

王 宏
傅新苗
冉 欣 编著

C++ Builder

程序设计教程



清华大学出版社
<http://www.tup.tsinghua.edu.cn>

C++Builder 程序设计教程

王宏 傅新苗 冉欣 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书主要介绍了 C++Builder 的一些基础知识, 包括 C++Builder 的基本操作, C++Builder 对 C++ 的实现方式和对面向对象程序设计的支持, 同时, 本书还详细介绍了可视化组件库在 C++Builder 中的应用。在此基础上, 对 C++Builder 在多媒体、数据库、图形图像等领域的应用做了比较深入的介绍。本书最后还介绍了 C++Builder 对多线程应用程序的支持情况。本书在介绍上述内容的同时, 提供了大量的应用实例, 其中不乏一些综合性较强的例子。

本书适合各种层次的 C++Builder 爱好者, 尤其对大学生、软件开发人员有很大的指导作用。同时, 它也可以作为 C++Builder 培训班的教材和参考书。

版权所有, 翻印必究。本书封面贴有清华大学出版社激光防伪标签, 无标签者不得销售。

书 名: C++Builder 程序设计教程
作 者: 王宏 傅新苗 冉欣 编著
出 版 者: 清华大学出版社 (北京清华大学学研楼, 邮编 100084)
<http://www.tup.tsinghua.edu.cn>
责任编辑: 童隆裴
印 刷 者: 北京市清华园胶印厂
发 行 者: 新华书店总店北京发行所
开 本: 787×1092 1/16 印张: 33.5 字数: 794 千字
版 次: 2000 年 3 月第 1 版 2000 年 3 月第 1 次印刷
书 号: ISBN 7-302-01322-5/TP·513
印 数: 0001~6000
定 价: 48.00 元

前 言

在基于图形界面的操作系统 Windows 取得了巨大成功的同时，基于图形界面的可视化各种开发工具也就应运而生，其中 Inprise 公司的 C++Builder 就是其中一员。Inprise 的前身之一是 Borland 公司。在中国大陆，Borland 公司的 TC 和 BC 系列软件曾经得到过非常广泛的应用。Inprise 公司在其多年开发 C++语言开发工具的基础之上，推出了基于图形界面的、可视化的 C++开发工具——C++Builder。经过不断的发展，现在 C++Builder 的最新版本为 4.0 版。

C++语言作为在计算机业界久经考验的语言，以其简洁明快的编程风格、功能强大的编程能力和众多的开发工具而著称于世。C++Builder 提供了对标准 C++的完全支持。使用任何一个 C++开发工具所能实现的功能也都能通过 C++Builder 来实现。而且，C++Builder 的编译器还提供了与当今主流 C++开发工具（包括 Microsoft 公司的 Visual C++、原 Borland 公司的 Borland C++和 Turbo C++等）的良好兼容，这种兼容为用户的代码最大限度地重用提供了可能。同时 C++Builder 中还对标准 C++做了一定的扩充，以适应新形势的需要。这些扩充进一步丰富和完善了 C++语言的编程能力。

C++提供了对 VCL（Visual Component Library）的良好支持。VCL 最初是由 Inprise 公司开发的一套基于 Windows 程序设计的可视组件库，这个组件库最初应用于 Inprise 公司的 Delphi 中，并获得了巨大的成功。现在，Inprise 公司把这套可视组件库应用于 C++Builder 中，使 C++Builder 在拥有 C++强大的编码能力的同时，还拥有了强大的用户界面设计能力。同时 C++Builder 还是一个跨平台的开发工具，它可以在 Windows 95/98、Windows NT 4 等操作系统上实现兼容，用户无需为每一种开发平台而设计专门的程序。

在国外，使用 C++Builder 进行应用开发已经成为一种潮流，介绍 C++Builder 的书籍也很多。在我国台湾，C++Builder 也得到了高度的重视，并有大量的书籍介绍这方面的知识。但是在国内，介绍这方面的书籍还很少。鉴于此，作者在长期使用 C++Builder 的基础之上，编写了本书。本书较为全面地介绍了 C++Builder 的使用、C++Builder 的程序设计和 C++Builder 在各个领域的应用等方面的知识，希望能收到抛砖引玉之效果。

由于作者能力有限，本书错误在所难免，欢迎广大读者批评指正。

作 者

1999 年 5 月于北京

目 录

第 1 章 C++Builder 的安装	1
1.1 C++Builder 概述	1
1.2 C++Builder 的安装	2
1.2.1 C++Builder 的运行环境	2
1.2.2 C++Builder 的安装步骤	2
1.2.3 C++Builder 的卸载	11
1.3 C++Builder 的启动	13
第 2 章 C++Builder 的开发环境	14
2.1 C++Builder 的操作界面	14
2.1.1 主窗口	15
2.1.2 对象监视器	30
2.1.3 窗体	34
2.1.4 代码编辑器	39
2.2 C++Builder 中的组件索引	40
2.2.1 Standard 选项卡	40
2.2.2 Additional 选项卡	41
2.2.3 Win32 选项卡	41
2.2.4 System 选项卡	42
2.2.5 Internet 选项卡	43
2.2.6 Data Access 选项卡	44
2.2.7 Data Controls 选项卡	44
2.2.8 Midas 选项卡	45
2.2.9 Decision Cube 选项卡	46
2.2.10 QReport 选项卡	46
2.2.11 Dialogs 选项卡	47
2.2.12 Win3.1 选项卡	48
2.2.13 Samples 选项卡	48
2.2.14 ActiveX 选项卡	49
2.3 建立应用程序	49
2.3.1 选择组件	50
2.3.2 设置属性	50
2.3.3 编写程序代码	52

2.3.4	程序编译运行	54
2.3.5	文件的保存	54
2.4	项目文件的管理	55
2.4.1	项目的概念	55
2.4.2	项目管理器	57
2.4.3	项目管理器的设置	58
2.5	集成调试器	61
2.5.1	断点操作	62
2.5.2	设计执行方式	64
2.5.3	加入观察窗	65
2.5.4	Call Stack 窗口	66
2.6	C++Builder 开发环境的设置	67
2.6.1	组件板的设置	67
2.6.2	设置 IDE	68
2.7	C++Builder 的帮助信息	71
2.7.1	C++Builder 的联机帮助信息	71
2.7.2	C++Builder 中的代码洞察功能	73
2.8	其它软件公司提供的组件在 C++Builder 中的应用	73
第 3 章	C++基础知识	77
3.1	C++的发展历程	77
3.2	一个简单 C++程序	79
3.3	数据类型	79
3.3.1	简单数据类型	80
3.3.2	复合数据类型	81
3.4	常量	86
3.4.1	整型常量	86
3.4.2	浮点常量	87
3.4.3	字符常量	87
3.4.4	常量的定义	88
3.5	变量	89
3.5.1	变量的说明和定义	90
3.5.2	变量名	90
3.6	指针	91
3.6.1	指针的类型及其定义	91
3.6.2	指针的初始化	92
3.6.3	字符串指针	93
3.6.4	引用类型	94

3.7 表达式	95
3.7.1 运算符	95
3.7.2 类型转换	102
3.8 控制语句	103
3.8.1 if 语句	103
3.8.2 switch 语句	105
3.8.3 循环语句	109
3.8.4 跳转语句	112
3.9 函数	115
3.9.1 函数的定义	115
3.9.2 函数的参数	117
3.9.3 函数的返回值	120
3.9.4 函数的调用	121
3.9.5 函数指针	123
3.9.6 内部函数和外部函数	126
3.9.7 局部变量和全局变量	126
3.9.8 动态存储变量和静态存储变量	127
3.10 编译预处理	128
3.10.1 文件包含	129
3.10.2 宏定义	129
3.10.3 条件编译	132
3.11 流	134
3.12 重载	135
3.12.1 函数重载	136
3.12.3 运算符重载	137
3.13 文件	138
第 4 章 面向对象的程序设计	141
4.1 面向对象程序设计的基本概念	141
4.1.1 类与对象	141
4.1.2 封装	142
4.1.3 继承	143
4.1.4 多态性	144
4.2 C++Builder 中的类	145
4.2.1 类的定义	145
4.2.2 类的数据成员和方法	149
4.3 构造函数和析构函数	153
4.3.1 构造函数	154

4.3.2	析构函数	157
4.3.3	对象数组	158
4.3.4	类的转换	160
4.3.5	友元	161
4.4	类与指针	162
4.4.1	this 指针	163
4.4.2	指向类对象的指针	164
4.4.3	指向类成员的指针	164
4.5	继承	166
4.5.1	派生类	166
4.5.2	访问控制	167
4.5.3	成员覆盖	168
4.5.4	多重继承	169
4.5.5	构造函数的继承	170
4.6	多态性和虚拟函数	171
4.6.1	多态性	171
4.6.2	滞后绑定	172
4.6.3	虚拟函数	172
4.6.4	抽象基类	174
4.6.5	虚拟函数与滞后绑定的关系	175
4.6.6	虚拟函数的数据封装	175
4.6.7	虚拟析构函数	175
4.7	操作符重载	176
4.7.1	单目运算符的重载	176
4.7.2	双目运算符的重载	180
4.7.3	几个特殊运算符的重载	181
4.8	C++Builder 的可视组件库	183
4.8.1	可视组件库概述	183
4.8.2	自定义组件类	184
4.8.3	修改一个已存在的组件类	184
4.8.4	创建 Windows 组件	185
4.9	异常处理	186
4.9.1	C++Builder 中异常处理的基础知识	186
4.9.2	异常处理在程序中的应用	188
第 5 章	标准组件	192
5.1	窗体的初步操作	192
5.1.1	窗体的基本操作	192

5.1.2 窗体对象的属性	193
5.1.3 窗体对象的方法	196
5.1.4 窗体对象的事件	198
5.2 标准组件概述	199
5.3 MainMenu 和 PopupMenu 组件	199
5.3.1 MainMenu 和 PopupMenu 组件的主要属性	199
5.3.2 MainMenu 和 PopupMenu 组件的主要方法	202
5.3.3 MainMenu 和 PopupMenu 组件的主要事件	203
5.4 Label 组件	204
5.4.1 Label 组件的重要属性	204
5.4.2 Label 组件的重要方法	205
5.4.3 Label 组件的重要事件	206
5.5 Edit 组件	206
5.5.1 Edit 组件的主要属性	206
5.5.2 Edit 组件的主要方法	208
5.6 Memo 组件	209
5.7 Button 组件	210
5.7.1 Button 组件的主要属性	210
5.7.2 Button 组件的主要方法	211
5.8 CheckBox 组件	211
5.8.1 CheckBox 组件的主要属性	211
5.8.2 CheckBox 组件的主要方法	212
5.9 RadioButton、RadioGroup 和 GroupBox 组件	213
5.9.1 RadioButton 组件	213
5.9.2 RadioGroup 组件	213
5.9.3 GroupBox 组件	214
5.10 ListBox 组件	214
5.10.1 ListBox 组件的主要属性	215
5.10.2 ListBox 组件的主要方法	219
5.11 ComboBox 组件	221
5.12 ScrollBar 组件	225
5.13 菜单设计	227
5.13.1 菜单编辑环境	227
5.13.2 菜单设计	228
5.13.3 快捷键及热键设计	230
5.13.4 快捷菜单设计	231
5.13.5 菜单模板	231

第 6 章 设计用户界面的增强型组件	233
6.1 BitBtn 组件	233
6.2 SpeedButton 组件	237
6.3 StringGrid 组件	238
6.3.1 StringGrid 组件的主要属性	239
6.3.2 StringGrid 组件的主要方法	242
6.3.3 StringGrid 组件响应的主要事件	243
6.3.4 DrawGrid 组件简介	244
6.4 Bevel、Shape 和 Image 组件	244
6.4.1 Bevel 组件	244
6.4.2 Shape 组件	245
6.4.3 Image 组件	246
6.5 MaskEdit 组件和 StaticText 组件	249
6.5.1 MaskEdit 组件	249
6.5.2 StaticText 组件	255
6.6 Chart 组件	251
6.6.1 创建图表的一般步骤	251
6.6.2 Chart 组件的主要属性	254
6.7 ScrollBox 组件	255
6.7.1 ScrollBox 组件的主要属性	256
6.7.2 ScrollBox 组件的主要方法	257
6.8 Splitter 和 CheckListBox 组件	257
6.8.1 Splitter 组件	257
6.8.2 CheckListBox 组件	259
6.9 OpenFileDialog 和 SaveDialog 组件	259
6.9.1 OpenFileDialog 对话框的主要属性	260
6.9.2 OpenFileDialog 组件的主要方法	263
6.9.3 SaveDialog 组件	264
6.10 OpenPictureDialog 和 SavePictureDialog 组件	264
6.10.1 OpenPictureDialog 组件	265
6.10.2 SavePictureDialog 组件	265
6.11 FontDialog 和 ColorDialog 组件	265
6.11.1 FontDialog 组件	266
6.11.2 ColorDialog 组件	267
6.12 PrintDialog 和 PrinterSetupDialog 组件	268
6.12.1 PrintDialog 组件	268
6.12.2 PrinterSetupDialog 组件	271
6.13 FindDialog 和 ReplaceDialog 组件	271

6.13.1 FindDialog 组件	271
6.13.2 ReplaceDialog 组件	272
6.14 用于文件管理的 Win3.1 组件	274
6.14.1 DirectoryListBox 组件	274
6.14.2 FileListBox 组件	275
6.14.3 DriveComboBox 组件	277
6.14.4 FilterComboBox 组件	278
第 7 章 Win32 选项卡下的组件	280
7.1 TabControl 组件	280
7.1.1 TabControl 组件的属性	281
7.1.2 TabControl 组件响应的事件	283
7.2 PageControl 组件	283
7.2.1 PageControl 组件的主要属性	284
7.2.2 TabSheet 的主要属性	285
7.2.3 PageControl 组件的主要方法	287
7.3 ImageList 组件	289
7.3.1 ImageList 组件的主要属性	290
7.3.2 ImageList 组件的主要方法	291
7.4 RichEdit 组件	292
7.4.1 RichEdit 组件的主要属性	292
7.4.2 RichEdit 组件的主要方法	295
7.5 ProgressBar 和 StatusBar 组件	296
7.5.1 ProgressBar 组件的主要属性	296
7.5.2 ProgressBar 组件的主要方法	298
7.5.3 StatusBar 组件	299
7.6 Updown、HotKey 和 DateTimePicker 组件	301
7.6.1 UpDown 组件	301
7.6.2 HotKey 组件	303
7.6.3 DateTimePicker 组件	303
7.7 TreeView 和 ListView 组件	304
7.7.1 TreeView 组件的主要属性	305
7.7.2 TreeView 组件的主要方法	308
7.7.3 ListView 组件	308
7.8 ToolBar 和 CoolBar 组件	311
7.8.1 ToolBar 组件	312
7.8.2 CoolBar 组件	312
7.9 HeaderControl 组件	313

第 8 章 图形操作	315
8.1 图形组件	315
8.1.1 Image 组件的使用	315
8.1.2 ImageList 组件的使用	318
8.1.3 FontDialog(字体对话框)组件的使用	319
8.1.4 ColorDialog(颜色对话框)组件和 ColorGrid (颜色表格) 组件的使用...	321
8.2 图形组件	323
8.2.1 TPen 对象	323
8.2.2 Brush (画刷) 的使用	326
8.2.3 Canvas (画面) 的使用	328
8.2.4 画图函数	330
8.3 简单绘图器的设计	333
8.4 图形设计的综合使用	339
8.5 C++Builder 与 OpenGL 程序接口	352
8.5.1 OpenGL 概述	352
8.5.2 OpenGL 的工作顺序	352
8.5.3 OpenGL 库函数	353
8.5.4 OpenGL 的颜色	353
8.6 OpenGL 程序设计实例	353
第 9 章 开发数据库应用程序	367
9.1 C++Builder 中数据库应用程序的结构	367
9.2 设计数据库应用程序的基本组件和辅助工具	370
9.2.1 C++Builder 中用于开发数据库应用程序的组件概述	370
9.2.2 C++Builder 提供的用于设计数据库 应用程序的工具软件	371
9.3 数据集组件及 TDataSource 组件	372
9.3.1 TTable 组件	372
9.3.2 TTable 组件在应用程序中的应用	377
9.3.3 TQuery 组件	383
9.3.4 TQuery 组件在应用程序中的应用	386
9.3.5 TStoredProc 组件	389
9.3.6 TDataSource 组件	391
9.4 数据查找	392
9.4.1 使用 GotoKey 方法查找数据	393
9.4.2 使用 FindKey 方法查找数据	393
9.4.3 不精确查找	394
9.4.4 用 TQuery 组件的动态 SQL 语句进行查询	395

9.5 数据控制组件	396
9.5.1 TDBGrid 组件	396
9.5.2 TDBNavigator 组件	399
9.5.3 TDBText 组件	401
9.5.4 TDBEdit 组件	401
9.5.5 TDBMemo 组件	402
9.5.6 TDBImage 组件	402
9.6 快速报表	403
9.6.1 一个简单的报表	403
9.6.2 TQuickRep 组件	405
9.6.3 创建报表	406
9.7 BDE 和 Database Desktop 简介	410
9.7.1 BDE 简介	410
9.7.2 Database Desktop 简介	413
9.8 BDE 程序设计	420
9.8.1 BDE 程序设计概述	420
9.8.2 BDE API 函数	421
9.8.3 BDE 程序实例	426
9.9 数据库窗体向导的应用	429
9.9.1 利用向导生成简单窗体	430
9.9.2 利用向导生成主/从结构的窗体	433
第 10 章 多媒体应用程序	437
10.1 TAnimate 组件的使用	437
10.1.1 TAnimate 组件的重要属性	437
10.1.2 TAnimate 组件的主要方法	440
10.1.3 TAnimate 组件响应的事件	440
10.1.4 TAnimate 组件的实际应用	440
10.2 TMediaPlayer 组件的使用	444
10.2.1 TMediaPlayer 组件的重要属性	445
10.2.2 TMediaPlayer 组件的重要方法	451
10.2.3 TMediaPlayer 组件的响应事件	456
10.3 多媒体应用程序实例	456
10.4 游戏制作	458
第 11 章 应用程序的开发	470
11.1 C++Builder 应用程序模板的使用	470
11.1.1 多文档模板的使用	471
11.1.2 应用向导 (Application Wizard) 的使用	472

11.1.3 其它模板的使用	474
11.2 设计多卡片应用程序	475
11.3 对话框的设计	482
11.3.1 MessageDlg 函数	482
11.3.2 MessageDlgPosHelp 函数	483
11.3.3 ShowMessage 函数	484
11.3.4 InputBox 函数	484
11.3.5 MessageBox 函数	485
11.3.6 MessageDlgPos 函数	486
11.4 多窗体界面	487
11.5 多文档应用程序的设计	493
11.5.1 多文档应用程序的概念	493
11.5.2 多文档的图片文本浏览器	494
第 12 章 开发多线程应用程序	503
12.1 线程与进程	503
12.2 C++Builder 中的线程类	504
12.3 线程管理	508
12.3.1 线程的优先权	508
12.3.2 线程的本地存储	509
12.4 线程同步	510
12.4.1 临界区	511
12.4.2 互斥变量	511
12.4.3 信号量	512
12.4.4 死锁	513
11.5 线程和可视组件库	513
12.6 应用线程的例子	514

第 1 章 C++Builder 的安装

1.1 C++Builder 概述

在计算机技术迅速发展的今天，程序设计语言也在不断地更新换代。

在 Microsoft 推出基于可视化原理的 Windows 系列操作系统以后，广大程序员面对的一个新课题，就是如何迅速高效地开发出基于 Windows 操作系统、界面友好、功能强大的应用程序来。显然，如果仍然使用陈旧的开发工具，已不能适应新形势的需要。

C++Builder 就是这样一个用于在 Windows 环境下进行快速开发的程序设计工具。它是 Inprise 公司在原 Borland 公司的 Borland C 系列软件的基础上，结合了可视化编程的思想而精心设计的一个以 C++语言为后台程序设计语言、以可视组件库为基础、同时支持 Microsoft 公司的 MFC (Microsoft Fundamental Class) 类库和 Borland 公司的 OWL(Object Windows Library)类库的优秀开发工具。

C++语言作为在计算机业界久经考验的语言，以其简洁明快的编程风格、功能强大的编程能力、众多的开发工具而著称于世。C++Builder 提供了对标准 C++的完全支持，用户使用任何一个 C++开发工具所能实现的功能，也都能通过 C++Builder 来实现。而且，C++Builder 的编译器还提供了与当今主流 C++开发工具（包括 Microsoft 公司的 Visual C++、原 Borland 公司的 Borland C++和 Turbo C++等）的良好兼容性，这种兼容性为用户的代码最大限度地重用提供了可能。同时 C++Builder 中还对标准 C++做了一定的扩充，以适应新形势的需要，这些扩充进一步丰富和完善了 C++语言的编程能力。

C++提供了对 VCL (Visual Component Library) 的良好支持，VCL 最初是由 Inprise 公司开发的一套基于 Windows 程序设计的可视组件库，这个组件库最初应用于 Inprise 公司的 Delphi 中，并获得了巨大的成功。现在，Inprise 公司把这套可视组件库应用于 C++Builder 中，使 C++Builder 在拥有 C++强大的编码能力的同时，还拥有了强大的用户界面设计能力。这无疑是强强携手，必然会给用户带来莫大的好处。同时 C++Builder 还是一个跨平台的开发工具，它可以在 Windows 95/98、Windows NT 4 等操作系统上实现兼容，用户无需为它们专门设计程序。

C++Builder 本身也在不停地发展、不断地完善，现在最新的版本是 C++Builder 4.0 版，新版的 C++Builder 较之以前的版本，不仅优化了开发环境、重新设计了编译器，而且还提供了很多功能强大的新组件、提供了对更多的数据库引擎的支持、提供了对 Windows NT 5 等最新一代操作系统的支持，同时众多的中间软件商也加入到 VCL 库的开发中，Inprise 公司在其发布的 C++Builder 4.0 软件中，同时向用户提供了一个由其它软件商开发的可视组件库。这些可视组件库极大地扩展了 C++Builder 的应用领域，大大地减轻了程序员的编

程负担。用户还可以通过网络等多种渠道获得为 C++Builder 设计的最新可视组件库。而且用户还可以自行开发可视组件，以备在今后的程序开发中使用。

最新版的 C++Builder 4 较之以前的版本来说有了很大的改进，其中最为明显的是大大地扩展了可视组件库。现在，C++Builder 4 中支持的组件与 Delphi 4 中支持的组件是相同的。也就是说，用户通过 Delphi 可以实现的应用程序，通过 C++Builder 也是可以实现的。同时，大量第三方开发者的加入，使得 C++Builder 中可以使用的组件日益增多，在伴随 C++Builder 4 提供给用户的光盘中，Inprise 还收录了大量来自第三方开发者的组件。

C++Builder 4 还大大优化了编译器，使用户开发时间大大缩短，应用程序的性能大为提高。集成开发环境也做了相应的改进，即使是初次接触 C++Builder 的人，也可以很快掌握它的使用。

1.2 C++Builder 的安装

1.2.1 C++Builder 的运行环境

C++Builder 4 有几种安装方式，每种安装方式对系统配置的要求不尽相同，尤其是对硬盘的要求有很大的不同。C++Builder 4 安装的最低配置如下所示：

- CPU：主频 100MHz 以上
- 内存：24MB（推荐使用 32MB 或者更高，内存的多少主要影响应用程序的编译速度）
- 可用硬盘空间：130MB 以上
- 操作系统：Windows 95/98 或者 Windows NT 4

1.2.2 C++Builder 的安装步骤

选中含有 C++Builder 的光盘，单击鼠标右键，在如图 1-1 所示的快捷菜单中选择“自动播放”菜单项，就可以启动安装程序。

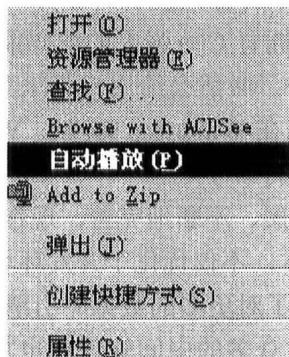


图 1-1 选择“自动播放”菜单项

安装程序自动启动后的初始界面如图 1-2 所示。

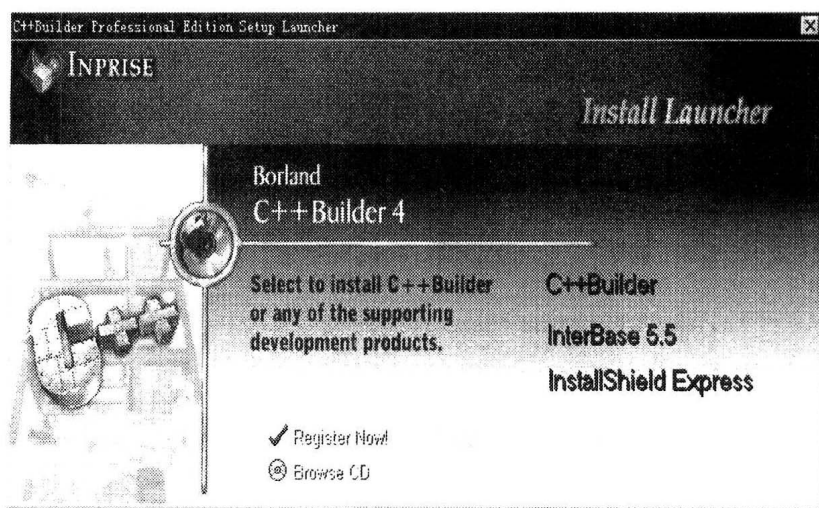


图 1-2 安装程序启动界面

在图 1-2 所示的界面中，用户可以进行下面一些操作：

Register Now: 单击该选项，用户可以立即通过 Internet 进行注册，这时将启动注册程序，如图 1-3 所示。注册后，可以获得最新的技术支持信息和产品升级信息，还可以参加 Inprise 公司组织的技术讨论会等。

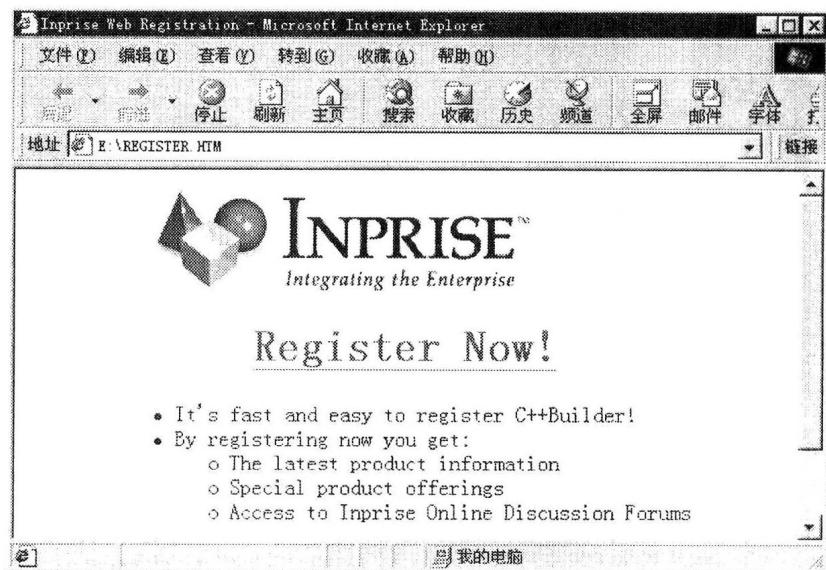


图 1-3 启动注册程序

Browse CD: 单击该选项，可以浏览光盘上的所有文件和目录列表。

C++Builder: 单击该选项，将启动 C++Builder 主程序的安装。

InterBase 5.5: 单击该选项，将启动 InterBase 5.5 的安装，启动 InterBase 5.5 时，Inprise 随 C++Builder 提供一个数据库管理系统。