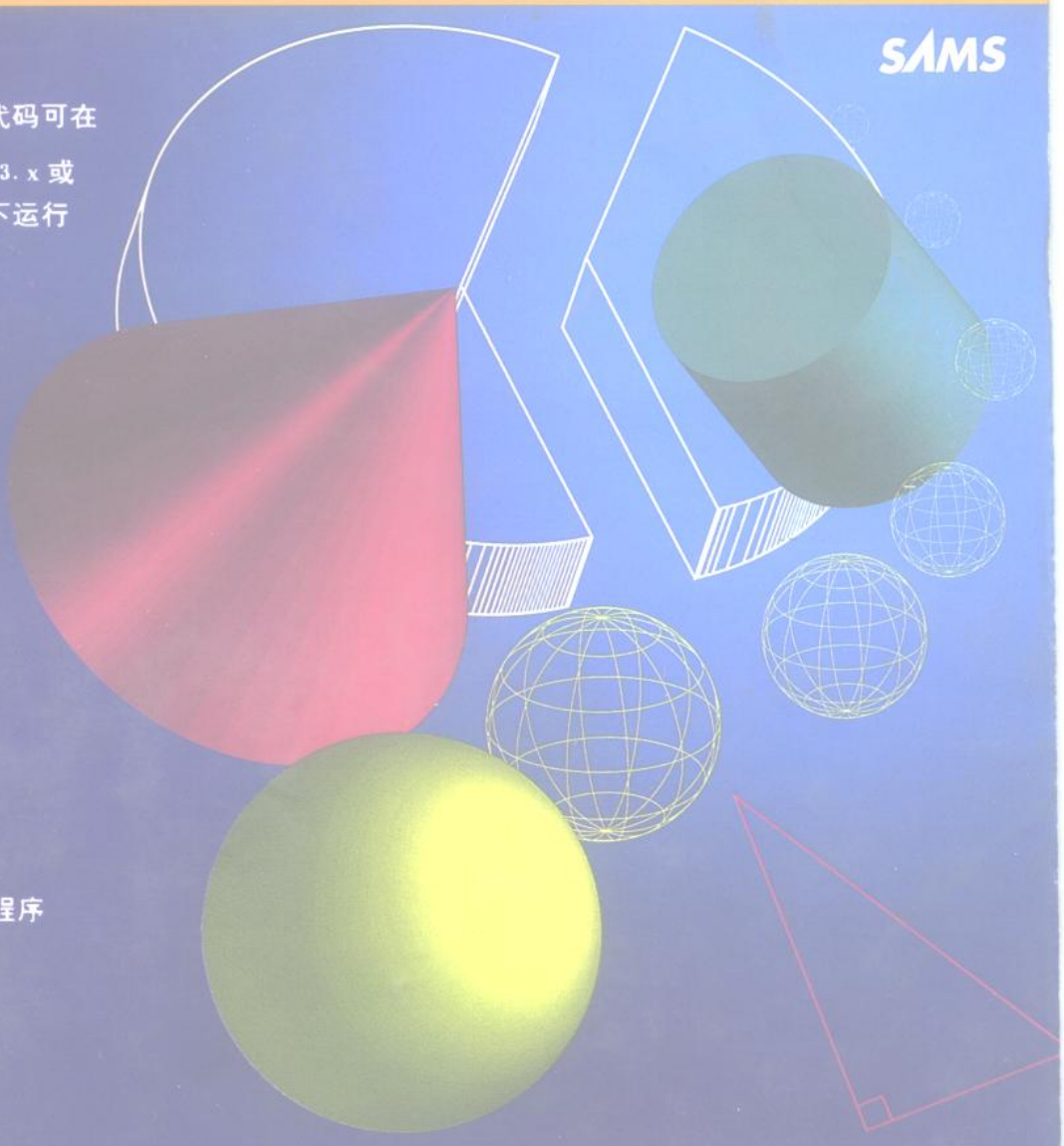
SAMS

软盘中的源代码可在
Visual Basic 3.x 或
更高的版本下运行



- 用 Visual Basic 建立绘图工具软件
- 建立自己所需的图形程序
- 揭示绘制 Bezier 曲线、调色等关键技术

VISUAL BASIC
图形程序设计

[美] Robert W. Stewart 著
冯建华 于志宏 郑海声 译
王 华

清华大学出版社

SAMS

北京科海培训中心

Visual Basic 图形程序设计

[美] Robert W. Stewart 著

冯建华 于志宏 郑海声 译

王 华 审校

清华大学出版社

(京)新登字 158 号

Graphics Programming with Visual Basic

书　名：Visual Basic 图形程序设计
译　者：冯建华　于志宏　郑海声
出版者：清华大学出版社(北京清华大学校内，邮编 100084)
印刷者：北京门头沟胶印厂
发　行：新华书店总店北京科技发行所
开　本：16　印张：29.5　字数：717 千字
版　次：1995 年 10 月第 1 版　1995 年 10 月第 1 次印刷
印　数：00001～5000
书　号：ISBN 7-302-02082-5/TP・971
定　价：59.00 元(带盘)

致 谢

感谢 Grace Buechlein 在组织本书的过程中所表现的耐心和付出的努力。

感谢本书的编辑 Brad Jones 和 Angelique Brittingham 在编辑本书初稿时所作的工作，以及在编辑过程中提出的许多建议、咨询和意见。

特别要感谢 Mary Inderstrodt 和 Sandra Doell 对本书书稿的编辑工作。还应该感谢 Sams 出版社的其他成员，他们在本书的制作过程中所给予的帮助。

还应该感谢技术编辑 K. L. Murdock 确保了本书及磁盘中的所有程序都能正确执行。

最后要感谢 Sandra Guinan 在本书的初始阶段所作的编辑工作。

关于作者

Robert W. Stewart(Barrie,Ontario)大约在 16 年前就开始了图形艺术家的生涯。他在加拿大的广告界及报界有着丰富的工作经验。他擅长于计算机编程和桌面出版。他现在是一个独立的咨询顾问，安装基于网络的图形制作系统。Robert 居住在加拿大安大略的多伦多。

简 介

《Visual Basic 图形程序设计》包括了许多图形工具以及它们的 Visual Basic 程序。本书介绍了用以创建简单的绘图工具的代码。这些工具中的绝大部分都是用简单的 Basic 编程语句编写的。本书还介绍了怎样模拟提供给最终用户使用的绘图工具，以前这些工具只能依靠诸如 C++之类的程序设计语言来产生。书中的这些程序应该被看作是起点，而不是最终产品，你可以试着将这些程序中的某些部分纳入你自己的程序。你会觉得这些例程对开发自己的 Visual Basic 程序有所裨益。

程序介绍

以下章节介绍了书中列出的所有程序的信息。

第 1 章的 Begin 程序

第 1 章介绍了几种使用 Basic 语言中 Line 和 Circle 语句的方法。这两个语句是所有的 Basic 语言都具备的标准图形函数。本章所给出的一些例子，可使你更好地理解 Line 语句的 Step 关键字以及圆和圆弧的 Start(起点)和 End(终点)位置。

第 1 章中的 Pset 程序

第 1 章还展示了 Basic 中 Pset 语句的若干例子，以及图形算法的基础知识。

第 1 章的 MoveTest 程序

使用鼠标在屏幕上移动对象的方法有许多种。本章所示的 3 个简单例程可以帮助你更好地理解其中的基本原理。

第 2 章中的 Artisan 程序

Artisan 是一个中型绘图程序，它使用了许多基本的绘图工具：形状(shape)、缩放(zoom)、线(line)、圆(circle)、方框(box)和文字(text)工具以及指针。该程序的其余部分都只是为了起装饰作用。虽然其中的控制对象可以起作用，但也都是为了增加视觉效果。例如，标尺确实可以缩放，但其中的尺寸并不是设成标准度量制。本程序中的主要部分是使用 Basic 语句绘制图形(第 15 章中的 ArtAPI 程序使用了更精致的代码及工具，并使用了 Windows API 函数)。不过，你还是应该仔细地看看 Artisan 程序，因为第 15 章中 ArtAPI 程序的许多例程都位于第 2 章的 Artsian 程序中。

Artisan 程序分布在第 1 章至第 4 章。本书首先给出声明段和各种窗体，以利于你理解每个程序中使用的变量和控制对象。因为所有的绘图操作都是在 Artwork 图片框控制对象中完成的，所以，首先给出该控制对象的各种事件过程，接着讨论 BoundingBox 过程，这些过程是对 Artwork 图片框的 MouseDown，MouseMove，MouseUp 事件的扩充。

第 3 章中的 Artisan 程序

第 3 章给出 Bézier 曲线的各种控制对象，它们用 7 个过程调用来完成计算、绘制及移动 Bezier 曲线的各项任务。当你对每个过程都很熟悉后，就可以通过组合不同的例程来缩短这些过程(尽管这样做是不必要的)。

第 4 章中的 Artisan 程序

第 4 章按字母顺序给出控制对象的事件和过程，从 C(ColorBar 图片框事件)开始，直到 N(N_H Clear 过程)结束。

第 5 章中的 Artisan 程序

第 5 章列出从 N(Node 图片框事件)到 Z(Zoom 过程)的程序和控制对象事件。

第 6 章中的 Animator 程序

本章给出一个基本公式，说明线框动画模型的基本知识。

可以给 Animator 程序加入许多功能。在熟悉了此程序后，就可以再加三个图片框控制对象，以执行生成前视图、顶视图和侧视图，或者从其他的视图平面构造线框对象。还可以将每一个单元存入位图中，然后再将这些单元串起来，以生成实时动画。Visual Basic 3.0 专业版的 PicClip VBX 控制对象可以实现该功能。

第 7 章中的多重查找和替换程序

多重查找和替换是每个字处理程序必备的内部功能。我售出的这类简单的查找替换文字编辑器比任何其他程序都要多。它使用了一个很简单的 Loop 过程，只要单击一次按钮就可以查找并替换多个单词。它是如此简单，以至于我几乎没有编入这一功能，因为我以为那些伙计们肯定在他们的字处理程序中包括进了这一基本的功能。其实不然！

第 8 章中的 API 绘图函数

如果你要在你的程序中使用 Windows API 绘图函数，你可以使用一个方便的小工具。我已将该工具写进了本书。这个工具给你展示大多数(但不是全部)API 图形调用的基本用法，请特别注意 PtInRect、FloodFill 和 Ellipse 函数。

第 9 章中的填色与调色程序

如果你有一块能够显示 256 种或更多颜色的 VGA 卡，那么第 9 章就可以让你更好地见识纯调色或喷涂填色。该章展示如何将 255%的 RGB 颜色值转换为 100%的颜色值，如何在一个调色板内显示各种颜色，如何在两种颜色之间产生颜色阴影，如何创建可显示多于 256 种自定义纯色或抖色的调色板，如何用调合的颜色或喷涂的颜色填充复杂对象。

第 10 章中的 Mini-Bézier 程序

Mini-Bézier 程序所用的代码结构与 Artisan 程序用来绘制 Bézier 曲线的结构完全相同。

我把这个简化版本包括进来，是为了让你能够很容易地明白我是怎样逐点绘制 Bézier 曲线的。预绘制曲线所使用的两个过程在采用递归方式绘制 Bézier 曲线的过程中可能并不需要。不过，在两种方法中，算法都是一样的，只是采用预绘制过程速度稍稍快一些。

第 11 章中的绘制圆角程序

在 APIdraw 程序中，可以使用 API 的 RoundedRect 函数绘制带圆角的矩形。如果试图用同样的调用来移动该对象，那么该对象就会在各种绘图模式之间来回切换，并发生闪烁。第 11 章告诉你如何用 Basic 的 Line 和 Circle 语句绘制同样形状，但不闪烁的圆角矩形。

第 12 章中的 Text Alignment 程序

Visual Basic 提供了简单的在窗体上显示文字的方法。当你想在用户图形软件包的演示部分处理各种文字时，Windows API 文字函数在功能上就会显得略有不足。如果不涉及文字的 Glyph 轮廓并设计自己的文字模块，就不会有更简便的方法对文字进行处理。但对这些模块代码的解释会使本书显得过于冗长。本章展示了发生鼠标按下事件时设置插入点的三种简单的文字对齐方法：左对齐，右对齐或文字居中。

第 13 章中的节点连接程序

这是所有绘图包的核心和灵魂。第 2 章的 Artisan 程序中没有包括该程序，这是因为在第 2 章必须有一个明确的目标，想要让节点和对象做些什么。设计 Artisan 程序是为了避免调用 API 函数而只使用标准的 Basic 语言。当涉及填充对象时，就达到了 Basic 语言绘图能力的极限。你必须求助于 Pset 或 Line 语句的循环来填充复杂对象（因为在 Windows 中 Basic 的 Paint 语句被闲置不用了）。这时候，你就必须一头扎进 Windows API 绘图库中。本程序所用的源代码虽然简单，但是仍需要大量的变量存储单元。当你加入可拆分的对象节点和 Bézier 曲线连接时，你就可以深入到代码中。本书的第 1 章介绍了在开始一个绘图程序之前必须对所用的变量和数组作出安排。

第 14 章中的 PrintAPI 和 Scaltest 程序

PrintAPI 是一个简单的小程序，该程序测试打印机的 Windows API 图形函数。它使用 API 调用在一张标准的信笺页上画方框、图及直线。你可以将图形传送至打印机，然后加进额外的代码调整设备（监视器或打印机）的坐标比例。第二个程序（Scaltest ）包含了若干种缩放某个窗口区域的方法（既用了 Basic 方法，又用了 API 方法）。你还应该运行一下附带磁盘中 16MISC 目录下的 APIScale 程序，看一看缩放窗口内选定对象的例子。

第 15 章中的 ArtAPI 程序

ArtAPI 是第 2 章中 Artisan 程序的改进版本。不同之处是该程序用 Windows API 的绘图函数取代了 Basic 语句。从该章的简介开始对程序作了全面的说明。在探讨 ArtAPI 程序之前，一定要读一下第 2 章的 Artisan 程序和第 13 章的节点连接程序。

附录 A

附录 A 包含了 Artisan 程序的全局变量清单，以及该程序中所用的文字和调色板窗体。

附录 B

附录 B 包含了与 Windows NT，Windows 95 有关的信息。

附录 C

附录 C 是对附带磁盘中 16MISC 目录下程序的简介。对这些附加程序代码的解释显示在每个程序的 Visual Basic 代码窗体中。这些程序展示了实现下列功能的例子：

- 怎样创建自定义的工具栏和按钮
- 怎样在 Visual Basic 2.0 和 3.0 中创建自定义光标
- 怎样创建标签风格的对话窗体
- 怎样在运行时确定窗体的位置和比例
- 怎样给其他程序增加菜单项
- 怎样使用少量的或不用资源显示位图图象
- 怎样使用 DPI 打印那些采用自定义比例尺寸的窗体或控制图象
- 怎样创建 Visual Basic 和 API 视窗
- 以及……

图形用户界面(GUI)

一个类似 Windows 的图形用户界面与 Basic 的图形语句相似，因为它采用隐含的代码创建图形对象。Windows 包含了一个 API 函数库，你可以利用这些函数完成超出标准 Basic 语句所能实现的功能。DOS 操作系统中的最后一个 Basic 版本有大约 200 个函数，而 Windows API 库则有 1000 多个。仅在图形库中，你就有若干种方法简单地绘出比直线和圆更复杂的图形，而且不用编写很长的例程就可以达到与 Basic 同样的效果。不过你必须对如何将这些直线和圆(或其他对象)放置在监视器屏幕上有所了解。在你充分理解这些以后，你就可以着手实现 Windows API 中的这些高级功能了，以后的章节中会讲述上述内容。

Basic 和 Windows

你在使用 Visual Basic 开发 Windows 程序方面有了一个良好的开端。到目前为止的最新版本有 400 多条不同的 Basic 语句。将这些和前面提到的 API 函数结合起来，你就可以创建一些强有力的应用程序。Visual Basic 在执行速度方面可能没什么特别，但在快速开发 Windows 程序上，它却得到了一致的首肯。

机器代码、解释器和编译器

有两种计算机语言：低级的和高级的。低级语言通常被称作机器代码或汇编语言。最基础的就是机器代码，它使用一些十六进制的指令直接与 CPU 对话。再向上一级是汇编语言，它使用一些简写(缩写)了的机器过程操作 CPU，这让人稍能容忍。

再向上一级就是高级语言。高级语言仅仅是一种称呼而已，它并不是指你正在使用的语言有多么高级。你不应该对使用这些语言有任何犹豫。Visual Basic 可以被称为高级语言，它被划入解释语言一类。与之相对的就是 C、Pascal 以及类似的语言，它们采用高级的编译技术进行处理。两者之间的差别在于像 Basic 这样的解释程序是将每一条语句翻译成隐含的机器代码之后，CPU 再执行这些代码。Visual Basic 需要有一个单独的文件来将子程序翻译成机器代码，这个文件就是 VBrun，内部翻译器代码就被称作 P-code 解释码。因此你的程序并不真是百分之百编译的 .EXE 文件。

编译语言（例如 C）被转换成汇编程序，生成最终的机器代码。虽然在操作上与 P-code 解释代码类似，但是编译的代码就执行速度而言远远胜过解释的代码。解释的代码读进每一条 Basic 命令语句，然后随着代码的输入每次执行一行。而编译的代码被编译成单个文件。

在开始之前应该知道哪些内容

第 1 章让你理解在使用简单的图形语句时屏幕坐标系统是怎样运作的，还将给你上一堂三角学复习课（让你回忆一下在学校中所学过的数学知识）。不过，此节之后不会有什么小测验了。

在每一段源代码清单之后都有一段对每行代码所作的简短解释。在这些摘要说明中，每一代码行的头几个单词都是以黑体给出，后面再跟一段解释。

你应该已经对 Visual Basic 的基本概念比较熟悉了，本书假设你超出了初学者的水平。不过，书中有些部分还是回顾了一下基本概念，以确保你理解这些基本知识，如果你已经对 Visual Basic 有了基本的理解，那就再好不过了。

程序

为了让所有的程序都运行正常，我付出了极大的努力。许多程序都是用标准的函数和子程序编写的。

测试这些代码使用了许多机器，包括以下这些：i386sx 16，386AMD40，i486DX33，486DCLcyrix 以及 Pentium 60。

鉴于这些程序的图形特性，应该有一台分辨率高于标准 VGA 的监视器。大部分图形程序都是针对 Super VGA 监视器创建的。本书中的全部程序窗体都是设计成在 800×600 分辨率的 SVGA 监视器上运行的。

附注： *所有程序的代码都包括在附带的磁盘中。另外还包括一些其他的程序。有关本书附带的磁盘所包含的内容的信息，请参见附录 C。*

目 录