BOOKS FOR PROFESSIONALS BY PROFESSIONALS™

Comprehensive
VB.NET Debugging

VB.NET
调试全攻略

- 全面描述面向Visual Basic与Visual Basic .NET开发人员的功能强大的调试技术。
- 讲述从 Windows Forms 到ASP.NET，再到XML Web服务的应用程序类型。
- 讨论了有深度但又易于理解的调试场景，包括多线程、继承及资源管理等。

(美) Mark Pearce 著
谢 俊 尹浩琼 译

清华大学出版社

VB.NET 调试全攻略

(美) Mark Pearce 著

谢俊 尹浩琼 译

清华大学出版社

北 京

内 容 简 介

本书系统全面地介绍了各种功能强大的调试技术，包括所有重要的调试工具和策略，以及 Windows Forms、ASP.NET、Web 服务、Windows 服务和 SQL Server 等多种应用程序的调试方法，最后还深入浅出地讨论了多线程、继承和资源管理等调试场景。

本书适合不熟悉调试技术或者希望全面掌握调试技术的 Visual Basic 与 Visual Basic .NET 开发人员阅读。

EISBN：1-59059-050-3

Comprehensive VB.NET Debugging

Mark Pearce

Original English language edition published by Apress L. P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA.

北京市版权局著作权合同登记号　图字：01-2003-7371

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。

图书在版编目(CIP)数据

VB.NET 调试全攻略/(美)皮尔斯(Pearce, M.)著；谢俊，尹浩琼　译. —北京：清华大学出版社，2004

书名原文：Comprehensive VB.NET Debugging

ISBN 7-302-08400-9

Ⅰ.V…　Ⅱ.①皮…②谢…③尹…　Ⅲ.BASIC 语言—程序设计　Ⅳ.TP312

中国版本图书馆 CIP 数据核字(2004) 第 026833 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：曹康

文稿编辑：王军

封面设计：康博

版式设计：康博

印 刷 者：北京密云胶印厂

装 订 者：北京国马印刷厂

发 行 者：新华书店总店北京发行所

开　　本：185×260　**印张**：23　**字数**：588 千字

版　　次：2004 年 5 月第 1 版　2004 年 5 月第 1 次印刷

书　　号：ISBN 7-302-08400-9/TP·6039

印　　数：1～4000

定　　价：45.00 元

前　言

本书主要讲述在使用 Visual Basic (VB).NET 创建桌面、网络和 Web 应用程序时，如何找出、理解、修复以及(最好是)预防 bug。书中探讨了新的跨语言及跨组件调试工具的强大功能，并说明了如何深入或横穿整个应用程序，从而在 bug 所在的任何级别上找出它们。

随着 VB .NET 的问世，许多传统的调试规则都已经改变，预示着一场变革风暴即将来临。

时代的变迁

回顾个人计算的黑暗时代(Dark Age)，在这期间，人就只是人而已，而代码都是以鲜血为代价编写出来的，创建一个可行而稳定的 Windows 应用程序需要付出大量非常艰苦的劳动。Windows 本身相对仍然不太成熟，而且因为缺乏可以用于生产程序的简单工具使其发展受到抑制。1991 年，Visual Basic 1.0 及其后续产品(以后统称为 VB.Classic)陆续出现，戏剧性地拯救并改变了软件开发的世界。

在软件开发史上，Windows 程序设计第一次变得易于理解，而从不认为自己是开发人员的人们也能够施展必要的手段将他们的软件思想变为现实。部门业务流程被大批量自动化，而且通常不需借助于内部技术部门的力量。当这些“非主流”程序员与重视 VB.Classic 带给 Windows 程序设计的非凡生产力的无数专业开发人员胜利会师之时，结果真是蔚为壮观。Windows 软件应用程序在接下来的 10 年中蓬勃发展，其动力是开发的速度和逐渐壮大的开发人员队伍。有一种相当有名的观点，即开发人员及其无数应用程序的可用性促进了 Windows 本身的广泛采用。VB.Classic 完全有可能成为 Windows 真正的应用程序。

这种程序设计的爆炸结果之一是，许多管理人员甚至开发人员开始认为，很多软件开发流程相对简单，而且 VB.Classic 在幕后所做的所有困难工作意味着应用程序的设计与开发可以变得更加迅捷。结果导致了另一次爆炸的出现——被 bug 肆虐的软件大量涌现。由于自身的不稳定和不可靠，Windows 应用程序已经变得声名狼藉。最终用户对每隔一小时就重启一次然后停止错误的应用程序已经习以为常。

由于对可能出现的 bug 类型缺乏了解，开发人员已经在他们的程序中引入了许多难以解决而且莫名其妙的缺陷。理解并解决个人计算涉及的许多典型问题已经花费了人们近 10 年的时间。

如果着眼于未来，可以预见历史可能重演。VB.Classic 可以简化 Windows 桌面应用程序的创建，而公共语言运行库(common language runtime，CLR)的设计目的则是简化桌面尤其是网络应用程序的创建及部署。这些新的应用程序全部使用了公共类库，可以使用至少一打不同语言中的任意一种或几种来编写。虽然与 VB.Classic 相比，学习曲线要明显困难得多，但是回报也肯定要大得多。可以肯定的是，只要 IDE 向导的魔杖一挥，从 XML Web 服务与操作系统服务到 Web 页与桌面程序的一切都将变为可用。部署将变得和 XCOPY 命令一样简单，而 DLL 地狱也将会只存在于远古的神话中(尽管它可能被策略地狱所取代)。

一个全新的世界在向我们招手，用其闪亮的新技术来引诱我们投入它的怀抱。CLR 及.NET Framework 在网络应用程序和 Internet 方面的潜力就像当初 VB.Classic 在 Windows 方面的潜力一样深不可测。一轮关于富有趣味性和革新性的分布式应用程序的全新浪潮已经蓄势待发。了解网络和网络应用程序的开发人员由此将身价大涨。

面临的挑战

上面说得天花乱坠……现在是接受现实检验的时候了。这种很酷的技术前景面临的几个主要问题之一是，网络资源与本地资源的透明是不同的。网络资源一般不会与本地机器上的资源具有同样的可用性、延迟以及可靠性。编写软件时必须将所有这些因素考虑在内，使用.NET 提供的一些非常方便的编程接口可以隐藏这些烦琐的细节，但是这样做甚至会使这些问题变得更加糟糕。此外，如果将应用程序放到雷阵雨定期肆虐的 Internet 世界中，那么它下沉的速度将会比 Titanic(泰坦尼克号)还要快。

这种技术前景的另一个主要困难是，.NET 的默认安全模型只给从本地 Intranet 下载的代码授予最低的权限，而给基于 Internet 的代码授予的权限就更少了。如果没有从总体上很好地理解这个安全模型及改变安全策略的含义，调试安全与权限问题很快就会变成一个噩梦。

所以，VB.NET 使得开发人员构建软件系统，尤其是网络应用程序与 Web 应用程序，比过去容易很多。然而，它同时也引入了很多产生 bug 的新途径。这给开发人员带来了令人畏惧的挑战，即尝试构建更加复杂的系统，而且使该系统在出错的时候仍然很容易诊断和修复。如果不小心，使我们的软件应用程序变得越分布式化、越复杂，我们自掘坟墓的可能性就越大。

规模更大的系统，尤其是分布式应用程序，必须使用与小型系统不同的方法进行设计与调试。我们必须吃一堑才能长一智。以防火墙、缓存、加速器、转换器、网关以及通信基础设施的形式引入中间件元素，意味着软件应用程序必须在与其周围环境合作的过程中克服几道障碍。软件需求的复杂度似乎与日俱增，而可靠性、可用性、安全性和集成性变得对业务更加关键。

“通过钝(blunt)对象进行程序设计”的思维课程在使用 VB.Classic 的时代通常占据主导地位，但是现在已经远远不够。VB.NET 的规则已经改变，而且为了补偿，交付可靠且缺陷低的软件的过程必须相应地改变。如果开发人员不了解这一点，那么天上的乌云将变得更加浓密，更加诡异，最终 bug 将会像雨点一样落在应用程序上。技术的进展只会使我们倒退的脚步加快。

本书旨在讲述与在有趣的 VB.NET 新世界中设计与调试软件相关的陷阱及危险。无疑，它并不是对语言或框架的一种批评。每项技术都有给开始使用这项技术的开发人员带来问题的方面。不应该让这些危险的因素蒙蔽了您的双眼，以至于看不到这种新语言及其类库潜在的好处。如果能够掌握并驾驭乌云，就可以体验到功能强大、效率高而且配备有一些可以在整个应用程序中使用的优秀调试工具的开发环境带来的好处。

本书内容安排

本书分为 4 个部分，共 15 章。建议首先阅读开始的 4 章，因为这 4 章中包含了对本书其余

部分所涉及到的内容的全面介绍。除此之外，建议应该能够自如地阅读每章，而不必参考前面或后面的内容。每一章的最后都有一个篇幅不长的“插曲”部分，讲述的是调试世界中一些有趣的故事。

第 I 部分：VB.NET 环境中的调试

这个部分首先站在战略的高度上概述了.NET 环境中的调试工作，然后更加详细地讲述了 VB.NET 中功能强大的新语言特性如何使您犯错误。

- 第 1 章讨论了 VB.NET 及新的.NET 环境战略上的调试难点。演示了一些需要了解的挑战与危险，从而为防止缺陷打下坚实的战略基础。
- 第 2 章介绍了一些由于不小心引起的讨厌的语言意外。表面上，VB.NET 与 VB.Classic 有一些相似之处，而这些相似之处会蒙蔽开发人员，使他低估了转向 VB.NET 语言的危险性。

第 II 部分：调试工具

在研究一些其他有用的调试工具之前，这一部分较详细地讲述了 Visual Studio .NET 调试器。最后以检查 VB.NET 的跟踪与检测工具结束。

- 第 3 章全面介绍了 Visual Studio .NET 调试器的功能。
- 第 4 章更加详细地讲述了 Visual Studio .NET 调试器，包括对各种调试器的设置及它更加高级的功能。
- 第 5 章研究了一些有用的调试工具，比如其他的.NET 调试器、Ildasm 实用程序和 Performance Monitor。
- 第 6 章详细讲述了桌面及网络.NET 应用程序的跟踪与检测。

第III部分：调试应用程序

这一部分说明了如何对每一种最常见的 VB.NET 应用程序类型配置并实现调试。还讨论了一些高级的 VB.NET 调试技术。

- 第 7 章研究了 Windows Forms 应用程序的调试，包括类和控件库的调试。
- 第 8 章研究了 XML Web 服务的调试。
- 第 9 章研究了 ASP.NET 应用程序的调试。
- 第 10 章研究了 Windows 服务的调试。
- 第 11 章研究了如何使用 VB.Classic 进行跨语言调试。
- 第 12 章研究了如何使用 SQL Server 进行跨语言调试。

第IV部分：调试常见情况

这一部分广泛讨论了 VB.NET 应用程序中的错误处理机制，并且还分析了多线程及分布式应用程序的调试。

- 第 13 章研究了 VB.NET 的异常机制和错误处理工具，并且说明了如何有效地保护应用程序。

- 第 14 章研究了多线程应用程序的调试，重点强调应通过完善的设计来预防多线程 bug。
- 第 15 章研究了分布式应用程序的调试与监控，重点强调远程调试、调试远程处理以及提供可靠的错误诊断方法。

本书面向的对象

本书假定的读者对象是正在使用或者已经长时间使用 VB.NET 的中高级 Visual Basic 开发人员。本书同样适用于正在学习高级程序设计课程的学生和正在寻求在现实世界中掌握调试技能的刚刚毕业的学生。开发团队领导者和非技术管理人员应该发现，本书对于理解正在埋头苦干的开发人员所面临的可靠性及性能问题非常有价值。

本书还假定读者以前对 VB.NET 语言、CLR 以及.NET Framework 有一些认识和了解。本书不适合用作这些主题的入门书。尽管我经常讨论某个特定功能的微妙之处，但您可能(可能不)希望了解如何使用它，您应该广泛阅读文档并进行实验，从而掌握所讨论的功能。尽管可以为每个主题提供更多的介绍，但是这将导致本书的厚度变为现在的 3 倍！所以，我们将重点放在如何避免与调试问题上，而对功能的介绍相对简短，只在确实有必要的时候才进行详细介绍。

最后一个假定是开发人员要负责修复自己的 bug，最好是在这些 bug 到达测试人员之前，否则也一定要在软件投入生产之前修复 bug。如果您认为应该由测试人员或最终用户负责找出您的一些 bug，那么希望本书能够改变您的想法，或者至少向您展示了截然不同的观点。

源代码

本书包含了许多示例程序。可以从 Apress Web 站点的 Download 部分下载它们，站点的 URL 是 *http://www.apress.com*。每一章的源代码以 ZIP 文件的形式提供，每个示例程序位于各自的 ZIP 文件中。

本书中列出的代码都使用最新的 Visual Studio 2003 的 Release Candidate 和.NET Framework 的 1.1 版测试过，而且还使用 Visual Studio 2002 和 1.0 版本的.NET Framework 进行过测试。Apress 站点上可用的代码还将另外使用 Visual Studio 2003 和.NET 1.1 的发布版本进行测试，而且将包括最新的.NET 服务包在本书出版之后进行的修订以及改动。由于本书的编写和出版需要一定的时间，所以这个方法很有必要。

目　　录

第Ⅰ部分　VB.NET 环境中的调试

第II部分　调 试 工 具

第III部分　调试应用程序

第Ⅳ部分　调试常见情况

第Ⅰ部分　VB.NET环境中的调试

“历史上，除了手枪和龙舌兰酒之外，就数电脑使人犯的错误最多”。

—— Mitch Radcliffe

- 第 1 章　战略性调试问题
- 第 2 章　VB.NET 语言中的新设计