

Visual Basic 3.0 for Windows 程序设计指南



郑庆华 编著
徐鹏 吴晖

西安电子科技大学出版社

Visual Basic 3.0 for Windows

程序设计指南

郑庆华 编著
徐鹏 吴晖

西安电子科技大学出版社

1995

(陕)新登字 010 号

内 容 提 要

本书以最新的 Visual Basic 3.0 为基础,从以下三个方面全面深入地阐述了 Visual Basic 程序设计的基本环境、基本方法、系统环境资源应用及其扩展功能:一、Visual Basic 程序设计基础,主要介绍 Visual Basic 程序设计的基本环境、基本方法及过程,内容涉及 VB 环境、控制图、界面及菜单设计、输入/输出、文件系统、图形功能等。二、Visual Basic 和 Windows 环境,主要介绍 VB 应用程序如何实现 Windows 的许多高级功能,内容涉及 Clipboard 的使用、动态数据交换 (DDE)、Visual Basic 和 DOS、定时器事件编程等。三、扩展 Visual Basic,主要介绍 Visual Basic 和 Windows、Borland C++ 间的接口,即调用动态链接库 (DLL) 函数的方法。

本书内容图文并茂、条理清晰、理论与实例并举。其中所举实例均为作者在开发 VB 应用程序过程中总结和积累下来的经上机验证的典型实用例子。

本书既适于 Windows 程序设计的初学者,又适于有一定经验的 VB 编程及开发人员,还可作为大专院校有关专业的 Visual Basic for Windows 程序设计课程教材。

Visual Basic 3.0 for Windows

程序设计指南

徐鹏 编

责任编辑:段成安、李惠群

西安电子科技大学出版社出版发行

西安市长青印刷厂印刷

新华书店经销

开本 787×1092 1/16 印张 21.2/16 字数 494 千字

1995 年 1 月第 1 版 1995 年 10 月第 2 次印刷 印数 5 001—10 000

ISBN 7-5606-0366-1/TP·0110

定价:19.00 元

前 言

Microsoft 公司 Windows 的推出,特别是 Windows 3.1 及中文版 Windows 3.1 的推出,立即以其新颖的图形用户界面、卓越的多任务操作系统性能、高层次的软件开发平台而风靡全球。学习、使用、研究和开发 Windows,已成为一股新的技术热潮。可以毫无疑问地认为:Windows 已经而且还将使计算机(尤其是 PC 机)的操作方式以及软件开发思想和过程产生深刻的变革,Windows 操作系统必将成为下一代 PC 机操作系统的主流。

分析 Windows 成功的原因,我们认为它主要依赖于以下三点:首先,Windows 提供了功能强大而又使用方便的“图形用户界面”,通过与 MS-DOS 的密切配合,使得 PC 机的用户界面焕然一新。Windows 通过提供“菜单”、“图标”、“窗口”、“对话框”等“图形用户界面”的使用工具,不仅替代了 MS-DOS 的命令行,免去记忆命令的麻烦,而且使得各种各样的软件具有几乎一致的运行和操作方式,从而使得计算机的操作方式越来越规范化、简单化和生活化。其次,Windows 操作系统提供了一个多道程序或多个任务并发执行的系统环境,真正利用和发挥了 386/486 等高档微机硬件所提供的内存资源、CPU 多任务切换和保护机制及其它系统资源。各任务或应用程序之间既能很方便地切换,又可随时相互交换信息。第三,Windows 提供了一个高层次的软件开发平台和一系列日益完善的应用软件。Microsoft 在推出 Windows 系统的同时,还推出了其软件开发工具包 SDK 和设备驱动程序开发包 DDK。各种程序设计语言也纷纷支持 Windows,如 Microsoft C++ 7.0 和 Borland C++ 3.1。Visual Basic for Windows 是由 Microsoft 公司自行设计、旨在简化和普及 Windows 应用程序设计的一种使用方便而又具有与 Borland C++ 功能相媲美的 GUI 式面向对象程序设计语言。

一、关于 Visual Basic

Visual Basic for Windows(简称 VB)是一种具有很好的图形用户接口(Graphic User Interface,简称为 GUI)的程序设计语言,同时又是一种完全支持结构化程序设计的面向对象程序设计语言。Visual Basic 综合运用了 Basic 语言结构化特点及 GUI 可视设计工具(各种控制图、下拉式菜单、窗口、对话框等),既具有 Windows 优越的图形工作环境,又汲取了 Basic 语言编程的简便性,既是一种可供专业程序设计者使用的专门的 Windows 应用程序设计语言,又是一种可供非计算机专业的计算机爱好者学习和掌握 Windows 编程的最简单易学的程序设计语言。自从 Microsoft 公司 1991 年推出 Visual Basic 1.0 以来,又相继推出了 VB 2.0 和 3.0 版本,已发展成为一套完全独立的 Windows 开发系统,在功能上可实现 Borland C++ 3.1 或 Microsoft C++ 7.0 的 Windows SDK 程序设计语言 95% 以上的功能,而其简单易学性可使任何一个从未实践过 Windows 程序设计的初学者很快入门并迅速掌握。

作为一种 Windows 下的 GUI 式 OOP 语言,Visual Basic 强调可视性、反馈性和调试功能。所谓可视性(Visuality)指的是用户设计图形界面就如同在白纸上设计图案一样方便直

观。反馈性强调的是 Visual Basic 和用户之间的交互性，不论是在程序设计阶段，还是在程序运行期间，VB 和用户间始终具有良好的交互和反馈。此外，Visual Basic 还向程序设计者提供了一个除了具有普通程序调试器功能以外的新调试功能——“邻近窗口”和“观察窗口”。Visual Basic 具有的可视性、反馈性和程序调试功能正是其既简单易学又实用灵活的原因所在。

Visual Basic 是一种面向对象的程序设计语言(Object Oriented Programming, 简称 OOP), 因而具有 OOP 语言所具有的对象建立、对象可编程、对象访问控制、对象继承、对象的多态性及对象可预见性等特征。Visual Basic 集中处理两种类型的对象: 一种是程序界面的载体——表格窗对象, 另一种是在表格上可任意定义的各种控制图对象——命令按钮、选择按钮、图像框、文本框……等共 22 个标准控制图对象。我们对 Visual Basic 编程可以作一个这样简单的概述: 在表格窗上设计好各种所需的控制图对象, 即设计好应用程序的界面, 并为这些对象设计相应的事件处理函数, 再利用 VB 提供的调试工具调试应用程序, 直至程序能正确运行为止。

Visual Basic 可以实现 Windows 的绝大多数高级功能, 如多文档界面(MDI)、对象的链接与嵌入(OLE)、动态数据交换(DDE)、动态链接库(DLL)函数的调用、TrueType 字型字体技术等。尤其是动态链接技术, 使得 VB 可调用 Windows 的各种系统资源, 实现和 Borland C++ 间的接口, 从而使得 Visual Basic 可以实现远远超过其自身, 并可与 C++ 相提并论的系统功能。此外, 就 VB 语言本身而言, 也具有许多特色, 如动态数组、静态和动态运行错误处理、丰富的图形图像处理功能和丰富的数据类型, 等等。

总之, Visual Basic 以其简单易学性、功能实用性、接口开放性而成为一种 Windows 下最简单易学而又功能卓越的 GUI 式面向对象程序设计语言。

二、本书的主要内容

如何让越来越多的计算机爱好者学习和掌握 Windows 编程? 我们认为, 最重要的是选择一种既方便易学、又不失强大功能的程序设计语言, 而 Visual Basic 正是您最佳的选择。

本书以最新的 Visual Basic 3.0 为基础, 从以下三个方面系统地阐述 Visual Basic 程序设计的基本环境、基本方法、系统环境资源调用及其扩展功能: 一、Visual Basic 程序设计基础, 主要介绍 Visual Basic 程序设计的基本环境、基本方法和基本过程, 内容涉及 VB 环境、控制图、界面及菜单设计、基本 I/O、文件系统、图形功能等。二、Visual Basic 和 Windows 系统, 主要说明 VB 应用程序如何实现 Windows 的许多高级功能, 内容涉及剪贴板(Clipboard)的使用、动态数据交换(DDE)、Visual Basic 和 DOS 间的接口, 此外, 还介绍了 VB 中定时器对象的使用和编程, VB 的动态运行错误处理等。三、扩展 Visual Basic, 主要说明 Visual Basic 和 Windows 及 Borland C++ 间的接口, 即调用动态链接库(DLL)函数的方法。

具体内容安排如下:

第一章: 介绍面向对象程序设计语言的基本概念及特征, 并阐述了 Visual Basic 是一种 GUI 式的 OOP 语言。

第二章: 介绍 Visual Basic 程序设计环境及程序设计的基本过程, 即对 VB 作一个总体概述。

第三章：说明 Visual Basic 程序设计语言的元素，包括数据类型、基本语句、静态与动态数组、子程序及其函数等。

第四章：介绍 VB 应用程序菜单的制作及控制方法，内容涉及下拉式菜单、选择按钮、检查框、列表框、混合框及滚动条等各种 Windows 菜单的建立及控制方法。

第五章：介绍 VB 的输入/输出功能，内容涉及 VB 的输入/输出方式控制、字型字体设定等。

第六章：VB 图像、图形功能介绍，内容涉及控制图的装饰、对象坐标系的定义、各种图形方法使用、色彩控制等。

第七章：介绍鼠标事件及其响应的事件函数，包括鼠标基本事件，以及由基本事件复合而成的复杂事件，如鼠标拖曳等。

第八章：介绍 Visual Basic 环境的资源调用及编程方法，内容包括多重表格应用程序设计、Clipboard 对象的应用、定时器应用、动态运行错误处理、外部事件响应及 VB 和 DOS 接口等。

第九章：VB 文件系统介绍，内容包括文件系统对象及其编程，VB 支持的三种类型文件——顺序文件、随机文件、二进制文件的使用及编程。

第十章：介绍如何实现 VB 应用程序和其它 Windows 应用程序间的动态数据交换 DDE，详细说明了和 DDE 相关的连接属性、连接事件和连接方法及 DDE 动态运行错误处理方法。

第十一章：Visual Basic 扩展功能的介绍，内容涉及 VB 控制图的制作与装入、VB 调用 DLL 的方法及实现等。

第十二章：详细介绍 VB 应用程序的排错及调试方法。

本书的附录也非常有特色，附录 A 分 7 张表，从各个不同的侧面给出了 VB 所支持的函数、方法、语句、对象、事件及属性。附录 B 较为详细地给出 VB 支持的系统函数的使用方法及语法规则。附录 C 列出了 VB 应用程序在运行过程中可能出现的动态运行错误及其相应的错误信息。附录 D 给出了 VB 系统提供的全局常量文件 CONSTANT.TXT 的具体内容。这些附录的内容都是程序设计者在程序设计过程中经常要查阅的内容。

值得指出的是，本书在阐述每一章所涉及的基本原理的基础上，配以大量应用实例，这些实例都是作者在使用、开发 VB 应用程序过程中总结和积累下来的、经过上机验证的既短小精悍又实用典型的例子。

本书内容由浅入深，由宏观到微观，内容安排层次分明，在具体写作上力求图文并茂、理论和实践并举。

本书在编著过程中，参阅了大量的 Microsoft 公司关于 Visual Basic Reference 的权威资料，作者在此基础上，又仔细研究了 Visual Basic 中本身的 Help 信息。可以说，这是一本全面深入地介绍 Visual Basic 使用及编程技术的参考工具书。

本书第一章、第二章、第六章、第七章、第八章、第十章、第十一章由郑庆华执笔，第三章、第四章、第五章由徐鹏执笔，第九章、第十二章由吴晖执笔，全书由郑庆华统稿。

三、本书的读者对象

既然 Visual Basic 是学习 Windows 编程的最佳选择，因而本书最适宜于那些渴望学习

075/47/05

Windows 的初学者(不论是计算机专业的还是非计算机专业的计算机爱好者)。

其次,本书作为一本全面深入地介绍 Visual Basic 使用及编程方法的技术参考工具书,也可供已有一定 VB 编程经验的程序设计者所用,尤其是书中第八章以后的内容及附录部分可使您节省大量查阅 Help 的时间。

此外,本书亦可作为大专院校开设 Visual Basic for Windows 程序设计课程的教材。

四、作者简介

郑庆华:1990年毕业于西安交通大学计算机系计算机软件专业,并于同年被推荐攻读本校计算机组织与系统结构硕士学位,1993年获该专业工学硕士学位,后留校任教于计算机系,并于次年(94年)被推荐攻读西安交通大学系统工程研究所“智能控制”专业的博士。

自90年以来,作者一直从事于计算机软件的开发,主要成果有:国家级火炬计划项目“CAIT 图文系统”、“CAIT 鼠标造字系统”的研制和实现;国家“七五”重点科技攻关项目“0530 多微处理机操作系统”的设计和实现;此外,还成功地开发了“通用报表生成及其图形分析系统”和“通用图文文档信息系统”两个商品化软件。

近年来,作者已发表论文8篇,主要获奖项目有:国家教委科技进步二等奖一项,校优秀科技成果二等奖一项,全国大学生课外学术、科技作品“挑战杯”竞赛二等奖一项。

徐鹏、吴晖:西安交通大学计算机系在校研究生。

最后,作者向那些为本书的拟稿、编著、审阅、校稿过程中作出过各种支持和帮助的亲人、师长和学友致以最诚挚的谢意!

郑庆华

1995年1月5日

目 录

第一章 面向对象的程序设计语言和 Visual Basic

§ 1.1 什么是面向对象的程序设计语言	1
§ 1.2 Windows 下面面向对象程序设计的基本原理	2
1.2.1 窗口	2
1.2.2 菜单	3
1.2.3 对话框	3
1.2.4 消息驱动的基本原理	3
§ 1.3 Visual Basic 概述	5
1.3.1 什么是 Visual Basic	5
1.3.2 Visual Basic 的功能特点	5
1.3.3 Visual Basic 是一种 GUI 式的 OOP 语言	6
本章小结	8

第二章 VB 开发环境及程序设计概述

§ 2.1 VB 开发环境	9
§ 2.2 对象及其三要素	15
2.2.1 对象三要素	15
2.2.2 对象定义及操作	17
§ 2.3 一个完整应用程序的设计过程	17
2.3.1 应用程序界面的建立	17
2.3.2 对象属性的设置	18
2.3.3 事件函数及其编程	20
2.3.4 应用程序的运行及调试	21
2.3.5 生成可执行的 .EXE 文件	22
2.3.6 应用程序的存盘	24
2.3.7 应用程序的调入	24
§ 2.4 怎样利用 VB 的 Help 功能	25
2.4.1 利用“Help”功能	25
2.4.2 在线帮助	27
§ 2.5 VB 编程实例	27
本章小结	31

第三章 VB 程序设计语言元素

§ 3.1 程序注解、数制表示及语句书写规范	32
§ 3.2 变量的类型、定义、作用域	32
3.2.1 变量名及其类型	33

3.2.2	变量定义	34
3.2.3	自定义类型和缺省类型	35
3.2.4	常量定义	36
3.2.5	变量的作用域	37
§ 3.3	基本语句	39
3.3.1	VB 的操作符	39
3.3.2	基本语句	40
§ 3.4	静态数组和动态数组	44
§ 3.5	子程序及其函数	46
§ 3.6	常用系统函数	48
§ 3.7	实例	51
本章小结		57
第四章	菜单的制作与控制	
§ 4.1	多级下拉式菜单的制作及控制	58
4.1.1	下拉式菜单的建立	58
4.1.2	菜单的控制	60
§ 4.2	选择按钮的建立及控制	61
§ 4.3	检查框的建立及控制	63
§ 4.4	列示框的建立及控制	64
§ 4.5	混合框的建立及控制	65
§ 4.6	滚动条的建立及控制	67
§ 4.7	菜单建立及控制编程实例	68
本章小结		73
第五章	输入输出	
§ 5.1	输入函数 InputBox \$	74
§ 5.2	多方式输出函数 MsgBox	75
§ 5.3	标签输出	79
§ 5.4	字型、字体的设置	81
§ 5.5	表格和图像框的输出方法 Print	86
§ 5.6	数字、日期、时间的格式化输出	89
§ 5.7	报表方式打印	91
§ 5.8	Tab 顺序控制	92
§ 5.9	文本框的使用及控制	94
§ 5.10	命令按钮的使用和控制	97
§ 5.11	输入输出程序设计实例	99
本章小结		103
第六章	VB 图像、图形功能	
§ 6.1	控制图装饰	104
6.1.1	LoadPicture 图像装入函数	104

6.1.2	控制图的动态调整	105
6.1.3	控制图的隐藏和显示	107
6.1.4	控制数组及控制图的动态增减	107
§ 6.2	建立对象的坐标系	109
6.2.1	对象坐标系的含义	109
6.2.2	对象坐标系的建立	110
§ 6.3	绘图	113
6.3.1	图形方法和属性	113
6.3.2	画点及取点的颜色	113
6.3.3	各种造型线及矩形框绘制	114
6.3.4	圆、椭圆及圆弧的绘制	118
§ 6.4	绘图色彩控制	121
6.4.1	RGB()函数	121
6.4.2	QBcolor()函数	121
6.4.3	直接对颜色赋值	121
§ 6.5	对象坐标系和图形绘制的关系	122
§ 6.6	应用实例	123
本章小结	129
第七章	鼠标事件的响应	
§ 7.1	鼠标事件	131
§ 7.2	鼠标事件函数的参数	132
7.2.1	鼠标位置参数(x, y)	132
7.2.2	鼠标按钮参数 Botton	132
7.2.3	移转参数(Shift)	137
7.2.4	复杂鼠标事件	138
§ 7.3	鼠标的拖曳与放下	138
7.3.1	拖曳图标(肖像)的设置	139
7.3.2	自动拖曳的实现	139
7.3.3	人工拖曳方式的实现	142
§ 7.4	鼠标事件应用实例	144
本章小结	145
第八章	环境应用	
§ 8.1	多重表格应用程序的设计	147
8.1.1	表格处理方法	148
8.1.2	多重表格程序设计	149
§ 8.2	全局模块	156
§ 8.3	文本块标示及其操作	157
§ 8.4	Clipboard 对象的应用	158
8.4.1	利用 Clipboard 实现文本块的剪贴	159

8.4.2 利用 Clipboard 实现其它数据信息的剪贴	159
§ 8.5 定时器应用	160
8.5.1 定时器控制图的建立及属性设置	161
8.5.2 定时器事件的响应或服务	161
8.5.3 利用定时器设计一个报时钟	162
§ 8.6 利用 DoEvents()函数执行外部事件	163
§ 8.7 响应键盘按键事件	165
§ 8.8 程序运行错误处理	169
8.8.1 运行错误的产生	170
8.8.2 运行错误的处理	171
8.8.3 集中式错误处理	173
8.8.4 滞后式错误处理	176
§ 8.9 Visual Basic 和 DOS	177
8.9.1 在 VB 应用程序中启动 DOS 应用程序	177
8.9.2 退出 Windows 并返回 DOS	179
本章小结	180
第九章 文件系统	
§ 9.1 文件系统的对象	182
§ 9.2 文件控制对话框的建立	185
§ 9.3 文件结构和种类	189
§ 9.4 顺序存取文件	190
§ 9.5 随机存取文件	193
§ 9.6 二进制文件	195
§ 9.7 文件系统应用实例	197
本章小结	200
第十章 应用程序间的通信	
§ 10.1 动态数据交换 DDE	201
§ 10.2 VB 对象的连接属性	202
10.2.1 LinkTopic——连接标题	202
10.2.2 LinkMode——连接模式	203
10.2.3 LinkItem——连接项目	203
10.2.4 LinkTimeOut——连接时间设定	204
§ 10.3 设计状态下 DDE 的实现	204
10.3.1 从其它应用程序获取数据	204
10.3.2 向其它应用程序传送数据	206
§ 10.4 连接事件	206
10.4.1 客户事件	207
10.4.2 服务事件	208
§ 10.5 自行设计 DDE	210

10.5.1	自行设计 DDE 的方法	210
10.5.2	一个自行设计 DDE 的实例	213
§ 10.6	连接方法	215
10.6.1	VB 客户程序方法	216
10.6.2	VB 服务程序方法	217
10.6.3	其它 Windows 应用程序中的 DDE 方法	217
§ 10.7	DDE 错误及处理	218
§ 10.8	向其它应用程序传递按键	219
	本章小结	221
第十一章	扩展 Visual Basic	
§ 11.1	用户控制图的装入	223
§ 11.2	DLL 简介	224
§ 11.3	DLL 的声明和调用	225
11.3.1	DLL 过程式函数的声明	225
11.3.2	DLL 函数的调用和执行	226
§ 11.4	获取屏幕及打印机的分辨率	231
§ 11.5	用 Visual Basic 制作音乐	233
§ 11.6	用 Visual Basic 处理串行输入及输出	239
	本章小结	245
第十二章	程序的调试与跟踪	
§ 12.1	VB 应用程序的错误分类	247
§ 12.2	VB 的三种模式	248
§ 12.3	中断执行	250
12.3.1	断点(Break Point)	250
12.3.2	Stop 语句	250
§ 12.4	程序跟踪	251
12.4.1	单步跟踪	251
12.4.2	单过程(函数)跟踪	251
12.4.3	跳跃执行	252
§ 12.5	调试信息窗口	252
12.5.1	观察窗口	253
12.5.2	邻近窗口	254
	本章小结	256
附录 A	Visual Basic 的函数、语句、对象、方法、事件、属性	257
附录 B	Visual Basic 系统函数	270
附录 C	动态运行错误的错误号及其错误信息	297
附录 D	Visual Basic 全局常量文件 CONSTANT.TXT 的内容	301

第一章

面向对象的程序设计语言和 Visual Basic

纵观程序设计语言的发展,从最初的纯机器代码发展到用符号表示机器指令的汇编语言,并在此基础上产生了 COBOL、FORTRAN、Basic、Pascal、C 和 FoxBase 等高级语言。这些高级语言的确大大地提高了编程的方便性,使得语言更加专业化,但它们都没有从根本上改变编程的思维方式,即都是通过一系列指令序列来处理数据,其最明显的特点:它们产生动作的指令和被处理的数据是分离的;程序被动处理数据,而不是数据主动地驱动程序的执行。

面向对象程序设计(Object Oriented Programming, 简称 OOP)是一种全新的程序设计方法,它第一次从根本上改变了半个世纪以来的编程模式。其基本思想为:通过建立和具体实际相对应的软件对象,并通过对这些对象的组合来创建具体的应用。所谓对象是指在传统的数据结构(如 Pascal 中的结构)中加入处理这些数据的过程和函数所构成的实体,因此,对象是数据和程序的结合体。

正如自顶向下和模块化程序设计方法给软件产业带来巨大的发展一样,面向对象技术同样会使软件行业产生深刻的变革。它不仅改变了人们编程的思维方式,而且也将改变软件设计的发展方向。面向对象的程序设计给软件的发展带来了以下益处:

(1) 可重用性:如果把软件设计比喻为产品设计,那么对象就相当于产品的零件。程序设计者利用软件对象可以设计出多种软件产品,而这正是面向对象程序设计技术的最本质的特征。

(2) 可靠性:首先,对象是数据和处理数据的过程或函数的结合体,系统提供的标准对象其所含的数据及其相应的函数或过程都是经标准测试的,因此,以对象为基础的程序设计所产生的应用程序也更为可靠。其次,由于应用程序的大量代码来源于可重复使用并具有高可靠性的库,因此,每一应用程序的新增代码明显减少,这就使得用户设计的应用程序更加可靠。

(3) 连续性:虽然面向对象的程序设计语言在编程模式上相差甚大,但它并没有摒弃传统的做法,它不仅采用了传统程序设计语言的语言元素,而且利用它们来模拟建立对象,从而使得一个熟悉传统程序设计语言的编程者能很快掌握面向对象程序设计的规律。

Windows 下的程序设计语言如 Borland C++、Microsoft C++、Visual C++ 及 Visual Basic 等都是面向对象的程序设计语言,虽然风格各异,但都具有共同的思维和编程模式。

§ 1.1 什么是面向对象的程序设计语言

虽然“对象”和“面向对象的程序设计”早在 80 年代初就提出了,但直到现在还有人在

曲解其本质特征和含义。有些程序号称是面向对象的，实际上只不过是高层次的编程环境。例如通过回答一些提问或填写几项内容而建立的菜单驱动程序根本不是面向对象的。

面向对象的程序设计语言必须具备以下特征：

(1) 对象：对象是一种将数据和过程或函数合为一体的数据结构。虽然大多数编程系统中都提供了数据和过程之间的关系，但只有面向对象系统才将数据及其处理过程和函数溶合在同一对象中。

(2) 可编程性：面向对象的语言必须是可编程的面向对象系统，以实现对象的控制和应用。Visual Basic 就是可编程的。

(3) 访问控制：对象必须既具有能对它的内部某些元素实施保护，使它们只能被对象内部的过程或函数使用，同时又能同其它外部元素进行联系，以便对对象进行操作。对象中能被外部其它程序所访问的元素在 Visual Basic 中被称为对象属性或方法。

(4) 继承性：通过对现有对象进行增加或修改来产生新的对象，这是面向对象的程序设计语言所必须具有的功能。对现有对象可以增加数据和过程，也可以对其中的过程进行重新定义。一般地，最初的类被称为基类(Base Class)，从基类扩展出来的类称为派生类(Derived Class)。显然，修改现有类进行派生是为了产生针对性更强、实用性更好的新类。

(5) 多态性：正如生态系统一样，通过对象派生或继承构成了对象类簇。类簇中的类都拥有同名的过程或函数，如 Visual Basic 图形框对象类中的 Circle 方法。多态性是指同名的过程或函数在同一类簇的不同对象中，具有不同的功能和表现形式。如 Circle 方法可以实现圆、椭圆、圆弧、椭圆弧、扇形等各种图形的绘制功能。

(6) 可预见性：对象的建立和清除对编程者必须是可预见的。在 Visual Basic 中程序设计者既可以静态地创建对象和清除对象，也可动态地建立和删除对象。

从上述面向对象程序设计语言所具备的特征我们看到：面向对象的程序设计以对象为基础，通过对对象的修改或增加产生出更具针对性的对象，并提供各种对象操作方法和手段，以实现对象的联系和应用。面向对象技术使程序设计者可以忽视对象的内部机制和具体实现，而将精力集中在数据结构的组织和对象的相互联系上，即注重于程序的整体结构，从而使得软件设计者可方便、高效、可靠、多产地设计出各种软件产品。

§ 1.2 Windows 下面面向对象程序设计的基本原理

对象是数据和程序的结合体，在 Windows 环境下，对象一般是指窗口、菜单、对话框等，这些对象是程序的基本元素，Windows 通过消息传递实现各对象之间的联系和作用，因此以消息或事件驱动(Event-driven)对象的执行是 Windows 下程序设计的基本点。

本节将具体说明上述元素及消息驱动的基本原理。

1.2.1 窗口

窗口是 Windows 应用程序基本的 I/O 设备，它是应用程序访问屏幕的唯一途径。窗口由标题栏、菜单栏、滚动条、边界及用户区等部分所组成，窗口或在程序设计时创建，或在程序运行时动态创建。

尽管窗口是由应用程序自己创建的，但必须由应用程序和 Windows 共同管理，Win-

Windows 管理标准窗口的元素,如边界、滚动条和标题栏等,并且完成对窗口的直接操作。应用程序维护窗口的其它部分,主要是维护窗口的用户区。

每一窗口必须有一窗口事件(如位置移动、大小改变等)的处理函数,在 Windows 中称为窗口函数或事件函数。由 Windows 系统实时响应任何改变窗口的行为,并以消息方式发送给窗口函数,窗口函数接收此消息并作出相应的操作或处理。这些消息或者指定函数应执行的动作,或者要求函数提供辅助信息。

1.2.2 菜单

菜单是 Windows 应用程序实现交互式用户界面和接收输入的主要手段之一。菜单是应用程序提供的一系列命令选项,对用户来说,这些选项是一组命令,可以看见并执行它们。

和窗口一样,菜单也是由应用程序在程序设计或执行时建立的,但必须由 Windows 和应用程序协调管理,Windows 负责显示和管理菜单,应用程序则实现菜单项功能的处理。当用户选取菜单时,由 Windows 系统将相应的消息发送给应用程序的菜单命令处理函数,从而实现菜单的操作。

1.2.3 对话框

对话框是一种用于应用程序和用户之间对话的临时窗口,对话框可以包含多个控制,一种控制是一种简单的窗口,完成简单的输入输出。例如,编辑控制对话框允许用户输入和编辑文字。对话框中的控制帮助用户输入各种信息(如文件名)、选择有关的选项或激活一个命令。

1.2.4 消息驱动的基本原理

Windows 和应用程序不仅一起完成对象的建立和处理,而且一起实现消息的处理。Windows 系统是个多任务的操作系统,为了实现多任务的运行和管理,Windows 在系统内部建立了一个系统消息队列,专门用于存放系统接收的各种消息,同时也为每个应用程序建立一个应用程序的消息队列,并负责将系统消息队列中的消息分发至各个相应的应用程序消息队列中,应用程序则不断地检测其消息队列,一旦发现就把此消息发送给相应的处理函数,如此循环,直至结束。其原理可用图 1.1 表示。

我们以应用程序处理用户按键为例说明消息驱动的过程和原理。用户按键可理解为一个事件,当用户按下和释放一个键时,Windows 首先接收到该事件消息,并存放至系统消息队列中,接着 Windows 将该消息从系统队列拷贝至相应的应用程序消息队列中,应用程序平时处于消息等待循环中,一旦发现其消息队列中有消息,则立即获取消息,并发送相应的消息函数(亦即事件函数),窗口函数处理该消息(例如将打入字符显示至窗口上)后,又回到消息循环等待状态。该过程可用图 1.2 表示。

从上述分析我们可以看到,Windows 下的面向对象程序设计和通常 DOS 下的结构化程序设计方法最大的区别在于:前者是一种消息驱动式体系结构,是由 Windows 主动给应用程序发送消息,并由 Windows 系统调用应用程序函数处理消息。而传统的程序设计方法则只能是应用程序调用操作系统资源,这是一种单向调用,是一种非消息驱动式体系结构。

可以说，Windows 系统调用应用程序以及两者协调实现各种消息事件的处理，是 Windows 面向对象体系结构的一大基石。

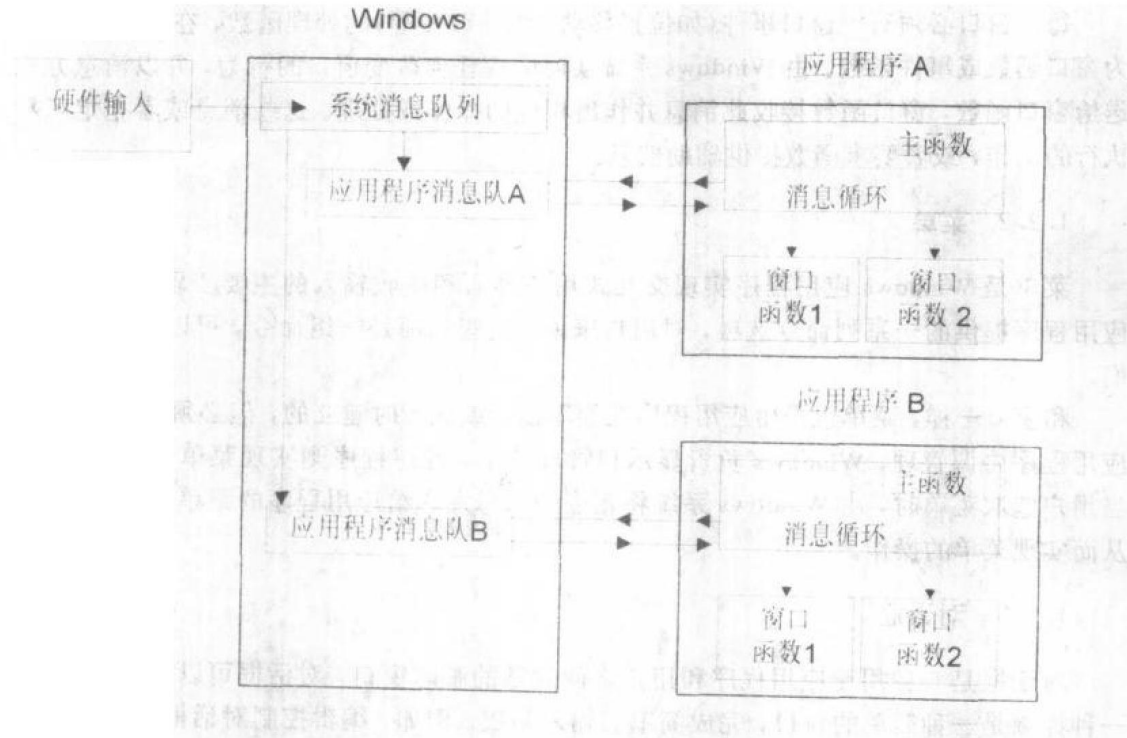


图 1.1

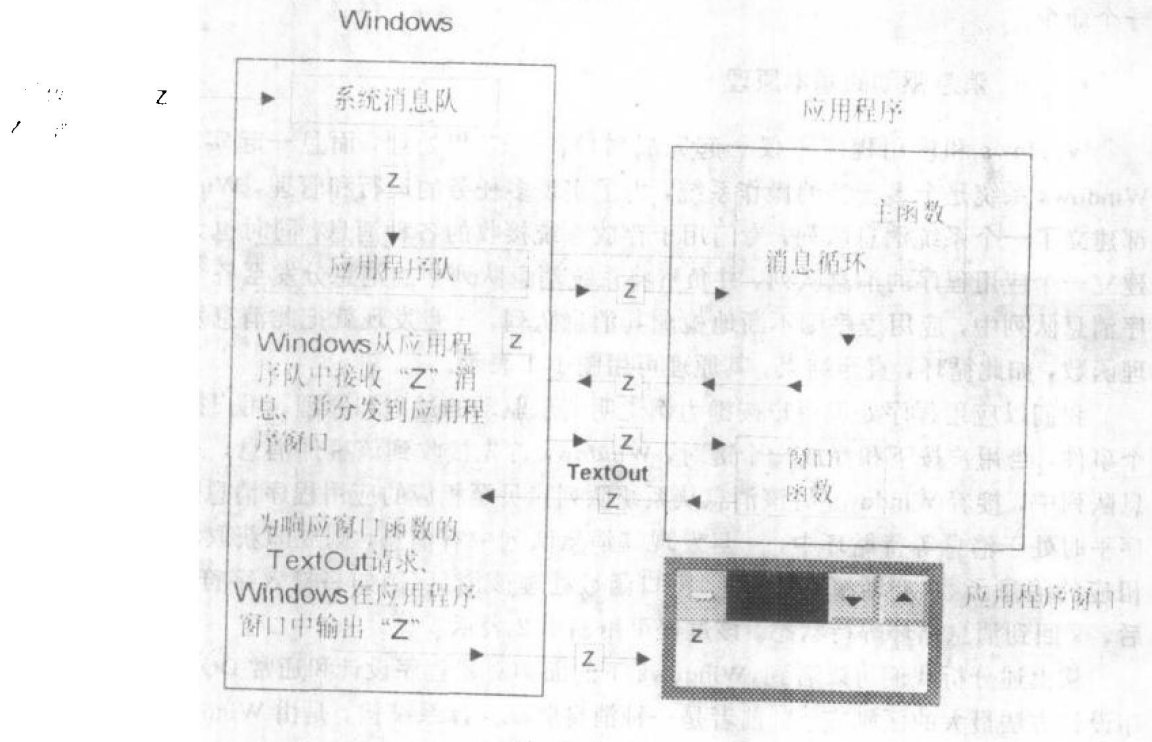


图 1.2

§ 1.3 Visual Basic 概述

1.3.1 什么是 Visual Basic

Visual Basic(简称 VB)是 Windows 系统下一种具有很好的图形用户接口(Graphic User Interface, 简称为 GUI)的程序设计语言,具有功能强大而又使用方便的各种图形界面——各种控制图、下拉式菜单、窗口、对话框等设计工具,同时提供了一系列简练而又功能完善的高层次程序设计元素。Visual Basic 强调可视性、反馈性和调试工具。所谓可视性(Visuality)指的是用户设计图形界面仿佛就像在白纸上设计图案一样直观方便。而反馈性强调的是 Visual Basic 和用户之间的交互性,不论在程序设计期间,还是在程序运行期间,VB 和用户程序间始终具有良好的交互和反馈。此外,VB 还向程序设计者提供了一个功能强大、使用方便的调试工具。

VB 集中处理两种类型的对象:一种是称之为表格的窗口;另一种是在表格上绘制的各种控制图。VB 为这两种对象提供了标准的事件处理函数或过程,程序设计者可在此基础上编写实际的事件处理程序以响应用户的动作。VB 以事件驱动方式执行应用程序,一个 VB 应用程序一般都由一个消息等待循环、若干对象以及和对象相应的事件函数组成。消息等待循环用以接收由 Windows 发来的各种用户请求事件,如窗口内容改变、大小调整、

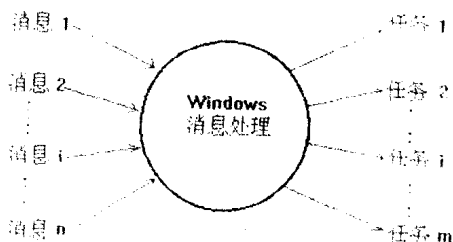


图 1.3

光标移动或按键等,Windows 将此事件或消息传递给相应的事件函数,并调用事件函数处理之。其工作模型可用图 1.3 示意。

消息等待循环一般形式为:

```

Do While Not DoEvents() '若无消息则等待
  If Get Message(& msg,null,0,0) then '若有消息则获取
    Translate Message(& msg) '将消息发送至消息队列
    Dispatch Message(& msg) '消息处理
  End if
Loop '继续等待消息
  
```

从上面分析我们看到,各对象的事件函数在无消息的情况下均处于不活动的静止状态,只有当相应的事件产生时,才被驱动执行。由此可见,Visual Basic 是一种真正的以事件驱动的面向对象的程序设计语言。

1.3.2 Visual Basic 的功能特点

Visual Basic 使用方便,具有功能强大的图形用户界面和程序开发接口,使程序设计者可以充分利用其图形环境和 Windows 系统资源。其主要功能如下:

(1) 创建 Windows 应用程序所需的各种对象,包括命令按钮、选择按钮、检查框、列