

程序设计经典书系



Borland C++ Builder 6

程序设计经典

详细介绍Borland C++ Builder语法、程序设计基本素养、Windows Form应用程序设计、窗口事件探讨、文件存取技巧、声音的基本原理、多媒体程序设计、指针与字符串深入探讨、对象与类及Paradox、Access、SQL Server数据库程序设计和Internet程序设计、Borland C++ Builder应用程序的封装。各单元都有精心设计的实用范例，让您体会出其设计精神。

余明兴 吴明哲 黄世阳 编著
黄丰隆 纪旺松 潘能煌



科学出版社
www.sciencep.com

程序设计经典书系

Borland C++ Builder 6

程序设计经典

余明兴 吴明哲 黄世阳
黄丰隆 纪旺松 潘能煌 编著

科学出版社

北京

图字: 01-2003-6989 号

内 容 简 介

本书共 18 章, 对 Borland C++ Builder 6 作了较为完整的介绍, 内容包括其开发环境、设计流程、数据类型与操作符、输入输出、选择结构、循环结构、绘图、键盘及鼠标事件、数组、指针与字符串、函数、多模块程序、文件、多媒体、链表、对象和类、数据库及网络编程。本书内容详尽, 范例较为经典。

本书可作为中、初级程序员用书或本科教材。

本书繁体字版名为《Borland C++ Builder 6 程式设计经典》, 由文魁资讯股份有限公司出版, 版权属吴明哲所有。本书简体字中文版由文魁资讯股份有限公司授权科学出版社独家出版。未经本书原出版者和本书出版者书面许可, 任何单位和个人均不得以任何形式或手段复制或传播本书的部分或全部。

图书在版编目(CIP)数据

Borland C++ Builder 6 程序设计经典/余明兴等编著. —北京: 科学出版社, 2004

(程序设计经典书系)

ISBN 7-03-012399-9

I.B... II.余 III.C 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(2003)第 096126 号

策划编辑: 李佩乾 / 责任编辑: 王日臣

责任印制: 吕春珉 / 封面制作: 东方人华平面设计部

科学出版社 出版

北京东黄城根北街16号

邮政编码: 100717

<http://www.sciencep.com>

新蕾印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2004 年 1 月第 一 版 开本: 787×1092 1/16

2004 年 1 月第一次印刷 印张: 36 1/2

印数: 1—4 000 字数: 557 000

定价: 65.00 元 (含光盘)

(如有印装质量问题, 我社负责调换〈路通〉)

序

自从微软公司推出 Windows 操作系统,由于它提供了简单易用的图形化界面,取代了传统 DOS 环境下要强记那些冷冰冰的命令,而且在 Windows 下所开发的应用程序具有较强的用户亲和力,因此吸引了全球无数个人计算机用户的眼光,形成计算机主流。

UNIX 操作系统是用 C 语言开发的,由于 C 语言具有结构化、节省内存、执行效率高等特点,因此,个人计算机或工作站程序开发人员大都使用 C 语言来开发各种系统或应用程序,无形中使得 C 语言成为程序语言的主流。尤其 C 语言的延伸——C++ 作为一种面向对象的程序设计语言,适合开发大型应用程序。笔者经过多年授课经验,觉得 C++ Builder 的集成开发环境与 Visual Basic 极相似,学过 Visual Basic 的人很容易上手。因此 C++ Builder 是初学者用来学习 C++ 程序设计的最佳途径。因此,目前很多程序设计人员仍钟情于 Borland C++ Builder。

近年来,市面上有关 C++ Builder 的书籍,大都是写给有基础的程序设计人员使用,C++ 初学者是较难接受的。因此,我们一直在寻求一本难易适中的 C++ Builder 入门书,让初学者能在 Windows 环境下学习 C++ 程序设计,而且又能将 C++ Builder 所提供的常用组件发挥出来,让老师能够拿来当教科书,学生学起来快速轻松无挫折感。然而翻遍市面上的计算机书籍都无法兼顾。因此笔者尝试以循序渐进方式,引导初学者由最基本的 C 语言程序设计并融入 C++ Builder 提供的常用组件去设计 Windows 应用程序。希望本书能对初学者有所帮助。C++ Builder 6 功能强大,单单一本书无法全部介绍完。本书主要是针对在 Windows 环境下学习 C++ 程序设计的初学者而编写的,较偏重基本程序设计能力的培养以及对 C++ Builder 常用组件的应用。由于 C++ Builder 提供的组件有数百个,无法在本书中详细介绍,我们只挑选初学者较常用到以及较重要的来介绍。

本书所有范例程序均存放在附书光盘中。凡是努力过的必会有收获,本书经过用心编写,自然有其特色,但虽经多次校对,难免百密一疏,希望读者不吝指正,以期再版时修订。

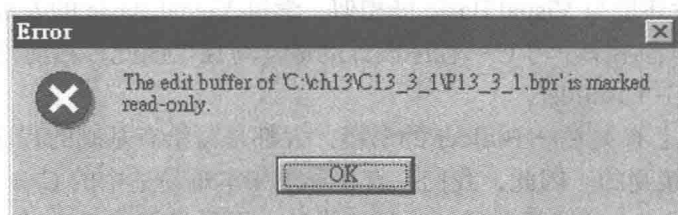
余明兴、吴明哲、黄世阳

黄丰隆、纪旺松、潘能煌 谨记

附书光盘使用说明

本书所有程序范例都收录在附书光盘中，以方便读者参阅使用。

举例说明文件的存放规律：第 13.3 节的第一个范例程序存放在 `ch13\c13_3_1` 文件夹内，同一节第二个程序存放在 `ch13\c13_3_2` 文件夹，依次类推。由于范例程序执行时会产生一些相关文件，并会保存到磁盘驱动器中。因此，程序是无法直接在只读的光盘下执行的，建议你想执行该程序时，先将该程序的整个文件夹或整章的文件夹拷贝到硬盘 C 中。由于拷贝附书光盘中的文件夹或程序到硬盘 C 内都会默认转换成只读文件，因此当你打开该程序项目时，会出现下列信息，表示该文件是只读文件。



因此需要取消文件的只读属性。

目 录

第1章 C++ Builder 6 集成开发环境	1
1.1 程序语言的演进	1
1.2 C 语言的演进	2
1.3 C++ Builder 6 介绍	2
1.4 对象和事件基本观念	5
1.5 软硬件需求	7
1.6 鼠标的基本操作用语	7
1.7 C++ Builder 6 的启动	8
1.8 C++ Builder 6 集成开发环境	10
1.9 窗体上的控件	14
1.10 项目管理员	22
1.11 项目群组	25
1.12 离开 C++ Builder	27
第2章 程序设计基本流程	28
2.1 如何开发应用程序	28
2.2 如何编辑一个简单的程序	29
2.3 执行程序	40
2.4 保存程序文件	41
2.5 删除复选框	43
2.6 打开复选框	44
2.7 显示窗体或程序代码	44
2.8 封装程序	46
2.9 习题	54
第3章 运算符与表达式	55
3.1 内存存储单位	55
3.2 变量的命名规则	56
3.3 基本数据类型	57
3.4 C++ 运算符	59
3.5 文字数据表示法	62
3.6 数字系统	63
3.7 数字系统的转换	65
3.8 习题	66

第 4 章 输出与输入界面	68
4.1 Form 窗体	68
4.2 Label 标签组件	75
4.3 Button 按钮组件	76
4.4 Edit 编辑组件	79
4.5 数值与字符串间的转换函数	80
4.6 对象与变量的命名	81
4.7 Edit 组件的应用	83
4.8 Change 事件应用	85
4.9 InputBox 函数	86
4.10 MessageBox 函数	87
4.11 ShowMessage 函数	92
4.12 实例	93
4.13 习题	94
第 5 章 选择结构	96
5.1 关系表达式与逻辑表达式	96
5.2 算法	97
5.3 结构化程序设计	99
5.4 选择语句	103
5.5 多向选择	110
5.6 CheckBox 复选框组件	113
5.7 Radio Button 选项按钮组件	116
5.8 嵌套选择语句	120
5.9 GroupBox 框架组件	124
5.10 Radio Group 组件	133
5.11 实例	137
5.12 习题	140
第 6 章 循环结构	141
6.1 for 循环语句	141
6.2 break 语句与 continue 语句	145
6.3 Image 图像组件	147
6.4 条件循环	155
6.5 嵌套循环	159
6.6 局部变量与全局变量	160
6.7 Timer 定时器组件	163
6.8 ScrollBar 滚动条组件	168
6.9 Panel、SpeedButton、BitBtn 组件	174
6.10 习题	177

第7章 绘图	179
7.1 Canvas 画布	179
7.2 颜色设置	180
7.3 绘图方法	183
7.4 Shape 形状组件	189
7.5 LoadFromFile 与 SaveToFile 函数	194
7.6 实例	198
7.7 习题	201
第8章 键盘与鼠标事件	202
8.1 键盘常用事件介绍	202
8.2 KeyPress 事件	202
8.3 KeyDown 和 KeyUp 事件	206
8.4 鼠标事件介绍	212
8.5 Click 事件	213
8.6 DblClick 事件	214
8.7 MouseDown、MouseUp 与 MouseMove 事件	214
8.8 拖曳与放置	220
8.9 实例	223
8.10 习题	227
第9章 认识数组	230
9.1 数组	230
9.2 排序与查找	235
9.3 结构	242
9.4 ListBox 与 ComoboBox 组件	246
9.5 菜单的设计	255
9.6 习题	265
第10章 指针与字符串	267
10.1 指针是什么	267
10.2 指针的声明	267
10.3 指针运算符	268
10.4 指针与数组	272
10.5 字符与字符串	274
10.6 AnsiString 类型	277
10.7 实例	284
10.8 习题	286
第11章 函数	288
11.1 函数	288
11.2 系统函数	289
11.3 一般函数	296

11.4	传值调用与传址调用	301
11.5	在函数间传递数组数据	305
11.6	递归函数	306
11.7	范围运算符	309
11.8	Define 宏与 Inline 函数	310
11.9	Function Overloading	312
11.10	习题	313
第 12 章	多单元项目程序设计	316
12.1	多单元的项目	316
12.2	如何调用应用程序 (一)	328
12.3	如何调用应用程序 (二)	330
12.4	习题	334
第 13 章	文件	335
13.1	文件的种类	335
13.2	文件的存取	336
13.3	顺序文件	340
13.4	随机文件	346
13.5	磁盘驱动器、目录和文件列表框	355
13.6	习题	359
第 14 章	多媒体 AVI 与 Wav 播放	361
14.1	语音基本知识	361
14.2	播放 AVI 影音文件	367
14.3	在 Windows 下播放与录制语音文件	368
14.4	多媒体播放组件	372
14.5	内存放音	383
14.6	Wav 文件格式说明	392
14.7	TOleContainer 与多媒体	394
第 15 章	指针与链表	398
15.1	动态数据结构	398
15.2	链表	400
15.3	实例	419
15.4	习题	424
第 16 章	对象与类	426
16.1	对象基本概念	426
16.2	类的定义	432
16.3	数据成员的初值设置与存取	435
16.4	类数组与类指针	438
16.5	对象间自变量的传递	440
16.6	Constructor 与 Destructor 函数	442

16.7	静态成员	447
16.8	New 和 Delete 运算符	450
16.9	This 指针	453
16.10	Overloading 运算符	454
16.11	Friend 函数与 Friend 类	457
16.12	类继承	460
16.13	VCL 对象继承实例	465
第 17 章	Paradox、Access 与 SQLServer 数据库程序设计	468
17.1	数据库介绍	468
17.2	数据库基本概念	469
17.3	建立数据库	469
17.4	利用数据库向导建立关联	481
17.5	数据感知	499
17.6	使用数据库命令编写程序	507
17.7	存取 Access 2000& 2002 数据库	518
17.8	存取 SQL Server 2000 数据库	526
17.9	习题	532
第 18 章	Internet 程序设计	534
18.1	网络组件介绍	534
18.2	浏览器的制作	536
18.3	FTP 组件介绍	541
18.4	FTP 实例	543
18.5	更多网络组件实例	556
附录 A	ASCII 表	580
附录 B	虚拟按键对照表	566
C++ Builder 6	常用组件表	567
C ++ Builder 6	主要用字索引表	570

第1章 C++ Builder 6 集成开发环境

自从微软公司推出 Windows 后,由于它提供了易学易用的图形化界面,替换了以往传统 DOS 环境下要死记那些冷冰冰的命令,而且在 Windows 下所开发的应用程序,其操作方式都统一,无形中提高了用户的亲和力,吸引了全球无数个人计算机用户的眼光,形成计算机的主流,这对个人计算机用户不啻是一大福音。

同时许多软件公司为了让学习程序设计的初学者,也能轻松地在 Windows 环境下编写程序,陆续推出具有集成开发环境 (IDE) 的工具。通过这些工具,在短时间内不必编写程序便能建构出所需要的输出输入接口,而设计者只要着重在程序主体上的流程,不但缩短了程序设计时间,也无形中带给程序设计者在窗口环境下学习程序设计一片晴朗的天空。C++ Builder 4 以后的版本已对 Project Manager、VCL、ActiveX、Debugging、DataBase、Client Dataset 等加强其功能外,并支持 MTS、解决千年虫问题、CORBA、Multi-tier,以及新建了 Class Explorer、New Compiler Feature、New Runtime Library feature、New Dockable Tool Window、New Dynamic Array Template Class 等功能,使得 C++ Builder 程序设计能力如虎添翼。

1.1 程序语言的演进

计算机系统是由硬件与软件组合而成。“硬件”就是由中央处理器、内存、输出输入接口等装置所构成的实体设备。至于“软件”就是程序 (Program),而“程序”是用来指挥计算机为我们解决问题的命令集,而这些程序就是使用“程序语言”所编写而成的。我们就是通过所编写的程序来命令硬件为我们解决较复杂的计算和处理庞大的数据。

最早期的程序语言就是“机器语言” (Machine Language),即使用 0 与 1 来编写程序,由于计算机只能处理 0 与 1,因此能马上执行,但是机器语言人们不易看懂而且编写很不方便,一般人很难接受。因此,希望能发展出一套较易于一般人所能接受的计算机语言,只要通过“语言翻译器” (Language Translator) 便自动产生计算机能认识的机器语言,经过计算机专家的改良而发展出以助记符号方式来编写程序,我们将这种语言称为“汇编语言” (Assembly Language) 或“低级语言” (Low-Level Language),这种语言的语法较机器语言易懂且易编写,但是它和机器有关,也就是机器不相同,语言的写法也不尽相同,可移植性 (不同平台之间程序移植) 不高,编写程序时也要先了解计算机内部缓存器和内存的运行才能编写,对一般计算机初学者还是很困难。因此,这种语言只适用于计算机专业人员来编写有关计算机系统或输出输入接口的驱动程序。

针对以上的缺点,计算机专家又开发出目前为大多数人所能接受的“高级语言” (High-Level Language),这种语言可移植性高而且语法更接近人类的自然语言与数学表

示式,其中以 C 语言与基础语言 (BASIC) 是目前的主流,其中 BASIC 是最容易为初学者所接受也是最普及的一种高级语言。至于 C 语言则是理工科或程序设计人员用来学习程序设计的利器。

1.2 C 语言的演进

世界公认 C 语言的鼻祖是 1968 年 Martin Richards 在英格兰剑桥开发出 BCPL (Basic Combined Programming Language)。1970 年 Ken Thompson 于美国 Bell 实验室根据 BCPL 设计出 B 语言,是一种没有类型的语言,使用在当时 DEC 公司 PDP-7 计算机上的一些系统软件,这些软件就是 UNIX 操作系统的前身。1972 年 Dennis Ritchie 于美国 Bell 实验室,为新型 PDP-11 计算机重新改写 UNIX 操作系统,而由 B 语言发展成 C 语言,这时 C 语言已能提供:整型、浮点型、数组、结构、文件等数据类型。同时 Ritchie 和 Brian Kernighan 于 1978 年出版《The C Programming Language》一书,因而奠定了 C 语言的完整结构,我们将这种版本的 C 语言称为 K&R C 语言。随后 C 语言百家争鸣,产生出多种版本的 C 语言如: Lattice C、MS-C、Quick C 等,为了使 C 语言能标准化,于 1983 年,美国国家标准协会 (ANSI) 制定了一套 ANSI 的 C 语言标准,于 1988 年开始实施。

1987 年美国 Borland 公司结合了 K&R C 和 ANSI C 推出 Turbo C,深受当时程序设计者的喜爱,随着信息科技的进步,面向对象程序设计开始流行,该公司又于 1989 年陆续推出 Turbo C++、Borland C++, 1999 年 3 月推出 Borland C++ Builder 4。同年 Microsoft 推出 Visual C++ 6.0。由于 C++ Builder 的集成开发环境比 Visual C++ 更容易学习,假若你是位刚学 C 语言的初学者建议使用 C++ Builder 较易学习。当然更高级的应用,由 INPRISE (原 Borland) 公司的报告可得知, C++ Builder 5 的功能是不逊于 Visual C++ 6.0 的。因此,目前很多企业里的 C 程序设计人员仍习惯使用 Borland C++ Builder。INPRISE 公司于 2002 年推出的 Borland C++ Builder 6,除保有 Builder 5 原有的功能外,着重在 Internet 和集成企业数据库以及电子商务应用。

1.3 C++ Builder 6 介绍

自从 Windows 推出以后,计算机的用户界面改善了很多,计算机已不再是少数人的专利品,大部分的人只需经过短时间的学习就能操作计算机,不用再去背那些冷冰冰的 DOS 命令而直接在屏幕上拉一拉、点一点鼠标便能做到。但是 Windows 刚出来时,希望能在 Windows 下写出一个漂亮的程序,对程序设计师简直是一个梦魇,必须面对微软窗口软件开发工具 (Software Development Kits, SDK) 内数百个 Windows 函数 (Windows API) 以及厚厚几大本程序设计手册,真是令人望而却步。所幸,经过软件公司不断改进与努力,引进 RAD (Rapid Application Development, 快速应用程序开发环境) 程序开发工具集成在 IDE (Integrated Development Environment) 集成开发环境中,配合视觉开发工具,让程序设计者能够在窗口环境快速开发出窗口的相关应用程序。目前 INPRISE 公司的 C++ Builder 是在窗口环境下开发系统软件或应用软件的最佳选择。

Borland C++ Builder (简称 BCB) 发展至今已第 6 版, 它是一种可视化的程序语言, 所谓“可视化”, 是指开发图形应用界面的方法, 其方法就是在建立用户界面 (User Interface) 时, 不必编写程序来描述输入或输出界面的外观和配置, 只要使用工具箱的工具, 在程序设计阶段便可实现, 是属于一种 “What you see is what you get” 可视化的设计观念。C++ Builder 是以 C 语言为基础, 再加入许多对象的概念以及 Windows GUI 有关的功能。C++ Builder 6 的功能简述如下:

(1) 引导 C++ 软件开发进入新境界

Borland C++ Builder 提供开发人员一个稳定、高效的 C++ 开发环境, 协助用户运用新兴 Web Services 技术, 开发高效率的 e-Business 应用系统。C++ Builder BizSnap 技术, 是一个以 Web Services 为基础的开发平台, 运用业界标准 SOAP/XML Web Services, 将企业内的应用系统, 能够与下游客户及上游产业供货商进行天衣无缝地集成, 大幅简化了传统企业对企业 (Business to Business, B2B) 间应用系统集成的复杂度。C++ Builder 的 WebSnap 技术是针对 Web 应用所提出的开发平台, 运用新式软件组件技术, 有效率简化开发过程中各环节的复杂度, 协助开发人员快速开发出功能丰富、高效能的 Web 应用程序。C++ Builder 的 DataSnap 技术, 提供一个高效能的数据存取解决方案, 帮助用户集成企业内各类操作流程, 以及企业外众多企业伙伴。

(2) Web Services 完美集成企业与企业间的异质系统

C++ Builder 的 BizSnap 技术提供完整的 Web Services 开发环境, 想要进入 e-Business 的新时代, 一点也不难, BizSnap 相关组件与工具可以轻松转换现有企业信息成为 XML 文件, 让企业在 B2B 的潮流中完全不受限制与约束。不论企业伙伴使用何种应用系统, C++ Builder BizSnap 采用业界标准 SOAP/XML, 让企业间的沟通集成, 丝毫不因多变、异质 IT 解决方案而受阻, BizSnap 让 C++ Builder 能够根据标准的 SOAP/XML Web Services 与 Microsoft .NET、Microsoft BizTalk 以及 Sun Microsystems ONE 等应用服务相集成。

(3) 快捷强大的 Web 应用程序开发能力

C++ Builder 的 WebSnap 技术让用户在开发 Web 应用程序时享有 RAD 工具独有的便利性, WebSnap 的调试服务器, 无须借助额外 Web 服务器, 即可在 C++ Builder 集成开发环境内提供 Web 应用程序完整调试功能, 用户可以使用 Microsoft FrontPage 或 Macromedia Dreamweaver 等知名网页制作工具制作画面及编写服务器端脚本语言, 根据 WebSnap 相关组件与 VBScript、JScript 等任何 Microsoft ActiveScript 所支持的脚本语言直接集成, 同时获得脚本语言的弹性与 C++ Builder 的执行效率。WebPage 外观设计向导, 以可视化工具与软件组件结合方式, 快速开发集成数据库及动态网页之 Web 应用程序。

(4) 建构高效能多层数据库 Web Services 应用系统

C++ Builder 的 DataSnap 技术让用户能够运用 SOAP、CORBA、COM、TCP/IP 等分布式运算标准, 开发多层分布式数据库应用系统。搭配 DataSnap 所开发出的中间件 (Middleware), 使企业内客户端应用程序无需任何组态设置, 降低客户端维护成本, 搭配 Web Services 更能够提供企业外的相关厂商, 流畅地交换企业信息。DataSnap 中间件内各项技术, 能对数据库连接 (Connection) 作有效的管理, 并让带宽控制达到最佳的效果, 亦为 Web Services 集成各式企业级数据库提供了最佳的途径, 将企业内应用系统延

伸成为 e-Business 应用系统。

(5) 轻易连接企业内数据库

C++ Builder 6 内建多种数据库存取方式,可跨 Linux 平台的 dbExpress 驱动程序,提供高速存取 Oracle、IBM DB/2 与 Informix、Borland InterBase 及 MySQL 的能力,新版 Borland Database Engine 提供高效能 SQL Link 驱动程序存取 Oracle、IBM DB/2 与 Informix、Microsoft SQL Server、Sybase 与 Borland InterBase,并针对 Oracle 对象关联延伸功能提供完整的支持。dbGo 系列组件可通过 Microsoft ADO 存取各类数据来源。无论用户使用 dbExpress、Borland Database Engine、dbGo,都可运用 C++ Builder 功能丰富的数据感知组件(Data aware component),快速开发出以数据为中心的窗口应用程序或 Web 应用程序。

(6) 实例企业级 e-Business 解决方案

Sun Microsystems 所提出之服务器端组件结构 Enterprise JavaBean (EJB) 是 Java 2 Enterprise Edition (J2EE) 结构中的核心,Java 跨越操作系统平台的特性以及自动内存管理特性,使得 Java 在服务器端应用获得业界广泛的支持。C++ Builder 6 能够根据 Borland Enterprise Server SIDL 功能,直接使用挂载于 Borland Enterprise Server 内的 EJB 对象,用户可以运用 C++ Builder 工具生成开发窗口应用程序、Web 应用程序或 SOAP/XML Web Services,桥接符合 Sun Microsystems J2EE 兼容认证的 Borland AppServer,将平台无关性、负载平衡、群集(Clustering)容错等企业级 Java 解决方案直接与微软窗口环境相集成。

(7) 运用快速开发工具(RAD)快速进入市场

C++ Builder 提供符合 ANSI/ISO 标准、功能强大的 C++ 编译器,新版本中增加支持 STLPort,并集成功能完整、高效率的可视化开发环境,包含画面设计、对象浏览器、项目管理员以及图形化调试器。C++ Builder 依循面向对象方法,建构以 VCL/CLX 软件组件为基础的开发结构,让程序代码可重复使用的能力大幅提高,缩短了软件开发周期。

(8) 强大的调试功能

C++ Builder 强化了原来断点管理的功能,支持多线程调试能力,亦能够对 COM+对象进行调试。内建 CodeGuard 向导自动侦测内存漏失的错误程序,C++ Builder 6 Enterprise 版针对多层分布式开发环境提供远程调试的能力,开发人员可以通过网络直接对远程的 COM 对象进行调试,让复杂的多层应用系统调试变得更容易。

(9) 高开发效率的集成开发环境

C++ Builder 大幅提高 Code Completion 向导执行速度,自动列出组件特性、方法列表框,减少错误发生,Code Insight 向导将常用程序代码定义于程序样版中,按下单一键即可插入程序代码中,全新 Object TreeView 窗口让用户更方便地选择用户程序中用到的组件,Object Inspector 增添关联功能,若组件某特性的值为其他的组件,可直接在 Object Inspector 视图与设置子组件的特性值。

(10) 运用 CORBA 集成异质环境

CORBA 是由 800 余家 IT 厂商参与的 OMG (Object Management Group) 组织所制定的分布式对象规格,也是众多厂商异质解决方案间沟通集成的桥梁,Borland VisiBroker 是全球使用最广泛、分发数量最多的 CORBA ORB 产品,C++ Builder 6 Enterprise 版内建

VisiBroker 4.5, 提供完整 C++ IDL 编译器及原生 CORBA 程序代码产生向导, 并集成于 C++ Builder 集成开发环境与项目管理员内, 运用 CORBA 技术代表用户的应用程序功能与规模将不再受限于窗口环境, 根据 VisiBroker 用户可以直接与 Java、UNIX 平台上的 CORBA 对象相集成。

(11) 精简快速的 InterBase 数据库

C++ Builder 内带的 InterBase 6.5 是功能完整的系统型数据库管理系统, 支持标准 ANSI SQL, 以最少的系统资源提供最佳的执行效能。支持各种操作平台的 InterBase 6.5, 几乎不需要专职的数据库系统管理员, 简单易用的特色领先众多的关系数据库管理系统, 却能提供企业级数据库相同的强大威力。

1.4 对象和事件基本观念

面向对象 (Object oriented) 就是把现实世界中的现象, 以更接近自然的形式在计算机世界中反映出来。每一个对象都有属于自己的属性 (Property) 和方法 (Method), 对象与对象间就是靠着方法来传递对象的消息 (Message)。

1. 对象

任何具体或抽象的事物, 基本上都可看作“对象” (Object)。例如: 一个人可视为一个大对象; 身体中的头部、胸部、腹部、四肢等部位可视为小对象; 胸部又可再细分成心、肺等更小的对象。在 C++ Builder 中的对象是由一些程序代码和材料组合, 对象可以被视为一个完整的章。对象可以是应用程序的一部分, 例如下列都可视为一个对象:

- ① 窗体 (Form)、打印机、屏幕、鼠标、键盘等。
- ② 利用组件模板的组件在窗体上所产生的控件: Label、Button、Edit、Image 等组件。
- ③ 菜单、数据库管理员、报表设计等。
- ④ 提供类别 (Class) 让用户产生自己的组件。

2. 属性

C++ Builder 中的每个对象都有其特性, 我们利用这些特性来描述这个对象的外观, 例如: 窗体的特性包括窗体的大小、前景色、背景色、标题等, 我们将这些特性称为“属性” (Property)。每个对象的属性大都有其默认值, 有些对象的属性在程序设计阶段或执行阶段都可以改变这些对象的属性值, 对象便会按照您的设置表现出来。如: “球”是一个对象, 然而我们却能分辨出篮球和乒乓球, 因为它们的大小不同、材质不同、颜色不同、名称也不相同。所以, 不同类的球 (对象) 虽有共同的属性, 但属性内容不同, 因此可造出不同的球出来。例如图 1.1 中, 不同标题名称的命令按钮, 即视为不同的对象。

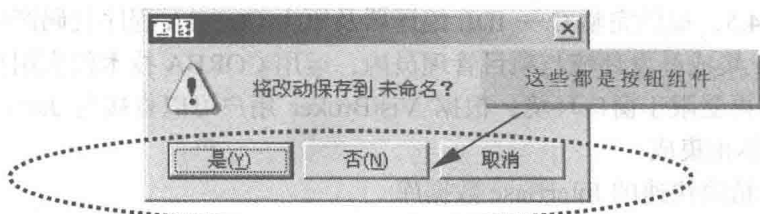


图1.1

3. 事件

在 C++ Builder 中只要在某个有作用的对象上按下键盘、移动鼠标或按鼠标一下都会产生一种反应，C++ Builder 将此反应称为“事件”（Event）。当 C++ Builder 收到属于某个对象的事件时，C++ Builder 会根据此事件做适当的处理，其处理的方式就是去执行该对象相对应的程序代码我们称为“事件函数”，而这些程序代码就是我们用 C++ Builder 所提供的命令所设计出来的。

4. 方法

所谓方法（Method），是指为了在对象完成某件事或某项目标，所采取的处理方式。所以在 C++ Builder 中的每个对象都有一些方法用来改变该对象的外貌或内部行为。有了对象的方法，我们可以在不需要知道真正的数据结构（隐藏在对象内）以及其他相关的事项，就可以达成所要的需求。例如，在窗体对象中提供了清图、打印、画点、画线等功能，这些功能通称为“方法”。例如，我们只要使用窗体所提供的列表方法，就可以打印出窗体而不必理会要如何控制打印机才能打印窗体。

5. 事件驱动

C++ Builder 所提供的“事件驱动”（Event driven）就像消防队的工作一样。平时若没有状况发生时，便在驻地做救火的准备或训练工作，随时待命。当有状况发生，知道火灾的地点、楼层的高度及该处是否有易燃物等，再携带必要的装备前往现场火。像这些因事件（火灾）发生，才做适当处理（救火）的观念称为“事件驱动”。同样，Windows 的应用程序也是如此，当你执行应用程序后，该应用程序就进入待命状态，当有按下键或鼠标时就表示有事件要发生，Windows 将此事件相关信息传给该应用程序，该程序会根据送来的信息做适当的处理。

传统的程序设计，写程序者主导整个程序的流程，用户只能按照既定的流程来操作，是属于顺序的。C++ Builder 的设计观念是将所有流程都交给用户来主控，完全由用户对窗口内所做出反应来决定或由系统事件来做决定。程序设计者是不决定整个程序的流程，是属于非顺序的。但是处理事件的方法和步骤到最基层时，仍然是属于顺序的，就好像消防队员进入火场灭火的基本操作方法是一定。所以，善用 C++ Builder 的特点很容易设计出用户接口以及变化多端的事件驱动处理程序式，可以大大地缩短开发应用程序的时间。

1.5 软硬件需求

在安装与启动 Borland C++ Builder 之前,我们先要了解 Borland C++ Builder 所需要的软硬件需求,以免执行时发生问题,以下是 C++ Builder 在硬件及软件方面的基本需求。

1. 硬件需求

- 至少 Intel Pentium 或兼容的 166 MHz CPU 以上微处理器(建议使用 Pentium III 以上)。
- 显示卡:使用 VGA 或更高级的显示卡。
- 硬盘: C++ Builder 标准版完全安装至少占 550MB,专业版 650MB,企业版 750MB,建议使用 20 GB 以上大容量硬盘。
- 主存储器: 128 MB RAM 以上(建议用 256MB RAM 以上)。
- 软盘: 3.5 英寸。
- 窗口环境兼容鼠标。
- 只读光驱(CD-ROM)。
- 16-Bits 声卡、喇叭、麦克风。

2. 软件需求

- Microsoft Windows 98、2000 中文版(Service Pack 2 以上)或 Windows XP。
- 标准版:提供 C++入门者一个能快速学习窗口软件开发的环境。
- 专业版:完整提供专业人员开发窗口平台下各类程序,包含商业数据库应用、Internet/Web 应用以及系统作用域的应用(本书为 C++ Builder 入门书,虽然是以专业版为主,不过除了 17、18 章之外,所有范例都适用于标准版)。
- 企业版:针对企业及用户提供各类数据库延伸连接驱动程序,以建构主从或多层结构应用系统以及电子商务系统。

1.6 鼠标的基本操作用语

在 Windows 作业环境下,大部分操作都是通过鼠标来控制,当然在 C++ Builder 开发环境下也不例外。例如:按命令按钮、拖曳工具图标、窗口的缩放等操作都是借着下列五种鼠标的基本动作来完成。

1. 鼠标基本操作术语

操作用语	操作方式
移到(Move)	移动鼠标指针到某个对象上
单击(Click)	移鼠标指针到某个对象上,按一下鼠标左键再放开
双击(Double Click)	移鼠标指针到某个对象上,连续快按鼠标左键两下
拖曳(Drag)	移鼠标指针到某个对象上后,按住鼠标左键不放并移动鼠标,将对象搬移到目的地的过程
右击(Click)	当鼠标指针在某个对象上,按一下鼠标右键,会出现与操作该对象有关的快捷菜单,以方便选择