

C++ Builder C++ Builder C++ Builder C++ Builder C++ Builder C++ Builder C++ Builder C++ Builder C++ Builder C++ Builder

李伟生 编著

# 深入 C++ Builder 编程



- ▶ 熟悉Borland C++ Builder 的集成开发环境
- ▶ 了解VCL 的体系结构
- ▶ 通过Data Access、Data Controls 和ADO组件开发数据库应用程序
- ▶ 使用Internet 和FastNet 组件编写网络应用程序



西安电子科技大学出版社

<http://www.xduph.com>

296

TP312C  
L354

# 深入 C++ Builder 编程

李伟生 编著



A0988647

西安电子科技大学出版社

2 0 0 1

## 内 容 简 介

Borland C++ Builder 5 是 Inprise 公司推出的新一代面向对象、可视化的快速应用程序开发环境。本书先对 C++ Builder 5 的使用、开发的基本步骤及技术要点作了总结性的介绍, 然后对 VCL 的结构层次以及 VCL 中常用的组件作了全面系统的介绍。由于数据库和网络应用程序是当前计算机软件业应用开发的热点, 因此, 本书主要对数据库和网络等方面的应用作了深入的介绍。

本书适合于有一定编程经验的初、中级读者, 尤其适合于通过 C++ Builder 5 来开发数据库和网络应用程序的读者朋友。

### 图书在版编目 (CIP) 数据

深入 C++ Builder 编程 / 李伟生编著. —西安: 西安电子科技大学出版社, 2001.9

ISBN 7-5606-1062-5

I. 深… II. 李… III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2001) 第 063103 号

策 划 臧延新

责任编辑 戚文艳

出版发行 西安电子科技大学出版社 (西安市太白南路 2 号)

电 话 (029)8227828 邮 编 710071

http://www.xduph.com E-mail: xdupfxb@pub.xaonline.com

经 销 新华书店

印 刷 西安文化彩印厂

版 次 2001 年 9 月第 1 版 2001 年 9 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 27.375

字 数 653 千字

印 数 1~4000 册

定 价 36.00 元

ISBN 7-5606-1062-5 / TP · 0529

\*\*\* 如有印装问题可调换 \*\*\*

本书封面贴有西安电子科技大学出版社的激光防伪标志, 无标志者不得销售。

# 前 言

---

Borland C++ Builder 5 是 Inprise 公司推出的新一代面向对象、可视化的快速应用程序开发环境。它基于面向对象的 C++ 语言, 使用了包括 Windows GUI 在内的许多先进特性和设计思想。

使用 C++ Builder 5 可以开发通用的或基于客户/服务器模式的 32 位 Windows 应用程序。C++ Builder5 不仅继承了传统 C++ 应用程序开发工具高效和低层硬件控制能力强的特点, 而且还利用了 VCL 中所提供的组件, 使得该工具真正具有应用方便和真正可视化的特点。一直以来, C++ Builder 的各个版本都以其简单易用和快速高效的特点得到了众多的 C++ 程序员的青睐。

2000 年 1 月 26 日, Inprise 公司发布了 C++ Builder 5 这一新款产品。该产品一面世就引起了业界的强烈关注。与以往的 C++ Builder 版本相比, C++ Builder 5 主要作了以下改进:

- 引入了 ADO 和 InterBase 等新的组件技术, 前者可以直接与 ADO 数据库进行交互, 后者可以通过执行 InterBase API 函数直接访问 InterBase 数据库, 使得数据接口的应用程序可以不需要 BDE 的支持;

- 支持 MIDAS 应用程序的结构, 也支持远程数据模块, 使得新的 Internet 组件在浏览器与来自 MIDAS 应用程序服务器的数据进行交互的地方可以创建 Web 应用程序;

- CORBA 的升级, 使得它能与 VisiBroker for C++ ORB 4.0 版一起使用;

- 重新设计了编译和连接器, 使得应用程序的性能有了很大的提高;

- 增加了一些新的调试特性;

- VCL 中添加了许多结合新技术和响应开发者要求的新对象、属性和事件;

- 优化了集成开发环境, 允许用户定制并保存自己的环境设置。

为了推广 C++ Builder 5 的使用, 使更多的用户了解和应用 C++ Builder 5, 我们特意编写了此书。

全书的内容共分为 12 章。下面对各章进行简单的说明。

第一章和第二章是学习 C++ Builder 5 的基础, 分别介绍了 C++ Builder 5 的集成开发环境和面向对象编程的基础知识。

第三章和第四章介绍了 C++ Builder 5 中 VCL 的体系结构和 VCL 中常用组件的使用方法。

第五章主要介绍了 C++ Builder 5 的基本绘图组件的用法。

第六章主要讲解了 C++ Builder 5 的程序调试与异常处理。

第七章至第十章主要介绍了 C++ Builder 5 在数据库方面的应用, 内容涉及 C++ Builder 5 的数据库开发环境、常用数据库应用程序组件、快速报表组件和 ADO 组件等方面。

第十一章至第十二章主要介绍了 C++ Builder 5 在网络方面的应用, 内容涉及 WinSock 开发、WWW 开发和 FastNet 选项卡上常用网络组件的用法等方面。

本书适合于有一定编程经验的初、中级读者，尤其适合于通过 C++ Builder 5 来开发数据库和网络应用程序的读者朋友。

本书在内容的编排上力求深入浅出，除了对应用开发的常用组件的属性、方法和事件作了详细的说明外，还给出了大量的程序示例供读者参考。书中附带的程序示例都经过编者的挑选和测试，一些例子还给出了详细的分析说明。尽管如此，由于时间仓促，加之编者的水平有限，错误与疏漏之处在所难免，欢迎广大读者朋友批评指正。

编者  
2001年4月

# 第一章

## C++ Builder 5 集成开发环境综述

Borland C++ Builder 5 是 Windows 32 位操作系统平台上的应用程序集成开发环境 (IDE, Integrated Development Environment)。在 Borland C++ Builder 5 集成开发环境中包含了编辑、编译、连接、调试和跟踪等工具。本章将先对 C++ Builder 作一个总体性的介绍, 然后讨论组成 Borland C++ Builder 5 集成开发环境的菜单和工具。这是学习 C++ Builder 的基础。

### 1.1 Borland C++ Builder 概述

Borland C++ Builder 是一种快速应用程序开发 (RAD, Rapid Application Development) 工具, 它基于面向对象的 C++ 语言。C++ Builder 具有许多重要的特色, 它们是 C++ Builder 产生并飞速发展的原因。下面介绍它的一些主要特色。

#### 1.1.1 C++ Builder 产生的背景

图形用户界面 (GUI) 使微型计算机工业发生了革命性的变化。随着 32 位 Windows 操作系统的出现, DOS 用户长期使用的 C:\> 符号被带有图标和使用菜单及鼠标的程序的桌面所代替。

图形用户界面出现的重要性并不在于其外在形式, 而在于为图形用户界面开发的应用程序通常都有一个一致的用户界面, 也就是说, 用户可以将更多的时间用于掌握应用程序本身的实现, 不必去关心使用什么命令完成所需的操作, 而只需使用菜单和对话框就可以完成这些操作。

为了开发 Windows 各种版本的应用程序, 需要一种有效的可开发的基于 GUI 的应用程序的工具。很长时间内, 这种可开发 Windows 应用程序的工具非常少。在 Windows 操作系统刚刚推出之时, Borland C++ 3.1 是当时唯一可以在 Windows 环境中开发 Windows 应用程序的工具。这样, 开发 Windows 应用程序要比开发 DOS 应用程序难得多, 程序员必须了解诸如鼠标正在干什么, 用户在菜单的什么地方, 用户在哪一区域单击或双击了鼠标等情况。这样, 即使是一个专业的程序员, 开发一个简单的 Windows 应用程序也需要编写数百行的代码。

在这样的形势下, 社会化生产的迅速发展和日益激烈的商业竞争推动了优秀软件开发工具的发展, 再加之近年来计算机技术日新月异的飞速发展, 使得计算机的容量和计算速

度都得以大大提高，为优秀可视化编程工具的发展提供了相应的经济基础和硬件支持。随着 Microsoft 连续推出 Visual C++ 和 Visual Basic 之后，在可视化发展工具的优胜劣汰下，Borland 渐渐失去了以前的优势地位。

Borland 为了扭转不利的形势，在经过了几年的尝试和努力下，于 1993 年终于推出了一个令全世界程序员都为之振奋的可视化开发工具，这就是 Delphi。在 Delphi 的开发环境中，通过 VCL (Visual Component Library, 可视化组件库) 提供了大量预制的可重用组件，使用这些组件可以非常方便地开发 Windows 应用程序，Delphi 以快速应用程序开发工具 (RAD) 为特征，推出之后立即成为程序员所关注的焦点，成为挽救当时 Borland 公司的最大功臣。

在推出 Delphi 之后，由于 C/C++ 在计算机软件编程方面具有巨大影响，因此，迫切需要一种基于 C/C++ 的快速应用程序开发工具，鉴于此，Borland 公司乘胜追击，于 1997 年推出了基于 C/C++ 的 VCL 编译器 C++ Builder 1.0，1998 年推出了 C++ Builder 3，1999 年 4 月推出了 C++ Builder 4，2000 年 1 月又推出了本书将要介绍的 C++ Builder 5。

### 1.1.2 C++ Builder 的特点

C++ Builder 是 Borland (Inprise) 公司新一代面向对象、可视化的快速应用程序开发工具，它可以运行在 Windows 95/98 或 Windows NT 操作系统上。使用 C++ Builder 可以开发通用的或基于客户/服务器模式的 32 位 Windows 应用程序。C++ Builder 不仅继承了传统 C++ 应用程序开发工具高效和低层硬件控制能力强的特点，同时还利用了 VCL 中所提供的组件，使得该工具真正具有应用方便和真正可视化的特点。

另一方面，C++ Builder 也以其独特的优点吸引了众多的 C++ 程序员。为了满足广大程序员的开发需求，从而促使了 C++ Builder 本身的发展，使其逐渐成为一种集 C++ 语言的高效性和快速应用程序开发工具优点完美结合的产物。

和传统的 C/C++ 开发工具相比，C++ Builder 具有以下几个方面的优点：

(1) C++ Builder 代码的执行效率高。

C++ Builder 是基于 C++ 语言的，它具有高效的编译、连接和运行速度。同时，C++ Builder 具有双编译器引擎，使得它不仅可以编译 C/C++ 程序，而且还可以编译 Object Pascal 语言程序。它是目前众多 C/C++ 编译器中对 C++ 支持最齐全的编译器之一。

(2) C++ Builder 是优秀的可视化应用开发工具。

C++ Builder 提供了完全可视的程序开发工具，在 C++ Builder 中，应用程序设计和实现的基本单元称为组件。组件是一个可视化的软件单元，可以直接放置到开发环境中去，由多个这样的组件协同完成应用程序的各项功能。在 C++ Builder 的集成开发环境中提供了 200 多个各种功能的组件，这些组件覆盖了应用程序开发的各个方面。通过组件进行应用程序的开发具有快速和可视化的特点，能够缩短开发的时间，提高程序开发人员的工作效率。

(3) C++ Builder 具有强大的网络编程功能。

C++ Builder 提供了众多的 Internet 应用开发组件，根据组件的功能，可将组件分为以下几类：解决分布式网络应用程序进程间相互通信的组件，比如 ClientSocket 和 ServerSocket 等；电子邮件服务组件，比如 NMSMTP 和 NMPOP 等；Web 服务器应用程序组件，比如

TPageProducer 和 TQueryTableProducer 组件。这些组件几乎涵盖了 Internet 应用的全部功能，利用它们，编程人员可以方便地建立自己的 Internet 应用程序。

(4) C++ Builder 具有强大的数据库应用开发功能。

Borland 数据库引擎 BDE (Borland Database Engine) 是 Borland C++ Builder 数据库应用程序开发环境的核心和纽带。BDE 是基于 Windows 操作系统的 32 位数据库引擎，它提供数据库别名机制，为客户/服务器应用程序提供可统一的数据库接口 API (Application Program Interface, 应用程序公共接口)，这些数据库包括 dBase, Paradox, Text, FoxPro, Access, InterBase, Oracle, Sybase 和 Microsoft SQL Server, 它还支持 Microsoft 的 ODBC (Open Database Connectivity) 数据源。这样，基于 BDE 的数据库应用程序对数据库的访问方法和数据库的类型完全无关，应用程序开发人员可以直接修改数据库的数据结构或数据库的访问机制，而不需要修改数据库应用程序的任何代码。

总之，采用组件的思想进行 Windows 应用程序的开发彻底改变了传统 Windows 应用程序的编程方式。C++ Builder 为应用程序的开发在幕后作了大量的工作，应用程序开发人员无需深入了解操作系统的运行机制就可以进行编程。在编写应用程序时，编程人员只需把实现各种功能的组件放置在一起，通过比较少的手动编程就可以实现应用程序的整体功能。无论是初学者，还是具有一定编程经验的 Windows 程序员都可以迅速掌握 C++ Builder。

## 1.2 启动 C++ Builder 5 集成开发环境

启动 Borland C++ Builder 5 集成开发环境的方法有多种。在 C++ Builder 集成开发环境安装完成后，安装程序会在【开始】、【程序】菜单中添加【Borland C++ Builder 5】菜单及其菜单项，如图 1.1 所示。通过单击菜单项中的【C++ Builder 5】命令，即可启动 Borland C++ Builder 5 集成开发环境。另外，通过在资源管理器中找到 Borland C++ Builder 5 文件系统的工作目录，双击 Bcb.exe 文件名（默认安装模式下，该文件位于\Program Files \Borland \CBuilder5 \Bin 目录下），也可以启动该环境。为了启动方便，也可以将 Bcb.exe 文件用鼠标拖动到桌面上，这样就创建了一个启动 C++ Builder 5 的快捷方式，以后启动就可以只双击该快捷方式。



图 1.1 启动 Borland C++ Builder 5 集成开发环境



## 1.3 C++ Builder 5 集成开发环境

Borland C++ Builder 5 启动后，将显示出 C++ Builder 5 集成开发环境的主窗口。其集成开发环境主要包括以下几个部分：

- 菜单栏
- 工具栏
- 组件面板 (Component Palette)
- 窗体 (Form)
- 对象检视器 (Object Inspector)
- 代码编辑器 (Code Editor)

这一节将简单介绍 Borland C++ Builder 5 集成开发环境中的菜单栏、工具栏和组件面板，而作为编程工具的其他几个部分将在下一节进行介绍。

### 1.3.1 菜单栏

Borland C++ Builder 5 集成开发环境中的菜单与其它 Windows 应用程序中的菜单相似，它们为应用程序开发人员提供了编辑、开发、调试和管理 Borland C++ Builder 5 应用程序工程的各种菜单项和工具。菜单栏位于 Borland C++ Builder 5 集成开发环境的顶部，如图 1.2 所示。



图 1.2 Borland C++ Builder 5 集成开发环境菜单栏

#### 1. File 菜单

File 菜单中的菜单项主要用于对文件的常用操作，包括新建、打开、保存、打印和关闭 Borland C++ Builder 5 工程项目中各种文件的命令。各个菜单项的快捷键和功能如表 1.1 所示。

表 1.1 File 菜单的菜单项功能列表

菜单项	快捷键	功能
New		打开 New Items 对话框，通过访问模板和对象存储器中保存的对象来创建新的窗体、对话框或工程项目等
New Application		创建新的 C++ Builder 应用程序工程项目
New Form		为当前工程项目创建一个新的窗体和一个单元文件
New Frame		创建一个新的框架，并添加到当前工程项目中
Open		打开 Open 对话框，用于选定打开的窗体或工程文件
Open Project	Ctrl+F11	打开工程项目文件
Reopen		重新打开一个最近打开过的工程项目文件或模块
Save	Ctrl+S	保存当前编辑过的文件
Save As		将当前文件重命名或更改路径后存盘
Save Project As		将当前工程重命名或更改路径后存盘
Save All	Shift+Ctrl+S	保存工程中所有打开的文件

续表

菜单项	快捷键	功能
Close		关闭当前窗口
Close All		关闭所有打开的窗口
Include Unit Hdr	Alt+F11	使用#include 语句在代码编辑器中将一个已存在的 Unit 文件名称添加到当前 Unit 文件中
Print		打印当前窗口文件
Exit		退出

## 2. Edit 菜单

Edit 菜单中的菜单项主要包含了撤消、重做、复制、粘贴等编辑时的常用命令，它还包含了调节对象的对齐方式和尺寸等命令。各菜单项的快捷键和功能如表 1.2 所示。

表 1.2 Edit 菜单的菜单项功能列表

菜单项	快捷键	功能
Undo/Undelete	Ctrl+Z	在代码编辑器中使用 Undo 命令撤消最近的操作，在窗体中使用 Undelete 命令恢复前一次的删除操作
Redo	Shift+Ctrl+Z	撤消最近使用的 Undo 操作
Cut	Ctrl+X	剪切掉选中的对象，并放置到系统剪切板中
Copy	Ctrl+C	复制选中的对象，并放置到系统剪切板中
Paste	Ctrl+V	粘贴系统剪切板中的对象
Delete	Ctrl+Del	删除选中的对象
Select All	Ctrl+A	选中当前窗口中的所有对象
Align to Grid		将窗体中选定的对象与最近的网格点对齐
Bring to Front		将窗体中选定的对象放置在其它对象之前
Send to Back		将窗体中选定的对象放置在其它对象之后
Align		打开 Alignment 对话框，调整窗体中所选定的可视化组件的对齐方式
Size		打开 Size 对话框，设置窗体中选定组件的大小
Scale		打开 Scale 对话框，设置窗体中所有组件的缩放比例
Tab Order		打开 Edit Tab Order 对话框，设置窗体中组件的 Tab 顺序
Creation Order		打开 Creation Order 对话框，设置窗体中非可视化组件的创建顺序
Flip Children		对窗体中的组件进行水平方向的翻转
Lock Controls		锁定窗体中各个组件的位置，不允许移动或调整组件的大小
CORBA Refresh		更新存在的 CORBA 实现类
Use CORBA Object		使用 CORBA 对象

## 3. Search 菜单

Search 菜单中的菜单项主要包含了在 C++ Builder 集成开发环境的代码编辑器中的查找、替换和定位等功能。Search 菜单中各菜单项的快捷键和功能如表 1.3 所示。

表 1.3 Search 菜单的菜单项功能列表

菜单项	快捷键	功能
Find	Ctrl+F	打开 Find Text 对话框，在当前文本编辑器中寻找指定的字符串
Find in Files		在多个文件中查找指定的字符串
Replace	Ctrl+R	打开 Replace Text 对话框，在文本编辑器中寻找指定的字符串，并用新的字符串替换
Search Again	F3	重复执行最后一次 Find 或 Replace 命令
Incremental Search	Ctrl+E	在文本编辑器中进行增量寻找操作
Go to Line Number	Alt+G	打开 Go to Line Number 对话框，将光标移动到指定的行号
Go to Address		打开 Enter Address to Position to 对话框，设置程序停止运行的地址

#### 4. View 菜单

View 菜单中的菜单项主要用于打开或关闭 C++ Builder 集成开发环境的各个工具软件，比如工具管理器、对象检视器、调试窗口、断点设置和定制工具栏等。View 菜单中各菜单项的快捷键和功能如表 1.4 所示。

表 1.4 View 菜单的菜单项功能列表

菜单项	快捷键	功能
Project Manager	Ctrl+Alt+F11	打开工程管理器
Translation Manager		打开语言管理器
Object Inspector	F11	打开对象检视器
To-Do List		打开 To-Do List 对话框，用于查看完成当前工程项目的任务列表
Alignment Palette		打开 Align 工具栏，调整组件布局
Class Explorer		打开代码编辑器（Class Explorer 位于编辑器的左边）
Component List		打开 Components 对话框，显示 C++ Builder 5 集成开发环境下所有的组件列表
Windows List	Alt+O	打开 Windows List 对话框，显示当前工程打开的所有窗口
Debug Windows		打开集成调试环境中的各个窗口
Desktops		保存窗口布局和在不同的窗口布局之间切换
Toggle Form/Unit	F12	在窗体和代码编辑器之间进行切换显示
Units	Ctrl+F12	打开 View Unit 对话框，显示选定的单元文件
Forms	Shift+F12	打开 View Form 对话框，显示选定的窗体
Type Library		打开类型库编辑器
New Edit Window		显示新的代码编辑窗口
Toolbar		关闭、显示或定制工具栏

#### 5. Project 菜单

Project 菜单中的菜单项主要用于实现管理工程文件、编译、连接和创建应用程序等功能。Project 菜单中各菜单项的快捷键和功能如表 1.5 所示。

表 1.5 Project 菜单的菜单项功能列表

菜单项	快捷键	功能
Add to Project	Shift+F11	显示 Add to Project 对话框, 在当前工程中添加一个已创建的单元文件
Remove from Project		显示 Remove from Project 对话框, 从当前工程中删除指定的文件
Import Type Library		显示 Import Type Library 对话框, 在当前工程项目中导入选定的类型库
Add to Repository		显示 Add to Repository 对话框, 将当前工程添加到对象存储器中
View Source		在代码编辑器中显示当前工程文件
Languages		改变当前工程的语言设置
Edit Option Source		在代码编辑器中显示当前工程的 XML 格式文件 (*.bpr)
Export Makefile		导出连接文件
Add New Project		显示 New Items 对话框, 将对象存储器中选定的对象或工程添加到当前工程项目组中
Add Existing Project		显示 Open Project 对话框, 将选定的工程项目添加到当前工程项目组中
Compile Unit	Alt+F9	编译当前单元文件
Make Project	Ctrl+F9	编译和连接当前工程项目中发生了变化的所有文件, 并生成相应的目标文件
Build Project		无论文件是否发生了变化, 对工程项目中的所有文件都进行编译和连接, 并生成目标文件
Information for		打开 Information 对话框, 查看当前工程的编译信息和编译状态
Make All Projects		编译和连接当前工程项目组中发生了变化的所有工程, 并生成相应的目标文件
Build All Projects		无论当前工程项目组中的工程是否发生了变化, 对所有工程都重新进行编译和连接, 并生成相应的目标文件
Web Deployment Options		配置 Web 应用程序开发环境中的 ActiveX 控件或 ActiveForm 选项
Web Deploy		对 ActiveX 控件或 ActiveForm 进行配置
Options	Shift+Ctrl+F11	打开 Project Options 对话框, 对工程的窗体、编译、连接以及管理工程的各个选项进行配置

## 6. Run 菜单

Run 菜单中的菜单项主要包括了 C++ Builder 调试环境中的命令, 比如运行、跟踪、查看和添加断点等。Run 菜单中各菜单项的快捷键和功能如表 1.6 所示。

表 1.6 Run 菜单的菜单项功能列表

菜单项	快捷键	功能
Run	F9	编译和执行应用程序
Attach to Process		显示 Attach to Process 对话框, 对系统当前运行的应用程序进程进行监视
Parameters		弹出 Run Parameters 对话框, 对应用程序运行时的命令行信息进行设置
Register ActiveX Server		把开发的 ActiveX 控件注册到系统中

续表

菜单项	快捷键	功能
Unregister ActiveX Server		从系统中取消已注册的 ActiveX 控件
Step Over	F8	在调试环境中单步执行, 在同一单元文件中遇到函数时跳过该函数
Trace Into	F7	在调试环境中单步执行, 遇到函数时跟踪到该函数内部
Trace to Next Source Line	Shift+F7	在调试环境中单步执行到下一程序行
Run to Cursor	F4	在调试环境中运行到代码编辑器中光标所在行
Run Until Return	Shift+F8	在调试环境中遇到函数调用时运行到函数返回
Show Execution Point		在编辑窗口中定位应用程序执行到所在行的光标
Program Pause		暂停应用程序的执行
Program Reset	Ctrl+F2	终止应用程序的执行, 退出调试环境
Inspect		打开 Inspect 对话框, 在调试环境中检查变量和表达式的值
Evaluate/Modify	Ctrl+F7	打开 Evaluate/Modify 对话框, 在调试环境中计算或修改表达式的值
Add Watch	Ctrl+F5	打开 Watch Properties 对话框, 添加表达式属性后监视表达式的值
Add Breakpoint		显示和添加新的断点

### 7. Component 菜单

Component 菜单中的菜单项主要包括了创建新组件、安装组件等命令。Component 菜单中各菜单项的功能如表 1.7 所示。

表 1.7 Component 菜单的菜单项功能列表

菜单项	功能
New Component	打开 New Component 对话框, 创建新的组件
Install Component	打开 Install Component 对话框, 安装新的组件
Import ActiveX Control	打开 Import ActiveX 对话框, 显示系统已经注册的 ActiveX 组件, 并可把选中的组件添加到当前工程中
Create Component Template	显示 Create Component Information 对话框, 创建组件模板
Install Packages	打开 Project Options 对话框, 指定应用程序设计阶段安装在 IDE 中的包文件和应用程序运行时工程项目需要的包文件
Configure Palette	打开 Palette Properties 对话框, 对组件栏进行配置

### 8. Database 菜单

Database 菜单中的菜单项主要包括了创建新组件、安装组件等命令。Database 菜单中各菜单项的功能如表 1.8 所示。

表 1.8 Component 菜单的菜单项功能列表

菜单项	功能
Explorer	打开数据库辅助工具软件 SQL Explorer, 用于创建、编辑和浏览 BDE 别名数据库中的数据
SQL Monitor	打开数据库辅助工具软件 SQL Monitor, 查看通过 SQL 连接到远程数据库或通过 ODBC 接口连接到一个 ODBC 数据源的语句调用
Form Wizard	打开 Database Form Wizard 对话框, 创建数据库窗体

## 9. Tools 菜单

Tools 菜单中的菜单项主要包括了创建新组件、安装组件等命令。Tools 菜单中各菜单项的功能如表 1.9 所示。

表 1.9 Tools 菜单的菜单项功能列表

菜单项	功能
Environment Options	打开 Environment Options 对话框，配置 C++ Builder 集成开发环境
Editor Options	打开 Editor Properties 对话框，配置 C++ Builder 集成编辑环境
Debugger Options	打开 Debugger Options 对话框，配置 C++ Builder 集成调试环境
Translation Tools Options	打开 Translation Tools Options 对话框，配置交互式语言环境 (ITE)
Repository	打开 Object Repository 对话框，配置对象存储器
Translation Repository	打开 Translation Repository 工具，配置语言存储器
VisiBroker SmartAgent	打开或停止系统中 Smart Agent
IDL Repository	显示 IDL Repository 对话框，在工程项目中注册 IDL 文件
Configure Tools	打开 Tool Options 对话框，对本菜单所包括的几个工具软件进行配置
Database Desktop	运行数据库辅助工具软件 Database Desktop
Package Collection Editor	运行 Package Collection Editor 工具软件
Visual C++ Project Conversion Utility	打开 VC++ Conversion Wizard 对话框，将 VC++ 工程文件转化为 C++ Builder 工程文件
CodeGuard Configuration	运行 CodeGuard Configuration 工具软件，对使用了 CodeGuard 的工程项目的调试环境进行配置
Image Editor	运行 Image Editor 工具软件，编辑工程项目中使用的图像

## 10. Help 菜单

Help 菜单中的菜单项主要用于查看 Borland C++ Builder 5 中提供的联机帮助信息。Help 菜单中各菜单项的功能如表 1.10 所示。

表 1.10 Help 菜单的菜单项功能列表

菜单项	功能
C++Builder Help	打开 Borland C++Builder Help 对话框，显示帮助信息
C++Builder Tools	打开 Borland C++Builder Productivity Help 对话框，显示 C++ Builder 中工具软件的帮助信息
Windows SDK	打开 Win32 Developer's Reference 对话框，供应用程序开发人员参考
Borland Home Page	浏览 Borland 的主页信息
Borland Community Page	浏览 Borland 的公共交流信息
C++Builder Home Page	浏览 Borland 关于 C++ Builder 的主页信息
C++Builder Developer Support	浏览 Borland C++ Builder 对开发人员提供的技术支持信息
C++Builder Direct	打开 C++ Builder Direct 对话框，进行关于 C++ Builder 的一些软件信息的下载
Customize	打开 OpenHelp 工具，定制 C++ Builder 帮助系统
About	显示 C++ Builder 的版权和版本信息

### 1.3.2 工具栏

Borland C++ Builder 5 集成开发环境中的工具栏按钮用于对菜单中的一些命令进行快速访问。Borland C++ Builder 5 集成开发环境默认的工具栏按钮如图 1.3 所示。



图 1.3 Borland C++ Builder 5 集成开发环境菜单栏

工具栏中的按钮可实现对 Borland C++ Builder 5 集成开发环境中一些菜单项的快速访问，它们的功能是与相应的菜单项一致的。表 1.11 分别列出了工具栏中各按钮对应的菜单项。

表 1.11 工具栏按钮及其对应的菜单项

工具栏按钮	菜单项	工具栏按钮	菜单项
	【File】【New】		【File】【Open】
	【File】【Save】		【File】【Save All】
	【File】【Open Project】		【Project】【Add to Project】
	【Project】【Remove from Project】		【Help】【C++Builder Help】
	【View】【Units】		【View】【Forms】
	【View】【Toggle Form/Unit】		【File】【New Form】
	【Run】【Run】		【Run】【Program Pause】
	【Run】【Trace Into】		【Run】【Step Over】

### 1.3.3 组件面板 (Component Palette)

组件是开发 Borland C++ Builder 5 应用程序的基本组成元素，应用程序的开发主要通过集成开发环境中提供的组件来实现。在 C++ Builder 中提供了多种基本组件，依据各个组件的不同特性，C++ Builder 将这些组件分类放在了组件面板中的多个选项卡里。在进行应用程序设计时，编程人员只需把需要的组件放在窗体或数据模块上，就可以对组件进行相应的处理。集成开发环境中的组件面板如图 1.4 所示。有关 C++ Builder 常用组件的一般用法本书将在以后的章节中叙述，这里不再赘述。

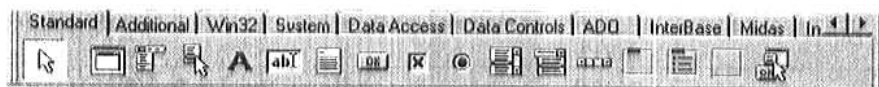


图 1.4 Borland C++ Builder 5 集成开发环境组件面板

## 1.4 C++ Builder 5 内置编程工具

Borland C++ Builder 5 的内置编程工具作为 C++ Builder 集成开发环境的一部分，主要包括了窗体、对象检视器、代码编辑器、类浏览器以及提供的一些辅助工具，比如图像编辑器、对象存储器和对齐工具栏等。下面主要对前几种工具作一简要介绍。

### 1.4.1 窗体

启动 Borland C++ Builder 5 或创建一个新的应用程序时，这个应用程序会自动创建一个新的窗体，如图 1.5 所示。窗体是 C++ Builder 程序设计的中心。一般说来，程序设计时的主要工作都是围绕着放置在窗体上的组件展开的。由于窗体中放置的是 C++ Builder 在集成开发环境中提供的各个组件，因而窗体也可以看作是一个包容组件的“容器”组件。

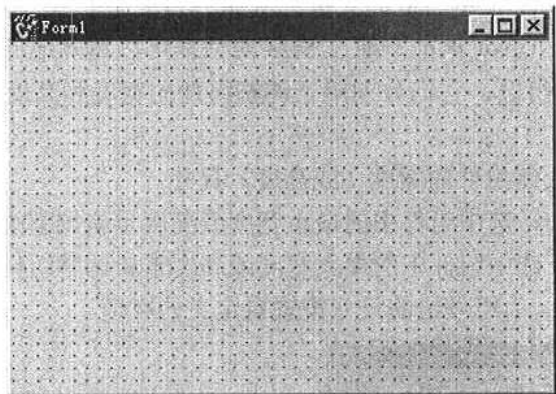


图 1.5 窗体

#### 1. 创建应用程序窗体和放置组件

在 Borland C++ Builder 5 中，通过以下的操作可以创建一个新的应用程序窗体：

- 启动 Borland C++ Builder 5 时，自动创建的新工程项目中包含一个新的窗体。
- 选择【File】、【New Application】菜单项，创建一个新的工程项目，其中包含一个新的窗体。
- 选择【File】、【New Form】菜单项或单击工具栏上的 New Form 快捷按钮，可以在当前的工程项目中添加一个新的窗体。

在应用程序设计阶段，常常根据应用程序的设计需要在窗体中放置多个相同或不同类别的组件，这些组件和窗体本身共同构成了应用程序的操作对象。将组件放置到窗体上的操作很简单，只需双击需要添加的组件，该组件就会出现在窗体的中央，另外，也可以单击组件，然后在需要出现组件的地方单击鼠标，把组件放在指定的位置。在窗体上放置了一个组件之后，可以用鼠标选中该组件，然后拖动到合适的位置。

#### 2. 改变组件尺寸

在实际应用中，在窗体上直接放置的组件的大小常常不能满足窗体界面设计的需要，



这时需要对组件的尺寸进行调整。C++ Builder 提供了多种调整组件尺寸的方法，下面分别作一简单介绍。



### 注意：

非可视化组件和有的可视化组件的尺寸是不允许改变的。

#### 1) 直接改变组件尺寸

选中需要改变尺寸的组件后，在组件的周围将出现一个边框，边框上有八个小黑块（锚点），如图 1.6 所示。其中，中间的四个锚点分别用于调整水平和垂直方向的尺寸，四周的锚点用于同时调整水平和垂直方向的尺寸。用鼠标拖动边框上不同位置的锚点，可使组件的大小发生相应的变化。

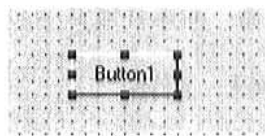


图 1.6 改变组件的尺寸

#### 2) 使用 Size 对话框改变组件尺寸

如果需要同时调整多个组件的尺寸，则可以使用 Size 对话框。在菜单栏中选择【Edit】、【Size】菜单项或者在窗体上单击鼠标右键，从弹出的快捷菜单中选择【Size】菜单项，就会打开如图 1.7 所示的 Size 对话框。通过 Size 对话框可以精确地调整选中的多个组件的大小，对话框中有两个分组框，它们分别用于调整组件的宽度和高度，其中各单选按钮项的意义如下：

- No change: 不改变组件的宽度（或高度）尺寸。
- Shrink to smallest: 选中组件中的最小宽度或高度重新调整组件组的尺寸。
- Grow to largest: 选中组件中的最大宽度或高度重新调整组件组的尺寸。
- Width 和 Height: 为组件组的尺寸设置自定义的取值。

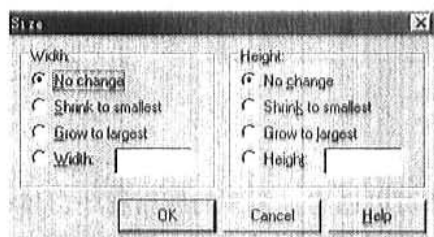


图 1.7 Size 对话框

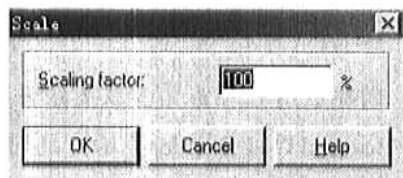


图 1.8 Scale 对话框

#### 3) 使用 Scale 对话框改变组件尺寸

在菜单栏中选择【Edit】、【Scale】菜单项或者在窗体上单击鼠标右键，从弹出的快捷菜单中选择【Scale】菜单项，就会打开如图 1.8 所示的 Scale 对话框。Scale 对话框与 Size 对话框的作用范围不一样，无论当前选中一个或多个组件，它都用于按比例调整窗体中所有组件的大小。Scaling factor 文本框中的比例因子的默认值是 100%，即不改变组件的大小，它的取值范围可在 25~400 之内，用户输入大于 100 的值和小于 100 的值则分别用于放大和缩小窗体上所有组件的尺寸。

### 3. 调整组件布局

组件的布局是影响窗体设计美观与否的一个重要因素，因此，在窗体中的组件个数较多时，编程人员常常需要对组件的布局进行调整。C++ Builder 中提供的调整组件布局的方