

<<Excel专业开发>>

图书基本信息

书名：<<Excel专业开发>>

13位ISBN编号：9787121183133

10位ISBN编号：7121183137

出版时间：2013-1

出版时间：电子工业出版社

作者：(美) 布伦 (Bullen,S.) 等

译者：王永

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<Excel专业开发>>

内容概要

《Excel专业开发(第2版Microsoft Excel及VBA应用开发权威指南)》编著者Rob Boverdeng。

《Excel专业开发(第2版Microsoft Excel及VBA应用开发权威指南)》基于Excel 2003及之后的版本和VSTO.NET 2008开发平台在第一版的基础上进行了较大的调整和更新，介绍了Excel专业应用程序开发的各种方法和技术。

主要包括Excel各类应用程序的结构和开发方法；用户界面等的开发；自定义类与对象的程序设计；VBA程序的设计、错误处理、调试及性能优化；图表编程、数据处理、数据库开发，以及在Excel应用程序中控制其他程序的技术等内容。

<<Excel专业开发>>

作者简介

作者：（美国）布伦（Stephen Bullen）（美国）Rob Bovey（美国）Dennis Wallentin（美国）John Green
译者：王永 谢青 罗龙艳 武建军 布伦（Stephen Bullen），和妻子Clare、女儿Becky，以及名叫Fluffy和Charlie的两只狗，生活在英国伦敦Woodford Green。

Stephen在牛津大学获得工程学、经济学和管理学硕士学位，是难得的精通商业和技术的双料专家。

Rob Bovey，Application Professionals软件开发公司的总裁，该公司专注于Microsoft Office、Visual Basic以及SQL Server等应用程序的开发。

Rob将其多年以来为团体客户开发财务、会计和行政信息系统的宝贵经验带到了Application Professionals。

Dennis Wallentin与妻子和两个女儿生活在瑞典Ostersund。

从1980年开始，Dennis就一直致力于Excel企业解决方案的开发，拥有企业管理和会计学硕士学位。

Dennis是XL—Dennis的创始人。

他为各种不同规模的公司，包括瑞典本土和国际公司的公共部门提供解决方案。

同时，他还为最新版的Excel程序和图书撰写评论，以及在瑞典的杂志上发表与Excel相关的文章。

在过去的几年中，Dennis专注于基于.NET技术的Excel企业解决方案的开发，其中也包括Visual Studio Tools for Office System（VSTO）。

John Green，是一名独立计算机顾问，工作生活地在澳大利亚悉尼市，他精于利用VBA对Excel、Access、Word及Outlook进行整合。

John有着30年以上的计算机从业经验，并获得了化学工程学士学位和MBA硕士学位。

王永，副教授，博士、硕士生导师。

主要从事管理信息系统、信息安全和计算机应用方面的教学与科研工作。

先后在北京邮电大学、香港城市大学做高级访问学者，从事管理数据分析与挖掘、信息安全方面的研究课题，发表论文50余篇，主持国家级、省部级科研项目十余项。

担任《Information Sciences》、《通信学报》等十余个国内外学术期刊的审稿人。

主译英文译著两部，主编教材多部。

谢青，讲师，主要从事大学计算机基础、Visual Basic程序设计、信息系统分析与设计等方面的教学科研工作。

主持参与了省部级、市级、校级项目科研课题多项。

熟悉Excel开发，参编和翻译书籍六部，其中Excel开发三部，发表文章十余篇。

武建军，讲师，研究方向是信息管理与信息系统。

曾先后在烟台东方电子有限公司、重庆电信计算机中心等多家IT及通信企业工作，担任软件测试工程师、软件工程师、软件架构师、软件项目管理等职位。

主持参与了《重庆市重要信息系统客户木马和僵尸网络监测系统》等多个IT项目的研究工作，出版《社区管理信息化应用技术规范研究》等专著与教材三部。

罗龙艳，讲师，长期从事办公自动化及Excel数据处理相关的教学科研工作，熟悉各类OA系统（尤其Microsoft Office），擅长Excel数据统计与分析，以及Excel在信息管理、预测与决策中的应用。

先后参与《Excel专业开发》（第1版）和《VSTO开发指南》的翻译。

书籍目录

第1章绪论 1.1关于本书 1.2本书适合的读者群 1.1.3Excel开发人员分类 1.4Excel——应用程序的开发平台
1.4.1用作数据输入和输出表示层的工作表 1.4.2用作简单数据存储的工作表 1.4.3VBA代码与用户窗体
1.4.4用作声明式编程语言的工作表 1.4.5Excel对象模型 1.5本书结构 1.6本书案例 1.7支持的软件版本 1.8
关于字体风格 1.9关于配套资源 1.10帮助与支持 1.11专业Excel开发网站 1.12读者反馈 第2章应用程序
的结构 2.1 基本概念 2.1.1 无编码的应用程序 2.1.2 自动化工作簿 2.1.3通用加载宏 2.1.4特定程序中的加载宏
2.1.5独立式应用程序 2.1.6可用的技术 2.2本章小结 第3章Excel与VBA开发的最佳惯例 3.1命名规则 3.1.1
一个命名规则惯例 3.1.2命名规则惯例 3.1.3过程 3.1.4模块、类和用户窗体 3.1.5工作表和图表 3.1.6 Visual
Basic工程 3.1.7 Excel用户接口的命名规则 3.1.8不必使用命名规则的特殊情况 3.2关于程序结构与组织的
最佳惯例 3.2.1应用程序的结构 3.2.2过程化应用程序的组织 3.3一般性应用程序开发的最佳惯例 3.3.1代
码注释 3.3.2代码的可读性 3.3.3 通用的VBA编程惯例 3.3.4修改控制 3.4本章小结 第4章工作表设计 4.1 工
作表用户接口设计的原则 4.2程序行和程序列：用户接口设计的基本技术 4.3预定义名称 4.3.1命名常量
4.3.2命名区域 4.3.3命名公式 4.3.4预定义名称的作用范围 4.4样式 4.4.1创建和使用样式 4.4.2修改样式
4.4.3在工具栏中添加样式下拉框 4.5用户接口中的画图技术 4.5.1使用边框创建特殊效果 4.5.2创建具有良
好格式的表格 4.5.3显示帮助文本的单元格批注 4.5.4使用图形 4.6数据检验 4.6.1数据的唯一性检验 4.6.2
级联列表 4.7条件格式 4.7.1创建动态表 4.7.2警示错误 4.8在工作表中使用控件 4.8.1 窗体控件的优点 4.8.2
Active控件的优点 4.9实例分析 4.9.1 隐藏行和列 4.9.2预定义名称 4.9.3样式 4.9.4用户接口的绘图技术
4.9.5数据检验 4.9.6条件格式 4.10本章小结 第5章函数、通用加载宏和特定应用的加载宏 5.1 应用程序的
四个阶段 5.1.1开发与维护 5.1.2 启动 5.1.3运行 5.1.4 关闭 5.2函数库加载宏 5.2.1用户自定义函数（UDF）
示例 5.2.2 UDF的命名规则 5.2.3让UDF看起来更像内部函数 5.2.4为函数库加载宏创建友好的名称和描
述信息 5.2.5 关于UDF的一些重要细节 5.2.6 VBA UDF问题 5.3通用加载宏 5.4特定应用的加载宏 5.4.1使
用表驱动的方法管理工作表用户接口（ul） 5.4.2使用VBA动态修改用户接口工作表 5.5 实例分析 5.5.1
功能特性 5.5.2应用程序的组织 5.6本章小结 第6章独立式应用程序 6.1独立式应用程序的结构 6.1.1 启动
与关闭 6.1.2 自定义用户界面 6.1—3处理与分析 6.1.4显示结果 6.2实例分析 6.2.1 PETRAS报表 6.3本章小
结 第7章使用类模块创建对象 7.1 创建对象 7.1.1类模块的结构 7.2创建集合 7.2.1创建集合对象 7.2.2用
地址表示类集合的不足 7.3捕获事件 7.4引发事件 7.4.1 家庭关系问题 7.4.2创建触发类 7.5 实例分析 7.5.1
PETRAS时刻表 7.5.2 PETRAS报表 第8章应用程序的结构 第9章XML介绍 第10章Office 2007 Ribbon
用户界面 第11章创建跨版本兼容的应用程序 第12章理解和使用Windows API调用 第13章用户窗体设计
与范例 第14章接口 第15章VBA错误处理 第16章VBA调试351 第17章优化VBA的性能 第18章数据库开发
简介 第19章Access和SQLSever数据库编程 第20章数据处理技术 第21章高级图表技术 第22章控制其
他Office应用程序 第23章Excel和VisualBasic6 第24章EXCEL和VB.NET 第25章使用VB.NET编写托管COM
外接程序 第26章使用VisualStudio的Office工具开发解决方案 第27章XLL和CAPI函数 第28章Excel和Web
服务 第29章帮助、安全、打包和发布

章节摘录

版权页：插图：注意，如果要对Excel对象库之外的对象使用早期绑定，必须先在VBE的菜单“工具>引用”中指定一个对相关对象库的引用。

例如，要创建一个早期绑定ADO对象的变量，必须首先指定一个对Microsoft ActiveX Data Objects 2.x库的引用（此处X表示ADO的版本号），应该尽可能地使用早期绑定。

相对后期绑定而言，早期绑定具有以下优点：性能更好编译时，若VBA已知对象变量的数据类型，则VBA可从对象中获取其属性和方法的内存地址，并将其与代码存储在一起。

在运行时，如果VBA NNT早期绑定的属性或方法，只需要执行对应存储位置的代码即可。

（为了说明方便，在描述上做了一定程度的简化。

实际上VBA存储的是执行代码相对于内存起始地址的偏移量，即相对于对象Vtable的偏移量。

）若变量为后期绑定，则VBA无法知道变量所包含的对象类型。

所以，在编译时就不能对属性及方法进行优化。

这意味着若VBA需要调用后期绑定变量的属性和方法，每次都必须先确定变量所包含的对象类型，然后再根据属性或方法的名称确定其对应的内存位置，最后才执行相应的代码。

显然，这个过程比早期绑定慢得多。

类型检查更严格在代码清单3—13所示的后期绑定示例中，如果不小心将对象变量所引用的ADO Connection对象写成了ADO Command对象，VBA不会就此给出提示。

你只有在代码中使用到Command对象不支持的方法和属性时，才会发现其中潜在的问题。

然而，如果对象采用的是早期绑定的方法，那么VBA会立即检测出当前所引用的对象有误，并给出“类型不匹配”的错误提示，这样可在代码尚未运行时就检测出所调用属性或方法的不正确性。

在编译时，VBA会在对应的对象库中查找要调用的属性名或方法名，如果没有找到，VBA就会抛出编译时错误。

具有智能感知的功能采用早期绑定的方式可使编程更为容易，因为VBA能确切地知道变量所代表的对象类型，在用户输入属性或方法名称的一部分时，VBA就会解析相应的对象库，并弹出一个下拉列表框，其中包含了相关的属性名和方法名，方便用户选择。

然而，在某些特殊情况下需要使用后期绑定而非早期绑定。

以下是两个最常见的必须使用后期绑定的情况：1.采用早期绑定的方式时，若程序中需要的对象库版本比计算机中现有的版本更高，则程序的兼容性会被破坏，无法正常运行。

这种情况非常普遍。

如果应用程序中引用的是新版的应用程序对象库，而当前运行程序的计算机中采用的却是旧版的应用程序对象库，此时就会产生“无法找到工程或库（Can't find project or library），的运行错误。

而且，这种情况非常容易引起误导，因为产生这种错误的根本原因是代码编写不当，而与对象库没有任何关系。

如果希望避免上述错误的出现，让应用程序在各种版本的对象库中均能正确运行，就需要在引用对象时使用后期绑定。

在创建新对象时，也只能使用带有独立参数ProgID的函数CreateObject，而不能使用“=New ObjectName”的语法格式。

2.当不能确定用户是否安装了应用程序所需要的支持软件，而且也不能自行安装这些软件时，只能采用后期绑定的方式。

在这种情况下，采用后期绑定的方式可以避免因为没有相应的对象库而产生的编译错误。

在应用程序运行时，它会检查在当前计算机中该对象库是否存在，如果不存在，应用程序就会自动退出。

提示：在编写使用后期绑定的代码时，可以先用早期绑定代替后期绑定，这样可充分利用代码的自动提示，提高编写的效率。

而在代码编写完毕后，在最后的测试和发布之前再将早期绑定更改为后期绑定。

编辑推荐

《Excel专业开发:Microsoft Excel及VBA应用开发权威指南(第2版)》对Excel应用程序开发各种技术的介绍全面而深入浅出,通俗易懂,是Excel中高级用户和开发人员的首选书籍,也是想了解与提高Excel应用程序设计的技术人员的最佳参考书。

名人推荐

随着Excel应用程序的日益复杂和Windows开发平台的愈加强大，Excel开发人员需要一本像这样的书来帮助他们将Excel解决方案提升到一个全新高度。

本书适用于那些希望利用微软的最新技术开发出功能强大的、艺术级的Excel应用程序的开发人员。

——Gabhan Berry，微软项目经理，Excel编程专家 本书第一版让我在Office开发方面获益最多，也是我最愿推荐的一本好书。

第二版在第一版的基础上对内容的广度和深度都进行了扩展。

本书的亮点是对每个问题的处理都能比你的预想更讲一步。

书中凝聚了作者当前实际应用的经验，不仅对某些特性的工作原理进行了剖析，而且还探讨了在专业工作中使用该特f出开产生的实际影响。

——Shauna Kelly，Thendara Green总监 本书对如何开发出设计良好、健壮和可维护的Excel应用程序的相关技术进行了介绍。

书中所述来源于作者几十年设计和开发应用程序的丰富经验。

此外书中还给出了一个贯穿全书的时间表应用程序开发案例，借此案例精要介绍了理论方法的实际应用。

每个严谨的Excel开发人员都应该认真阅读此书，定会获益匪浅。

我就是这样做的。

——Bill Manville，Bill ManxSille Associates应用开发人员 本书详尽解释了许多难以理解的概念，给出了某些复杂开发项目的多种解决方案并析其利弊，其对最佳开发实践的介绍尤其值得赞赏。

——Beth Melton，Microsoft Office MVP，独立承包商 本书是一本专为严谨的Excel开发人员准备的专业书籍，其内容不仅包括对象模型、工作表布局和代码语法，更重要的是对专业开发人员的工作设计思路进行了探讨。

本书对Excel的探讨相当深入，更为难得的是它还对成功的Excel开发背后的思维和逻辑过程进行了探索。

——Jon Pelfier，Microsoft Excel MVP，Pelfier Technical Services公司总裁

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>