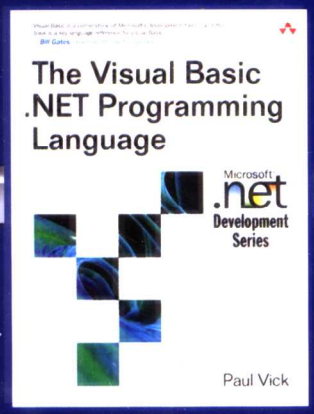
国外计算机科学教材系列

Visual Basic 是微软开发工具的基础，本书是学习 Visual Basic 的优秀参考书。

——比尔·盖茨

Visual Basic .NET 编程语言

The Visual Basic .NET Programming Language



英文版

[美] Paul Vick 著

電子工業出版社
Publishing House of Electronics Industry
http://www.phei.com.cn

国外计算机科学教材系列

Visual Basic .NET 编程语言

（英文版）

The Visual Basic .NET Programming Language

［美］ Paul Vick 著

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内容简介

本书由 VB.NET 语言大师 Paul Vick 撰写。在讲述了从 Visual Basic 到 VB.NET 的发展历程后，全面讨论了 VB.NET 语言的语法结构，探讨了 VB.NET 语言的基本概念、基本类型、数组和枚举、运算符、声明、异常、类与结构、方法、域和属性、事件和委派、继承、接口等，也讲述了如何从 COM 转换到 CLR，如何利用.NET 平台，如何充分利用 VB.NET 的面向对象的特征。全书除了提供基本的参考资料外，还提供了来自微软 VB.NET 设计团队的数百个代码范例。

本书适合 VB.NET 程序员和对 VB.NET 感兴趣的人士使用。

版权贸易合同登记号　图字：01-2005-3593

图书在版编目（CIP）数据

Visual Basic .NET 编程语言 = The Visual Basic .NET Programming Language /（美）维克（Vick, P.）著.
北京：电子工业出版社，2006.1
（国外计算机科学教材系列）
ISBN 7-121-02095-5

Ⅰ.V... Ⅱ.维... Ⅲ. BASIC 语言－程序设计－教材－英文　Ⅳ. TP312

中国版本图书馆 CIP 数据核字（2003）第 143970 号

责任编辑：赵红燕
印　　刷：北京天竺颖华印刷厂
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 × 980　1/16　　印张：27　　字数：622 千字
印　　次：2006 年 1 月第 1 次印刷
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 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

Praise for *The Visual Basic .NET Programming Language*

"There is no substitute to getting the inside scoop directly from a book written by the father of a programming language such as Bjarne Stroustrup for C++, James Gosling for Java and Alan Cooper for the original version of Visual Basic. Paul Vick, the father of Visual Basic .NET, explains the whys and hows of this exciting new language better than any other human being on the planet." —*Ted Pattison, Barracuda.NET*

"*The Visual Basic .NET Programming Language* includes nuances that in all my use and study of VB .NET, I haven't seen discussed anywhere else. For example, I learned that you can use the Imports statement to import an Enum name, so that you needn't refer to the enum in all its uses. In addition, I learned that the dictionary lookup operator, '!', works in VB .NET—I thought this one had been retired. In any case, if you're searching for a book that covers all the language syntax issues, and more, Paul Vick's book is a great place to look." —*Ken Getz, Senior Consultant, MCW Technologies, LLC*

"This book is an excellent stepping stone for Visual Basic developers wanting to get their toes wet in the .NET waters. Paul's presentation of the core topics all VB developers should tackle first is clear, concise, and unlike other books in the genre, does not overwhelm the reader. The VB6 vs. VB.NET task-oriented approach guides you through the new language and OO features, and then moves to basic threading and other CLR topics—as well as to the key points in the COM to .NET transition—in a well thought-out sequence. If you've been holding out on VB .NET, this is a great book to get you started."

—*Klaus H. Probst, Sr. Consultant/Architect, Spherion Technology Services, Microsoft MVP*

"There is no shortage of VB .NET books in the market, but this is the only book straight from the creators. While that is an excellent reason in itself for reading this book, it is the brevity and clarity of the content, along with the examples, that makes this book a must-have." —*Amit Kalani, Developer*

"Overall, I liked this book and it definitely benefited me. I learned new things I didn't see anywhere else and I'll certainly put these to good use in the future. Paul's book makes a great reference manual for intermediate and advanced VB .NET developers."

—*Philip Williams, System Engineer, LDC Direct*

"This book contains a lot of great information I have seen nowhere else and addresses issues that other books do not."

—*Ethan Roberts, .NET Architect, General Casualty*

"This book is full of useful information and provides a good historical background for the Visual Basic .NET Language."

—*Dave Vitter, Technical Lead Developer and author of* Designing Visual Basic .NET Applications *(Coriolis, 2001)*

Microsoft .NET Development Series

John Montgomery, *Series Advisor*
Don Box, *Series Advisor*
Martin Heller, *Series Editor*

The **Microsoft .NET Development Series** is supported and developed by the leaders and experts of Microsoft development technologies including Microsoft architects and DevelopMentor instructors. The books in this series provide a core resource of information and understanding every developer needs in order to write effective applications and managed code. Learn from the leaders how to maximize your use of the .NET Framework and its programming languages.

Titles in the Series

Brad Abrams, *.NET Framework Standard Library Annotated Reference Volume 1*, 0-321-15489-4

Keith Ballinger, *.NET Web Services: Architecture and Implementation*, 0-321-11359-4

Don Box with Chris Sells, *Essential .NET, Volume 1: The Common Language Runtime*, 0-201-73411-7

Mahesh Chand, *Graphics Programming with GDI+*, 0-321-16077-0

Anders Hejlsberg, Scott Wiltamuth, Peter Golde, *The C# Programming Language*, 0-321-15491-6

Alex Homer, Dave Sussman, Mark Fussell, *A First Look at ADO.NET and System.Xml v. 2.0*, 0-321-22839-1

Alex Homer, Dave Sussman, Rob Howard, *A First Look at ASP.NET v. 2.0*, 0-321-22896-0

James S. Miller and Susann Ragsdale, *The Common Language Infrastructure Annotated Standard*, 0-321-15493-2

Fritz Onion, *Essential ASP.NET with Examples in C#*, 0-201-76040-1

Fritz Onion, *Essential ASP.NET with Examples in Visual Basic .NET*, 0-201-76039-8

Ted Pattison and Dr. Joe Hummel, *Building Applications and Components with Visual Basic .NET*, 0-201-73495-8

Chris Sells, *Windows Forms Programming in C#*, 0-321-11620-8

Chris Sells and Justin Gehtland, *Windows Forms Programming in Visual Basic .NET*, 0-321-12519-3

Paul Vick, *The Visual Basic .NET Programming Language*, 0-321-16951-4

Damien Watkins, Mark Hammond, Brad Abrams, *Programming in the .NET Environment*, 0-201-77018-0

Shawn Wildermuth, *Pragmatic ADO.NET: Data Access for the Internet World*, 0-201-74568-2

www.awprofessional.com/msdotnetseries/

出 版 说 明

21世纪初的5至10年是我国国民经济和社会发展的重要时期，也是信息产业快速发展的关键时期。在我国加入WTO后的今天，培养一支适应国际化竞争的一流IT人才队伍是我国高等教育的重要任务之一。信息科学和技术方面人才的优劣与多寡，是我国面对国际竞争时成败的关键因素。

当前，正值我国高等教育特别是信息科学领域的教育调整、变革的重大时期，为使我国教育体制与国际化接轨，有条件的高等院校正在为某些信息学科和技术课程使用国外优秀教材和优秀原版教材，以使我国在计算机教学上尽快赶上国际先进水平。

电子工业出版社秉承多年来引进国外优秀图书的经验，翻译出版了“国外计算机科学教材系列”丛书，这套教材覆盖学科范围广、领域宽、层次多，既有本科专业课程教材，也有研究生课程教材，以适应不同院系、不同专业、不同层次的师生对教材的需求，广大师生可自由选择和自由组合使用。这些教材涉及的学科方向包括网络与通信、操作系统、计算机组织与结构、算法与数据结构、数据库与信息处理、编程语言、图形图像与多媒体、软件工程等。同时，我们也适当引进了一些优秀英文原版教材，本着翻译版本和英文原版并重的原则，对重点图书既提供英文原版又提供相应的翻译版本。

在图书选题上，我们大都选择国外著名出版公司出版的高校教材，如Pearson Education培生教育出版集团、麦格劳－希尔教育出版集团、麻省理工学院出版社、剑桥大学出版社等。撰写教材的许多作者都是蜚声世界的教授、学者，如道格拉斯·科默（Douglas E. Comer）、威廉·斯托林斯（William Stallings）、哈维·戴特尔（Harvey M. Deitel）、尤利斯·布莱克（Uyless Black）等。

为确保教材的选题质量和翻译质量，我们约请了清华大学、北京大学、北京航空航天大学、复旦大学、上海交通大学、南京大学、浙江大学、哈尔滨工业大学、华中科技大学、西安交通大学、国防科学技术大学、解放军理工大学等著名高校的教授和骨干教师参与了本系列教材的选题、翻译和审校工作。他们中既有讲授同类教材的骨干教师、博士，也有积累了几十年教学经验的老教授和博士生导师。

在该系列教材的选题、翻译和编辑加工过程中，为提高教材质量，我们做了大量细致的工作，包括对所选教材进行全面论证；选择编辑时力求达到专业对口；对排版、印制质量进行严格把关。对于英文教材中出现的错误，我们通过与作者联络和网上下载勘误表等方式，逐一进行了修订。

此外，我们还将与国外著名出版公司合作，提供一些教材的教学支持资料，希望能为授课老师提供帮助。今后，我们将继续加强与各高校教师的密切联系，为广大师生引进更多的国外优秀教材和参考书，为我国计算机科学教学体系与国际教学体系的接轨做出努力。

电子工业出版社

教材出版委员会

Preface

PERHAPS NO COMPANY'S fortunes have been more closely tied to those of a single programming language than Microsoft's has been with BASIC. Since Bill Gates and Paul Allen formed Microsoft over 25 years ago to sell their Altair MIPS BASIC interpreter, the corporation's successes have been linked with the BASIC programming language. From MS-DOS to Windows to the .NET Framework, BASIC has played a key role in attracting millions of developers to every major Microsoft platform.

And of all the incarnations of BASIC produced by Microsoft, none has been more successful than Visual Basic. The introduction of Visual Basic played a large part in the phenomenal success of the Windows platform and the Office productivity suite. The language also boasts the most developers—over 3 million at last counting—of any computer language, making it the most successful computer programming language ever. Much of the success of Visual Basic has come from the fundamental design tenets of the language: simplicity, straightforwardness, and ease of use. Visual Basic is a language designed for first-time programmers and experienced programmers alike. It makes learning computer programming easy by being approachable and understandable while providing maximum productivity and power suitable for the most advanced kind of applications. It is a versatile language that is an important part of any programmer's toolbox.

About This Book

This book contains a detailed description of the Visual Basic .NET programming language. It is intended to help new programmers to learn to

program in Visual Basic .NET and to serve as a reference for experienced Visual Basic .NET programmers. In general, this book does not cover topics that are unrelated to the language itself, such as how to use the Windows Forms libraries to do GUI programming or the underlying design of the Common Language Runtime. It focuses first and foremost on the language itself, and discusses topics outside the language only to the degree that they are necessary to understand some part of the language design.

The book is laid out as follows: After a general overview of the language is given, concepts are arranged in order from the simplest to the most advanced. The book is intended to be read sequentially, although programmers with some prior experience in Visual Basic might find that skipping back and forth through the book will work just as well. The progression of concepts through the book starts with the basics of the language such as statements and expressions, moves on to the object-oriented programming (OOP) features of the language such as classes, and then finishes with advanced topics such as inheritance and versioning. Readers who are familiar with prior versions of Visual Basic are still encouraged to read through each chapter; changes occurred throughout the language when moving to the .NET Framework, and so even familiar topics may contain new features.

In general, when a concept is discussed, it is discussed in detail. For the reader new to programming, the .NET Framework, or Visual Basic, some of the details may seem difficult to grasp. Advanced concepts are usually highlighted and are not required for a working understanding of the language—in practice, they can be skipped. Readers are encouraged not to get hung up on things that may not make sense, but instead to plow ahead and circle back later. Many times, advanced concepts earlier in the book may be simpler to understand once later concepts have been digested.

NOTE

The Visual Basic .NET language as discussed in this book corresponds to Visual Studio .NET 2003 and the .NET Framework 1.1.

A Short, Unofficial History of Visual Basic

The original BASIC programming language was defined by John G. Kemeny and Thomas E. Kurtz at Dartmouth College in 1964. BASIC is an acronym for **B**eginner's **A**ll-purpose **S**ymbolic **I**nstruction **C**ode, and the language was intended to be easy for beginners to learn, yet powerful enough to write general-purpose programs. In 1975, a couple of programmers named Bill Gates and Paul Allen developed a version of BASIC for the Altair MIPS and started selling it through a small corporation they set up. That version of BASIC helped launch the Microsoft Corporation to some extraordinary heights, and since then Microsoft has had a BASIC product available in one form or another.

In 1991, BASIC at Microsoft took a major step forward with the introduction of Visual Basic. Based on an idea originally developed by Alan Cooper, Visual Basic wedded a version of the BASIC language to the new Windows user interface, resulting in a powerful tool for developing Windows applications. The concept took off, and today Visual Basic is the most widely used programming tool in existence. The fourth version of the Visual Basic product marked the introduction of the Common Object Model (COM), a framework for developing components that could plug into one another. COM was also extremely successful, forming a foundation technology for Windows and being used to build millions of components.

As successful as COM was, it had limitations. One of the biggest was that it was not always designed to work well with languages other than Visual Basic. This meant that, for example, programmers who used C++ often found it difficult and time consuming to program against COM components, especially those written in VB. Another problem was that COM only defined how components were *supposed* to interact with each other, leaving the actual details of this interaction, especially in terms of such things as component lifetime and memory management, up to the component author. Although Visual Basic provided these services to its programmers, other popular languages, such as C++, did not. Because each C++ programmer was required to implement these services themselves, imple-

mentations were not always compatible or complete, resulting in not only more work but also more bugs (and, by extension, more user anguish).

To try to address some of the shortcomings of COM, Microsoft began developing a replacement technology, the Common Language Runtime (CLR). In addition to defining component interactions, the CLR also took on the job of managing such things as memory and component lifetime, taking over those responsibilities from the programmer. Microsoft also began developing the .NET Framework, a set of class libraries that provided capabilities equivalent to the Win32 API, on top of the CLR. Together, the .NET Framework and the CLR form the foundation of the next generation of Windows programming.

Although the CLR is intended as a replacement for COM, there are many differences between the two, some major, some extremely subtle. Because Visual Basic had become so integrated with COM, it was necessary to make corresponding changes to the language, some of which significantly altered the way programs were written in VB. As a result, Visual Basic .NET, a new and distinct version of the Visual Basic language, was developed. Although most of the language came through unchanged by the transition, some details of the language were modified, and many new concepts were added.

Where practical, things that have changed from previous versions of Visual Basic are pointed out in a box labeled Compatibility, but a full accounting of the changes in the language between Visual Basic 6.0 and Visual Basic .NET is beyond the scope of this book.

Style and Preferences

The style and practice of writing BASIC code has changed significantly since the language was introduced almost 40 years ago, and continues to evolve. More than any other programming language, BASIC has been adapted to almost every conceivable purpose and used in almost every possible computing environment. It is, perhaps, the most nonstandard computer language ever developed. Thus, the idea of suggesting a particular style or usage of the language may seem quaint, but is nonetheless part of the purpose of this book. As the previous section should have made clear,

Visual Basic .NET represents a huge leap forward for the Visual Basic language. Parts of the language have been dropped, other parts have been significantly altered, and still other parts are completely new. Where possible, the book offers suggestions on how best to use the language and provides rationale for those suggestions. Some suggestions are based on practical considerations; others are purely stylistic, bordering on pure personal preference. Readers are, of course, always free to ignore such suggestions—one of the joys of programming is the freedom to write code as one sees fit.

Acknowledgments

Without all the people who contributed to the creation of Visual Basic .NET, this book would not have been possible. In particular, my thanks go to the core group of people who worked together with me to come up with the language design for Visual Basic .NET, and especially Alan Carter, Sam Spencer, John Hamby, and Cameron McColl. They all managed to stick it out through what was often a long and extremely difficult transition from Visual Basic to Visual Basic .NET. Without their tireless work and passion for the product, we never would have made it.

Thanks also go to the larger Visual Basic .NET team, who translated the language designs into reality, and to the C# team, our partners in crime (as it were). Thank go especially to Anders Hejlsberg, Peter Golde, and Scott Wiltamuth, who all provided valuable insight into language design from a decidedly different vantage point. The CLR and .NET Framework teams also played an invaluable role in bringing Visual Basic .NET to fruition, and many thanks are due to the efforts of Brad Abrams and Jim Miller, who patiently spent many an hour explaining some aspect or another of the new platform!

This book itself would not have been possible without the enthusiastic support and enduring patience of Stephane Thomas and Michael Mullen at Addison-Wesley, and the editing skills of Martin Heller. Thanks go to Ken Getz, Amit Kalani, Rex Jaeschke, Rocky Lhotka, David Vitter, Phillip Williams, Ethan Roberts, Joe Hummel, and Klaus Probst for their willingness to review the book and offer helpful and enlightening comments and suggestions. And there would have been no way that I would have been

able to complete the book without the willingness of Julia Liuson and Paul Kuklinski to give me time to work on it (and to understand when other things didn't necessarily get done on time!).

And finally, none of this would have been possible without the love and support of my family, especially my wife, Andrea. I can never thank them enough.

Contents