

● 计算机编程与实例解析系列丛书

北京科海培训中心

C++Builder 4.0

编程与实例

刘 华 编著



科学出版社

北京科海培训中心

• 计算机编程与实例解析系列丛书

C++Builder 4.0 编程与实例

刘 华 编著

科学出版社

2000

内 容 简 介

本书从 C++Builder 4.0 基础知识入手,通过大量的实例,讲述了 C++Builder 编程的各个方面,并着重讲述了多媒体程序设计、Internet 应用程序设计、数据库应用程序开发、线程类和多线程应用程序设计、动态链接库技术、动态链接库的创建和调用、自定义组件、ActiveX 控件等高级技巧。

书中提供了丰富的实例代码,用户只需稍加修改,即可直接用于自己的应用程序中,方便而实用。

本书适合于 C++Builder 4.0 的中、高级用户。

图书在版编目(CIP)数据

C++Builder 4.0 编程与实例/刘华编著. —北京:
科学出版社,2000.4
ISBN 7-03-008394-6

I. C... II. 刘... III. C 语言-程序设计
N. TP312

中国版本图书馆 CIP 数据核字(2000)第 05376 号

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码:100717

北京朝阳科普印刷厂印刷

科学出版社总发行 各地新华书店经销

*

2000 年 4 月第 一 版

开本:787×1092 1/16

2000 年 4 月第一次印刷

印张:25.75

印数:1—5 000

字数:626 240

定价:34.00 元

目 录

第 1 章 C++Builder 4.0 概述.....	1
1.1 C++Builder 4.0 简介.....	1
1.1.1 C++Builder 4.0 的历史	1
1.1.2 C++Builder 4.0 的新特色	2
1.2 传统 Windows 应用程序开发工具与 C++Builder 之比较.....	3
1.2.1 传统 Windows 应用程序开发工具	3
1.2.2 C++Builder 和其他 RAD 工具	4
1.3 C++Builder 4.0 的安装.....	4
1.3.1 安装前的准备.....	4
1.3.2 安装 C++Builder 4.0	4
1.4 小结.....	8
第 2 章 C++Builder 初步.....	9
2.1 C++Builder 4.0 开发环境的定制.....	9
2.1.1 C++Builder 4.0 开发环境介绍	9
2.1.2 BPL 包的安装	16
2.1.3 OCX 控件的导入与使用	17
2.1.4 组件面板的定制.....	19
2.1.5 工程环境特殊定制和公共定制.....	21
2.2 C++Builder 4.0 开发工具介绍.....	30
2.2.1 应用程序界面设计器.....	30
2.2.2 代码编辑器的使用.....	33
2.2.3 图形编辑器的使用.....	35
2.2.4 数据库平台.....	37
2.2.5 对象查看器.....	37
2.2.6 对象存储器.....	38
2.3 帮助系统的定制.....	41
2.4 C++Builder 应用程序开发快速入门.....	42
2.4.1 C++Builder 工程的建立	43
2.4.2 应用程序界面的设计.....	49
2.4.3 应用程序功能的设计.....	52
2.4.4 程序的编译和调试.....	55
2.4.5 程序的运行和功能分析.....	57
2.5 小结.....	59

第 3 章 C++Builder 应用程序开发基础	60
3.1 传统的 Windows 应用程序开发	60
3.1.1 用 C 语言设计的 MS-DOS 程序	60
3.1.2 用 C++ 设计的标准 Windows 程序	62
3.1.3 消息循环是 Windows 程序的核心	65
3.1.4 面向过程和面向对象的程序设计	65
3.2 理解事件和事件驱动	67
3.2.1 事件	67
3.2.2 事件驱动	68
3.3 C++Builder 应用程序开发的特点	68
3.3.1 如何处理传统 Windows 程序中的重复代码	69
3.3.2 消息循环的“消失”——封装	69
3.3.3 C++Builder 开发程序的快捷方式	70
3.4 小结	70
第 4 章 对话框和对话框应用程序开发	71
4.1 对话框和对话框应用程序简介	71
4.1.1 对话框简介	71
4.1.2 对话框应用程序简介	72
4.2 对话框常用组件	72
4.2.1 几种按钮	73
4.2.2 编辑组件	79
4.2.3 显示组件	89
4.2.4 其他几种常用组件	103
4.3 标准对话框函数	107
4.3.1 MessageBox 函数	107
4.3.2 MessageDlg 函数	109
4.3.3 MessageDlgPos 函数	110
4.3.4 ShowMessage 函数	112
4.3.5 ShowMessagePos 函数	112
4.3.6 InputBox 函数	113
4.4 通用标准对话框组件	114
4.4.1 文件对话框	114
4.4.2 图形文件对话框	120
4.4.3 字体和颜色对话框	122
4.4.4 查找和替换对话框	125
4.4.5 打印对话框和打印机设置对话框	131
4.5 对话框应用程序的事件	131
4.5.1 鼠标事件和键盘事件	132

4.5.2 窗体事件的响应.....	139
4.5.3 文本编辑框事件的响应.....	143
4.6 对话框应用程序的设计.....	144
4.6.1 对话框应用程序概述.....	144
4.6.2 应用程序的运行.....	145
4.6.3 应用程序的完整代码.....	146
4.7 小结.....	151
第 5 章 Win32 应用程序常用组件.....	152
5.1 静态图像列表框和动画播放框.....	152
5.1.1 静态图像列表框.....	152
5.1.2 动画播放框.....	156
5.2 轨迹栏和进度栏.....	159
5.2.1 轨迹栏概述.....	159
5.2.2 进度栏概述.....	160
5.2.3 轨迹栏和进度栏使用举例.....	160
5.3 树状查看框和列表查看框.....	162
5.3.1 树状查看框.....	163
5.3.2 列表查看框.....	171
5.4 状态栏和组件集控制栏.....	176
5.4.1 状态栏的基本属性、事件和方法.....	177
5.4.2 组件集控制栏的基本属性和事件.....	182
5.5 Win32 常用组件综合应用举例.....	185
5.6 小结.....	192
第 6 章 文本编辑器的常用组件.....	193
6.1 菜单的设计和控制.....	193
6.1.1 菜单设计.....	195
6.1.2 菜单的动态控制.....	199
6.2 多页界面与多文档界面.....	201
6.2.1 多页界面 (MPI).....	202
6.2.2 多文档界面 (MDI).....	205
6.3 多文本编辑器.....	207
6.4 普通文本编辑器综合举例.....	212
6.4.1 程序界面的设计.....	212
6.4.2 子窗口的功能设计.....	214
6.4.3 主窗口的功能设计.....	223
6.5 小结.....	236
第 7 章 图形基础.....	237

7.1 C++Builder 的图形编程模式	237
7.2 画笔、画刷和画布	238
7.2.1 画笔组件	238
7.2.2 画刷组件	239
7.2.3 画布组件	240
7.3 Shape 组件	243
7.3.1 Shape 组件的属性	243
7.3.2 Shape 组件应用举例	244
7.4 图形控制工具——位图	244
7.4.1 位图的属性	244
7.4.2 位图的事件	245
7.4.3 位图的方法	245
7.4.4 位图的使用示例	245
7.5 绘图程序综合应用举例	246
7.5.1 头文件——Main.h	247
7.5.2 实现文件——Main.cpp	250
7.6 小结	261
第 8 章 多媒体程序设计	262
8.1 多媒体技术基础	262
8.2 动画播放框 (TAnimate)	264
8.3 媒体播放器 (TMediaPlayer)	264
8.3.1 媒体播放器的常用属性	265
8.3.2 媒体播放器的事件	269
8.3.3 媒体播放器的方法	272
8.4 媒体播放器应用举例	274
8.4.1 界面设计	275
8.4.2 部分功能代码解释	275
8.4.3 程序的完整代码	278
8.5 小结	291
第 9 章 Internet 应用程序设计	292
9.1 C++Builder 的常用 Internet 组件	292
9.1.1 C++Builder 的 Internet 组件面板的组成	292
9.1.2 客户端组件——TClientSocket	293
9.1.3 服务器组件——TServerSocket	295
9.2 Internet 应用程序综合举例	298
9.2.1 应用程序功能介绍	298
9.2.2 应用程序的界面设计	298
9.2.3 应用程序部分代码解释	299

9.2.4 应用程序的完整代码.....	301
9.3 小结	311
第 10 章 数据库应用程序的开发	312
10.1 BDE	312
10.1.1 BDE 能访问的数据库.....	312
10.1.2 BDE 的别名机制.....	313
10.1.3 BDE 的数据库别名管理.....	313
10.2 C++Builder 数据库应用程序的结构.....	315
10.3 数据库组件	317
10.3.1 数据访问组件.....	317
10.3.2 数据控制组件.....	322
10.4 数据库应用程序开发综合举例	327
10.4.1 例程数据库概述.....	327
10.4.2 例程功能概述.....	327
10.4.3 例程具体代码段解释.....	328
10.4.4 例程的完整代码.....	328
10.5 小结	334
第 11 章 高级开发技术	335
11.1 线程类和多线程应用程序的设计	335
11.1.1 线程类——TThread	335
11.1.2 创建线程类.....	338
11.1.3 线程同步.....	340
11.1.4 线程通信.....	342
11.1.5 多线程应用程序的设计	342
11.2 动态链接库技术概述	358
11.2.1 动态链接库的基本概念.....	358
11.2.2 动态链接库的加载和卸载.....	358
11.2.3 使用动态链接库的优点.....	359
11.3 动态链接库的创建和调用	359
11.3.1 非 VCL 动态链接库的创建.....	359
11.3.2 VCL 动态链接库的创建.....	364
11.3.3 动态链接库的调用	366
11.4 自定义组件	369
11.4.1 使用组件创建向导.....	369
11.4.2 手工创建组件.....	373
11.4.3 组件功能的测试.....	374
11.4.4 组件的安装.....	377
11.5 ActiveX 控件.....	377

11.5.1 创建 ActiveX 控件工程	378
11.5.2 设置 ActiveX 控件属性	384
11.5.3 注册 ActiveX 控件	384
11.6 小结	385
第 12 章 程序调试和安装程序制作	386
12.1 程序调试	386
12.1.1 程序的错误处理	386
12.1.2 应用程序的功能测试	390
12.2 安装程序的制作	392
12.2.1 应用程序安装前的处理	392
12.2.2 制作过程	392
12.2.3 应用程序的试安装	403
12.3 小结	404

第 1 章 C++Builder 4.0 概述

1.1 C++Builder 4.0 简介

1.1.1 C++Builder 4.0 的历史

C++Builder 产生的历史背景

C++Builder 的产生有其深刻的历史原因，首先是日益发展的商业竞争的需要，这是其发展的动力；计算机技术的飞速发展，如计算机容量的增大和计算速度的提高，则为其发展提供了硬件基础；而 C++程序员的要求则是其直接原因。具体体现在：

快速开发应用程序的需要

用户对快速开发应用程序的需要是 C++Builder 产生的根本原因和动力，众多 RAD (Rapid Application Development, 快速应用程序开发) 软件就是为适应这一潮流而出现的。

软件包重用性的需要

编写几乎完全相同的 Windows 应用程序外壳是传统 Windows 程序员不可避免的重重复劳动，大量标准 Windows 组件的重复开发也是完全没有必要的，这一要求也是 C++Builder 产生的重要原因。

数据库应用程序开发的需要

当前开发的商业应用程序几乎无一不涉及数据库应用，甚至许多应用就是围绕数据库而展开的，所以 C++Builder 具有强大的数据库应用开发功能自然是这一需求的产物。

Internet 应用程序开发的需要

网络和数据库应用可以说是商业应用的两大标志，几乎所有的 Windows 应用程序都支持这两点，否则它就不会有生命力；而作为应用最普遍、效率最高的 C++程序设计语言的开发工具，理应能快速设计出这样的应用。传统的开发工具(如 Borland C++ 5.0, Microsoft C/C++等)使 C++程序员在这一方面成为最蹩脚的程序员，而 C++Builder 强大的 Internet 和数据库应用开发功能几乎完全为程序员解决了这一难题，使 C++程序员在面对最新需求时充满信心。

降低程序管理复杂性的需要

由于程序界面友好性的要求和大量应用需求致使众多程序变得越来越庞杂，传统的应用管理工具和方法难以满足需要，C++Builder 的工程管理功能能完美地解决这一问题。

提高应用程序稳健性的需要

程序的稳健性是其优劣的重要标志之一，而它与程序的复杂程度是紧密相连的。若用传统的应用程序开发工具，随着应用程序复杂程度的提高，它的健壮性急速下降，这是传统应用开发工具对程序的管理不当而导致的。C++Builder 能很好地解决它们之间的矛盾。

在这种情况下，C++Builder 应运而生，其功能强大，使用方便，因而受到程序员的普遍欢迎，特别是 C++程序员。

C++Builder 的简单历史回顾

Borland 公司首先发布了基于 Pascal 的 RAD 开发工具——Delphi，受到热烈欢迎，而且版本迅速提高，这使 Borland 公司意识到 C++的 RAD 开发工具一定具有广阔的发展前景：其一是因为其 VCL（Visual Component Library，可视组件库）的不断成熟，其二是因为 C++与 Pascal 相比有更大的市场。正因为如此，Borland 公司迅速推出了 Delphi 的姊妹工具——C++Builder，它和 Delphi 外观相似，但它们有着本质的区别。首先 C++Builder 是基于 C++的，而 Delphi 是基于 Pascal 的，而且，前者可以直接编译 Borland C++ 5.0、Delphi 及 Visual C++等编写的程序，支持 Windows 95 和 NT 的新特性，支持 ActiveX，提供了 Internet 编程工具，具有强大的数据库应用程序开发功能。C++Builder 的功能升级步伐之快更是让人吃惊。

1997 年 6 月，Borland 中国公司在京正式发布 C++Builder 1.0。随后不久，在 1997 年末就推出了 C++Builder 2.0 版，1998 年上半年又推出了 3.0 版，最近，4.0 版又正式发布了。升级速度如此之快，实属罕见，这足以看出 Borland 公司对其新产品 C++Builder 的未来充满信心。

1.1.2 C++Builder 4.0 的新特色

C++Builder 是高性能的 C++可视化应用开发工具，与其他 C/C++开发工具相比，它具有许多新特色：

C++Builder 是高性能的 C++开发工具

C++Builder 是基于 C++的，它可以快速编译、连接和执行。使用 Borland C++ 5.0 的 OWL 库开发 Windows 应用程序的程序员对这一点都深有体会。同时 C++Builder 具有双编译器引擎，这使它不仅能编译 C/C++程序，还能编译 Object Pascal 语言程序。C++Builder 是目前众多 C/C++编译器中对 C++的支持最完备的编译器之一，从异常处理、模板、重载虚函数、布尔类型到名字空间等一应俱全。C++Builder 为学习者提供了最佳的 ANSI C++标准。

C++Builder 是优秀的可视化应用开发工具

C++Builder 不仅有完美的 C/C++编译器（这一点 Borland C++ 5.0/5.02 已经做得很好），更重要的是它具有完美的可视化应用开发工具。它使程序员从繁重的代码编写中解放出来，使他们能将注意力重点放在程序设计上而不是简单的重复劳动上。同时，它还提供完全可视的程序界面开发工具，从而使程序员对开发工具的学习周期大大缩短。

C++Builder 具有强大的数据库应用开发功能

对于传统的 C/C++程序员来说，访问数据库是最让人头痛的事，传统的 C/C++编译器对数据库应用程序的支持少得可怜。而 C++Builder 提供了强大的数据库处理功能，它使得程序员不用写一行代码就能产生功能强大的数据库应用程序，这些主要得益于 C++Builder 众多的数据库感知控件和底层的 BDE 数据库引擎。

C++Builder 的 BDE 数据库引擎是 Borland 公司成熟的数据库连接技术，它提供了三种数据访问方法：

- 直接存取 dBase, FoxBase, Foxpro, Paradox, MS Access 等多种文件型数据库生成的 DB, DBF 文件；
- 通过标准的数据库接口访问任何一种支持 ODBC 的数据库；
- 通过高效的 SQL Links 数据库驱动程序允许用户访问 Oracle, Informix, SyBase, MS SQLServer, DB2 和 Borland 的 InterBase。

C++Builder 具有强大的网络编程功能

C++Builder 具有众多的 Internet 应用开发控件，如 WebBroker, WebDispatcher, WebBridge 等，它们基本涵盖了 Internet 应用的全部功能，利用它们程序员可以方便地建立自己的 Internet 应用程序。

C++Builder 具有完美的中文支持能力

对于 C++Builder 的广大中文使用者来说，它还具有另一个突出优点。C++Builder 采用了双字内核结构，彻底解决了“半个字”的问题。而且它的 VCL 组件也和中文完全相容，甚至支持中文输入法的切换，这种对中文的高度兼容能力是许多软件所望尘莫及的。

总之，C++Builder 的出现是广大程序员的福音，因为它使 C++从此走上了可视化的应用程序快速开发阶段，它使许多其他 RAD 开发工具黯然失色。

1.2 传统 Windows 应用程序开发工具与 C++Builder 之比较

1.2.1 传统 Windows 应用程序开发工具

与 C++Builder 相比，传统的 Windows 应用程序开发工具具有许多缺点，这使得 C/C++ 程序员开发应用程序极其困难，且开发出来的应用程序难以维护。主要体现在以下几点：

传统的 Windows 应用程序开发过于复杂

传统的应用程序开发工具以 Borland C++ 5.0/5.02 和 Microsoft C/C++最具代表性，它们的 Windows 应用程序开发分别是基于 OWL (Object Windows Library) 和 MFC 的，代码的重用是基于类层次的，这对程序员的要求太高，只有熟练掌握了 OWL 库的程序员才可能设计出优秀的程序。

传统的 Windows 应用程序开发周期长

传统应用程序的开发需要经过系统分析、系统设计、编码调试、软件测试等许多阶段，这是应用程序开发的复杂性所决定的。从而应用程序的开发周期太长，开发效率很低，不能满足当前对应用程序快速开发的需要。

传统的 Windows 应用程序开发环境差

传统的所谓开发环境“可视化”有其局限性，譬如我们可以说 Borland C++ 5.0 的开发环境是可视的，它对工程的管理确实是可视的，其实它是可视的也只限于这一点。程序的设计过程是完全不可见的，这是加大程序复杂度的重要原因，从而导致开发效率很低。

1.2.2 C++Builder 和其他 RAD 工具

除 C++Builder 外, 还有其他一些 RAD 工具, 如 Visual Basic、Delphi 等。这些 RAD 工具之间的本质差别在于它们基于不同的语言, 由此产生的代码效率也会有所不同。

C++Builder 是基于 C++ 语言的, 它继承了 C++ 语言和 C++ 的传统编译器的优点, 因而产生的代码具有高效性的特点, 同时它吸收了其他 RAD 工具的优点, 因而是大型应用程序首选的开发环境。而 Visual Basic 是基于 Basic 语言的, 不可避免地将解释型语言的缺点带入 RAD 开发工具的产品中, 速度比较慢。

C++Builder 的版本有:

- Borland C++Builder Client/Server Suite;
- Borland C++Builder Enterprise;
- Borland C++Builder Professional;
- Borland C++Builder Standard;
- Learn to Program with Borland C++Builder。



本书中的示例程序都是在 Borland C++Builder 4.0 (Enterprise) 下编写、编译和运行的。

1.3 C++Builder 4.0 的安装

1.3.1 安装前的准备

安装前的主要工作是检查自己的硬件配置是否满足最低配置要求:

- IBM-PC 或兼容机, CPU 在 80486 以上;
- 32 位操作系统: Microsoft Windows 95/98 或 Windows NT 3.51/4.0;
- 16MB 以上 (建议 64MB) 的内存;
- CD-ROM;
- 150MB 以上的硬盘空间;
- 标准 VGA 显示器或更高分辨率的显示器;
- 鼠标或其他类型的输入设备。

1.3.2 安装 C++Builder 4.0

在硬件配置满足最低配置要求后, 可以进行 C++Builder 4.0 的安装。C++Builder 4.0 及其安装程序都是运行在 Microsoft Windows 95、Microsoft Windows 98 或 Windows NT 3.51/4.0 下的。在这几种操作系统下, 安装方法基本相同, C++Builder 4.0 的安装向导能很好地引导用户安装系统。下面以企业版在 Microsoft Windows 98 下的安装过程为例说明其安装方法。其安装过程主要包括:

1. 操作系统的启动;
2. 运行 Install.exe 安装文件;
3. 按照安装向导的提示完成程序安装。

运行安装向导 Install.exe

将光盘插入光盘驱动器中,打开资源管理器,查看安装盘(光盘)根目录,单击 Install.exe 以运行安装向导,如图 1-1 所示。C++Builder 4.0 及其附件安装程序主界面如图 1-2 所示。

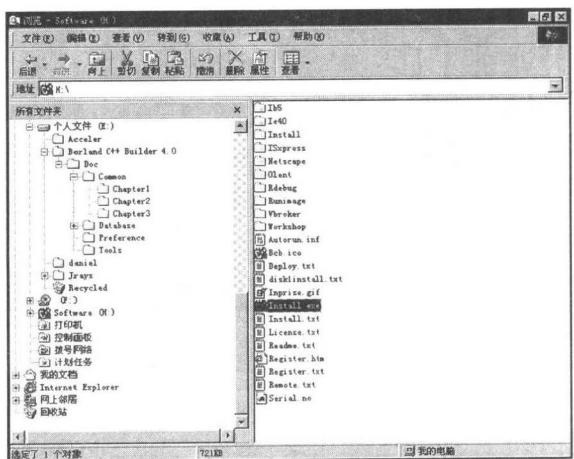


图 1-1 运行安装程序

有三种打开资源管理器的方式:



- 从【开始】菜单中运行 Explorer.exe;
- 直接使用 Windows 键+E 组合键;
- 单击【开始菜单/程序/Windows 资源管理器】菜单项。

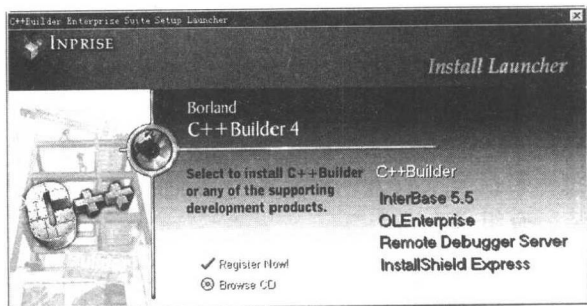


图 1-2 C++Builder 4.0 及其附件安装程序主界面

在运行 C++Builder 4.0 安装程序前最好关闭所有应用程序,以免发生冲突。

单击图 1-2 中的 C++Builder 就可以安装 C++Builder 4.0,进入 C++Builder 4.0 的安装向导,首先出现的是安装程序主界面,接下来的安装过程便完全在安装向导的指导下进行。单击【Next】按钮以继续进行安装。

接着将弹出要求回答安装的序列号和许可协议号的窗口,如图 1-3 所示。填写完序列号和许可协议号后单击【Next】按钮以继续安装。

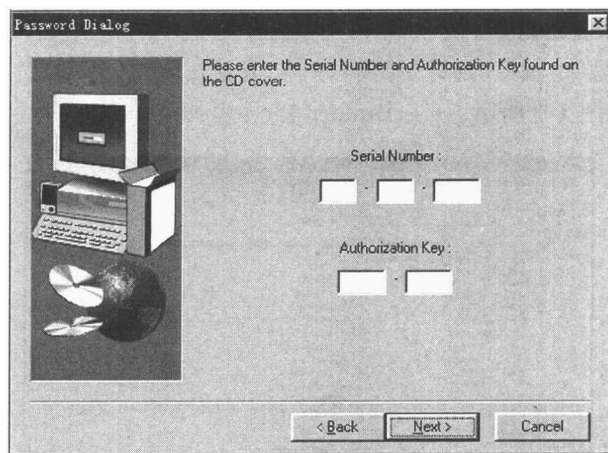


图 1-3 序列号、许可证号填写窗口

在接下来的版权声明和许可协议窗口中分别单击【Yes】或【Next】，即可进行下面的安装过程。

安装类型选择

C++Builder 4.0（企业版）共提供 4 种安装方式：

- **典型安装** 建议一般用户使用的安装方式。主要特点是不安装 OWL 库、MFC 库，以及 MS Help 帮助系统。
- **压缩安装** 对于磁盘空间比较小的用户，可以选择这种装方式，它可以大大节省空间，但 C++Builder 4.0 的许多功能也将无法使用。它仅安装程序运行所必需的最少选项。
- **用户自定义安装** 对 C++Builder 比较熟悉的用户可以选择此种安装方式，它可以根据用户的需要选择安装组件，既能满足用户需要，又能节省空间，但用户必须对 C++Builder 比较熟悉。
- **全部安装** 对于磁盘空间较大或需要较多功能的用户可以选择这种安装方式，它将为用户安装所有组件。

安装方式选择窗口如图 1-4 所示，根据具体情况选择合适的安装方式，然后单击【Next】以继续下面的安装过程。

接下来将显示安装组件窗口。C++Builder 4.0 安装盘上也提供了两种应用程序开发辅助工具：InterBase Client 和 VisiBroker for C++Builder。用户可以选定这两个选项，现在就安装，或者不选定，以后再安装。单击【Next】以继续安装程序。

安装目录选择

安装目录选择窗口如图 1-5 所示。有两种修改安装目录的方式，其一是直接在目录编辑框中键入，其二是单击【Browse】按钮，通过目录浏览窗口选择目录，选择好目录后单击【OK】以返回安装目录选择窗口，单击【Next】以继续安装过程。

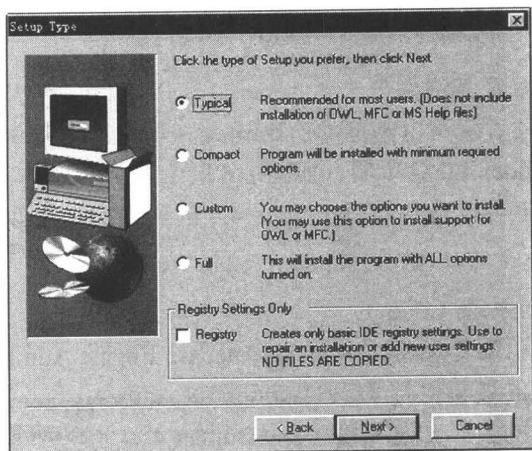


图 1-4 安装方式选择窗口

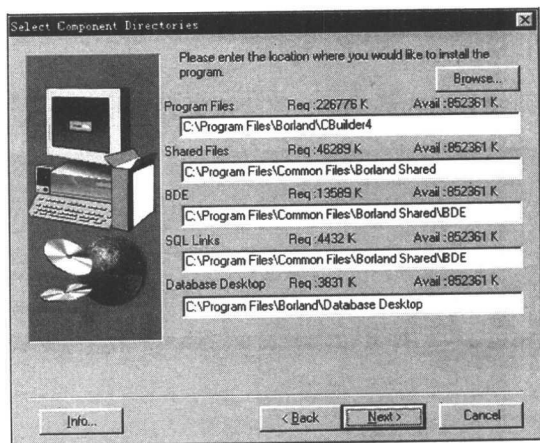


图 1-5 安装目录选择窗口

进入安装组件和安装目录确认窗口。单击【Back】可以返回上述目录选择过程重新作出选择；单击【Install】开始将文件复制到安装目录下。大约需要几分钟，具体时间长短取决于机器的配置。

在复制文件的过程中，安装向导程序一方面向用户提示当前工作文件复制进度，另一方面向用户简要介绍 C++Builder 4.0 的新功能和新特性。通过安装界面窗口，读者能大体了解 C++Builder 4.0 的一些新特点。在开发应用程序时，用户应尽量使用这些新特性，以加快应用程序开发速度，或者提高应用程序的性能。

系统环境的设置

复制了 C++Builder 4.0 的安装文件之后，安装向导便着手系统环境的设置工作。此时，用户需要等待一会儿。完成设置后，程序安装便同时完成，出现结束安装程序的提示窗口，选择重新启动系统（也可以不选择），单击【Finish】，C++Builder 4.0 便成功地安装了。

为了运行 C++Builder 4.0，读者必须重新启动系统。再次进入系统后读者就可以浏览

C++Builder 4.0 了。

浏览 C++Builder 4.0 及其组件

重新启动计算机后读者就可以浏览 C++Builder 4.0 的安装组件及其开发环境。从【开始】菜单中选择 C++Builder 4.0 菜单项，如图 1-6 所示。

在【开始】菜单项上可以看到 C++Builder 4.0 的所有安装组件和辅助开发工具。它们主要包括以下几项：

- 功能强大的帮助系统（Help）；
- Borland 数据库引擎管理器（Borland Database Engine Administrator）；
- C++Builder 4.0 的最新信息（C++Builder 4 Readme）；
- C++Builder 4.0 的集成开发环境（C++Builder 4）；
- 数据库操作平台（Database Desktop）；
- 图像编辑器（Image Editor）；
- Windows 监视器（WinSight32）；
- SQL 工具（SQL Monitor、SQL Explorer）。

我们将在下一章中介绍 C++Builder 4.0 的集成开发环境。



图 1-6 浏览 C++Builder 4.0 安装组件

1.4 小 结

在本章中，我们简要介绍了 C++Builder 和传统的 C/C++ 开发工具相比所具有的特色，并详细地讲述了 C++Builder 4.0 的安装过程。通过本章的学习，读者至少应该得到以下两点收获：

- 读者会对 C++Builder 的新特性有所了解；
- 能够顺利地安装 C++Builder 4.0。