件的事件	709
19.1.3	性能和可伸缩性	684	19.10	用 JavaScript 处理页面和服务器控件	710
19.1.4	ASP.NET 编译系统	684	19.10.1	使用 Page.Client Script.RegisterClient ScriptBlock	711
19.1.5	ASP.NET 应用程序的健康监控	685	19.10.2	使用 Page.Client Script.RegisterStartupScript	713
19.1.6	读写配置设置	685	19.10.3	使用 Page.Client Script.RegisterClient ScriptInclude	714
19.1.7	本地化	685			
19.1.8	访问数据的对象	685			

19.11 小结	714	20.9.8 用 AJAX 建立简单的 ASP.NET 页面	761
第 20 章 ASP.NET 3.5 的高级功能	715	20.10 小结	766
20.1 应用程序和页面	715	第 21 章 Silverlight 开发	767
20.1.1 跨页面传送	715	21.1 Silverlight 简介	767
20.1.2 ASP.NET 编译功能	719	21.1.1 Silverlight 浏览器 的要求	767
20.2 Master 页面	722	21.1.2 Silverlight 的两个版本	768
20.2.1 创建 Master 页面	723	21.1.3 安装 Silverlight	768
20.2.2 创建内容页面	725	21.2 开发 Silverlight 应用程序	769
20.2.3 在应用程序范围内 声明 Master 页面	727	21.3 Silverlight 示例	774
20.2.4 为 Master 页面提供 默认内容	728	21.3.1 简单的 Hello World! 示例	775
20.3 数据驱动的应用程序	728	21.3.2 处理多个元素	776
20.3.1 使用 GridView 和 SqlDataSource 控件	729	21.3.3 事件和 Silverlight	778
20.3.2 允许在 GridView 中 编辑和删除记录	734	21.4 小结	781
20.3.3 其他内容	737	第 22 章 Visual Studio Tools for Office	782
20.4 页面导航	737	22.1 VSTO 的各个版本	782
20.4.1 使用 SiteMapPath 服务器控件	739	22.1.1 Office 的自动化 功能和 VSTO	783
20.4.2 Menu 服务器控件	740	22.1.2 VSTO 项目类型	784
20.4.3 TreeView 服务器控件	741	22.2 Office 业务应用程序的 体系结构	785
20.5 成员和角色管理	744	22.3 使用 VBA 和 VSTO	786
20.6 个性化功能	747	22.4 创建文档模板(Word)	791
20.7 在 Vista 的 IIS 上配置 ASP.NET	749	22.4.1 给文档添加内容	793
20.8 使用 ASP.NET 的提供 程序模型	750	22.4.2 添加 Ribbon 和 动作面板	794
20.9 ASP.NET AJAX	756	22.4.3 激活动作面板	797
20.9.1 理解对 AJAX 的需求	757	22.4.4 更新内容控件	800
20.9.2 推出 AJAX 之前	757	22.5 创建 Office 插件(Excel)	802
20.9.3 AJAX 改变的方面	758	22.6 Outlook Form Regions	808
20.9.4 ASP.NET AJAX 和 Visual Studio 2008	759	22.7 小结	816
20.9.5 客户端技术	759	第 23 章 程序集	817
20.9.6 服务器端技术	760	23.1 程序集	817
20.9.7 用 ASP.NET AJAX 开发	760	23.2 清单	818