

WUTP



普通高等学校计算机科学与技术专业新编系列教材

Visual Programming Technology and Application

# 可视化编程技术及应用



陈和平 邵平凡 汤惟 主编

```
#include <stdio.h>
#include <stdio.h>
void main()
void main()
{
    void swap(int * ptr1,int * ptr2);
    void swap(int * ptr1,int * ptr2);
    int x,y, * ptr1, * ptr2;
    int x,y, * ptr1, * ptr2;
    printf("input x,y:");scanf("%d,%d",&x,&y);
    printf("input x,y:");scanf("%d,%d",&x,&y);
    printf("%d\t%d\n",x,y);ptr1=&x;ptr2=&y;
    printf("%d\t%d\n",x,y);ptr1=&x;ptr2=&y;
    if(x<y)
    if(x<y)
    swap(ptr1,ptr2);
    swap(ptr1,ptr2);
    printf("%d\t%d\n",x,y);
    printf("%d\t%d\n",x,y);
}
void swap(int * ptr1,int * ptr2)
void swap(int * ptr1,int * ptr2)
```

武汉理工大学出版社

Wuhan University of Technology Press



普通高等学校计算机科学与技术专业新编系列教材

Visual Programming Technology and Application  
**可视化编程技术及应用**

主 编 陈和平 邵平凡 汤 惟

武汉理工大学出版社

Wuhan University of Technology Press

## 【内容简介】

本书以 C++ Builder 6.0 为蓝本,主要介绍可视化编程的基本概念、技术以及可视化应用程序的设计方法。

本书共分八章。首先简要介绍了可视化编程的有关概念和 C++ 语言基础,然后详细介绍了窗体和常用组件的属性、方法和事件,在此基础上介绍了数据库编程技术及其实例编程方法,并结合实例介绍了多媒体、动态链接库和网络编程技术,最后的附录介绍了应用程序的打包与发布方法。

本书内容丰富,图文并茂,范例极具典型性和实用性,既可作为大专院校计算机专业和相关专业学生学习可视化编程技术的教材,亦可适合于学用 C++ Builder 编程技术的应用程序开发人员使用。

### 图书在版编目(CIP)数据

可视化编程技术及应用/陈和平,邵平凡,汤惟主编. —武汉:武汉理工大学出版社,2005.5

普通高等学校计算机科学与技术专业(本科)新编系列教材

ISBN 7-5629-2292-6

I. 可… II. ①陈… ②邵… ③汤… III. 程序设计-高等学校-教材  
IV. TP311

中国版本图书馆CIP数据核字(2005)第106860号

出版发行:武汉理工大学出版社(武汉市洪山区珞狮路122号 邮编430070)

<http://www.techbook.com.cn>

E-mail: [tiandq@mail.whut.edu.cn](mailto:tiandq@mail.whut.edu.cn) [duanchao@mail.whut.edu.cn](mailto:duanchao@mail.whut.edu.cn)

经 销 者:各地新华书店

印 刷 者:武汉理工大印刷厂

开 本:787×960 1/16

印 张:22.5

字 数:441千字

版 次:2005年5月第1版

印 次:2005年5月第1次印刷

印 数:1~3000册

定 价:32.00元

本书如有缺页、倒页、脱页等印装质量问题,请向出版社发行部调换。本社购书热线电话:(027)87397097 87394412

普通高等学校  
计算机科学与技术专业新编系列教材  
编审委员会

顾问:

卢锡城 周祖德 何炎祥 卢正鼎 曾建潮  
熊前兴

主任委员:

严新平 钟 珞 雷绍锋

副主任委员:

李陶深 鞠时光 段隆振 王忠勇 胡学钢  
李仁发 张常年 郑玉美 程学先 张翠芳  
孙成林

委员:(以姓氏笔画为序)

王 浩 王景中 刘任任 江定汉 朱 勇  
宋中山 汤 惟 李长河 李临生 李跃新  
李腊元 李朝纯 肖俊武 邱桃荣 张江陵  
张继福 张端金 张增芳 陈和平 陈祖爵  
邵平凡 金 聪 杨开英 赵文静 赵跃华  
周双娥 周经野 钟 诚 姚振坚 徐东平  
黄求根 郭庆平 郭 骏 袁 捷 龚自康  
崔尚森 蒋天发 詹永照 蔡启先 蔡瑞英  
谭同德 熊盛武 薛胜军

秘书长:田道全

总责任编辑:段 超 徐秋林

## 出版说明

当今世界已经跨入了信息时代,计算机科学与技术正在迅猛发展。尤其是以计算机为核心的信息技术正在改变整个社会的生产方式、生活方式和学习方式,推动整个人类社会进入信息化社会。为了顺应时代潮流,适应计算机专业调整及深化教学改革的要求,充分考虑到不同层次高校的教学现状,满足广大高校的教学需求,武汉理工大学出版社经过广泛调研,与国内近 30 所高等院校的计算机专家进行探讨,决定组织编写“普通高等学校计算机科学与技术专业新编系列教材”。

我们在组织编写新编本套系列教材时,以培养现代化高级人才为重任,以提高学生综合素质、培养学生应用能力和创新能力为目的,以面向现代化、面向世界、面向未来为准绳,注重系列教材的特色和实用性,反映最新的教学与科研成果,体现本专业的时代特征。同时,面对教育改革的需要、人才的需要和社会的需要,在编写本教材时,借鉴、学习国外一流大学的先进教学体系,结合国内的实际需要,吸取具有先进性、实用性和权威性的国外教材的精华,以更好地促进国内教材改革顺利进行。从时代和国际竞争要求的高度来思考,为打造一套高起点、高水平、高质量的系列教材而努力。

本套教材具有以下特色:

与时俱进,内容科学先进——充分体现计算机学科知识更新快的特点,及时更新知识,确保教材处于学科前沿,以拓宽学生知识面,培养学生的创新能力。

紧跟教学改革步伐,体现教学改革的阶段性成果——符合全国高校计算机专业教学指导委员会、中国计算机学会教育委员会制订的“计算机学科教学计划 2000”的内容要求。

实现立体化出版,适应教育方式的变革——本套教材努力使用和推广现代化的教学手段,凡有条件的课程都准备组织编写、制作和出版配合教材使用的实验、习题、课件、电子教案及相应的程序设计素材库。

本套教材首批 25 种预定在 2003 年秋季全部出齐。我们的编审者、出版者决不敢稍有懈怠,一定高度重视,兢兢业业,按最高的质量标准工作。教材建设是我们共同的事业和追求,也是我们共同的责任和义务,我们诚恳地希望大家积极选用本套教材,并在使用过程中给我们多提意见和建议,以便我们不断修订、完善全套教材。

武汉理工大学出版社

2002 年 10 月

# 前 言

20世纪90年代以来,可视化程序设计技术随着以Windows为代表的图形界面的出现而受到广泛重视,其“所见即所得”的特征在程序设计开发环境中体现为基于面向对象和组件的程序设计方法。

目前,各种可视化的开发工具不断涌现,从微软公司的Visual系列到Borland公司的C++ Builder等,一系列功能强大的可视化编程工具既简化了基于窗口的应用软件的开发工作,同时也使众多的编程人员得以从大量枯燥繁琐的界面设计和难度较大的程序底层支持功能实现中解脱出来,将其主要精力集中于程序核心逻辑及用户业务逻辑的实现上。

Borland公司的Turbo C/C++和Borland C/C++曾经是很多程序员踏入C和C++程序设计殿堂的入门语言,并在实际系统的开发中得到广泛应用。C++ Builder是Borland公司继Borland C++和Delphi之后开发的又一个通用的Windows应用程序开发工具,为使用者提供了强大的RAD(Rapid Application Development,快速应用程序开发)功能,与当今主流的C++开发工具(包括Microsoft的Visual C++等)具有良好的兼容性,任何一种采用C++开发工具实现的功能都可以在C++ Builder中高效实现。

C++ Builder以其具有的面向对象、可视化、组件齐全和简明易用的特性为多媒体应用、数据库应用和网络应用等提供了一个便捷、稳定和高效的应用开发环境,得到广大程序员的普遍认可。同时,C++ Builder上手也比较容易,是学习可视化程序设计方法的理想途径和工具。

本书作为高校计算机及相关专业的可视化程序设计教材,为具有初步的C/C++程序设计基础的学生所编写。希望通过本书的学习,使学生能较快地了解可视化程序设计的基本概念,并较好地掌握运用可视化编程技术进行实际应用系统的开发方法。本书也可以满足初、中级用户使用C++ Builder开发Windows应用系统的需要。

根据以上设想,作者在内容上做了周密安排。本书首先介绍了可视化编程的基本思想和C++ Builder 6.0的特点与定位,从第2章开始,每章都采用精心设计的具体应用实例来引导读者循序渐进、深入浅出地学习和掌握可视化程序设计方法,在对C++语言基础和C++ Builder的常用基本组件进行了适度介绍

之后,用较多的篇幅重点介绍了如何运用 C++ Builder 进行数据库应用系统开发、多媒体应用编程和网络应用编程等内容,并对动态链接库编程及应用程序的发布方法进行了专题讨论,使读者在学习完本书的主要内容之后能快速地切入 C++ Builder 的高级应用开发。

本书每章所附的思考题和习题是本书的重要组成部分,读者在学习时应予以充分重视。可视化程序设计方法不是看会的,必须通过自己动手实践才能逐步地从中悟出真谛。

关于本书中源代码的电子文档,读者可以从 <http://www.info.wust.edu.cn> 下载获得。

本书是集体智慧的产物,参加编写工作的有陈和平(第 1 章),邵平凡(第 2 章、第 5 章和附录),袁嵩(第 3 章),邓宏涛(第 4 章),黄颖(第 6 章),汤惟(第 7 章),李晓卉(第 8 章)。本书由陈和平、邵平凡、汤惟任主编,负责全书的组织和定稿。

本书在编写和出版过程中得到了武汉理工大学出版社田道全副社长和徐秋林副编审的大力支持和指导,并参考了许多作者的优秀著作和文献,编者在此真诚地向他们表示感谢。

由于水平有限,加之写作时间匆促,书中谬误和不足之处,敬请读者批评指正。

编 者

2005 年 6 月于武汉



# 目 录

<b>1 概述</b> .....	(1)
1.1 可视化编程概述 .....	(1)
1.2 可视化编程工具——Borland C++ Builder 6.0 .....	(3)
1.2.1 C++ Builder 的特点 .....	(4)
1.2.2 C++ Builder 的定位 .....	(4)
1.3 C++ Builder 6.0 的集成开发环境 .....	(6)
1.4 C++ Builder 6.0 工程的建立、运行与调试步骤 .....	(11)
1.4.1 Borland C++ Builder 6.0 一般编程步骤 .....	(11)
1.4.2 Hello, world! .....	(12)
习题与思考题 .....	(15)
<b>2 C++语言基础</b> .....	(17)
2.1 第一个 C++ 程序 .....	(17)
2.2 数据类型 .....	(20)
2.2.1 基本数据类型 .....	(20)
2.2.2 复杂数据类型 .....	(21)
2.3 运算符 .....	(23)
2.3.1 运算符分类 .....	(23)
2.3.2 表达式 .....	(24)
2.3.3 运算符的优先级与结合性 .....	(24)
2.4 基本语法 .....	(26)
2.4.1 复合语句 .....	(26)
2.4.2 分支语句 .....	(26)
2.4.3 循环语句 .....	(27)
2.4.4 函数 .....	(28)
2.5 面向对象的程序设计基础 .....	(31)
2.5.1 类和对象 .....	(31)
2.5.2 构造函数与析构函数 .....	(34)



---

2.5.3	类的继承	(37)
2.5.4	函数重载与运算符重载	(41)
2.5.5	虚函数与多态性	(42)
2.5.6	异常处理	(48)
2.5.7	try/finally 处理语句	(51)
	习题与思考题	(52)
<b>3</b>	<b>窗体和常用组件</b>	<b>(54)</b>
3.1	理解 VCL	(55)
3.1.1	属性	(55)
3.1.2	方法	(55)
3.1.3	事件	(55)
3.2	窗体和组件	(56)
3.2.1	窗体的属性	(56)
3.2.2	窗体的方法	(60)
3.2.3	窗体的事件	(60)
3.2.4	组件	(62)
3.3	文本组件	(63)
3.3.1	Edit 组件	(63)
3.3.2	MaskEdit 组件	(65)
3.3.3	Memo 组件	(66)
3.3.4	RichEdit 组件	(67)
3.4	按钮组件	(72)
3.4.1	Button 组件	(72)
3.4.2	BitBtn 组件	(73)
3.4.3	SpeedButton 组件	(74)
3.4.4	RadioButton 组件	(76)
3.4.5	CheckBox 组件	(76)
3.5	列表组件	(77)
3.5.1	ListBox 组件	(77)
3.5.2	ComboBox 组件	(82)
3.5.3	CheckListBox 组件	(84)
3.5.4	TreeView 组件	(85)
3.5.5	Listview 组件	(89)
3.5.6	ImageList 组件	(90)
3.5.7	DateTimePicker 组件	(90)

---

3.6	分组组件	(91)
3.6.1	GroupBox 组件	(91)
3.6.2	RadioGroup 组件	(92)
3.6.3	Panel 组件	(92)
3.6.4	ScrollBar 组件	(92)
3.6.5	PageControl 组件	(92)
3.6.6	TabControl 组件	(93)
3.6.7	HeaderControl 组件	(93)
3.7	信息反馈组件	(93)
3.7.1	Label 组件	(94)
3.7.2	StaticText 组件	(95)
3.7.3	ProgressBar 组件	(96)
3.7.4	StatusBar 组件	(98)
3.8	菜单、工具栏和状态栏	(98)
3.8.1	菜单	(98)
3.8.2	工具栏	(106)
3.9	对话框组件	(109)
3.9.1	TCommonDialog	(110)
3.9.2	OpenDialog	(111)
3.9.3	SaveDialog	(114)
3.9.4	OpenPictureDialog	(114)
3.9.5	SavePictureDialog	(115)
3.9.6	FontDialog	(115)
3.9.7	ColorDialog	(117)
3.9.8	PrintDialog	(118)
3.9.9	PrinterSetupDialog	(120)
3.9.10	FindDialog	(121)
3.9.11	ReplaceDialog	(122)
3.9.12	对话框综合实例	(122)
	习题与思考题	(126)
4	数据库编程技术	(130)
4.1	数据库技术基础	(130)
4.1.1	数据库技术的发展	(130)
4.1.2	数据库、数据库管理系统和数据库系统	(131)
4.1.3	数据模型	(131)

---

4.1.4	关系数据模型和关系数据库 .....	(132)
4.1.5	结构化查询语言 SQL .....	(134)
4.1.6	关系数据库应用系统设计 .....	(139)
4.2	C++ Builder 6.0 数据库访问技术 .....	(140)
4.3	C++ Builder 6.0 数据库开发工具 .....	(142)
4.3.1	相关概念 .....	(142)
4.3.2	Database Desktop .....	(143)
4.3.3	SQL Explorer .....	(146)
4.3.4	SQL Builder .....	(148)
4.3.5	SQL Monitor .....	(150)
4.3.6	BDE Administrator .....	(150)
4.3.7	数据库向导 .....	(153)
4.4	C++ Builder 6.0 数据库组件 .....	(163)
4.4.1	常见的数据集组件 .....	(163)
4.4.2	常见的数据控制组件 .....	(165)
4.4.3	常见的数据源组件 .....	(166)
4.4.4	常见的数据库组件应用 .....	(166)
4.4.5	ADO 组件 .....	(167)
	习题与思考题 .....	(169)
<b>5</b>	<b>数据库应用系统实例 .....</b>	<b>(170)</b>
5.1	基于 BDE 的桌面型 MIS 实例 .....	(170)
5.1.1	总体设计 .....	(170)
5.1.2	创建数据表 .....	(171)
5.1.3	创建数据录入界面 .....	(174)
5.1.4	创建主界面 .....	(177)
5.1.5	数据录入模块设计 .....	(181)
5.1.6	数据查询模块设计 .....	(202)
5.1.7	报表输出模块设计 .....	(211)
5.1.8	用户管理模块设计 .....	(215)
5.1.9	用户登录模块设计 .....	(219)
5.2	基于 ADO 的 C/S 模式 MIS 实例 .....	(223)
5.2.1	总体设计 .....	(223)
5.2.2	创建数据表 .....	(223)
5.2.3	主界面设计 .....	(225)
5.2.4	数据录入模块设计 .....	(227)

---

5.2.5	数据查询模块设计 .....	(233)
5.2.6	报表输出模块设计 .....	(236)
	习题与思考题 .....	(238)
<b>6</b>	<b>多媒体编程技术 .....</b>	<b>(240)</b>
6.1	基本概念 .....	(240)
6.1.1	图形与图像 .....	(240)
6.1.2	音频 .....	(241)
6.1.3	动画、文本和视频 .....	(242)
6.2	图形绘制 .....	(243)
6.2.1	图形组件 .....	(243)
6.2.2	图形绘制常用类 .....	(246)
6.2.3	文本绘制 .....	(256)
6.3	图像处理 .....	(257)
6.3.1	图像组件 .....	(257)
6.3.2	图像处理常用类 .....	(258)
6.3.3	图像旋转与翻转 .....	(259)
6.3.4	图像格式转换 .....	(262)
6.3.5	图像特效处理 .....	(266)
6.4	动画播放 .....	(267)
6.5	多媒体播放器 .....	(270)
	习题与思考题 .....	(274)
<b>7</b>	<b>动态链接库编程技术 .....</b>	<b>(276)</b>
7.1	动态链接库编程基础 .....	(276)
7.2	动态链接库的创建 .....	(277)
7.2.1	创建不使用库资源的 DLL .....	(277)
7.2.2	创建使用 VCL 的 DLL .....	(280)
7.3	动态链接库的调用 .....	(286)
7.3.1	静态调用 .....	(287)
7.3.2	动态调用 .....	(289)
7.4	资源型动态链接库的创建与实现 .....	(291)
7.4.1	资源文件的编写方法 .....	(291)
7.4.2	资源型动态链接库的创建 .....	(292)
7.4.3	资源型动态链接库的调用 .....	(293)
	习题与思考题 .....	(298)
<b>8</b>	<b>网络编程技术 .....</b>	<b>(299)</b>

---

8.1 网络基本知识 .....	(299)
8.1.1 TCP/IP 协议族 .....	(299)
8.1.2 客户/服务器编程模型 .....	(300)
8.1.3 套接字和端口 .....	(301)
8.2 网络组件 .....	(301)
8.2.1 Internet 组件页面 .....	(301)
8.2.2 FastNet 组件页面 .....	(304)
8.3 网络组件应用及编程 .....	(307)
8.3.1 客户/服务器通信编程实例 .....	(307)
8.3.2 Web 浏览器编程实例 .....	(313)
8.3.3 FTP 客户端编程实例 .....	(318)
习题与思考题 .....	(334)
附录 应用程序的打包与发布 .....	(336)
参考文献 .....	(348)

# 1 概 述

## 本章提要

本章概述了可视化编程技术的基本概念,详细介绍了 Borland C++ Builder 6.0 的集成开发环境及其特点,并结合实例讲解了使用 C++ Builder 6.0 进行可视化编程的基本方法和一般步骤。读者在学习完本章后,对可视化编程技术会有一个初步的了解和认识,并对学习和使用 C++ Builder 产生较浓厚的兴趣。

## 1.1 可视化编程概述

以图形方式观察和认识现实世界的客观事物,是人类最为便捷的一种认知方式。各种可视化编程工具不断发展,使得当今大多数的程序开发人员可以利用一系列高效的、具有良好可视化效果的编程工具来开发专业级的 Windows 和网络应用软件。

程序开发环境和工具的发展来源于程序设计方法的不断优化。就程序设计方法而言,主要有以下三种方法:

### (1) 结构化程序设计

结构化程序设计是针对面向过程的编程语言(如 Pascal、C 等)所提出的一种程序设计原则,包括使用自顶向下的分析方法和结构化的流程设计。结构化程序设计方法是一种面向过程处理的设计方法,该方法把数据和过程分为相互独立的实体,即所谓的“程序=数据结构+算法”的概念。结构化程序设计方法在 20 世纪 70 年代末至今的软件开发过程中发挥了很大的作用。但随着计算机技术、软件技术的飞速发展和计算机应用的日益普及,应用系统的复杂度愈来愈高,结构化程序设计方法的不足之处也随之显现出来。

首先,程序员在编程时必须考虑算法何时在何种数据结构上进行何种操作,对于不同的数据结构即使进行相同的处理或对于相同的数据结构进行不同的处理均要编制不同的程序,从而使得程序的可重用性欠佳;其次,由于数据和过程相分离,每一个过程与所对应的数据必须始终保持密切的关系,因此,维护程序与数据的相容性成了程序员的一种沉重负担;再者,数据和过程的分离还会带来系统维护上的困难,只要对某一数据结构进行修改,就必然要修改涉及该数据结构的所有程序模块,以致造成一系列的连锁反应;最后,随着软件开发规模的不断扩大,目标系统变得更为复杂,仅仅单靠使用结构化的程序设计方法,显然已不可能得心应手地驾驭软件开发的整个过程。

### (2) 面向对象的程序设计

面向对象的程序设计方法与技术起源于面向对象的编程语言(如C++、Java等),至今已成为计算机应用开发领域的主流。究其原因,主要是由于面向对象技术具有对现实世界最直接、最自然的描述以及极强的抽象与表达能力,并能将数据和对数据的操作封装到一起成为一个与现实世界中真实实体相对应的“软件实体”。有关面向对象的程序设计的其他特点(如继承性、多态性等)将在第2章中详细介绍。

面向对象的程序设计方法大大减轻和简化了软件开发工作,优化了程序结构,较为符合开发人员的思维习惯,在较大程度上适应了大规模化软件开发工作的需求。

### (3) 组件对象模型程序设计

组件对象模型程序设计是在面向对象程序设计技术基础上发展起来的。组件对象模型程序设计技术的目标是实现软件的功能模块化、编程语言的无关性和操作系统的无关性等,采用该技术,可极大地提高程序代码的可重用性,增加软件的可扩展性。

实际应用过程中,上述三种程序设计方法之间不存在着相互替代的关系,而是呈现一种“你中有我,我中有你”的包含关系。

程序设计思想与方法的不断改进直接促进了编程环境的更新换代。20世纪90年代初,可视化编程环境的出现,无疑给众多的程序员带来了福音。基于可视化对象的操作简化了传统的命令行方式,取而代之的是人性化的图形用户界面GUI(Graphical User Interface)。在图形用户界面下,用户不必再死记硬背各种命令,而仅以鼠标点击图形界面对象便可运行程序或进行各项操作。要实现这一点,若不借助于任何可视化开发工具,对于一个普通的程序员来说,将是一种极大的挑战,因为可视化窗口的程序设计要比传统的程序设计复杂得多。例如,在Windows环境下,采用C语言并配以SDK(Software Development Kit,软件开发工具包)来开发基于可视化窗口的程序,仅开启一个简单的窗口就需编

写 80 多行程序,如欲开发更为复杂的系统,所要耗费的人力和时间就可想而知了。正是在这种新的需求的推动下,可视化的编程工具(Visual Programming Tool)应运而生。

多媒体技术和图形图像技术的进步,使可视化技术得到广泛的重视。可视化技术的最大特点就是原来抽象的数字、表格、功能逻辑等用直观的图形、图像的形式表现出来。其方法是在建立用户界面时,不必编写程序来描述输入和输出界面的外观和配置,而是直接使用可视化编程环境提供的各种开发工具,通过对各种图形化对象的创建和设置来完成,在程序设计阶段即可实现所谓的“*What You See Is What You Get*”(所见即所得)的效果。

可视化编程是可视化技术的重要应用之一。所谓可视化编程,是指在软件开发过程中,用直观的具有一定含义的图形化的对象取代原来完全依赖人工进行的逐行编辑、运行、调试等操作。软件开发过程表现为以鼠标选取和拖放图形化的对象以及指定对象的属性并设计对象的行为(程序代码)的过程。这种可视化的编程方法易学易用、方便快捷,并可大幅度地提高编程工作的效率。

目前,各种可视化和基于面向对象的开发工具不断涌现,运行环境各存差异,其优势和不足各有千秋。从微软公司的 Visual 系列到 Borland 公司的 C++ Builder 等,一个又一个功能强大的可视化编程工具为编程人员简化了基于窗口软件的开发工作,使编程人员从以往枯燥繁琐的界面设计中解脱出来,从而能将主要精力集中到用户业务逻辑的实现上。

## 1.2 可视化编程工具——Borland C++ Builder 6.0

市面上的可视化编程工具可谓门类繁多、琳琅满目,常见的有:Microsoft 公司的 Visual Basic、Visual C++ 和 Visual Studio. NET、Borland 公司的 C++ Builder 和 Delphi,还有 Sybase 公司的 Powerbuilder 等等,让人颇有眼花缭乱之感。网上或专业报纸杂志上时常有人询问究竟应该学习哪一种编程工具为好的问题,答案自然是五花八门的,其间也夹杂了不少答复者个人的偏好。对于此问题,较为合适的观点应该是:从广义的角度来评价,多数可视化编程工具总体上差别并不太大,因此,选择学习哪一种编程工具都不会是白花工夫。任何一种流行的编程工具之所以能流行肯定会有其独到之处,只是学习者需要根据自己的基础和条件以及个人的目标来选择适合自己学习的编程工具,并养成良好的编程习惯。精通了一种可视化平台及其软件开发技术并达到了一定的熟练程度之后,再去学习其他的可视化编程工具自然就会觉得游刃有余了。对于大多数已经接触过 C/C++ 语言或有些初步编程基础的学习者来说,Borland C++ Builder 不失为一种较佳的选择。



Borland C++ Builder 有着诸多的优点和鲜明的个性,以下仅就其主要特点及定位加以介绍。

### 1.2.1 C++ Builder 的特点

Borland C++ Builder 是 Borland 公司推出的基于 C++ 语言的快速应用程序开发(Rapid Application Development, RAD)系列工具,是组件技术和面向对象的高效语言 C++ 融合的产物。Borland C++ Builder 6.0(以下简称为 C++ Builder 或 BCB)继承了以前版本的诸多优点,充分利用了已经发展成熟的 Delphi 的可视化组件库(Visual Component Library, VCL),并配以优秀的编译器和调试器,组成了一个功能强、效率高且非常成熟的可视化应用程序开发环境。C++ Builder 6.0 最主要的特点表现在以下几个方面:

#### (1) 高性能的 C++ 开发工具

C++ Builder 是基于 C++ 的编程工具,具有较高的编译、连接和执行速度,并带有双编译器引擎,不仅可以编译 C/C++ 程序,还能够编译 Object Pascal 语言程序。

#### (2) 优秀的可视化应用程序开发工具

C++ Builder 是优秀的可视化应用程序开发工具之一,它所提供的类别齐全的可视化程序开发工具集和功能强大的编程环境,不仅能使程序员明显提高软件开发的工作效率,而且还能使其对开发工具本身的学习周期也大为缩短。

#### (3) 强大的数据库应用程序开发功能

C++ Builder 提供了丰富的数据库感知控件和多种数据库引擎,为程序员开发功能强大的数据库应用程序提供了强有力的支持。C++ Builder 除了支持 Microsoft 的 ADO(Active Data Object)数据库连接技术,还提供了一种自己开发的数据库连接技术——BDE(Borland Database Engine)数据库引擎。

#### (4) 强大的网络编程能力

C++ Builder 6.0 具有众多的 Internet 应用程序开发组件,如 WebBroker、CppWebBrowser 及 WinSocks 等,所提供的这些组件基本上涵盖了 Internet 应用的绝大部分功能,程序员利用这些组件可以方便地建立自己的 Internet 应用程序。

### 1.2.2 C++ Builder 的定位

C++ Builder 定位于快速应用开发工具层面上,也就是说,C++ Builder 主要用于基于 Windows 环境的应用程序开发。由于 C++ Builder 具备强大的可视化开发能力,并且含有众多可以直接使用的 VCL(Visual Component Library)组件,因而使得开发者可以专注于逻辑代码设计,而不需要花太多精力