

IS46/90

Borland C++ Builder **使用技巧与配置指南**

钟建平 史惠茗 编著

抖斗书屋 审校

清华大学出版社

(京)新登字 158 号

内 容 提 要

C++ Builder 是 Borland 公司最新推出的 Windows 应用开发软件，是一种全新可视化开发工具。它采用 Object-Orient Language 和该公司的快速编译器技术及数据库技术，是程序设计人员的一个得心应手的工具。本书全面地介绍了 C++ Builder 的功能、使用方法，以及它们之间的协同使用。本书适用于计算机初、中级用户。

版权所有，翻印必究。

书 名：Borland C++ Builder 使用技巧与配置指南

作 者：钟建平 史惠茗

审 校：抖斗书屋 (Done Bookhouse)

出版者：清华大学出版社 (北京清华大学校内，邮编：100084)

<http://www.tup.tsinghua.edu.cn>

印刷者：北京市人民文学印刷厂

发行者：新华书店总店北京科技发行所

开 本：787 × 1092 1/16 印张：15.5 字数：321 千字

版 次：1999 年 1 月第 1 版 1999 年 1 月第 1 次印刷

书 号：ISBN 7-302-03303-X/TP·1773

印 数：0001~6000

定 价：26.00 元

前 言

C++ Builder 是 Borland 公司于 1997 年开始推出的全新可视化开发工具，是一种方便、快捷的 Windows 应用程序开发软件，是该公司继其 Delphi 可视化产品之后的又一力作。它使用了 Microsoft Windows 图形用户界面的许多先进特性和设计思想，采用了面向对象程序语言（Object-Oriented Language）和该公司的快速编译器技术及数据库技术。

Borland 公司的编程开发工具历来受到广大的程序设计人员的喜爱，90 年代以来，很多的程序设计者就是从开始使用该公司的 Turbo C，Borland C++ 3.1，Borland C++ 5.0 等 C，C++ 开发工具和 dBASE 等数据库产品走上软件开发设计之路的。Borland 公司的 C，C++ 开发产品系列以其使用方便、友好和快速、优质的编译器享誉世界。然而，随着近几年 Windows 操作系统的普及，越来越多的应用程序在向 Windows 环境转移。虽然 Borland 公司的 Borland C++ 3.1，Borland C++ 5.0 等 C++ 工具已经具有了开发 Windows 应用程序的能力，但对大多数习惯了 DOS 开发的普通程序人员来说，跨越这一步仍需花费相当大的努力和艰辛，他们迫切希望 Borland 公司尽快推出 C++ 的可视化开发产品系列。现在，Borland C++ 的老用户们不用遗憾了，1997 年，Borland 公司正式推出了其可视化的 Windows 开发产品 C++ 版——Borland C++ Builder 1.0，圆了广大用户的心愿。1998 年，C++ Builder 又升级到了 3.0 版。

C++ Builder 的语言基础是 C++，但它与传统的 C++ 语言有着根本的区别。C++ Builder 开发环境自动为应用程序生成应用程序框架，这一框架正是应用程序的“骨架”。在框架上可不附加任何其他程序代码，仍可以正常地按照设计意图运行。使用 Borland 公司的 C++ Builder 可视化工具进行 Windows 应用程序开发时，程序人员所做的工作非常简单和容易，编程人员只要将一些组件、事件和不多的程序代码加入框架中，就可以实现自己的目的。默认的应用程序是一个空白的窗体（Form），这个窗口具有 Windows 应用程序窗口的所有属性：可以任意放大，直至全屏，也可以任意缩小，直至最小状态。可以这么说，C++ Builder 应用程序框架提供了所有 Windows 应用程序共有的东西，为用户进行 Windows 应用程序的开发打下了良好的基础。

本书针对 C++ Builder 的使用方法和主要功能，由浅入深地介绍了使用 C++ Builder 进行编程的主要步骤。为了方便初学者的学习，本书侧重于介绍如何使用该工具创建 Windows 应用程序的过程，对组件的功用、开发环境选项的配置及一些使用工具、如调试器等都有比较详细完整的介绍，最后，还重点对工具的编译器、连接器选项做了完整的说明。

本书由史惠康策划编写，全书大部分内容由钟建平、史惠茗编写，另外提供资料的有赵俊才、马向英、王纪华、宋新波、郭美山、石利文等。在编写过程中，白燕斌、贾志东、贾少彬、韩友斌等提供了部分插图和程序，在此表示感谢。钟建平负责了全书的统稿工作。

尽管编写时力求详尽准确，但疏漏之处在所难免，敬请读者批评指正。

编 者

1998 年 5 月于抖斗书屋

目 录

第 1 章 C++ Builder 简介	1
1.1 C++ Builder 开发工具特色及主要功能	1
1.2 Borland C++ Builder 运行的硬软件环境及安装	5
1.2.1 系统的硬软件环境配置要求	5
1.2.2 系统安装	5
第 2 章 C++ Builder 在线帮助的使用	13
2.1 如何获得语言定义的联机帮助	13
2.2 获得可视组件库(VCL)和运行库(RTL)的参考数据	15
2.3 程序设计时某些特定程序信息的联机帮助	15
2.4 程序设计交互式教学课程的使用	17
第 3 章 C++ Builder 快速入门	19
3.1 C++ Builder 开发环境的启动	19
3.2 C++ Builder 开发环境基本组件	20
3.2.1 C++ Builder 可视组件	21
3.2.2 C++ Builder 不可视组件	26
3.3 用 C++ Builder 开发一个程序的基本步骤	28
3.3.1 产生一个新的空白窗体	29
3.3.2 将一个新的组件加入到窗体中	29
3.3.3 设置组件的属性值 (Properties)	30
3.3.4 为组件建立事件处理句柄	31
3.3.5 嵌入用户实际作业的程序代码	33
3.3.6 生成执行程序	33
第 4 章 组件概述	35
4.1 C++ Builder 组件	35
4.2 Standard 页中组件的功能	37
4.3 Win 95 页中组件的功能	38
4.4 Additional 页中组件的功能	40
4.5 DataAccess 页中组件的功能	41
4.6 DataControls 页中组件的功能	42
4.7 Win 3.1 页中组件的功能	43
4.8 Dialogs 页中组件的功能	44
4.9 System 页中组件的功能	45
第 5 章 建立窗体和菜单	47
5.1 什么是一个窗体	47
5.2 查看窗体和程序单元	47

5.2.1	查看窗体	47
5.2.2	查看程序单元	48
5.3	建立窗体	48
5.3.1	什么是组件	49
5.3.2	在窗体中使用组件	50
5.3.3	在窗体中设置组件的属性	57
5.4	控制运行期间的窗体	59
5.4.1	将一个窗体指明为项目的主控窗体	59
5.4.2	控制窗体在运行期间的创建顺序	60
5.4.3	指定在运行期间自动创建的窗体	61
5.4.4	在运行期间实例化窗体	62
5.5	使用预定义的窗体	62
5.5.1	将一个现有的窗体加到项目中	63
5.5.2	继承对象仓库中的窗体	64
5.5.3	共享窗体	65
5.6	建立对话框	65
5.6.1	使一个对话框成为有模式的或无模式的	65
5.6.2	设定一个对话框窗体属性	66
5.6.3	指定一个对话框窗体的标题	67
5.6.4	创建标准命令按钮	67
5.6.5	按 Esc 键执行按钮代码	68
5.6.6	按 Enter 键执行按钮代码	69
5.6.7	设置一个按钮的属性来自动关闭一个对话框	70
5.6.8	设置 Tab 按键的行走顺序	70
5.6.9	测试 Tab 按键的行走顺序	72
5.6.10	取消一个组件的 Tab 按键顺序功能	72
5.6.11	使组件失效	72
5.7	窗体连接	73
5.8	创建窗体模板	73
5.9	创建窗体菜单	75
5.9.1	打开菜单设计工具	75
5.9.2	建立菜单	76
5.9.3	不打开菜单设计工具来编辑菜单项	82
5.9.4	使用菜单设计工具的快捷菜单来构造菜单	82
5.9.5	使用菜单模板来构造菜单	83
5.9.6	将菜单存储为模板	84
5.10	连接菜单事件和程序代码	85
5.10.1	处理菜单事件	85
5.10.2	动态地改变菜单项	86

第 6 章 项目文件	89
6.1 什么是 C++ Builder 项目?	89
6.2 组成一个 C++ Builder 项目的文件类型	89
6.3 C++ Builder 项目中各种文件	90
6.3.1 项目文件 (Project files)	90
6.3.2 窗体文件 (Form files)	94
6.3.3 单元文件 (Unit files)	95
6.3.4 关于窗体和单元文件	96
6.3.5 桌面文件 (Desktop file)	96
6.4 保存和命名 C++ Builder 项目中的文件	97
6.5 项目管理器 (Project Manager)	103
6.5.1 项目管理器介绍	103
6.5.2 集成窗体和单元到项目中	106
6.5.3 添加窗体和单元文件	106
6.5.4 从其他项目和目录中共享文件	106
6.5.5 使用 Borland C++, C 或 Pascal 源代码单元	108
6.5.6 使用项目管理器查看窗体和单元	108
6.5.7 在项目组件之间导航	109
6.5.8 将一个窗体置于桌前	109
6.6 对象仓库 (Object Repository) 和项目	109
6.6.1 对象仓库介绍	110
6.6.2 对象仓库中对象的共享	112
6.6.3 对象仓库中的向导	113
6.6.4 使用对象仓库中的对象	113
6.6.5 将用户项目作为项目模板添加到对象仓库中	115
6.6.6 将用户窗体保存为窗体模板	116
6.6.7 使用项目模板	117
6.6.8 定制对象仓库	118
6.6.9 在一个共享环境中使用对象仓库	120
6.7 编译、建立和运行项目	121
6.7.1 单独编译一个文件	121
6.7.2 构造一个项目	123
6.7.3 建立项目	124
6.7.4 运行项目	124
第 7 章 环境选项设置	125
7.1 Preferences 页中各选项意义及配置	126
7.2 Library 页中各选项意义及配置	128
7.3 Editor 页中各选项意义及配置	131
7.4 Display 页中各选项意义及配置	135

7.5	Colors 页中各选项意义及配置	137
7.6	Palette 页中各选项意义及配置	140
第 8 章	使用集成调试器	142
8.1	程序错误的类型	142
8.2	设计一个调试方案	145
8.3	为开始一个调试过程作准备	145
8.4	控制一个程序的执行	148
8.4.1	执行到光标位置	149
8.4.2	单步执行程序代码	150
8.4.3	执行到断点位置	153
8.4.4	暂停调试程序的执行	155
8.5	重新开始一个新的调试过程	155
8.6	使用断点	155
8.6.1	设置断点	155
8.6.2	无效断点	158
8.6.3	断点管理	158
8.6.4	修改断点属性值	161
8.6.5	自定义断点和执行点颜色	162
8.7	检查程序数据值	162
8.7.1	监视表达式 (watch expression)	163
8.7.2	计算和修改表达式	166
第 9 章	一般项目选项设置	171
9.1	项目选项和其他选项概述	171
9.2	Forms 页主要选项意义及配置	172
9.3	Application 页主要选项意义及配置	174
9.4	C++页主要选项意义及配置	175
9.5	Pascal 页主要选项意义及配置	182
9.6	Linker 连接器页主要选项意义及配置	185
9.7	Directories/Conditionals 页主要选项意义及配置	187
9.8	为新项目改变默认设置	189
第 10 章	高级工程选项设置	190
10.1	编译器特定选项组	191
10.2	C++特定选项组	207
10.3	优化选项组	212
10.4	警告信息选项组	216
10.5	连接选项组	230

目 录

第 1 章 C++ Builder 简介	1
1.1 C++ Builder 开发工具特色及主要功能	1
1.2 Borland C++ Builder 运行的硬软件环境及安装	5
1.2.1 系统的硬软件环境配置要求	5
1.2.2 系统安装	5
第 2 章 C++ Builder 在线帮助的使用	13
2.1 如何获得语言定义的联机帮助	13
2.2 获得可视组件库(VCL)和运行库(RTL)的参考数据	15
2.3 程序设计时某些特定程序信息的联机帮助	15
2.4 程序设计交互式教学课程的使用	17
第 3 章 C++ Builder 快速入门	19
3.1 C++ Builder 开发环境的启动	19
3.2 C++ Builder 开发环境基本组件	20
3.2.1 C++ Builder 可视组件	21
3.2.2 C++ Builder 不可视组件	26
3.3 用 C++ Builder 开发一个程序的基本步骤	28
3.3.1 产生一个新的空白窗体	29
3.3.2 将一个新的组件加入到窗体中	29
3.3.3 设置组件的属性值 (Properties)	30
3.3.4 为组件建立事件处理句柄	31
3.3.5 嵌入用户实际作业的程序代码	33
3.3.6 生成执行程序	33
第 4 章 组件概述	35
4.1 C++ Builder 组件	35
4.2 Standard 页中组件的功能	37
4.3 Win 95 页中组件的功能	38
4.4 Additional 页中组件的功能	40
4.5 DataAccess 页中组件的功能	41
4.6 DataControls 页中组件的功能	42
4.7 Win 3.1 页中组件的功能	43
4.8 Dialogs 页中组件的功能	44
4.9 System 页中组件的功能	45
第 5 章 建立窗体和菜单	47
5.1 什么是一个窗体	47
5.2 查看窗体和程序单元	47

5.2.1	查看窗体-----	47
5.2.2	查看程序单元-----	48
5.3	建立窗体-----	48
5.3.1	什么是组件-----	49
5.3.2	在窗体中使用组件-----	50
5.3.3	在窗体中设置组件的属性-----	57
5.4	控制运行期间的窗体-----	59
5.4.1	将一个窗体指明为项目的主控窗体-----	59
5.4.2	控制窗体在运行期间的创建顺序-----	60
5.4.3	指定在运行期间自动创建的窗体-----	61
5.4.4	在运行期间实例化窗体-----	62
5.5	使用预定义的窗体-----	62
5.5.1	将一个现有的窗体加到项目中-----	63
5.5.2	继承对象仓库中的窗体-----	64
5.5.3	共享窗体-----	65
5.6	建立对话框-----	65
5.6.1	使一个对话框成为有模式的或无模式的-----	65
5.6.2	设定一个对话框窗体属性-----	66
5.6.3	指定一个对话框窗体的标题-----	67
5.6.4	创建标准命令按钮-----	67
5.6.5	按 Esc 键执行按钮代码-----	68
5.6.6	按 Enter 键执行按钮代码-----	69
5.6.7	设置一个按钮的属性来自动关闭一个对话框-----	70
5.6.8	设置 Tab 按键的行走顺序-----	70
5.6.9	测试 Tab 按键的行走顺序-----	72
5.6.10	取消一个组件的 Tab 按键顺序功能-----	72
5.6.11	使组件失效-----	72
5.7	窗体连接-----	73
5.8	创建窗体模板-----	73
5.9	创建窗体菜单-----	75
5.9.1	打开菜单设计工具-----	75
5.9.2	建立菜单-----	76
5.9.3	不打开菜单设计工具来编辑菜单项-----	82
5.9.4	使用菜单设计工具的快捷菜单来构造菜单-----	82
5.9.5	使用菜单模板来构造菜单-----	83
5.9.6	将菜单存储为模板-----	84
5.10	连接菜单事件和程序代码-----	85
5.10.1	处理菜单事件-----	85
5.10.2	动态地改变菜单项-----	86

第 6 章 项目文件	89
6.1 什么是 C++ Builder 项目?	89
6.2 组成一个 C++ Builder 项目的文件类型	89
6.3 C++ Builder 项目中各种文件	90
6.3.1 项目文件 (Project files)	90
6.3.2 窗体文件 (Form files)	94
6.3.3 单元文件 (Unit files)	95
6.3.4 关于窗体和单元文件	96
6.3.5 桌面文件 (Desktop file)	96
6.4 保存和命名 C++ Builder 项目中的文件	97
6.5 项目管理器 (Project Manager)	103
6.5.1 项目管理器介绍	103
6.5.2 集成窗体和单元到项目中	106
6.5.3 添加窗体和单元文件	106
6.5.4 从其他项目和目录中共享文件	106
6.5.5 使用 Borland C++, C 或 Pascal 源代码单元	108
6.5.6 使用项目管理器查看窗体和单元	108
6.5.7 在项目组件之间导航	109
6.5.8 将一个窗体置于桌前	109
6.6 对象仓库 (Object Repository) 和项目	109
6.6.1 对象仓库介绍	110
6.6.2 对象仓库中对象的共享	112
6.6.3 对象仓库中的向导	113
6.6.4 使用对象仓库中的对象	113
6.6.5 将用户项目作为项目模板添加到对象仓库中	115
6.6.6 将用户窗体保存为窗体模板	116
6.6.7 使用项目模板	117
6.6.8 定制对象仓库	118
6.6.9 在一个共享环境中使用对象仓库	120
6.7 编译、建立和运行项目	121
6.7.1 单独编译一个文件	121
6.7.2 构造一个项目	123
6.7.3 建立项目	124
6.7.4 运行项目	124
第 7 章 环境选项设置	125
7.1 Preferences 页中各选项意义及配置	126
7.2 Library 页中各选项意义及配置	128
7.3 Editor 页中各选项意义及配置	131
7.4 Display 页中各选项意义及配置	135

7.5	Colors 页中各选项意义及配置	137
7.6	Palette 页中各选项意义及配置	140
第 8 章	使用集成调试器	142
8.1	程序错误的类型	142
8.2	设计一个调试方案	145
8.3	为开始一个调试过程作准备	145
8.4	控制一个程序的执行	148
8.4.1	执行到光标位置	149
8.4.2	单步执行程序代码	150
8.4.3	执行到断点位置	153
8.4.4	暂停调试程序的执行	155
8.5	重新开始一个新的调试过程	155
8.6	使用断点	155
8.6.1	设置断点	155
8.6.2	无效断点	158
8.6.3	断点管理	158
8.6.4	修改断点属性值	161
8.6.5	自定义断点和执行点颜色	162
8.7	检查程序数据值	162
8.7.1	监视表达式 (watch expression)	163
8.7.2	计算和修改表达式	166
第 9 章	一般项目选项设置	171
9.1	项目选项和其他选项概述	171
9.2	Forms 页主要选项意义及配置	172
9.3	Application 页主要选项意义及配置	174
9.4	C++页主要选项意义及配置	175
9.5	Pascal 页主要选项意义及配置	182
9.6	Linker 连接器页主要选项意义及配置	185
9.7	Directories/Conditionals 页主要选项意义及配置	187
9.8	为新项目改变默认设置	189
第 10 章	高级工程选项设置	190
10.1	编译器特定选项组	191
10.2	C++特定选项组	207
10.3	优化选项组	212
10.4	警告信息选项组	216
10.5	连接选项组	230

第1章 C++ Builder 简介

1.1 C++ Builder 开发工具特色及主要功能

1997 年 6 月 26 日， Borland 中国公司在京正式发布了 Borland 公司的新产品—— Borland C++ Builder 1.0。为了满足广大的 C++ 程序员开发设计时的更高要求， Borland 公司加快了 C++ Builder 的功能升级步伐，于 1997 年末推出了 C++ Builder 2.0 版， 1998 年上半年，又将其提高到 3.0 版本。

C++ Builder 的推出和功能的不断升级，迎合了软件开发工具发展的潮流，将软件的可视化开发方式推到了一个新的高度，在快速应用开发工具（ RADT ）方面引发了一场革命。虽然 Borland 公司在这之前已在 1995 年推出了该公司的第一个可视化的开发工具 Delphi 1.0， 并已升级至现在的 4.0 版（ 1998 年下半年）， 然而大多数 Windows 程序员仍习惯使用 C 和 C++ 语言，尤其是广大的 Borland C++ 用户。现在，终于可以没有遗憾了， Borland 公司推出的 C++ Builder 开发工具，圆了 C++ 用户的梦。

初试 C++ Builder，对于使用过 Delphi 开发工具的用户来说，其开发界面（如图 1-1 所示）和环境犹如孪生的同胞兄弟，长得一模一样，不细看也许分不清是谁，界面的区别仅仅在于标题和图标上。简单地说， C++ Builder 就是 Delphi 的 C++ 版本。但实质上， C++ Builder 是基于 C++ 的，它支持 ANSI 92 的全部特性，可直接编译用 Borland C++ OWL， Delphi 及 Visual C++ 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，具有强大的数据库功能。

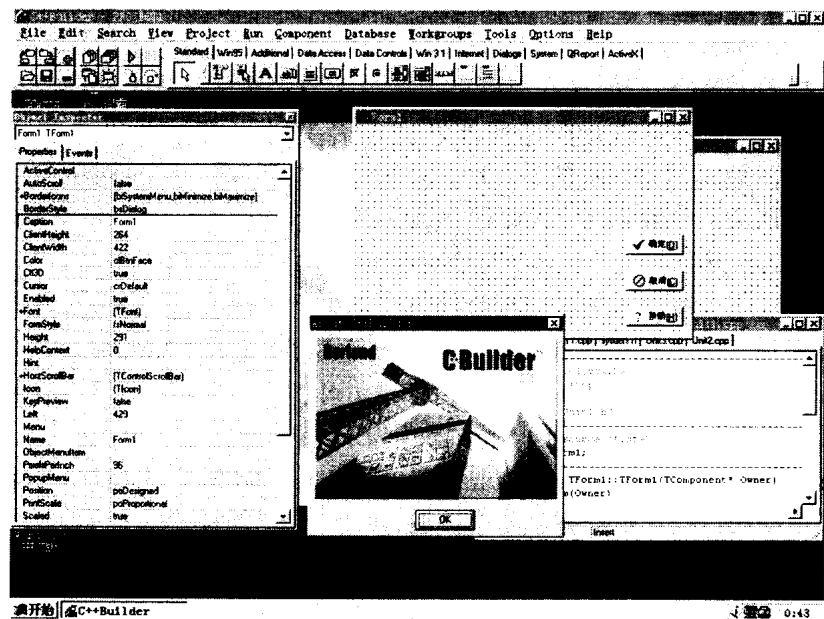


图 1-1 C++ Builder 的开发界面

C++ Builder 开发工具给使用 C/C++ 开发应用程序的人员带来了可视化编程的轻松感觉。C++ Builder 的可重用性可大幅度提高开发效率,充分发挥 C++ 语言的高效性、实时性、稳定性和底层控制能力,同时它还具有高性能的数据库连接能力。无论是想使用 C/C++ 编写应用程序来提高开发效率,或是想学习 C/C++ 最新的标准,或是想在窗体环境下学习使用 C/C++ 编写应用程序,或是从来没有学习过 MFC、OWL,而想编写基于窗体的 Windows 应用程序者的所有这些应用需要,在通过学习 Borland 的 C++ Builder 开发工具以后,都可获得圆满的解决。Borland 的 C++ Builder 是通向 Windows 可视化编程的又一值得选择的工具。

C++ Builder 是将可视化与 C++ 技术相结合的一种开发工具,为用户提供了一种全新的 Windows 应用程序开发方法。它将可视化开发环境、可视双向开发工具、可视组件库加入到 C++ 语言中,采用了具有弹性和可重用的完整的面向对象 (Object-Oriented) 程序语言,以及使用了目前世界上最快的编译器 (Compiler) 和领先的数据库技术,使开发人员不再需要在高效的底层控制与轻松的程序设计间做出艰苦的抉择。它可把开发者从繁杂的代码编写过程中解放出来,更多地关注于应用程序的功能。用户可以利用这个开发工具使用大家非常熟悉的 C++ 语言,以比从前快得多的速度设计出更完善的程序。

C++ Builder 使用了以下新技术:

● 优秀的可视化开发工具

可视化开发可以大大缩短学习周期,作为一名 C++ 的程序员,无论是使用 VC (Visual C++), BC (Borland C++) 还是其他开发工具,如果从头学起至少得半年时间才能进行真正的程序设计工作,如果使用 Borland C++ Builder 这种可视化开发工具,仅需几周时间学习便可着手进行开发工作。C++ Builder 在其集成环境中提供了可视化窗体设计窗口、对象监视器 (Object Inspector)、组件板 (Component Palette)、项目管理器 (Project Manager)、完全集成的程序代码编辑器 (Code Editor) 和集成调试器 (Integrated Debugger),给开发者提供了真正的可视化环境和对代码、资源的完全控制。再者,利用鼠标拖放进行应用程序设计的开发方式可以节省大量的时间,创建 Windows 应用程序就是从可视化的组件库中拖出一个组件放到设计的窗体中,填写好它的属性,事件及其句柄、参数都会在需要的时候自动创建。另外,大部分程序代码还可以自动生成,将开发人员从繁重的代码编写工作中解放了出来。

利用传统开发工具编写出来的应用程序或软件系统,维护起来是非常麻烦的。对于新接手的程序员来说,那一行行的源代码会使你一筹莫展。使用 Borland C++ Builder 这种以组件为中心的可视化开发工具编写的程序,可使程序员在长期的系统维护工作中得心应手、轻松自如。

像 Delphi 开发工具一样, C++ Builder 也具有 Borland 独有的双向工具特点,允许开发人员在鼠标拖动的可视化设计方面与直接编写 C++ 代码的设计方式之间随时切换,可以任意使用程序代码编辑器或用可视化设计窗口交互式进行修改, C++ Builder 会自动保持这两种设计方式的完全同步。

在数据库应用程序设计方面, C++ Builder 提供了数据感知功能,可以在程序设计阶段就与数据库建立连接,将数据库中真实数据显示出来,方便了程序设计工作。

● 高度的可重用性

Borland C++ Builder 和传统的 C/C++ 开发工具不一样的地方是以组件为中心的开发观念取代了以往以应用程序结构为中心的开发方式。

可重用的组件库（VCL — Visual Component Library）

C++ Builder 的组件库中提供了 100 多个组件。其中包含了 Windows 95 用户界面所用的全部组件；还包含了两组开发数据库程序的专用组件，可以使用户利用简单的鼠标拖动和点击方式进行各种数据库连接，并可将数据库中的数据以各种符合用户要求的不同方式出现在用户界面中，同时还提供了一组 Internet 组件和一组 ActiveX 组件。然而 Borland C++ Builder 更让人兴奋的则是它提供了这些 VCL 组件的原始程序代码。用户可以修改它们以符合自身的需要。此外，Borland C++ Builder 本身也提供了简单的用户界面让用户可以轻松地继承这些 VCL 组件特性生成新的组件。

当用户使用 Borland C++ Builder 开发应用程序的时候，可以直接应用 VCL 组件来构造用户界面。Borland C++ Builder 是一个以窗体为中心的用户设计界面，用户只需从 Borland C++ Builder 的组件板中选取需要的用户界面组件，再使用 Borland C++ Builder 的属性窗口设定组件的属性值来改变用户界面的表现方式，不再需要启动任何资源编辑器编辑用户界面后再切换回集成开发环境。当用户在窗口中加入了某个组件时，Borland C++ Builder 也已经在原始程序代码窗口中为用户产生了必要的程序代码，用户可以立刻到程序代码编辑器中编写处理这些用户界面的程序代码，或者编写应用程序本身的逻辑程序代码。

可重用和可定制的对象仓库

Borland C++ Builder 开创了新一代的可重用设计理论，利用 BDE（Borland 数据库引擎）的数据存储能力结合一个图形化的用户界面，提供了一个可重用的设计框架。在 Borland C++ Builder 的对象仓库中，包括了数据模块、项目、对话框、窗体等。Borland C++ Builder 的对象仓库功能使窗体甚至完整的应用程序具备了可重用性。所有对应用程序新的开发都可以由继承、参考乃至简单地复制等方式直接应用到开发者所设计的结构中，用户可以挑选最适合开发需要的对象结构，从而达到省时、省力、精确、最佳的目的。

● 高性能的 C++工具

高速的编译、连接和执行速度

传统的 C/C++ 开发工具在编辑、编译和连接上浪费了许多时间，往往一个程序需要编译数十分钟到数十个小时。在这种情况下，一个 C/C++ 程序员实在很难有非常高的开发效率。Borland 为了解决这个问题，花了许多精力在改善 C/C++ 程序代码的编译和连接时间上，使 Borland C++ Builder 的编译和连接速度比其他的 C/C++ 开发工具要快得多。

双编译器引擎

Borland C++ Builder 内部使用的 Borland C++ 5.02 版的 C/C++ 编译器让用户可以编

译 C/C++ 的程序代码。此外，Borland C++ Builder 还包含了一个可以编译 Object Pascal 语言的 Delphi 编译器。由于 Borland C++ 和 Delphi 使用相同的后端优化代码生成器，所以它们产生的程序代码可以很容易地连接在一起。

Borland C++ Builder 第一版虽然在 1997 年上半年才推出，但由于 Borland C++ Builder 可以编译 Object Pascal 的程序代码，因此它可以使用 Delphi 的组件。这成为 Borland C++ Builder 的有利条件，因为在 Internet 上已经有超过 1000 个 Delphi 组件了。其他的 C/C++ 开发工具中至今没有哪个有如此多的组件可供使用。

对 C++ 的完美支持

Borland C++ Builder 对 C++ 的支持能力是目前所有 C/C++ 开发工具中最齐全的。从异常处理、RTL、template 到 override virtual function return value，bool，namespace 等一应俱全。所以 Borland C++ Builder 不但可以让用户使用 C/C++ 开发应用程序，更可以让使用者学习最佳的 ANSI C++ 标准。对于原先利用 VC，BC 工具进行应用程序开发的程序员，以前的心血不会白费，Borland C++ Builder 可以直接编译原有的 C++ 代码，甚至可与 BC 使用相同的项目文件。Borland C++ Builder 也内附了 STL 的类别库供用户在程序中使用。

● 强大的数据库处理能力

编写处理数据库的应用程序是让大部分的 C/C++ 程序员伤脑筋的事情。因为 C/C++ 虽然在许多方面提供了强大的功能，但是唯独在数据库方面的支持似乎一直不受重视。

然而，现在 Borland 公司推出了 C++ Builder 开发工具，这对于所有使用 C/C++ 编写数据库应用系统的程序员而言，是一个好消息。因为 Borland C++ Builder 的数据库可以直接使用的感知组件超过 20 个。在许多情况下，用户甚至不须编写任何程序代码便可以产生一个复杂的应用程序。Borland C++ Builder 提供的数据库感知组件具有很高的使用价值。如果用户还不满足这些功能，也可以直接调用 Borland 数据库引擎 BDE/IDAPI 找寻更多的功能。

Borland C++ Builder 还提供了强大的 Borland 数据库引擎，这是一种非常成熟的数据库连接技术。它提供了三种访问数据库的方式，一是可以让用户直接存取 dBASE，FoxBase，Foxpro，Paradox 等文件型数据库生成的 DB，DBF 文件；二是提供了一个标准的 ODBC 接口，通过这个接口使用户可以存取任何一种支持 ODBC 的数据库；三是提供了一个高效的 SQL Links 数据库驱动程序，允许用户直接存取 Oracle，Informix，SyBase，MS SQLServer，DB2 和 Borland 的 InterBase。而 SQL Links 是一种速度非常快的数据库驱动程序，使用它用户可以拥有非常高的数据库存取能力。

● 完美的中文支持能力

对于广大的中国开发人员来说，使用 Borland C++ Builder 还有一个优越的功能就是它对中文具有完美的支持能力。Borland C++ Builder 采用双字节内核设计，完全解决了“半个字”的问题。除了它的集成开发环境支持 DBCS 之外，它的 VCL 组件也和中文完全相容。Borland C++ Builder 的 VCL 组件甚至直接支持中文输入法的切换，这种高度中文兼容能力是其他一些开发工具到目前还无法做到的。使用 Borland C++ Builder 不需要

担心中文的问题，因为它表现得比 Delphi 还好。

Borland C++ Builder 产品系列针对不同类型的开发者提供了与之相适应的不同版本，从开发建立有决策性能的 Client/Server 应用程序者到刚刚起步的初级程序员，Borland C++ Builder 总有一个版本适合你。Borland C++ Builder 的系列产品如下：

- **Borland C++ Builder Client/Server Suite**
适合开发单位、专家和增值供货商，以及需要在 Client/Server 解决方案中使用可视化的 C++ 编程工具去连接组织数据的所有开发者。
- **Borland C++ Builder Professional**
是为专业程序员特别设计的，适用于使用 C++ 建立商用 Windows 应用程序的增值供货商、独立软件商和专家。
- **Borland C++ Builder Standard**
适用于开始使用基于 C++ 的真正可视化开发工具的程序员。
- **Learn to Program with Borland C++ Builder**
适用于学生、程序设计爱好者和一切想学习 C++ 和 Windows 编程的人士。

1.2 Borland C++ Builder 运行的硬软件环境及安装

1.2.1 系统的硬软件环境配置要求

- 基于 Intel 486 以上的 CPU ；
- Microsoft Windows 95 ， Windows NT 3.51 或 4.0 的操作系统；
- 最小内存需要 16MB （建议 24MB ）；
- 分辨率相当于 VGA 以上的显示器；
- 鼠标器或其他 Windows 支持的定位设备；
- 所需硬盘空间（会因簇的大小而有所不同）：

LearnToProgram	75MB
Standard	75MB
Professional	100MB
Client/Server	130MB

1.2.2 系统安装

Borland C++ Builder 与它的安装程序都是运行于 Microsoft Windows 95 ， Windows NT 3.51 或 4.0 操作系统下的，显示卡硬件必需支持 VGA 以上的分辨率，因此在安装 Borland C++ Builder 软件之前必须检查是否满足必需的操作环境环境和必要的硬件要求，否则不能完成系统的安装。在符合安装要求的情况下，安装时安装程序会生成所需的目录，并从磁盘或光盘中复制所需的文件，更新必要的配置文件。安装程序在使用上非常简单明了，容易操作。下面以在 Windows 95 操作系统下的安装为例说明 Borland C++ Builder 的安装过程。

在 Windows 95 下安装 Borland C++ Builder ，通常是按以下 5 个步骤来进行：

1. 启动进入 Microsoft Windows 95 环境;
2. 将含 Borland C++ Builder 软件的光盘放入光驱或将第一张软盘放入软驱中;
3. 单击“开始”按钮, 然后指向“运行”并单击;
4. 在“运行”对话框中键入 C++ Builder 光盘中的 Setup.exe 安装程序, 或通过对话框中的“浏览”命令运行 C++ Builder 光盘中的 Setup.exe 安装程序;
5. 接着按照安装程序的提示来进行安装。

当安装程序完成安装后, 可以选择阅读 Readme.hlp 文件, 其中包含了许多 C++ Builder 的重要信息, 建议用户在运行 C++ Builder 前先仔细阅读它。

以下是 Borland C++ Builder 在 Windows 95 下的完整安装过程。

● 开始安装

安装既可以从光盘也可以从硬盘上进行。如果有足够的硬盘空间, 建议使用硬盘进行安装, 因为这样安装进行速度会更快一些。本安装过程以光盘安装为例进行说明。如果要 from 硬盘进行安装, 只需将 C++ Builder 的安装文件拷贝到硬盘, 其他步骤与光盘安装没有区别。

首先将安装盘放进光驱, 在 Windows 95 下单击“开始”按钮, 然后指向“运行”并单击, 在“运行”对话框中键入 C++ Builder 光盘中的 Setup.exe 安装程序 (如图 1-2 所示, 图中的盘符及路径是指机器上的光驱和光盘中 C++ Builder 安装程序 Setup.exe 的存储位置)。

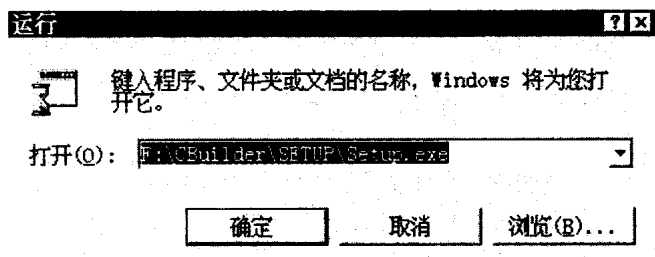


图 1-2 启动 C++ Builder 光盘中的 Setup.exe 安装程序

在“运行”对话框中单击“确定”按钮, 首先启动安装向导, 如图 1-3 所示。

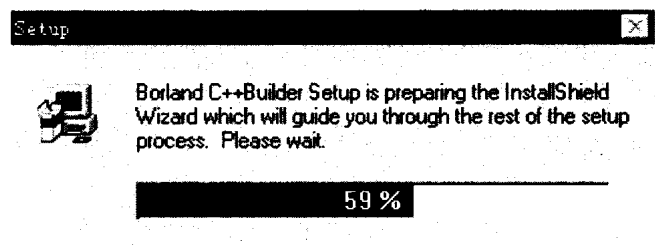


图 1-3 启动安装向导

C++ Builder 有三种版本: 标准版 (Standard Version)、专业版 (Professional Version) 和客户端/服务器版 (Client/Server Version)。安装过程是以客户端/服务器版为例, 因此在屏幕上显示着“ Client/Server ”字样。如果想退出安装, 可单击“ Cancel ”按钮。在以后的安装步骤中, 都设有“ Cancel ”按钮, 以提供随时退出安装的途径。如果继