

Visual Basic .NET

应用程序和 组件开发

Building Applications
and Components with Visual Basic .NET

[美] Ted Pattison- Dr. Joe Hummel 著
韩江译

Microsoft
.net
Development
Series

“VB 已经变得比以往任何时候都更加强大，你需要一个像 Ted Pattison 这样的人的经验，来帮助你掌握 VB 的新功能，并在面向对象程序设计的曲径中找到自己的路。”

——Francesco Balena, 《Microsoft Visual Basic .NET 程序设计》的作者

.NET 技术大系

Building Applications and Components
with Visual Basic .NET

Visual Basic .NET
应用程序和组件开发

[美] Ted Pattison, Dr. Joe Hummel 著

韩江译

電子工業出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

Visual Basic .NET 是专为 .NET Framework 设计的一种新的程序设计语言, 掌握它就能利用 .NET Framework 的新特性来开发网络应用程序和可重用的组件库。全书共分 12 章, 前 3 章介绍 .NET Framework 体系结构的核心组成部分, 包括公共语言运行时、Framework 类库和公共类型系统等; 后 9 章详述了 Visual Basic 最新版本中所添加的新编程特性, 如类、对象、继承、接口、委托、事件、构造函数、异常处理、装配件、无用存储单元收集和 COM 互操作等。

本书主要面向那些已经掌握程序设计基本原理、并使用过 Visual Basic、C++ 或 Java 的中、高级程序员, 书中还提供了丰富的实例, 因此也适用于具有一定编程经验并了解面向对象程序设计相关概念的读者。

Simplified Chinese edition copyright © 2004 by PEARSON EDUCATION ASIA LIMITED and Publishing House of Electronics Industry. Building Applications and Components with Visual Basic.NET, First Edition, ISBN: 0-201-73495-8 by Ted Pattison, Copyright © 2004 by Pearson Education, Inc.

All Rights Reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison Wesley.

This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macau).

本书中文简体字翻译版由电子工业出版社和 Pearson Education 培生教育出版亚洲有限公司合作出版。未经出版者预先书面许可, 不得以任何方式复制或抄袭本书的任何部分。

本书封面贴有 Pearson Education 培生教育出版集团激光防伪标签, 无标签者不得销售。

版权贸易合同登记号 图字: 01-2003-8578

图书在版编目 (CIP) 数据

Visual Basic.NET 应用程序和组件开发 / (美) 帕蒂森 (Pattison, T.) 等著; 韩江译. —北京: 电子工业出版社, 2004.9

(.NET 技术大系)

ISBN 7-121-00259-0

I. V… II. ①帕… ②韩… III. BASIC 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2004) 第 084612 号

责任编辑: 顾慧芳

印刷: 北京智力达印刷公司印刷

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

经销: 各地新华书店

开本: 787×1092 1/16 印张: 23.5 字数: 564 千字

印次: 2004 年 9 月第 1 次印刷

印数: 4000 册 定价: 45.00 元

凡购买电子工业出版社的图书, 如有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系。联系电话: (010) 68279077。质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书所获赞誉

“Ted 对 .NET 的底层机制非常熟悉。作者对于事件和委托等领域的深入研究，使得这本书成为一本非常有用的教材。”

——Gregory A. Beamer, Microsoft MVP

“《Visual Basic .NET 应用程序和组件开发》一书的内容非常独特；我觉得其他书都不如这本书对 OOP 特性介绍得全面和深入。对于希望掌握 OOP 编程知识的 VB 开发者来说，本书无疑是一本最好的学习指南。”

——Darshan Singh, PerfectXML.com 总编

“总体而言，本书非常出色——清晰、简洁，同时也介绍了 Visual Basic .NET 开发者必须掌握的所有细节。”

——Gerard Frantz, 顾问

“Ted 在介绍各种 OOP 概念和 CLR 及其相关概念方面做得相当不错。无疑我从这本书中学到了很多我原本不知道的关于 CLR 的知识，这些知识对我来说非常有用。”

——Paul Vick, Microsoft 公司 Visual Basic .NET 技术总监

“当 Ted 的第一本书于 1998 年首次面世时，我立即冲到书店买了一本，带着哈利波特式的狂喜。通过那本书我掌握了有关 VB6、COM 和 MTS 的所有我需要的知识。时隔五年之后，现在我终于能够看到 Ted 的关于 VB.NET 的新书了（并且 Ted 已经成为我的好朋友）。Ted 在这本书中秉承了他的一贯风格：通过结合了在 DevelopMentor 多年的工作经验和对 .NET 的全面深入的了解，Ted 的这本新书与以前的书一样深入浅出、内容丰富，并且讲解精辟。如果你想学习 Visual Basic .NET，那么这本书正是你所需要的。最后引用一段我在亚马逊网站对本书所做的评价：‘谢谢你，Ted，你的这本有关 Visual Basic .NET 和 .NET Framework 的专著使我们获益匪浅。’”

——George Bullock, MSDN

随着你逐渐深入地了解 .NET Framework 和 Visual Basic .NET，你将会发现这本书的内容非常有价值。

——摘自 Rocky Lhotka 为本书所写的序言

谨以此书献给我亲爱的妻子 Amy 和我可爱的女儿 Sophie 和 Daisy。
你们永远是我的生命中惟一不可缺少的财富。

——Ted

谨以此书献给我的妻子 Beth，是她使我最终能够完成这项有意义的工作。

——Joe

关于作者

Ted Pattison 于 1985 年毕业于 Vanderbilt 大学，获得物理学学士学位。作为一名专业的录音工程师在洛杉矶工作 4 年之后，他参加了美国加州大学 Irvine 分校研究生课程的学习，并于 1990 年获得了 MBA 学位。

1990 年，Ted 进入了 QuickStart Technologies 公司，并在那里开始了他作为一名软件开发者和教育工作者的职业生涯。在 QuickStart 期间，Ted 使用 Visual Basic 和 SQL Server 编写了几个基于 LAN 的业务解决方案。他同时也创作并教授面向开发者的培训课程。Ted 于 1991 年完成了他的第一个关于 Microsoft Access 的培训教程，当时这个软件还在测试之中。从那以后，Ted 撰写了十几本针对专业软件开发者的培训教程。

1994 年，Ted 离开 QuickStart 并加入了 DevelopMentor。在 DevelopMentor 期间，Ted 负责撰写课程并进行课程培训。Ted 非常幸运地得到了 Don Box 的指导，并与 Bob Beauchemin、Keith Brown、Tim Ewald、Fritz Onion、Jason Masterman、Chris Sells 和 Mike Woodring 等多名非常有才能的作者一起工作。Ted 率先在 DevelopMentor 开设了 Visual Basic 课程，并编写了公司第一个 Java 培训教程。

在 2003 年夏天，Ted 与 Jason Masterman 共同创办了 Barracuda.NET，这是一家致力于帮助开发者和软件公司成功开发网络应用程序的公司。可以访问 <http://Barracuda.NET> 来了解 Barracuda.NET 能够给你的公司提供哪些服务。

作为一名作者，Ted 曾出版了销量极好的《COM+与 Visual Basic 6.0 分布式应用程序设计》一书，他还是 MSDN 杂志的长期撰稿人，主编该杂志的 Basic Instincts 专栏。Ted 还参加了著名的 Band on the Runtime，这个乐队致力于在软件业中逐渐灌输一些幽默感。Ted 在这个乐队中的伙伴包括 David Chappell、Don Box、George Bullock、George Shepherd、Sarah Shor 和 Francesco Balena。

Joe Hummel 博士是 Lake Forest 大学计算机科学系的助理教授和 DevelopMentor 公司的 .NET 培训师。他在美国加州大学 Irvine 分校获得了计算机科学博士学位，并从 1993 年开始从事有关 DevelopMentor 和 Visual Basic 的工作。

译者序

.NET Framework 是 Microsoft 公司于 2002 年初发布的一个全新的开发平台。在 .NET Framework 发布两年之际，我们非常荣幸地为读者献上由 Ted Pattison 和 Joe Hummel 共同编著的《Visual Basic .NET 应用程序和组件开发》一书。Ted Pattison 是 Microsoft 公司的一位非常优秀的培训师，也是一位非常成功的计算机图书作者。在本书的出版过程中，多位知名的计算机专家都毫不吝惜地给予了本书极高的评价。本书的译者都具有多年的编程经验，在翻译本书的过程中，我们对于作者对 Visual Basic .NET 和面向对象程序设计的深刻理解都深怀敬佩，而作者深入透彻的讲解也让我们自己获益良多。

本书首先介绍了 .NET Framework 体系结构的核心组成部分，包括公共语言运行时、Framework 类库和公共类型系统等，然后对 Visual Basic 的最新版本中所添加的新的编程特性进行了详细的介绍，如类、对象、继承、接口、委托、事件、构造函数、异常处理、装配、无用存储单元收集和 COM 互操作性等。正如作者所说，在介绍这些内容的过程中，作者并不是简单地告诉读者“应该怎么做”，而是对“为什么这样做”进行了深入的分析。也就是说，本书不止是一本传授技能的书，它的价值体现在作者在书中引入了很多编程的思想。

本书所面向的读者是那些已经掌握程序设计的基本原理，并使用过 Visual Basic、C++ 或 Java 的中、高级程序员。对于初学者来说，阅读本书会有一些困难。如果你具有一些编程的经验，并且了解面向对象程序设计的相关概念，则本书可以帮助你尽快掌握使用 Visual Basic.NET 进行程序设计所需的知识和各种特性。随着书中讲述内容的不断深入，所讲述主题的难度也越来越大，对读者理解能力的要求也越来越高。但这并不意味着本书难以理解，事实上本书所使用的语言平实易懂，书中还提供了丰富的实例，非常利于读者理解和巩固所学的内容。

本书主要由韩江、郭大权、王嫣翻译，全书由韩江统一校稿。在翻译过程中，译者力求准确地反映原著的内容，但由于水平有限，加上时间紧迫，书中难免有错误之处，望广大读者批评指正。

译者

Francesco Balena 为本书写的序言

假如 Ted Pattison 是一个产品，那么它将是所有广告公司的梦想。因为，他属于那种可以进行自我推销的产品：你根本不需要为它做促销，而是只需要让人们知道有这样一种产品存在就可以了。从这点上来说，如果序言的主要目的是为了说服本书潜在的购买者成为真正的读者，那么我想我并不需要写太多的内容，我要说的完全可以用一句话来概括：想学习 Visual Basic .NET 吗？买这本书吧。句号。

还想听听更多的建议？还没有拿定主意？好吧，如果是这样，我就对本书的内容和它的作者多做一些介绍，并在其中穿插一些我个人的回忆。

在上一个千年快结束时（时间并没有过去多久，但这样说更让人印象深刻）所召开的 VBITS 会议上，我第一次见到了 Ted，当时我们都在一些会议和全天的专家研讨会上发言。正如我无数次地对他说的那样，让我这个不太适应美国英语的人来应付他那像机关枪一样快的演讲，可不是一件容易的事。不过，我并没有错过参加他的会议，即使是我已经很熟悉他所讲的主题。我这样做的原因很简单：Ted 生来就具备做一名老师所需的才能，而他严谨的治学态度、恰当的代码示例以及详尽的解释说明都使他的这种天生的才能更加突出。因此，我总是对他的发言以及他的内容广泛的谈话非常感兴趣。

当我在阅读《COM 与 Visual Basic 6.0 分布式应用程序设计》(Programming Distributed Applications with COM and Visual Basic 6.0) 及其第二版（包括 COM+ 方面的内容）的时候，我并没有因为看到他的讲授才能如此适于写作而感到吃惊。这本书被许多希望离开 RAD 程序设计的安全港湾，转向面对企业级应用程序开发挑战的 Visual Basic 6.0 程序员作为经典著作摆在案头。

Ted 的所有演讲和书籍（包括你正在阅读的这本书）都有一个共同的特点：他并不力求使演讲或书籍的内容覆盖到所讲述语言、产品或技术的每一个关键字或每一个次要的细节。相反，他更注重为读者提供有关基本概念的坚实基础。现在 Visual Basic 有如此多的新特性——如继承、构造函数、属性、委托，等等，大多数程序员都对何时以及如何使用这些特性感到困惑，在这种情况下，Ted 的治学方法就显得更加有价值。

你不需要熟悉如何使用 Visual Basic 的早期版本进行面向对象的程序设计。事实上，使用 Visual Basic 的早期版本，你甚至一个类都不需要编写，就可以创建一个中等规模的业务应用程序。在实践中，Visual Basic 6.0 的类只是一种创建 COM 组件的方法，其语言本身并没有改进面向对象的程序设计过程。这种状况现在已经改变，如果你想开发稳定的、高效的、可定制的应用程序，就不能忽视对象的输入和输出。

我相信，这本书可以极大地帮助开发者进入 Visual Basic .NET 和 Microsoft .NET Framework 的新天地。VB 已经变得比以往任何时候都更加强大，你需要一个像 Ted Pattison

这样的人的经验，来帮助你掌握 VB 的新功能，并在面向对象程序设计的曲径中找到自己的路。

Francesco Balena
Microsoft MSDN 意大利区域经理
《Microsoft Visual Basic .NET 程序设计》的作者
VB-2-The-Max, www.vb2themax.com 的创始人

Rocky Lhotka 为本书写的序言

计算机业是循环往复的。我们在选择是采用瘦客户机终端及基于浏览器的界面，还是采用胖客户机界面之间不断往复。我们曾经从过程或模块设计发展到面向对象的程序设计，可是现在我们又回到了 Web 服务形式的过程设计。

但是，在这背后有一种趋势并不是循环往复的，即基于组件的程序设计方式。不管我们选择的是瘦客户机还是胖客户机，我们都使用组件来开发应用程序。在开发基于过程、基于面向对象或基于 Web 服务的应用程序时，都要使用组件。

组件是软件开发的通用元件。Visual Basic 和较早的 VBX 组件对上世纪 90 年代早期组件概念的普及发挥了很大的作用。而在这之后向 COM 和 ActiveX 组件的转换，使得组件的使用更加广泛，从 UI 窗口部件扩展到了各种预构建的库。

更重要的是，Visual Basic 4 使我们可以开发属于自己的组件。这是在 7 年之前，从那以后组件得到了广泛的使用。不仅是 Visual Basic，在 C++ 和 Java 程序设计中普遍使用了组件技术。

实际上，现在所有的应用程序都是由各种组件所组成的，这些组件互相配合以提供所需的功能。我们使用组件来开发 Windows 和 Web 用户界面、以管理中间层代码以及访问数据。如何创建 MTS 或 COM+ 代码呢？你可以创建一个组件。如何创建 Web 服务呢？你可以创建一个组件。

Microsoft 的“组件对象模型”（COM）技术在过去的几年中成为最受欢迎的基于组件的技术。COM 使基于组件的程序设计变得相对简单，尤其是在使用 Visual Basic 进行程序设计时。大多数开发者不仅可以使使用组件，而且还可以创建组件。

最近，Microsoft 开发了 .NET Framework。 .NET Framework 是一个强大的开发平台。它既包含 COM 的基于组件的特性，同时又克服了 COM 的局限性，特别是克服了部署和版本方面的局限性。

更重要的是，.NET Framework 将基于组件和面向对象的概念相融合，并使之成为一个完整的整体。在 COM 中组件之间不能相互继承，但是在 .NET 中经常会遇到在一个组件中继承另一个不同的组件中的类的情况。即使另一个组件是使用其他编程语言开发的，也可以进行继承。所以，现在不仅 Visual Basic .NET 是完全面向对象的，而且底层的开发平台也是面向对象的了。

当然，功能的增强也带来了一些复杂性。这也是作者编写本书的原因。

成功地使用 .NET Framework 和 Visual Basic .NET 的新功能的关键，在于理解这些功能是什么，以及如何实现这些功能。在本书中，Ted 非常成功地以清晰和系统的方式讲述了 Visual Basic .NET 的特性。

通过阅读本书，我学到了很多，并且使我对一些关键概念的理解得到了增强。这本书介绍了很多使用 Visual Basic .NET 和 .NET Framework 开发应用程序和组件所需要的基本知识。

随着你逐渐深入地了解 .NET Framework 和 Visual Basic .NET，你将会发现这本书的内容非常有价值。不管是对于 Windows 开发者和 Web 开发者，还是对于以数据为中心的开发者和面向对象的开发者，这本书都非常有用。无论你是要使用组件还是要创建组件，你都会发现 Visual Basic .NET 是一个功能强大并有趣的工具，我相信这本书对你充分使用 Visual Basic .NET 的功能将有很大的帮助。

Rocky Lhotka
Magenic Technologies

前 言

.NET Framework 是 Microsoft 公司于 2002 年初发布的一个开发平台。使用 .NET Framework 是非常具有战略性的,因为它使软件公司开发的软件能够应用 Microsoft 开发分布式应用程序的最新技术。这个平台还提供了一些功能,可以使应用程序在网络环境中的部署更简单,更经济,也更安全。

Visual Basic .NET 是专门为 .NET Framework 设计的一种新的程序设计语言。一旦你掌握了如何使用 Visual Basic .NET 来编写和测试代码,你就能够利用 .NET Framework 的很多新特性。而这些新特性也使你能够开发网络应用程序和可重用的组件库,这样就可以充分利用 Microsoft 所提供的资源。

对于使用过 Visual Basic 早期版本的开发者来说,会觉得 Visual Basic .NET 语言的一些部分看起来很熟悉。例如,Visual Basic .NET 使用与以前相同的语法来声明变量,并使用 If 语句和 While 循环等结构来控制程序执行的流程。因为 Visual Basic .NET 感觉很像 Visual Basic 的早期版本,所以 Visual Basic 开发者可以继续以与过去相同的风格来编写程序。版本之间的相似性也使得将代码从 Visual Basic 6 移植到 Visual Basic .NET 更容易。

虽然 Visual Basic .NET 在很多地方都与 Visual Basic 的早期版本很相似,但它们绝不是一回事。Visual Basic .NET 远比 Visual Basic 的任何早期版本功能强大,它们之间具有很大的差别。如果你要从 Visual Basic 6 转到 Visual Basic .NET,则需要学习很多新的知识,这是因为这种新的程序设计语言具有很多过去 Visual Basic 中所没有的编程特性。如果你要从 C++ 或 Java 等面向对象的程序设计语言转到 Visual Basic .NET,这个过程可能会相对简单一些,但是你仍然要学习很多细节。本书的目的就是为读者提供这些必须学习的细节。

本书的读者对象

《Visual Basic .NET 应用程序和组件开发》的目标读者是那些希望使用 Visual Basic .NET 开发 .NET Framework 应用程序和组件库的开发者。本书适用于具有 Visual Basic 或 VBA 的早期版本使用经验的开发者。它同样也适用于从 C++ 和 Java 转到 Visual Basic .NET 的开发者。

本书读者需要具有什么经验

本书的典型读者应该是具有 Visual Basic、C++ 或 Java 开发经验的中高级开发者。你应该已经知道如何创建包含域和方法的简单的类定义。你应该了解使用封装的重要性,以及如何将类成员定义为私有的。最后,你还应该理解类、对象以及对象引用之间的区别。

了解计算机科学基本知识也会对学习本书有所帮助（但这不是必需的）。如果不了解作用范围、生存期、调用堆栈、堆和线程的概念，那么要想深入地学习如何开发 .NET Framework 软件就会非常困难。如果你缺少这方面的知识，我建议你考虑先从基础学起。

掌握“组件对象模型”（COM）的知识也会对学习本书有所帮助（但这不是必需的）。COM 为 Visual Basic 6 开发者使用组件 DLL 开发应用程序提供了底层支持。理解 COM 可以使开发者更熟练地使用 Visual Basic 6。尽管 .NET Framework 使用一种新的更灵活的架构来代替 COM，但同时大部分基本规则仍然与 COM 相同。不过，有一些基本规则发生了变化，以使开发更加容易。你还必须了解 COM 以理解 Visual Basic .NET 和 Visual Basic 6 之间的互操作性。

本书的主要内容

对于大多数已经能够熟练使用 Visual Basic 6 的开发者来说，转到 Visual Basic .NET 仍需要付出很多努力。你必须学习一个全新的平台，并熟悉一组新的开发工具。你还必须学习 Visual Basic .NET 程序设计语言。这个过程中你将面对很多挑战，因为 Visual Basic .NET 与 Visual Basic 的任何早期版本都非常不同，它的功能也要强大得多。在转到 Visual Basic .NET 的过程中，你必须学习新的概念、新的关键字和新的语法。

本书的目的就是帮助你尽快地转到 Visual Basic .NET。本书将首先介绍 .NET Framework 体系结构的核心组成部分，然后介绍在 Visual Basic 的最新版本中所添加的新的编程特性。下面列出本书的主要内容：

- .NET Framework 和“公共语言运行时”（CLR）
- 开发 CLR 应用程序和组件库
- Framework 类库（FCL）
- 公共类型系统
- 新的面向对象程序设计（OOP）特性
- 支持新 OOP 特性所需的新的 Visual Basic 关键字和语法
- 在程序设计中委托和事件
- 使用结构化的异常处理
- 使用值和对象管理内存
- 命名、开发、部署和修改装配件 DLLs
- 与 COM 和 Visual Basic 6 进行互操作

本书没有涉及的内容

《Visual Basic .NET 应用程序和组件开发》没有包含太多的一步一步的操作指南。因此，对于那些只关心如何完成一个特定的任务，而不关心为什么这样做的读者来说，也许不会对本书感兴趣。尽管笔者理解读者对于学习“如何做”的需要，但是本书的目的是希望使你能够理解使用 .NET 和 Visual Basic .NET 时“为什么这样做”。

虽然本书主要介绍如何使用 Visual Basic .NET 进行面向对象的程序设计，但并不对有关面向对象设计（OOD）的问题进行深入讨论。你将学到很多新的 OOP 特性，新的语法要使用这些特性。不过在介绍这些内容时，本书将着重讲述语法，以及语法如何影响代码的行为。如果你想学习设计模式或 UML 建模等高级的 OOD 知识，则需要参考其他资料。

本书也不包含与创建分布式应用程序相关的内容。需要指出的是，本书中并不介绍如何使用 ASP.NET 或 .NET Remoting。尽管本书包含一些对于创建客户端和服务器应用程序的开发者来说非常重要的内容，但你还是需要参考其他资料来学习如何使你的应用程序可以在网络中进行通信。

如果你希望阅读一本包含大型应用程序实例的书以作为入门的教材，那么本书就不太适合你的要求。书中给出的大多数代码段都很短，一般介于 5 到 20 行之间。在需要列出代码的地方，作者会给出尽量少的代码，以使你的注意力集中在所讲述的问题上。而且，代码段中一般都会省略无关的内容（如错误处理等）。基于以上原因，本书的风格不适合那些希望可以直接将代码示例复制粘贴到应用程序中的读者。总之，本书的目的是教给你如何能钓到鱼，而不是简单地把鱼给你。

本书的在线支持

我为这本书开设了一个支持网站，网址是 <http://Barracuda.net/Publications/VBNET>。在这个网站上，你可以下载本书中出现的代码示例。下载之后你就可以方便地运行、测试和调试这些 Visual Basic .NET 代码，从而学习本书各章中讨论的关键问题和语法。我正在写本书的第二版，我也准备在这个网站上发表一些新的章节。

作为一个普通人，我离完美还差得很远。因此，本书的支持网站中将有一个勘误表，其中列出本书中的印刷错误和技术错误。如果你发现了本书中的错误并想告诉我，请给我发 e-mail: VBErrata@Barracuda.net。

致谢

一本有用的书很少是由一个人单独完成的。当然，在过去的两年里我也并不是独自在创作《Visual Basic .NET 应用程序和组件开发》。在此期间很多人都给予了我大量的帮助。

首先，我要感谢 Joe Hummel 博士与我一起从事这项工作，有了他的帮助，这本书的完成提前了好几个月（甚至更长）的时间。感谢 Joe 对书中技术错误的更正，对不恰当的表述方式的修改，以及对本书整体结构制定所给予我的帮助。希望将来我有机会与 Joe 博士在更多的项目上进行合作。

感谢 Francesco Balena 和 Rocky Lhotka 为我题写了本书的序言。我感到非常幸运能有这样的好朋友，他们是我生命中的奇迹。他们已经出版的书对 Visual Basic 的发展发挥了很大作用，我们期待着他们在将来给我们带来更多的作品。如果有机会，你应该听听 Francesco 演奏的萨克斯或 Rocky 讲的笑话。我保证你不会失望。

我还将感谢本书的坦率和积极的审阅小组。感谢 Greg Beamer、George Bullock、James

W. Cooper、Gerard Frantz、Martin Heller、Rocky Lhotka、Jason Masterman、Fritz Onion、Ethan Roberts、Darshan Singh、Doug Turnure、Jason Whittington、Jim Wilson、Mike Woodring 和 Paul Vick。我坚持认为具有批评精神的审阅人是一名技术作者最好的朋友。我的观点是，如果一个审阅人不能提出任何问题，那么他的话就没有价值。从这一点来看，我的审阅人没有让我失望。我对他们再次表示感谢。

感谢所有帮助我形成对软件开发和.NET Framework 的思考方法的业内人士。非常感谢 Don Box，在我的职业生涯之初他给了我莫大的帮助。感谢 Craig Andera、Bob Beauchemin、Niels Berglund、Scott Bloom、Keith Brown、Calvin Caldwell、Henk de Koning、Peter Drayton、Tim Ewald、Jon Flanders、Andrew Gayter、Justin Ghtland、Ken Getz、Ian Griffiths、Martin Gudgin、Stu Halloway、Simon Horrell、Paul Kirby、John Lam、Jose Mojica、Ted Neward、Jeff Prosis、Brian A.Randell、Brent Rector、Jeffrey Richter、Steve Rodgers、Dave Schmitt、Chris Sells、George Shepherd、Dan Sinclair 和 Aaron Skonnard。DevelopMentor、Microsoft 和 QuickStart 等公司中还有很多其他非常有才能的人值得我感谢。在过去的十年中能够得到他们的反馈并与他们交换意见，对于我来说是非常有价值的。

最后，我要感谢 Addison-Wesley 的工作人员，是他们不知疲倦地工作使这本书最终以出版。衷心感谢我的责任编辑 Stephane Thomas，她一直热心并积极负责这个项目，并总是骗她的老板说我的进度没有拖延。感谢 Amy Fleischer、Jill Hobbs 和 Michael Mullen 等其他几位编辑，他们为本书的出版做出了很多贡献。最后，我要感谢 John Montgomery、Martin Heller、Stacey Giard，以及 Microsoft 的其他员工，是他们与 Addison-Wesley 的共同努力使 Microsoft .NET 开发系列丛书成为一个整体。我觉得自己非常幸运，可以在这样好的一家公司工作。

Ted Pattison
Manhattan Beach, CA
2003 年 9 月

目 录

第 1 章 .NET Framework	1
1.1 .NET Framework 之路	1
1.2 .NET Framework 的体系结构	3
1.2.1 受控代码和装配件	4
1.2.2 属性	6
1.2.3 受控语言的选择	7
1.3 公共语言运行时环境	8
1.3.1 AppDomain	10
1.3.2 受控执行	12
1.4 Framework 类库	14
1.4.1 系统提供的类库	14
1.4.2 Microsoft.VisualBasic 装配件	15
1.4.3 FCL 的应用程序架构	18
1.5 转到.NET	20
1.5.1 转到.NET 的动机	20
1.5.2 转到.NET Framework 所面临的挑战	22
1.6 总结	23
第 2 章 使用 Visual Basic .NET 进行开发	25
2.1 入门知识	25
2.2 Visual Basic .NET 编译器	26
2.2.1 编写一个简单的基于控制台的应用程序	27
2.2.2 编写与使用组件库	28
2.2.3 使用 ILDASM.EXE 实用程序检查装配件	30
2.2.4 理解名称空间	31
2.2.5 导入名称空间	33
2.3 使用 Visual Studio .NET 进行开发	36
2.3.1 引用和名称空间重访	37
2.3.2 创建基于窗体的应用程序	39
2.3.3 创建一个 Windows 应用程序	40
2.3.4 调试项目	43
2.4 对 Visual Basic 的主要更改	45
2.4.1 新的便利功能	46

2.4.2	消除不一致和特异性	47
2.4.3	改进的范围规则	48
2.4.4	结构化的异常处理	49
2.4.5	严格类型检查	50
2.4.6	在 Option Strict 下使用 Boolean 值进行编程	54
2.4.7	逻辑比较和按位操作	55
2.5	总结	57
第 3 章	公共类型系统	58
3.1	公共类型系统的动机	58
3.2	公共语言规范	59
3.3	基本的 CTS 概念	60
3.3.1	继承	61
3.3.2	共享成员与实例成员	62
3.3.3	线程、方法执行和调用堆栈	63
3.3.4	值类型和引用类型	64
3.3.5	使用 Is 操作符对引用进行比较	66
3.4	核心 CTS 类型	67
3.4.1	基本类型	67
3.4.2	在基本类型之间进行转换	70
3.4.3	Object 类	73
3.4.4	装箱	74
3.4.5	使用 Object 类进行设计	78
3.4.6	在编程中使用字符串	79
3.4.7	连接字符串	81
3.4.8	使用 String 类的方法对字符串进行比较	83
3.4.9	使用 “=” 操作符对字符串进行比较	85
3.4.10	字符串转换	87
3.4.11	格式化字符串	88
3.5	在程序设计中使用时数组	89
3.5.1	使用 ReDim Preserve	92
3.5.2	定义数组的元素类型	94
3.5.3	数组类和多维数组	97
3.5.4	System.Array 类	98
3.6	总结	101
第 4 章	类	102
4.1	设计类	102
4.2	对类型和成员进行封装	103