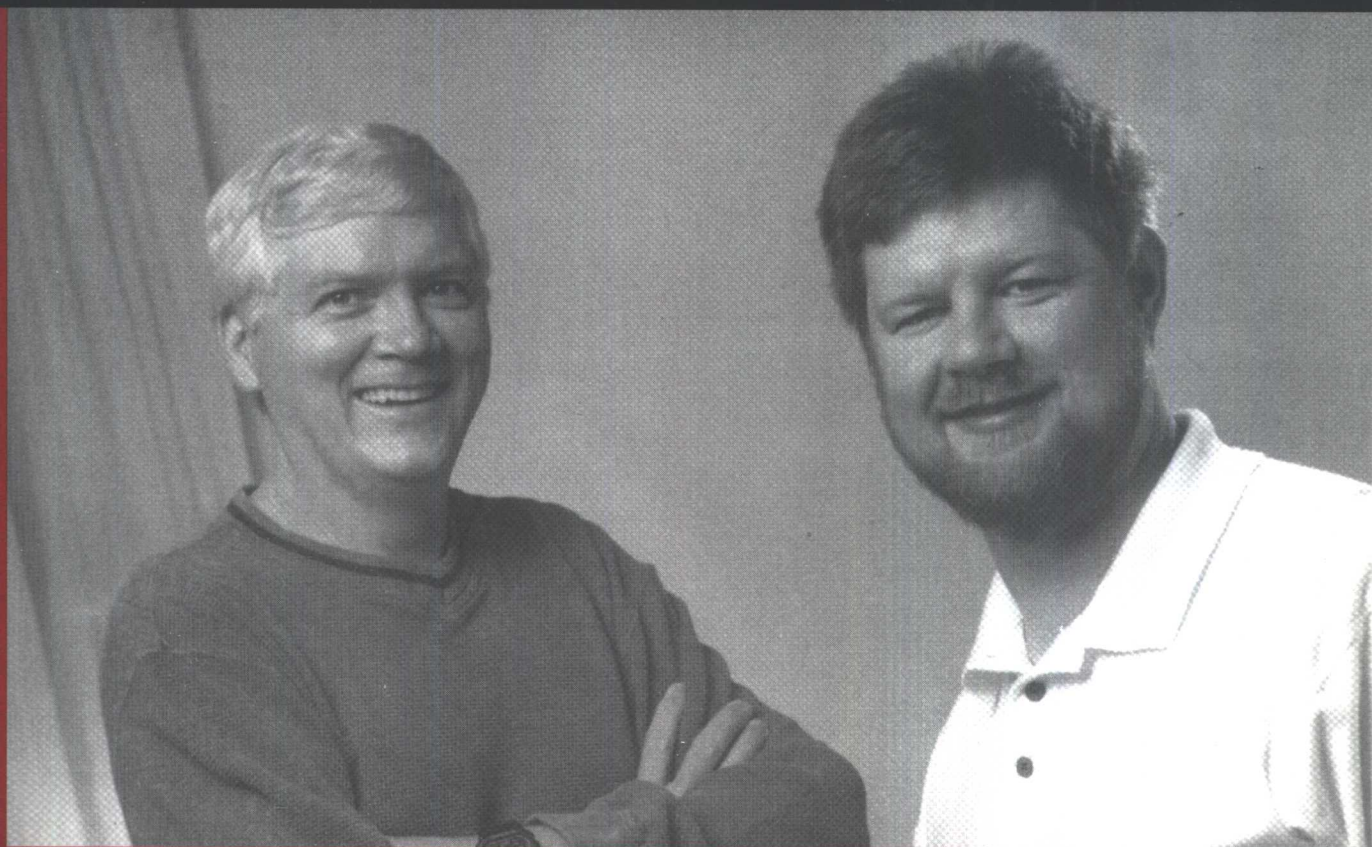


PROGRAMMER TO PROGRAMMER™



Billy Hollis  
Rockford Lhotka 等著  
康 博 译

# VB.NET

## 程序设计教程



清华大学出版社  
<http://www.tup.tsinghua.edu.cn>



# VB.NET 程序设计教程

Billy Hollis  
Rockford Lhotka 等著  
康 博 译

158220.123

清华大学出版社

(京)新登字 158 号

北京市版权局著作合同登记号:01-2001-2572

### 内 容 简 介

Microsoft 已经将它的未来与 .NET Framework 紧密联系在一起,而 .NET Framework 是第一个完全按 Internet 思想设计的开发平台。VB.NET 是基于此框架的一种开发工具,Microsoft 在 VB.NET 中添加了许多新功能,使它成为完全面向对象的编程语言,并成为功能有很大改进的 Web 设计工具。

本书主要涉及的内容包括:.NET Framework 和 VS.NET IDE,VB.NET 中的新功能,面向对象功能,使用新功能创建 WinForms,创建 Web Forms 和 Web Services,使用 ADO 访问数据等内容,并提供了大量的编写应用程序实例。

本书适合有经验的 VB 开发人员,使他们可以深入了解这些新功能,并充分利用 VB.NET 的功能来设计和编写实际的应用程序。

Billy Hollis, Rockford Lhotka: VB.NET Programming

EISBN: 1-861004-91-5

Copyright©2001 by Wrox Press Ltd.

Authorized translation from the English language edition published by Wrox Press Ltd.

All rights reserved For sale in the People's Republic of China only.

本书中文简体字版由清华大学出版社和英国乐思出版公司合作出版。未经出版者书面许可,不得以任何方式复制或抄袭本书的任何部分。

**版权所有,翻印必究。**

**本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。**

书 名: VB.NET 程序设计教程

作 者: Billy Hollis Rockford Lhotka 等著 康博 译

出 版 者: 清华大学出版社(北京清华大学学研大厦,邮政编码:100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑: 宋方 杨海儿

印 刷 者: 世界知识印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787 × 1092 1/16 印张: 25 字数: 593 千字

版 次: 2001 年 10 月第 1 版 2001 年 10 月第 1 次印刷

书 号: ISBN 7-302-04878-9/TP·2897

印 数: 0001 ~ 5000

定 价: 40.00 元

# 前 言

Microsoft 已经将它的未来与 .NET Framework 紧密联系在一起,VB.NET 很可能会成为未来几年中基于此框架且应用最广泛的开发工具。这意味着成千上万的 Visual Basic 开发人员需要变换开发工具,而 VB.NET 是最主要的一种工具。VB 开发人员需要迅速掌握 VB.NET 的主要内容,而本书正是为这种需要而编写的。

我们写此书的目的是告诉有经验的 VB 开发人员有效使用 VB.NET 应该掌握的重要知识。熟练的 VB 开发人员会对 VB.NET 中的许多新技术也非常感兴趣,这些新技术包括:

- 完全继承
- 其他新的对象功能,如参数化的构造函数和共享成员
- 结构化的错误处理
- 新的线程模型

## 本书简介

首先向读者介绍了基本的 VS.NET 环境,它为所有 .NET 开发人员提供了一系列强大的功能。前两章主要介绍 .NET Framework,详细论述它的开发情况以及 VS.NET IDE 中的一些特性。第 3 章讲述 VB.NET 中的新特性,重点强调了名称空间的使用和完全继承,以及错误处理和无用存储单元收集例程的变化。

第 4 章的内容包括 Win Forms,如何将旧代码移入 VB.NET 环境,并举例说明 VB6 开发人员没用过的一些新功能。

第 5 章进一步阐述了适用于 VB.NET 开发人员的面向对象的新功能,并详细分析了继承性及其在应用程序中的使用。

第 6 章介绍提供给 VB 开发人员的另一个新功能 Web Forms。首先讲解 Web Forms 的结构和创建方法,然后举一个实例讲解 Web Services 及其安装方法。

第 7 章讨论如何使用 ADO 访问数据,并从客户机和服务器的角度详细阐明了 SOAP 协议和 XML 的使用方法。

第 8 章主要讨论高级编程技巧,包括在 VB.NET 中创建中间层组件,在 VB.NET 中执行线程、编写控制台应用程序、安装、创建 Windows 服务程序、监视文件系统以及从命令行执行操作的方法。

第 9 章讲述 VB 的交互功能以及如何迁移到 VB.NET,这包括如何从 .NET 中使用 COM 对象,调用 Windows API 和使用 Migration 工具。

第 10 章介绍了两个开发实例——一个是创建 Web Forms Payment Calculator,另一个是在 SQL Server 中为 pub 数据库中 Titles 表创建一个查询屏幕。这两个实例应用程序都使用了本书前面提到的技术和工具,并能够巩固以前学过的知识。

第 11 章是本书的结尾部分,为读者提供了获得更多 VB.NET 资源的方法,以及目前可以采用,保证工程能够顺利地从 VB 演化到 VB.NET 的步骤。

## 本书的读者

本书介绍的是一个还在开发阶段的产品,主要针对的是有一定 VB、ASP 和 VBScript 经验的开发人员。这不是一本 VB 语言的学习教程,而是对 VB6 和 VB.NET 的编程技术进行比较——因此要想充分利用本书,熟悉 VB6(至少是 VBA 或 VBScript)是非常必要的。

本书主要面向有熟练经验的 VB6 开发人员,他们想了解未来的 .NET 平台,并且想知道如何迅速、高效地使用 VB.NET。

由于 VB.NET 定位为一种有潜力的网络开发工具,因此本书也适合于那些使用 ASP.NET(以前称为 ASP+)以及打算使用 VB 语言开发 ASP.NET 站点的网络开发人员。ASP.NET 与 VB.NET 的集成程度远远超过了 ASP 和 VB6 的集成。全面掌握 VB.NET 对于任何一个用 ASP.NET 进行网络开发的人来说都将受益匪浅。

事实上,由 Microsoft 推出的、应用于 ASP 开发的 Visual Interdev 开发工具已经被 VS.NET 环境代替。在 .NET 环境中,与 Visual Interdev 具有相同功能的是 VS.NET IDE。

.NET Framework 提供了一种强大的、适合于 Windows 和 Internet 的新编程方法,这种方法会将使用其他语言和平台的有经验的开发人员吸引到 .NET 平台和 VB.NET 语言上。本书将告诉他们,由于书中给出了许多编程实例和技术,因此能够迅速地从其他平台和语言过渡到 VB.NET。

## 使用本书的要求

在本书的第 1 章给出了运行全部样例代码所需的软硬件,大致地说,至少应具备以下条件:

- Pentium II, 450MHz 的计算机, 128MB 的 RAM 和 3GB 的硬盘空间。
- Windows 2000
- Internet Explorer 5.5
- IIS
- SQL Server 2000

当然,也可以从网上下载 VS.NET beta 1,网址是

<http://msdn.microsoft.com/net>。

## 欢迎读者反馈意见

我们努力使本书对读者有很大用处,并希望知道读者的意见。一直渴望知道读者想知道什么以及需要知道什么。

非常感谢读者对我们的工作所提供的任何反馈意见,在未来的编写过程中我们还要广泛吸纳批评和表扬意见。如果读者有何建议,请通过 e-mail 与我们联系: [feedback @ wrox.com](mailto:feedback@wrox.com) 或者将反馈意见链接到我们的 web 站点: <http://www.wrox.com>。

# 目 录

|                                             |    |
|---------------------------------------------|----|
| 第 1 章 开篇 .....                              | 1  |
| 1.1 各主题的 80/20 法则 .....                     | 2  |
| 1.2 VB.NET 中的变化及本书内容安排 .....                | 2  |
| 1.3 VB.NET 测试版所需安装的软件 .....                 | 4  |
| 1.3.1 软件的获得 .....                           | 5  |
| 1.3.2 环境要求 .....                            | 6  |
| 1.3.3 准备安装 .....                            | 8  |
| 1.3.4 安装 Visual Studio.NET .....            | 10 |
| 1.3.5 配置远程 IIS 服务器 .....                    | 21 |
| 1.4 确保软件正常运行——Hello World 程序 .....          | 22 |
| 1.5 小结 .....                                | 27 |
| 第 2 章 .NET Framework 入门 .....               | 28 |
| 2.1 一个适应未来的应用范围广、功能强大的开发平台 .....            | 28 |
| 2.2 一个重要的警告 .....                           | 29 |
| 2.3 避免混淆——.NET Enterprise Servers 的作用 ..... | 29 |
| 2.4 我们现在拥有的工具所存在的问题 .....                   | 30 |
| 2.4.1 显示层 .....                             | 30 |
| 2.4.2 中间层 .....                             | 31 |
| 2.4.3 数据层 .....                             | 32 |
| 2.4.4 有关 DNA 模型的讨论 .....                    | 32 |
| 2.5 .NET 的起源 .....                          | 33 |
| 2.6 .NET Framework 简介 .....                 | 34 |
| 2.7 公用语言运行时 .....                           | 35 |
| 2.7.1 主要的设计意图 .....                         | 35 |
| 2.7.2 元数据 .....                             | 38 |
| 2.7.3 多语言集成和支持 .....                        | 39 |
| 2.7.4 公共类系统 .....                           | 40 |
| 2.7.5 名称空间 .....                            | 42 |
| 2.7.6 .NET Framework 中应用程序的结构 .....         | 43 |
| 2.7.7 安装和执行 .....                           | 43 |
| 2.8 下一层——.NET Class Framework .....         | 47 |
| 2.9 用户和程序界面 .....                           | 49 |
| 2.9.1 用户界面 .....                            | 49 |

|                                         |           |
|-----------------------------------------|-----------|
| 2.9.2 编程界面 .....                        | 51        |
| 2.10 作为.NET“元语言”的 XML .....             | 52        |
| 2.11 COM 的作用 .....                      | 53        |
| 2.12 DNA 的作用 .....                      | 53        |
| 2.13 其他优点 .....                         | 53        |
| 2.13.1 更少出错——整个错误类有望消失 .....            | 54        |
| 2.13.2 更好的性能 .....                      | 54        |
| 2.13.3 一些潜在的缺点 .....                    | 54        |
| 2.14 第一步——Visual Studio.NET .....       | 55        |
| 2.14.1 所有语言的公共 IDE .....                | 55        |
| 2.14.2 公用语言规范 .....                     | 55        |
| 2.14.3 多语言工程的管理 .....                   | 56        |
| 2.14.4 Visual Interdev 的出现 .....        | 56        |
| 2.15 对 Visual Basic 的影响 .....           | 56        |
| 2.16 .NET 对我们的影响 .....                  | 57        |
| 2.16.1 一系列编程模块 .....                    | 57        |
| 2.16.2 减少 Internet 开发的障碍 .....          | 57        |
| 2.16.3 预先编写的功能库 .....                   | 58        |
| 2.16.4 更简单的部署 .....                     | 58        |
| 2.17 警告 .....                           | 58        |
| 2.18 小结 .....                           | 58        |
| <b>第3章 IDE 和 VB.NET 的语言特性 .....</b>     | <b>59</b> |
| 3.1 Hello World 应用程序 .....              | 59        |
| 3.1.1 声明 .....                          | 60        |
| 3.1.2 声明、初始化和终止 .....                   | 61        |
| 3.1.3 Windows Form Designer 生成的代码 ..... | 65        |
| 3.1.4 事件处理代码 .....                      | 68        |
| 3.2 新的 IDE 特性 .....                     | 70        |
| 3.2.1 主页 .....                          | 70        |
| 3.2.2 窗口的定位 .....                       | 71        |
| 3.2.3 主窗口 .....                         | 72        |
| 3.2.4 对宏的支持 .....                       | 83        |
| 3.2.5 集成化的调试 .....                      | 83        |
| 3.3 语言和语法方面的改变和特性 .....                 | 85        |
| 3.3.1 名称空间 .....                        | 85        |
| 3.3.2 语言和语法上的改变 .....                   | 90        |
| 3.3.3 委托 .....                          | 118       |

|                                         |     |
|-----------------------------------------|-----|
| 3.3.4 属性 .....                          | 119 |
| 3.4 小结 .....                            | 120 |
| 第4章 VB.NET 新的 Windows UI 功能 .....       | 121 |
| 4.1 Windows Forms .....                 | 121 |
| 4.2 Windows Forms 的重要性 .....            | 121 |
| 4.3 Windows Forms 基础 .....              | 122 |
| 4.4 Windows Forms 的结构 .....             | 123 |
| 4.4.1 Control 类 .....                   | 124 |
| 4.4.2 RichControl 类 .....               | 125 |
| 4.4.3 ScrollableControl 类 .....         | 126 |
| 4.4.4 ContainerControl 类 .....          | 126 |
| 4.4.5 Form 类 .....                      | 127 |
| 4.4.6 UserControl 类 .....               | 129 |
| 4.4.7 与 Web Forms 保持一致 .....            | 129 |
| 4.5 Windows Forms 与 VB6 窗体的比较 .....     | 129 |
| 4.5.1 使用 Windows Forms 的可视化设计器的区别 ..... | 129 |
| 4.5.2 对话框的区别 .....                      | 130 |
| 4.5.3 从属窗体 .....                        | 134 |
| 4.5.4 Cancel 按钮和 Default 按钮的窗体属性 .....  | 135 |
| 4.5.5 窗体和控件在定位和布局上的区别 .....             | 136 |
| 4.5.6 控件的新属性 .....                      | 137 |
| 4.5.7 运行期间添加新控件 .....                   | 139 |
| 4.6 重要控件小结 .....                        | 140 |
| 4.6.1 控件的层次结构 .....                     | 140 |
| 4.6.2 与 VB6 的控件对应的 VB.NET 控件 .....      | 140 |
| 4.6.3 新控件 .....                         | 141 |
| 4.6.4 现有控件的改进 .....                     | 149 |
| 4.7 多文档界面(MDI)窗体的区别 .....               | 150 |
| 4.7.1 创建 MDI 父窗体 .....                  | 150 |
| 4.7.2 MDI 父窗体的区别 .....                  | 150 |
| 4.7.3 MDI 子窗体的区别 .....                  | 151 |
| 4.7.4 在 MDI 父窗体中排列子窗口 .....             | 151 |
| 4.7.5 VB.NET 中的一个 MDI 例子 .....          | 151 |
| 4.8 使用定制的属性和方法 .....                    | 152 |
| 4.9 Windows Forms 的可视化继承 .....          | 153 |
| 4.10 GDI+ .....                         | 157 |
| 4.10.1 System.Drawing 名称空间 .....        | 158 |

|              |                                      |            |
|--------------|--------------------------------------|------------|
| 4.10.2       | 在 Windows Form 中使用 GDI+ 功能 .....     | 158        |
| 4.10.3       | System.Drawing.Drawing 2D 名称空间 ..... | 161        |
| 4.10.4       | System.Drawing.Imaging 名称空间 .....    | 161        |
| 4.10.5       | System.Drawing.Text 名称空间 .....       | 162        |
| 4.10.6       | 对 VB6 及其早期版本的修改 .....                | 162        |
| 4.11         | 创建定制的 Windows Forms 控件 .....         | 163        |
| 4.12         | 深层的内容 .....                          | 165        |
| 4.13         | 小结 .....                             | 166        |
| <b>第 5 章</b> | <b>VB.NET 的新面向对象功能 .....</b>         | <b>167</b> |
| 5.1          | 面向对象和面向组件合并的概念 .....                 | 168        |
| 5.2          | VB.NET OO 的实现 .....                  | 168        |
| 5.2.1        | 创建类 .....                            | 169        |
| 5.2.2        | 对象的生存周期 .....                        | 174        |
| 5.2.3        | 继承性 .....                            | 179        |
| 5.2.4        | 共享或类成员 .....                         | 193        |
| 5.2.5        | 事件 .....                             | 195        |
| 5.3          | 接口 .....                             | 199        |
| 5.4          | 与对象交互 .....                          | 202        |
| 5.4.1        | 对象的声明和实例化 .....                      | 202        |
| 5.4.2        | 终止对象 .....                           | 204        |
| 5.4.3        | 前期和后期的绑定 .....                       | 204        |
| 5.5          | 跨语言的继承性 .....                        | 206        |
| 5.5.1        | 创建 VB.NET 基类 .....                   | 207        |
| 5.5.2        | 创建 C# 子类 .....                       | 207        |
| 5.5.3        | 创建客户应用程序 .....                       | 208        |
| 5.6          | 可视化的继承 .....                         | 209        |
| 5.7          | 小结 .....                             | 209        |
| <b>第 6 章</b> | <b>VB.NET 的新 Web 功能 .....</b>        | <b>211</b> |
| 6.1          | .NET 的接口层 .....                      | 211        |
| 6.2          | Web 开发的预备知识 .....                    | 211        |
| 6.2.1        | HTML .....                           | 212        |
| 6.2.2        | DHTML .....                          | 213        |
| 6.2.3        | 浏览器特性的总结 .....                       | 213        |
| 6.2.4        | XML .....                            | 214        |
| 6.2.5        | Active Server Pages .....            | 214        |
| 6.3          | Active Server Pages 的问题 .....        | 215        |

|                                                |            |
|------------------------------------------------|------------|
| 6.4 Web Forms 概述 .....                         | 216        |
| 6.4.1 活动的 Web Forms .....                      | 216        |
| 6.4.2 ASP.NET 作为运行时引擎 .....                    | 219        |
| 6.4.3 Web Form 的剖析 .....                       | 220        |
| 6.4.4 Web Forms 事件模块 .....                     | 225        |
| 6.5 服务器控件 .....                                | 228        |
| 6.5.1 需要服务器控件的原因 .....                         | 228        |
| 6.5.2 HTML 服务器控件 .....                         | 229        |
| 6.5.3 ASP.NET 服务器控件 .....                      | 230        |
| 6.5.4 验证控件 .....                               | 232        |
| 6.5.5 定制 Web Form 控件 .....                     | 234        |
| 6.6 Web Forms 中的数据绑定 .....                     | 237        |
| 6.7 最后一个例子——使用 Web Forms 的小应用程序 .....          | 237        |
| 6.7.1 第一步——初始化工程 .....                         | 237        |
| 6.7.2 第二步——把控件放在 Web Form 上 .....              | 237        |
| 6.7.3 第三步——放置一个标签来显示窗体的结果 .....                | 238        |
| 6.7.4 第四步——在窗体上放置验证控件 .....                    | 238        |
| 6.7.5 第五步——在按钮的 Click 事件中插入代码 .....            | 239        |
| 6.8 把现有的 VB 应用程序转换为 Web Forms .....            | 241        |
| 6.8.1 典型的设计方案 .....                            | 242        |
| 6.8.2 降低对事件的依赖 .....                           | 242        |
| 6.8.3 把代码移到中间层 .....                           | 242        |
| 6.9 Web Services 概述 .....                      | 242        |
| 6.9.1 Web Services 的用途 .....                   | 243        |
| 6.9.2 理解 SOAP 协议 .....                         | 243        |
| 6.9.3 在 VB.NET 中创建 Web Service .....           | 244        |
| 6.9.4 在 Visual Basic.NET 中使用 Web Service ..... | 246        |
| 6.9.5 例子——创建和使用 Web Service .....              | 247        |
| 6.10 小结 .....                                  | 248        |
| <b>第 7 章 VB.NET 中的数据访问 .....</b>               | <b>250</b> |
| 7.1 从 ADO 到 ADO.NET 的改进 .....                  | 251        |
| 7.1.1 与 ADO 的比较 .....                          | 251        |
| 7.1.2 ADO 和 ADO.NET 的类似之处 .....                | 251        |
| 7.2 ADO.NET——重要的概念 .....                       | 252        |
| 7.2.1 ADO.NET 类的位置 .....                       | 252        |
| 7.2.2 替换 Recordsets .....                      | 252        |
| 7.2.3 对 DataSet 的说明? .....                     | 252        |

|       |                                           |     |
|-------|-------------------------------------------|-----|
| 7.2.4 | 在 VB6 中使用 Recordset 处理相关数据 .....          | 253 |
| 7.2.5 | DataSet 如何提供更高的灵活性 .....                  | 254 |
| 7.2.6 | ADO.NET 中的数据流概述 .....                     | 254 |
| 7.3   | 管理支持程序 .....                              | 256 |
| 7.3.1 | 当前可用的支持程序 .....                           | 256 |
| 7.3.2 | 由管理支持程序执行的类 .....                         | 257 |
| 7.4   | DataSet 的结构 .....                         | 261 |
| 7.4.1 | DataTable .....                           | 262 |
| 7.4.2 | DataColumn .....                          | 263 |
| 7.4.3 | DataRow .....                             | 263 |
| 7.4.4 | 约束 .....                                  | 265 |
| 7.4.5 | DataRelation .....                        | 265 |
| 7.5   | DataSet 的处理 .....                         | 266 |
| 7.5.1 | 手工创建 DataSet .....                        | 267 |
| 7.5.2 | 由数据库创建 DataSet .....                      | 268 |
| 7.6   | 使用 DataView .....                         | 276 |
| 7.6.1 | 在 DataView 中得到数据的一个子集 .....               | 277 |
| 7.6.2 | 给 DataView 排序 .....                       | 277 |
| 7.7   | Typed DataSet .....                       | 278 |
| 7.8   | ADO.NET 的 Exception 类 .....               | 280 |
| 7.9   | 通过 ADO.NET 访问 XML .....                   | 281 |
| 7.10  | 通过 DOM 访问 XML .....                       | 283 |
| 7.11  | 小结 .....                                  | 283 |
| 第 8 章 | 高级论题 .....                                | 284 |
| 8.1   | 中间层组件 .....                               | 284 |
| 8.1.1 | 类库和 DLL .....                             | 285 |
| 8.1.2 | Web Services .....                        | 296 |
| 8.1.3 | 事务组件 .....                                | 296 |
| 8.1.4 | 使用 Microsoft Message Queue (MSMQ) .....   | 300 |
| 8.2   | .NET 线程 .....                             | 304 |
| 8.3   | 控制台应用程序 .....                             | 315 |
| 8.4   | 新打印模式 .....                               | 317 |
| 8.4.1 | PrintDocument 和相关的对象 .....                | 317 |
| 8.4.2 | 打印的实现 .....                               | 318 |
| 8.5   | .NET 中的安装 .....                           | 324 |
| 8.6   | 使用 VB 创建一个 Windows NT/Win2K Service ..... | 326 |
| 8.6.1 | 创建 Windows 服务程序 .....                     | 326 |

|                                          |            |
|------------------------------------------|------------|
| 8.6.2 安装服务程序 .....                       | 329        |
| 8.7 监视文件系统 .....                         | 332        |
| 8.8 命令行选项 .....                          | 333        |
| 8.8.1 vbc 命令行参数 .....                    | 334        |
| 8.8.2 \ bin 目录中的工具概述 .....               | 335        |
| 8.9 小结 .....                             | 338        |
| <b>第 9 章 交互操作和迁移</b> .....               | <b>339</b> |
| 9.1 COM/DCOM/COM+ 的交互操作性 .....           | 339        |
| 9.1.1 在 .NET 中调用 COM 组件 .....            | 340        |
| 9.1.2 在 COM 上调用 .NET 装配件 .....           | 351        |
| 9.2 调用 Windows API .....                 | 355        |
| 9.3 使用迁移工具 .....                         | 360        |
| 9.4 小结 .....                             | 365        |
| <b>第 10 章 更多的 VB.NET 开发例子</b> .....      | <b>366</b> |
| 10.1 展开基于 Web 的付款计算器 .....               | 366        |
| 10.2 用继承来的类生成 SQL .....                  | 371        |
| 10.3 小结 .....                            | 381        |
| <b>第 11 章 总结</b> .....                   | <b>382</b> |
| 11.1 建议 .....                            | 382        |
| 11.1.1 目前 Visual Basic 工程的语法定约 .....     | 382        |
| 11.1.2 从条件中终止过程动作 .....                  | 383        |
| 11.1.3 终止隐含对象的实例化 .....                  | 383        |
| 11.1.4 迁移到 Visual Basic.NET 中的后备选项 ..... | 384        |
| 11.2 Visual Basic 迁移工具的局限性 .....         | 384        |
| 11.2.1 处理 DHTML 页面和 WebClasses .....     | 385        |
| 11.2.2 处理数据 .....                        | 385        |
| 11.2.3 转换后期绑定对象的问题 .....                 | 385        |
| 11.2.4 使用内部命名的常数来代替值 .....               | 385        |
| 11.2.5 结构中固定长度的字符串的问题 .....              | 386        |
| 11.3 从理解概念开始 .....                       | 386        |
| 11.3.1 开始使用 VB6 的 SOAP 工具箱 .....         | 387        |
| 11.3.2 理解继承性和相关对象的概念 .....               | 387        |
| 11.4 资源 .....                            | 387        |
| 11.5 致谢 .....                            | 388        |

# 第 1 章 开 篇

在 2000 年 7 月的 PDC(Professional Developers Conference)会议上,微软宣布启动 .NET——这是微软技术方向的重大转变,对于那些使用微软工具开发软件的人来说更是一个重大转折。微软将它的未来与 .NET Framework 紧密联系在一起,从当前以 COM 为基础的世界向一个分布性更强、更开放的动态环境转变。

目前,VB 是世界上应用最广泛的编程语言。显然,微软以 Visual Basic .NET 的形式使 VB 向 .NET Framework 转变。当 VB 开发人员转入 .NET 环境时,VB.NET 对于他们来说就是一种很自然的转变。在未来几年中,它很有可能成为在 Framework 中用得最广泛的开发工具。

微软也引入了新的 C# (读作“C sharp”)语言,并进行了大量的宣传和包装。毋庸置疑,会有许多开发人员使用 C#。但对于一般 VB 开发人员来说,转而使用 VB.NET 是很自然的,因为它保留了 VB 目前的基本语言结构和语法。

.NET Framework 与现有的开发环境相比有了很大的改变,VB.NET 与以前版本的 VB 也存在一些本质的差异。同时,现在大多数 VB 开发人员已经掌握了足够的知识,无需再熟悉新的语法规则和语言结构,如 C#。

或许更重要的是,VB.NET 和 C# 语言具有相同的基本功能。客观上讲,谁也没有明显的优势。C# 提供了一些 VB.NET 不具备的特性(如操作符重载),而 VB.NET 也提供了一些 C# 不具备的特性(如后期绑定语法)。在选择 VB.NET 和 C# 时,开发者应根据哪种语言学习起来更容易,哪种语言让人觉得更愉快,效率更高两个方面来考虑。

.NET 产品已经开发三年多了。部分产品已经面市,如 COOL、NGWS (Next Generation Windows Services)、Windows DNA、COM + 2.0 和 Visual Studio 7.0。这些产品已经合并到 .NET 旗下,因此具有一致性。

此外,.NET 环境建立在大量现有软件的基础上,包括 Windows 32 位操作系统、COM、COM +、MSMQ,以及旧版本的 Visual Studio。.NET Framework 也建立在大量新的标准或现有的标准上,如 HTML、CSS、XML 和 SOAP。

在引入了创建 32 位应用程序、类和 COM 控件的功能后,.NET 就无可争议地成为 VB4 以来对 VB 系列产品冲击最猛烈的产品。事实上,.NET 是一种更大的、涉及面更广的变革,因为它提供了一种全新的内部对象交互机制,这种机制不直接依赖于 COM。

这就意味着成百上千的 VB 开发人员将需要从 VB6 转到 VB.NET,这是一个大的飞跃。本书通过详细讲述以下内容,帮助开发人员顺利实现这一转变:

- 探讨 VB.NET 的新功能
- 举例说明不同于 VB 以前版本的重大改变和区别
- 最重要的是,本书提供了一些有经验的开发人员对这些变化的理解,必要时将给出

背景知识。

VB.NET 和 .NET Framework 是近几年微软推出的、最令人兴奋的、最具革命性的技术。希望通过本书,开发人员也能享受到这种快乐,并帮助开发人员迅速适应这个新环境。

## 1.1 各主题的 80/20 法则

对于微软开发人员来说,.NET 代表了基本平台的改变。大多数面向组件的概念从 COM 转向 .NET 时,在新的环境中还会有许多重大的改变和内容的增强。

现在开发人员可以与 Win32 API 进行交互,而在 .NET 中大多数交互式操作要用到 .NET 系统类库。事实上,任何 .NET 程序都要利用此系统类库进行操作,因此任何一个想在 .NET 环境中工作的开发人员都要牢牢掌握系统类库。

在这些平台的改变之上的是 VB 本身的改变。从功能的角度来看,现在我们有继承和自由线程功能。除了功能上的改变外,许多语法和编码也发生了改变,这将对语言日常的使用产生影响。

VB IDE 已经和其他所有的 .NET 开发工具一起集成到 Visual Studio.NET IDE 中,这个集成的 IDE 在某些方面与 VB6 IDE 类似,但有一些根本区别。

事实上,其中的变化非常多,一本书无法全部涵盖。我们不打算讲解每一个变化,而是选用了由 Pareto 提出的“80-20”法则,即“用小的投入产出大的效果”——对于我们而言,就是如果讲解 80% 的功能,就提供了一种在大多数情况下都能用得上的资源。在本书中,我们列出了开发人员会遇到的最常见的变化,重点讲解 IDE 的变化、语法的变化以及最重要的新特性。我们也会在一定程度上涉及 .NET 系统类库的内容,但只讨论目前 VB 开发人员最常遇到的类。

## 1.2 VB.NET 中的变化及本书内容安排

在 VB.NET 中会涉及很多新技术,它对有经验的 VB 开发人员来说非常有价值,如:

- 完全的面向对象(OO)的功能。
- 代码继承
- 方法/操作符重载
- 参数化的构造函数
- 共享成员
- 结构化的错误(例外)处理
- 新的线程模型

这些技术都基于微软的 .NET Framework,第 2 章将讲述 .NET,重点讲解对于 VB 开发人员很重要的细节。以上所讲的新技术也会涉及到,必要时还会把这些新技术与 .NET Frame-

work 联系起来。

.NET Framework 提供了人们期盼已久的新功能,但从旧版 VB 向 VB.NET 的迁移也出现了障碍,其中有很多语法上的差别和其他不兼容的方面。微软的 .NET Framework 要求对 VB 进行重大的结构改变——包括使语法和数据类型合理化,让 VB 与其他语言保持一致,还要从传统的 VB 窗体转向 WinForms。开发人员必须了解这些棘手的问题,才能迅速、有效地实现向 VB.NET 的飞跃。以上是第 3 章的主要内容,它讲述了新的 VS.NET IDE 的主要轮廓,并探讨了语法和编码方面的变化。

多年以来,VB 开发人员一直在使用 Ruby 窗体引擎工作。这种窗体环境可以使 VB 开发人员迅速、轻松地建立带有 Windows 用户界面的应用程序。.NET Framework 提供一种通用的窗体环境 WinForms;VB.NET 使用它而不是 Ruby。而 WinForms 提供的 Windows GUI 功能与旧式 VB 窗体引擎一样强大,这是一个全新的环境,提供了许多新功能和新方法去做我们已经做了多年的事情。第 4 章重点讲述这个新环境,演示如何创建常用的 GUI 界面,以及 Winforms 令人激动的一些新功能。

自从 VB 4.0 版推出以来,VB 开发人员就能用类模块编写面向对象程序,建立可重复使用的 COM 组件。从那时起,人们就发现:缺乏一个完全的面向对象环境——特别是缺少继承性——这是在 VB 中创建复杂系统时一个重要的制约因素。所谓继承性,就是创建一个新类,它继承了现有类的属性、方法和代码。

VB.NET 通过提供完全的面向对象功能,包括继承、方法的重复加载,方法的忽略等,正面致力于这个问题。这些功能完全都集成在 .NET 环境中,对 VB 开发人员来说也有很大的好处。以上新功能将在第 5 章中讨论,并举例说明 VB 开发人员应如何利用面向对象编程的所有优点。

除了改变传统的 VB 开发技术外,VB.NET 还向成千上万没有利用 Active Server Pages 和相关技术的开发人员提供了网络编程环境。第 6 章简述了 ASP.NET 环境——用 VB.NET 开发基于 Web 的屏幕和软件与用以前版本的 VB 开发 Windows 屏幕和软件是非常类似的。ASP.NET 使 VB 开发人员更容易开发网络系统。

VB.NET 中简化迁移的关键技术是 Web Forms 和 Web Services。虽然这些技术的许多特性对于 VB 开发人员来说都很熟悉,但还存在一些显著的差别和应避免的陷阱,例如,当与服务器进行交互时过度使用事件的问题。ASP.NET 本身就是一个很大的题目,但第 6 章可以使读者从 VB 开发人员的角度理解一些概念。

多年以来微软开发了一系列数据访问技术,它们都适用于 VB 开发人员。我们已经用过 DAO,接下来是 RDO,最近又推出了多种版本的 ADO。.NET 附带了一个全新的数据访问技术,称为 ADO.NET(以前叫 ADO+)。通过交互操作机制,使 VB.NET 程序仍可以用旧的数据访问技术成为一种可能,但大多数 VB.NET 开发人员很可能会使用 ADO.NET,因为它提供了一种最直接、最简便的方法。第 7 章主要阐述了 ADO.NET,并举例说明如何用这个新技术读取、创建、更新和删除数据。

整个 .NET 平台(包括 VB.NET)都非常依赖于 XML 技术。第 7 章讲述开发人员如何在 VB.NET 中使用 XML,或者通过 ADO.NET 或者直接从 VB 中使用 XML。

.NET Framework 的一个主要目标是简化安装,并删除“DLL Hell”(见第 8 章,它是令 VB 开发人员最头疼的问题之一)。VB.NET 不仅解决了大多数“DLL Hell”问题,还提供了 Visual Basic 以前附带的、强大而全面的安装技术——这较之以前的 Setup Wizard 或者 Package and Deployment Wizard 技术前进了一大步。

第 8 章还会涉及到一些更高级的技术,如创建中间层和事务处理组件、创建控制台应用程序(VB.NET 可以很容易与标准的控制台输入/输出机制 stdin 和 stdout 进行交互)以及 .NET Framework 附带的一系列强大的外部工具——包括 VB 编译器本身(该程序可以根据需要在命令行上直接运行)。我们也会看到其他一些内容,如用 C# 进行交叉语言开发,如何编写程序监视文件系统的变化,以及如何用 VB 创建 Windows NT 服务器。

很明显,大多数开发人员都曾编写过基于 COM 的基本应用程序,或者使用过其他技术如 ADO。第 9 章讲解交互操作性和升级,说明 VB.NET 程序如何利用现有的 COM 组件,或根据需要在 COM 组件中使用 VB.NET 程序。该章还简单地介绍了微软的升级向导。此向导会帮助将现有的 VB6 应用程序升级到 VB.NET。

第 10 章介绍如何在 VB.NET 中建立一个样例应用程序。我们将用 WinForms 技术创建一个带有标准 Windows GUI 用户界面的应用程序。然后介绍如何创建一个基于 Web Forms 和 ASP.NET 的 Web 界面。通过这个应用程序,将向读者介绍继承和其他面向对象特性的用法,以及 ADO.NET、Web Services 和其他 .NET 技术。

本书的结尾部分为第 11 章,列出了学习更多的 .NET Framework 和 VB.NET 知识所应采取的步骤,并把程序从 VB6 顺利地转到新的环境。另外,本章还列出其他一些有价值的资源,以加速学习者的学习进程。

现在我们要回到第 1 章,介绍如何安装 Visual Studio.NET。

## 1.3 VB.NET 测试版所需安装的软件

.NET Framework 与 VB.NET 既相互独立又相互联系。.NET Framework 包括 CLR(Common Language Runtime 公用语言运行时刻)、系统类库和其他工具。而 VB.NET 是 Visual Studio .NET(VS.NET)包的一部分,是开发人员使用的一种开发工具和 IDE。

VB.NET 要求在微软的 .NET Framework 环境下运行,甚至包括 VB 编译器。也可以只用 .NET Framework SDK(Software Developer's Kit)进行 VB 开发工作,但这样做就意味着没有 IDE 及其附带的所有集成工具。但是对于那些习惯于在文本编辑器中工作的人来说——或者具有 ASP、Unix、Open VMS 编程背景的人——也可以不用 VS.NET 开发应用程序,而是完全依赖于 .NET Framework SDK。

但本书的意图主要是讲解创建 VB.NET 应用程序时如何使用 VS.NET。VS.NET 是一种非常强大的 IDE,它可以使开发过程更简单、更高效。创建一个 .NET 应用程序需要完成许多工作,我们可以手工去做,但用 VS.NET 效率会更高。

要建立一个完整的 VB.NET 测试平台,必须安装 VS.NET 和 .NET Framework SDK。有时

也需要升级操作系统和操作环境的组件,才能满足.NET环境和VS.NET的最低要求。

### 1.3.1 软件的获得

获得.NET Framework SDK和VS.NET的途径很多,可以免费下载或者花一点钱买一张CD。

有关订购CD的信息,请到网址<http://msdn.microsoft.com/net>上查询。

#### 下载软件

几乎任何人(受到出口限制)都可以从MSDN网站下载.NET Framework SDK,MSDN的网址是<http://msdn.microsoft.com/net>。这个站点还连接了下载VS.NET的网址,但这个网址只允许MSDN Universal的订阅者或者MCSP组织者下载。

#### ① NET Framework SDK

NET Framework SDK需单独下载,大约100MB,可以安装在任意一个Windows 32位的操作系统上,包括Windows 95。下载的文件是一个自解压的安装文件——只需在计算机上运行它,便可安装SDK。

注意.NET Framework SDK需要一些环境支持——在运行Framework安装程序前必须安装该软件。

#### 注意:

安装.NET Framework SDK最简单的方法就是只下载下面要讨论的WINUPD CD。此CD不仅收录了SDK,还包括升级需要的所有软件,以确保计算机能够满足SDK有效工作的最低要求。

#### ② VS.NET

下载VS.NET的网址是<http://msdn.microsoft.com/vstudio/nextgen/beta.asp>,这个文件更大、更复杂。如果没有CD,那么要分以下3步下载:

- CD1
- CD2
- WINUPD(Windows Component Update CD)

CD1,CD2和WINUPD CD分别是245MB,121MB和358MB(压缩格式)。

#### 注意:

WINUPD CD包含.NET Framework SDK,因此无需单独下载SDK。

这些文件都是压缩格式,解压缩后会形成目录结构用于创建安装盘。另外也可以更易于把压缩文件解包在目录结构,从硬盘直接安装。

#### 注意:

从硬盘安装时,CD1和CD2必须被解包在同一个目录中,不然安装程序无法找到CD2。

从光盘安装,要求把上面提到的3个压缩文件分别放到各自的目录中,然后用刻录机将它们分别刻到3张光盘上。