

C++ Builder 4 编程技术

刘振安 刘大路 编著
张蕊 秦俊

人民邮电出版社

内 容 提 要

本书以最新的 C++ Builder 4 版本为对象，介绍 C++ Builder 的编程技术。全书共分 12 章和一个附录。第一章是令人耳目一新的 C++ Builder；第二章是 C++ Builder 4 编程的优点；第三章是 C++ Builder 4 应用基础；第四章是窗体的属性及设计；第五章是常用组件控制；第六章是鼠标和键盘的控制；第七章是文件管理和文本编辑器；第八章是图形图像处理；第九章是奇妙的多媒体世界；第十章是数据库控制；第十一章是如何使用自定义的类；第十二章是程序的发行。附录中给出虚拟键值表。本书写作密切结合实际，以实用为前提，用最少的篇幅介绍尽量多的内容。读者可以按照本书的指导，一边上机，一边学习。

本书可作为各种培训班和读者自学 C++ Builder 编程的教材，也可以作为广大计算机工作者学习 C++ Builder 编程的常备参考书。

前 言

自 Windows 问世以来，其漂亮的图形用户界面(GUI)、高效的内存管理、多进程和多线程的并发机制及丰富的多媒体服务，深深地吸引了广大用户。随着计算机硬件功能的不断提高，不仅 Windows 已经成为软件发展的趋势，而且 Windows 程序的设计方法也在不断完善。例如，传统的 Windows 程序设计方法，使用的仍然是 C 语言的程序设计技术。尽管传统的 Windows 程序是模块化的，但其结果代码有时很长，也很麻烦。虽然 C++ 的 Windows 编程技术简化了编程，但仍然没有在程序结构上有所突破。Windows 软件开发是一个复杂庞大的工程，因此又使许多程序员望而却步。Visual Basic 的出现，使开发 Windows 应用程序变得简单方便。但好景不长，由于 Basic 自身的缺陷，使许多工作难以实现。虽然 Borland C++ 及 Visual C++ 等可视化的 C 语言功能强大，可是其所依赖的 OWL 和 MFC 类库却非常复杂，熟悉起来相当不易。Delphi 这一快速程序开发工具的推出，以其高效、舒适、方便的开发环境迅速风靡全球，不过，要使用 Delphi，还要先学习 Pascal 语言的有关知识。

Borland 公司集数年开发 Borland C++ 和 Delphi 之经验，推出了 C++ Builder 这一划时代的 C++ 语言开发工具，它结合了 Borland C++ 和 Delphi 的优点，把可视化开发环境、开发工具和可视组件库加入到 C++ 语言中，实现了真正的面向对象的开发。

C++ Builder 作为一种全新的面向对象的可视化编程语言，与以往的 C++ 和 VB 等相比，不论是编程技巧、文件结构、语言组织，还是事件处理方法等，都有其独特的特点。由于 Windows 采用的是消息驱动，传统的 Windows 编程总是要处理一大堆各种各样的消息循环，使用大量的 switch 和 case 语句，就是专业的程序员，稍有不慎，程序也会出错，这也是一般软件人员对 Windows 望而却步的原因。C++ Builder 则不同，它采取了把 Windows 的这些消息都封装在 VCL 类库中的方法，使得应用程序不必再处理繁琐的消息循环，只要编写组件相应的消息处理函数即可，这样一来，Windows 编程就变得非常方便容易。相信读者接触 C++ Builder 后，一定会对之爱不释手，并迅速转移到使用 C++ Builder 来进行 Windows 程序的设计和开发。

本书是以最新的 C++ Builder 4 版本为对象，介绍 C++ Builder 的编程技术。

参加本书编写工作的有：刘振安、刘大路、张蕊、秦俊、苏仕华、赵晓东、徐峰、颜廷荣、孙宝其和孙敏等。

把当今国际上最新的计算机技术及时地介绍给广大读者，是科技工作者的光荣使命，但由于我们才疏学浅，疏漏之处在所难免，敬请同行及读者批评指正。

刘振安
1999 年 7 月

目 录

第一章 令人耳目一新的 C++ Builder	1
1.1 C++ Builder 的迷人之处	1
1.2 安装并运行 C++ Builder 4	2
1.3 熟悉 C++Builder	9
1.4 可视编程的基本概念	9
1.5 简单的 C++ Builder 编程实例	10
第二章 C++ Builder 4 编程的优点	18
2.1 C 程序编制 Windows 基本窗口	18
2.2 C++ 设计 Windows 应用程序	23
2.3 ObjectWindows 设计 Windows 应用程序	33
2.4 C 与 C++编程比较	38
2.5 C++ Builder 的编程特点	38
第三章 C++ Builder 4 应用基础	44
3.1 集成开发环境简介	44
3.2 C++ Builder 主窗口	45
3.3 对象查看器	48
3.4 项目管理器窗口	50
3.5 对象模板和专家工具	55
3.6 菜单设计器	56
3.7 集成调试器	56
3.8 C++ Builder 外接工具	57
第四章 窗体的属性及设计	59
4.1 C++ Builder 基本文件及其功能	59
4.2 窗体属性	61
4.3 利用组件设计窗体	62
4.4 动态改变窗体属性	65
第五章 常用组件控制	69
5.1 文字的输出输入控制	69
5.2 按钮对象	78
5.3 单选钮及复选框	81
5.4 列表框及组合框	83
第六章 鼠标和键盘的控制	89
6.1 鼠标的控制	89
6.2 键盘控制方法	98
6.3 Borland C++ Builder 中的集合	103

第七章 文件管理和文本编辑器	110
7.1 文件管理的基本概念	110
7.2 文件管理	112
7.3 文件组件	114
7.4 使用文件组件进行文件操作实例	116
7.5 文本编辑器的开发	122
第八章 图形图像处理	145
8.1 Windows 图形设备接口	145
8.2 Canvas 对象的使用	146
8.3 图形图像组件	164
8.4 使用 OpenGL	173
8.5 图形的动画制作	179
第九章 奇妙的多媒体世界	184
9.1 多媒体技术简介	184
9.2 C++ Builder 的多媒体控件 TMediaPlayer	185
9.3 使用多媒体控件编制简单的应用程序	187
9.4 功能强大的媒体播放器	191
第十章 数据库控制	200
10.1 C++ Builder 的数据库功能	200
10.2 数据库的链接	202
10.3 数据库的建立	204
10.4 利用数据库组件编程	207
第十一章 如何使用自定义的类	211
11.1 定义自己的类	211
11.2 界面设计	213
11.3 程序的实现	217
11.4 程序运行结果	218
第十二章 程序的发行	219
12.1 Express For C++ Builder 4 的使用	219
12.2 安装盘的运行结果	224
附录 虚拟键值表	226

第一章 令人耳目一新的 C++ Builder

何谓 C++ Builder?它又有何特点?它的界面和编程思想如何?不要着急,本章将一一对此进行解释,并给出安装 C++ Builder 的方法,以便读者一边实践一边学习。

本章主要介绍如下内容:

- (1) C++ Builder 4 的特点。
- (2) 安装 C++ Builder 的机器配置。
- (3) C++ Builder 4 的界面组成。
- (4) 可视编程的基本概念。
- (5) C++ Builder 4 编程实例。
- (6) 编译产生 C++ Builder 4 的可执行 EXE 文件。

1.1 C++ Builder 的迷人之处

自 Windows 问世以来,其漂亮的图形用户界面(GUI)、高效的内存管理、多进程和多线程的并发机制及丰富的多媒体服务,都深深地吸引了广大的用户。随着硬件功能的不断提高,Windows 已经成为软件发展的趋势。但是与 MS-DOS 相比,Windows 软件开发的确是一个复杂庞大的工程,因此又使许多程序员望而却步。

Visual Basic 的出现,曾令人们欣喜若狂,开发 Windows 应用程序开始变得简单方便。但好景不长,由于 Basic 自身的缺陷,又使许多工作难以实现。虽然 Borland C++ 及 Visual C++ 等可视化的 C 语言功能强大,可是其所依赖的 OWL 和 MFC 类库却非常复杂,人们熟悉起来相当不易。Delphi 这一快速程序开发工具(Rapid Application Development)的推出,以其高效、舒适、方便的开发环境迅速风靡全球,不过,要使用 Delphi,还要先学习 Pascal 语言的有关知识,这对一般的 C/C++ 程序员来说,简直是一种痛苦。

Borland 公司集数年开发 Borland C++ 和 Delphi 之经验,推出了 C++ Builder 这一划时代的 C++ 语言开发工具。它结合了 Borland C++ 和 Delphi 的优点,把可视化开发环境、开发工具和可视组件库加入到 C++ 语言中,实现了真正的面向对象的开发。

C++ Builder 是一个优秀的独立的开发系统,与其他工具相比,具有如下的特点:

(1) C++ Builder 是一个可视化的快速应用程序开发工具,具有高效、最优化的源代码编译器,可视化的快速应用程序开发环境和可扩展的数据库技术,适合各种类型的可视化程序开发。

(2) 因为它使用的程序语言是 C++, 所以是一个面向对象的语言,具有灵活、自由、简洁、高效的特点,由于 Windows 自身是用 C 语言编写,所以 C++ 可直接调用 Windows API,另外,还增强了异常状况的处理能力以及类的封装能力和特性。

(3) C++ Builder 具有完整的基于组件设计与可视化的开发环境,提供了所有可视组件对

象的完整代码，可以让程序设计人员很快产生程序原型及用户界面，实现代码的重用。

(4) 拥有独立，高效的编译技术，执行文件 EXE 或 DLL 都与 Windows 95 的操作环境无关，支持多语言开发代码，对于 Power Builder、Visual Basic、Borland C++、Visual C++ 和 Delphi 代码都可重新编译利用。

(5) 有强大的数据库支持能力，提供了存取 dBASE 或 FoxPro 数据库数据的能力。提供了 Client/Server 支持，可以轻松自如地开发客户/服务器程序。

(6) C++ Builder 4 为 Windows 98 增加了许多新特征，包括项目管理器、类浏览器、编辑器、实时库、窗口拖放功能和调试等诸多方面。

(7) C++ Builder 4 提供扩展的 ActiveX 支持及 VCL 增强功能。VCL 可以控制两位数年代的解释，提供了对 2000 年的支持。

(8) C++ Builder 4 提供与微软产品的兼容性，可以很容易地将微软 Visual C++ 5.0 的项目转换为 C++ Builder 的项目。

1.2 安装并运行 C++ Builder 4

本书是介绍 C++ Builder 4 版本，安装 C++ Builder 4 的机器配置必须符合下列最低要求：

- ① CPU 为 80486 以上的个人计算机；
- ② 16MB 以上的内存，最好大于等于 32MB 内存；
- ③ Windows 95、Windows 98、Windows NT3.5 或以上版本的操作系统；
- ④ 安装 C++ Builder 4 所需硬盘空间与安装方法有关。例如，完全安装需要 275MB 的空间。

我们使用的环境是奔腾机，内存 32MB，Windows 98 系统。安装很简单，只要运行安装光盘中的安装程序并按提示进行操作即可。下面给出安装步骤。

1.2.1 选择安装方式

选择安装方式的步骤如下：

(1) 选择安装光盘中的 Setup.exe 安装程序(可以在启动时自动运行)，进入如图 1.1 所示的 C++ Builder 4 安装界面。安装光盘包括 C++ Builder 4、InterBase 5.5 和 InstallShield Express 三个软件的安装程序。选择 C++ Builder 4 安装程序后，进入图 1.2 所示的界面，开始安装前的准备工作，并给出图 1.3 所示的画面。

(2) 单击图 1.3 的“Next”按钮，给出图 1.4 所示的 Password 对话框。

(3) 单击图 1.4 的“Next”按钮，给出图 1.5 所示的选择安装方式。

(4) 选择 Full 安装方式后，询问用户在安装 C++ Builder 4 时，是否需要同时安装 InterBase Client，如图 1.6 所示。



图 1.1 安装光盘的安装界面示意图



图 1.2 安装准备示意图

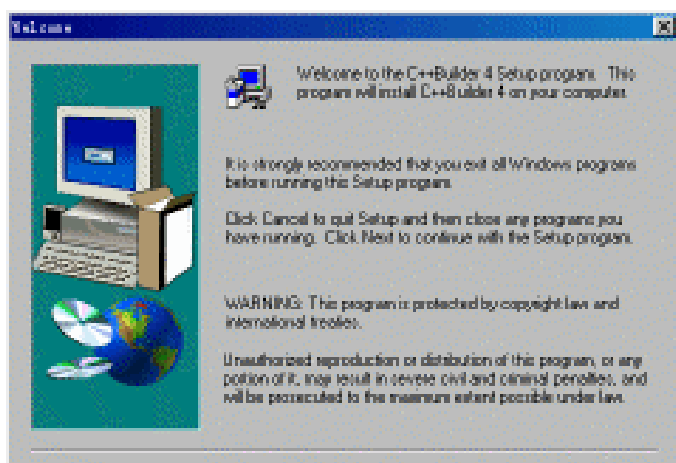


图 1.3 欢迎界面示意图

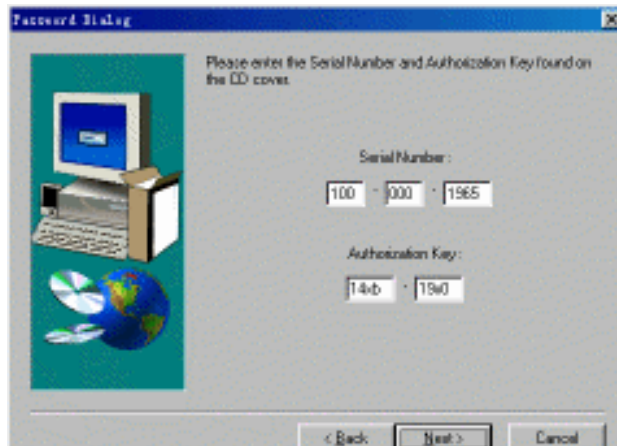


图 1.4 询问 Password 示意图

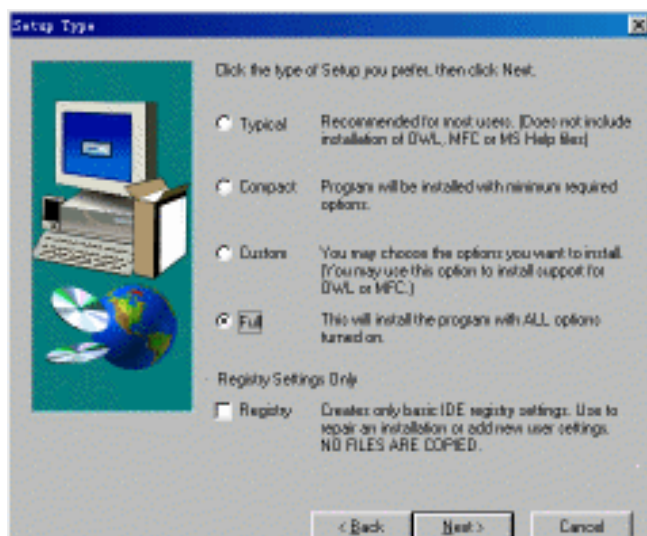


图 1.5 选择安装方式

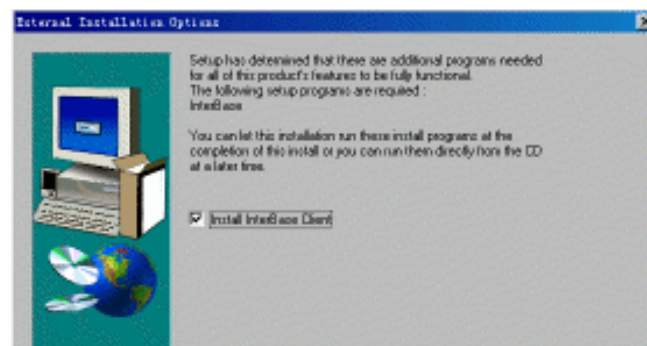


图 1.6 询问是否同时安装 InterBase Client

1.2.2 选择安装目录及程序组

选择安装目录及程序组的方法如下：

- (1) 单击图 1.6 所示对话框中的“Next”按钮，给出图 1.7 所示的图形界面。
- (2) 图 1.7 给出默认选择。如果需要改变安装目录，选择“Browse”按钮。
- (3) 选择安装目录之后，按“Next”按钮，继续设置程序组，给出图 1.8 所示的安装设置位置和安装界面。

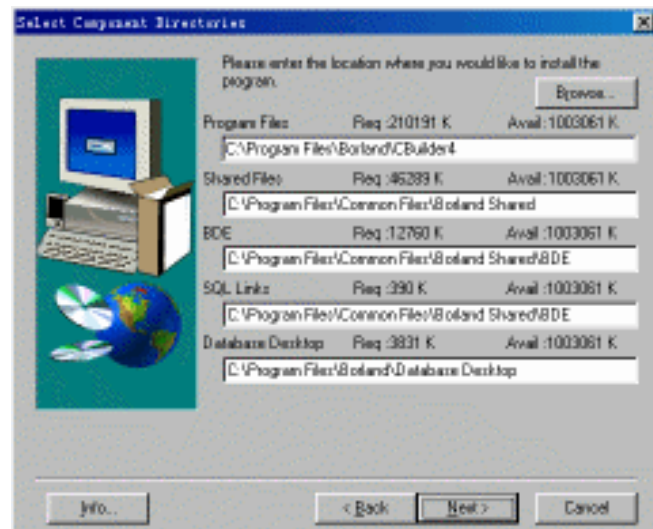


图 1.7 设置安装目录

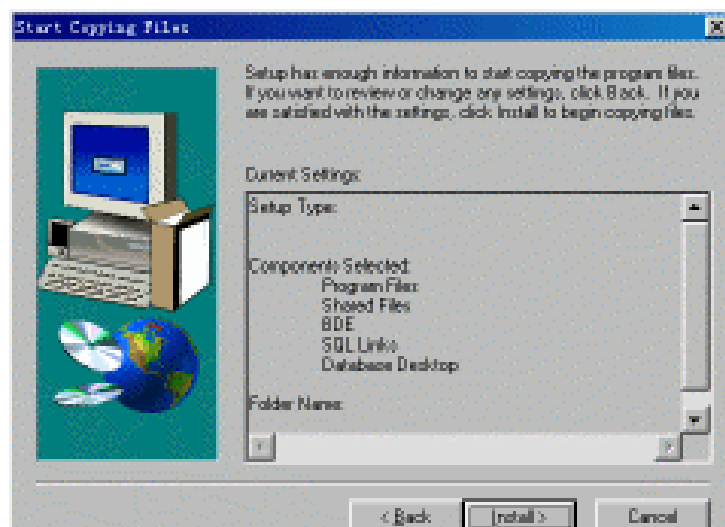


图 1.8 安装信息及安装界面

1.2.3 开始安装

下面按安装选项进行安装。步骤如下：

- (1) 如果用户又有了与图 1.8 所示的报告不同的方案,可以按“Back”按钮返回上述相应步骤进行修改。
- (2) 如果用户接受图 1.8 所示的安装方案,单击“Install”按钮,进入图 1.9 所示的安装界面进行安装。
- (3) 安装过程中给出它的主要特点,图 1.9 是典型安装画面示意图。
- (4) 完成安装并进入图 1.10 所示的 InterBase Client Setup 安装程序对话框。
- (5) 完成安装,给出图 1.11 所示的界面。
- (6) 如果要安装 InstallShield Express,返回图 1.1 所示的界面,选该安装项进行安装。图 1.12 是安装示意图。

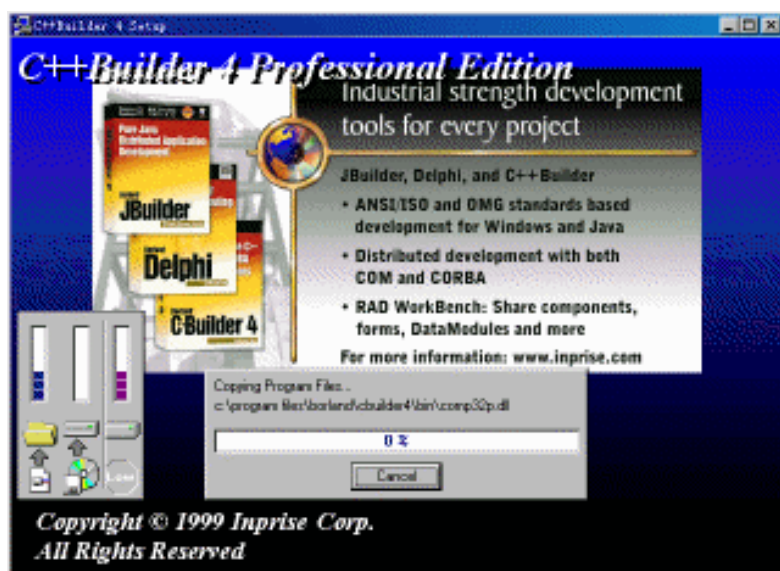


图 1.9 安装界面

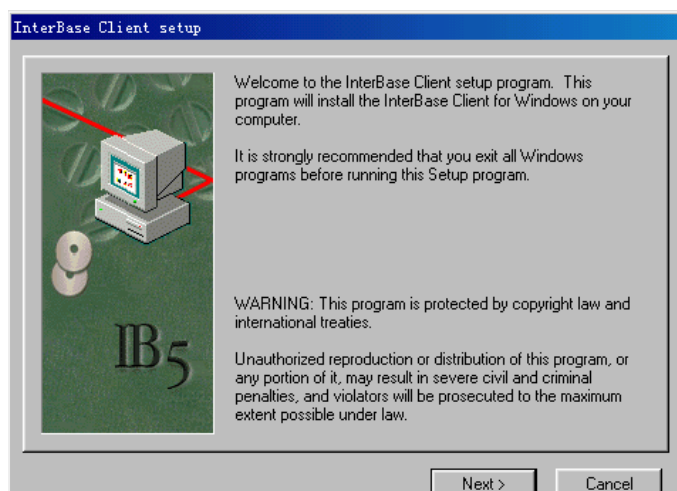


图 1.10 InterBase Client Setup 安装程序对话框

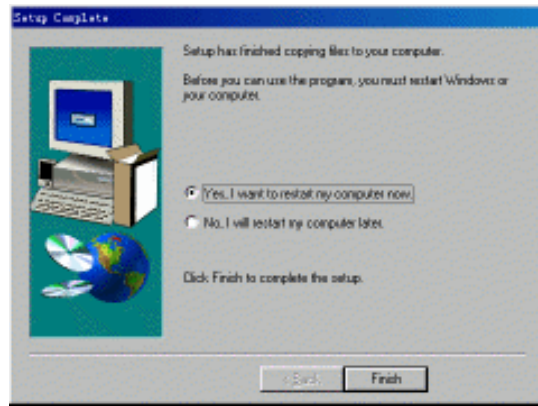


图 1.11 完成安装的界面图

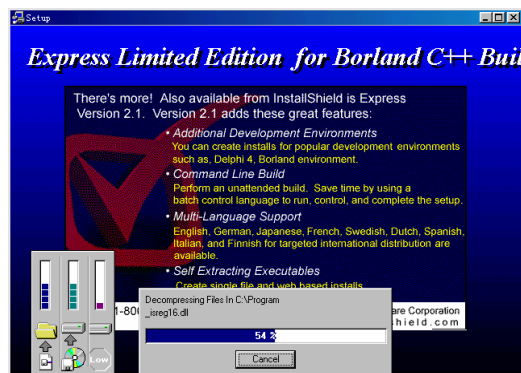


图 1.12 安装 InstallShield Express

运行 C++ Builder 的过程也很简单，先单击 Windows 的“开始”按钮，再在弹出菜单中选择“程序”菜单项，然后依次从级联菜单中选择 Borland C++ Builder 4 菜单项和 C++ Builder 4 菜单命令，如图 1.13 所示，即启动 C++ Builder，出现图 1.14 所示画面，然后给出图 1.15 所示的界面。

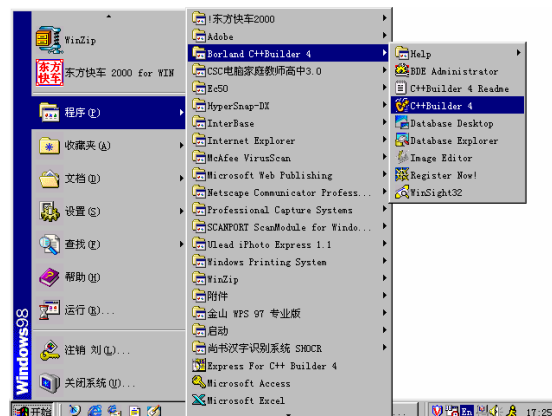


图 1.13 C++ 程序组示意图



图 1.14 启动 C++ Builder 画面

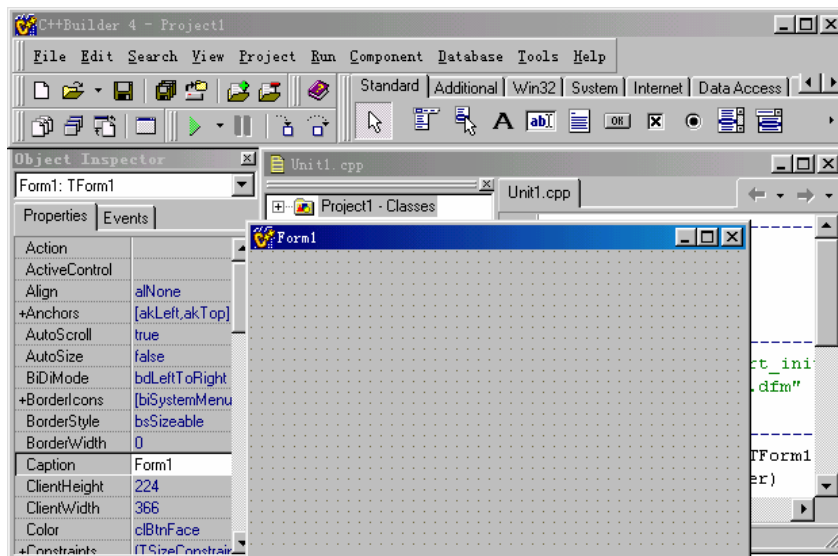


图 1.15 C++ Builder 4 的界面示意图

1.3 熟悉 C++Builder

初次进入 C++ Builder 4 时,其界面如图 1.15 所示。整个 C++ Builder 界面分以下四个部分:

1. 主窗口

如图 1.15 所示,标题为 C++ Builder 4 - Project1 的是主窗口,主窗口包括标题栏、菜单栏、工具栏和组件板。工具栏位于屏幕左端,菜单栏的下面。4 个工具栏上有执行各种动作的快捷按钮,这些按钮是菜单中某些功能的快捷方式。组件板位于工具栏右端,菜单栏的下面,包括 Standard、Additional、Win32、System、Internet、Data Access、Data Controls、QReport、Dialog、Win31、Samples 和 ActiveX 等选项卡。用组件板可以快速选择需要的组件到表格窗口中。

2. 对象查看器

标题为 Object Inspector 的窗口就是对象查看器。对象查看器有 Properties 和 Events 两个选项卡。Properties 可以设置和改变属性,Events 选项卡上定义了对对象接收到事件时要执行的操作,如图 1.15 左下方所示。

3. 窗体(表格)窗口

标题为 Form1 的是窗体窗口。因为它显示供对位用的点格,所以也称表格窗口,与 VB 及 Delphi 的表格窗口一样,都是用来开展设计工作的。

4. 代码编辑窗口

标题为 Unit1.cpp 的是代码编辑窗口,最初隐藏于表格窗口下,可单击“查看”栏的第三个按钮(Toggle Form/Unit)或按“F12”键来实现表格窗口和代码编辑窗口之间的切换。代码编辑窗口就是进行编程工作的窗口。

关于 C++ Builder 界面的各个组成部分以后还将详细介绍,下面一节将先简要说明有关的基本概念。

1.4 可视编程的基本概念

C++ Builder 是一种可视化快速应用程序开发工具,它基于“可视组件库”(VCL),同 Microsoft 的“微软基础类库”(MFC)和 Borland 公司 C++的“对象库”(OWL)相比,它是建立在组件(Components)、属性(Properties)和事件(Events)等概念之上,是一个面向对象的函数库,完全支持所有面向对象编程的概念标准,如继承、封装、多态性等,下面就介绍 VCL 的这些关键概念。

1. 窗体

窗体是设计时的窗口,是窗口的框架。可以把窗体视为一个容器,它里面盛装的材料就是构成窗口的各个部件。

2. 组件

组件是构成窗口的各个部件。C++ Builder 4 为用户提供了丰富的组件，使得用户可以像搭积木一样构成一个窗口，因而设计窗口变得很简单。从语言的角度上讲，组件是 C++ Builder 预定义的一种类。

3. 属性

顾名思义，属性就是组件的性质。属性说明了组件的功能及运行时的显示方式，例如表格窗体的 Caption 属性决定了在窗体标题栏中显示的标题。属性可以分为如下三类：

- (1) 设计属性：是在设计时就能显示和发挥作用的属性。
- (2) 运行属性：是运行时才得以发挥作用的属性。
- (3) 只读属性：是只能查看不能改变的属性。

从语言角度上讲，组件是类，而属性是类中的数据成员，这些数据成员有语言中已有类型，也有系统自定义的类型。

4. 事件

事件指的是组件接收到激励时要执行的动作。激励可以是外部激励，如鼠标操作等，也可以是内部激励，如组件某个属性的改变等。从语言角度上讲，组件是类，而事件则是类中的成员函数，只不过这种成员函数没有返回值，在 C++ Builder 的编程语言中被称作事件处理过程。因此，从一定意义上讲，属性和事件构成了组件。

5. 控件

控件是具有控制作用的组件。因此，所有的控件都是组件，而组件不一定是控件，在运行时可以看见的组件就是控件，不可见的组件为非控件。读者在后续章节中会逐步接触到控件和非控件的例子。

1.5 简单的 C++ Builder 编程实例

下面举两个简单的例子来说明如何应用以上概念进行 Builder 编程。由于 C++ Builder 已经帮助完成了所有的初始化及文件的组织和管理，所以程序员只要在需要的地方填入相应的代码即可。

【例 1.1】 在窗口中显示文字的例子。

如果要在窗口中显示“Welcome to C++ Builder !”，程序的编制步骤如下：

- (1) 用“File”菜单下的“New Application”命令开始一个新的应用程序。
- (2) 设置窗体窗口属性。这可以在对象查看器中单击 Caption 属性，把缺省值 Form1 改为 Welcome。
- (3) 添加组件。从组件页 Standard 中选取一个 Label 控件加到表格窗体中。
- (4) 设置组件属性。单击 Object Inspector 标题栏以激活对象查看器，在对象查看器中单击 Label1 的 Caption 属性右侧，键入新的属性值“Welcome to C++ Builder!”。并把 Font 属性设为 MS-Sans-Serif，粗斜体，18 号字体。窗体的设置如图 1.16 所示。

然后单击标准工具栏的第二个按钮将程序存盘，屏幕上首先出现的是“Save Unit1 As”窗口，缺省单元文件名为 Unit1。

在“文件名”文本框中键入文件名 `Welcome` 以保存单元文件。而后屏幕上又出现一个“Save Project1 As”窗口，缺省项目文件名为 `Project1`，如图 1.17 所示。现在是以缺省文件名 `Project1` 保存项目文件。

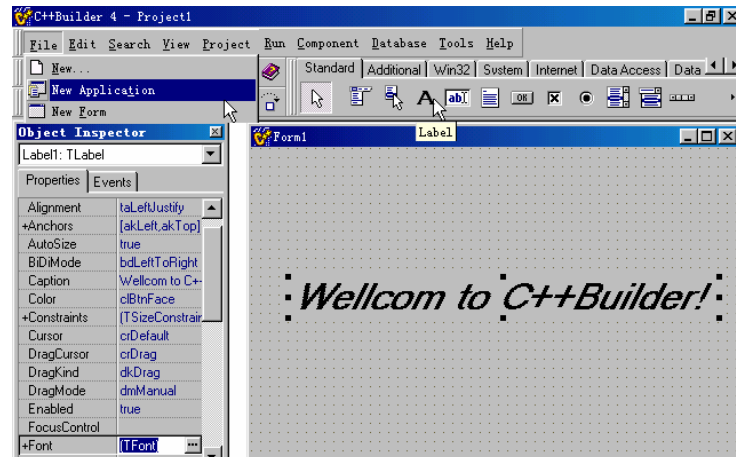


图 1.16 窗体的设置

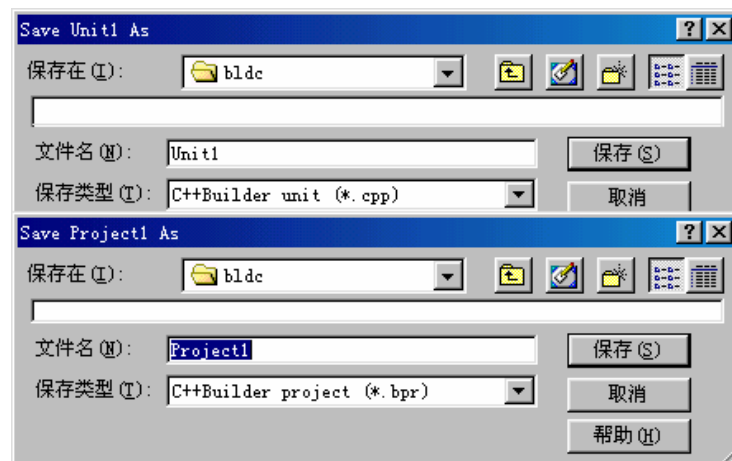


图 1.17 Save As 窗口

注意：存储一个应用文件要分别存储单元文件和项目文件两个文件，而且这两个文件名不能相同。文件保存好以后，使用 `Run` 菜单中的 `Run` 选项或按 `F9` 键来编译运行该项目文件，得到运行结果如图 1.18 所示。

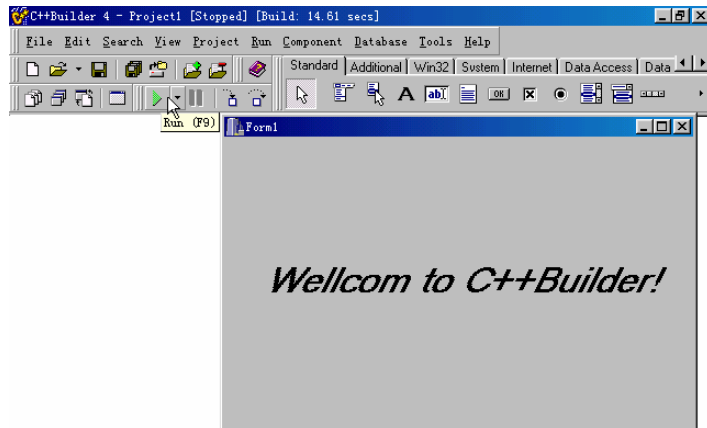


图 1.18 程序运行结果

【例 1.2】 在窗口中使用鼠标控制按钮显示不同文字的例子。

使用上一个例子的方法为窗体设置文本框和命令按钮各一个。如图 1.19 所示，选中对象 Edit1 并且修改属性 Text 的内容为“表演程序”，作为程序运行后的初始显示内容。显示字体大小等参数通过单击图中的 TFont 的按钮，弹出图 1.20 所示的“字体”对话框设置。然后使用同样的方法把命令按钮上的名称改为“抬起/放下”，如图 1.21 所示。

如图 1.22 所示，选中窗体中的命令按钮，然后单击“Object Inspector”对话框中的“Events”选项卡，分别双击 OnMouseDown 和 OnMouseUp 右边的空栏，栏中自动出现该事件的名称，同时在编程窗口 Uint1.cpp 中出现该事件函数的编程位置。如图中所示，分别在这两个函数体中输入如下两句：

Edit1->Text="鼠标放下";

Edit1->Text="鼠标抬起";

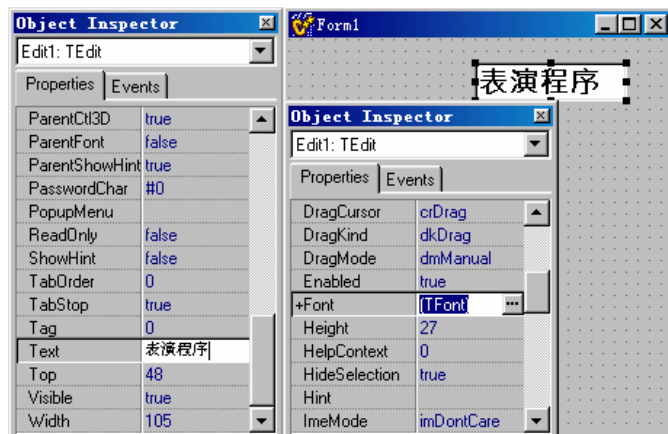


图 1.19 为窗体设置文本框和命令按钮