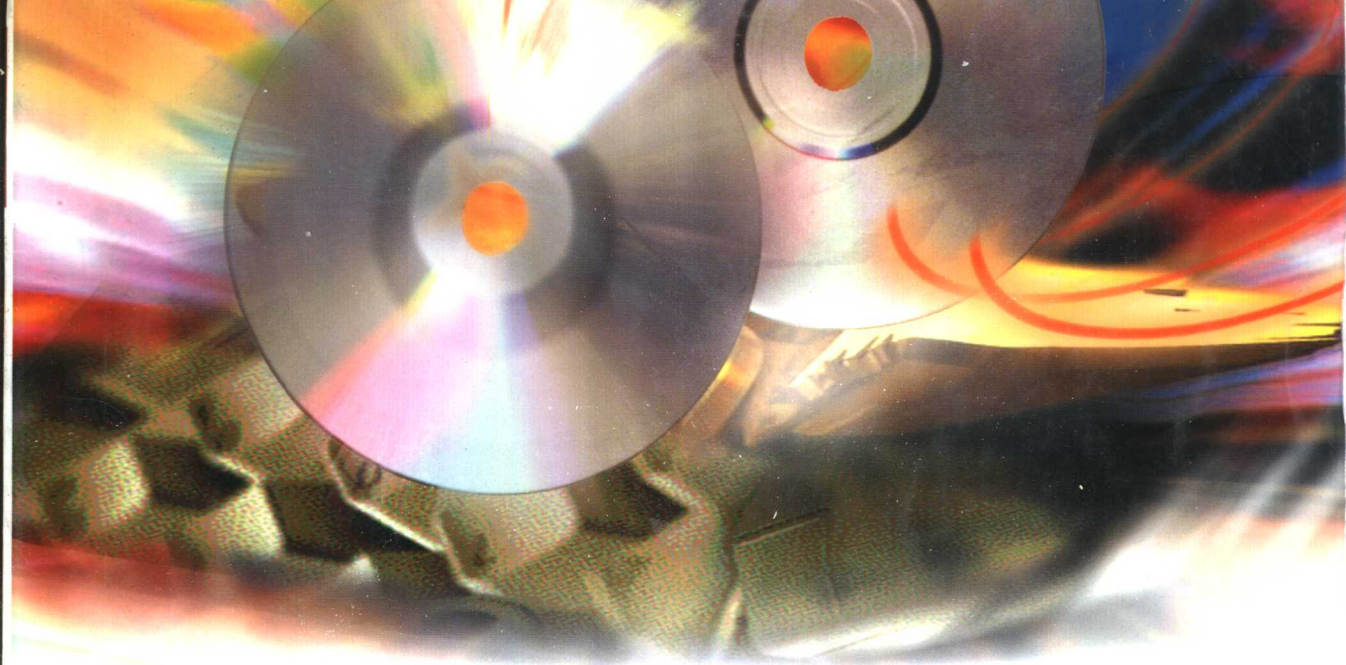




 万水计算机实用编程技术系列

Borland C++ Builder 5

高级开发技术

李冬 王宏 等编著



中国水利水电出版社
www.waterpub.com.cn

万水计算机实用编程技术系列

Borland C++ Builder 5 高级开发技术

李 冬 王 宏 等编著

中国水利水电出版社

内 容 提 要

本书针对 C++ Builder 5 在多媒体、数据库和网络的应用开发方面所涉及的基础知识、开发步骤及程序设计技巧作了比较深入的介绍。书中不仅对上述三个方面所涉及的组件作了较为详细的介绍,而且还对 OpenGL、DirectX、BDE 和 ODBC 等非组件技术也作了比较详细的介绍。书中的例子都在 Windows 98 平台下测试通过,并就其中的部分例子给出了较为详细的解释说明。

本书适合对 C++ 语言规范和 C++ Builder 5 开发环境比较熟悉的读者,尤其适合那些通过 C++ Builder 5 开发多媒体、数据库和网络应用程序的读者们,同时本书还是广大中专院校学生学习 C++ Builder 5 的一本好的参考书。

图书在版编目(CIP)数据

Borland C++ Builder 5 高级开发技术/李冬等编著. —北京:中国水利水电出版社, 2000.7

(万水计算机实用编程技术系列)

ISBN 7-5084-0404-1

I. B… II. 李… III. C 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2000)第 64420 号

书 名	Borland C++ Builder 5 高级开发技术
作 者	李冬 王宏 等编著
出版、发行	中国水利水电出版社(北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: sale@waterpub.com.cn 电话: (010) 63202266 (总机)、68331835 (发行部)
经 售	全国各地新华书店
印 刷	北京蓝空印刷厂
规 格	787×1092 毫米 16 开本 27.5 印张 622 千字
版 次	2000 年 7 月第一版 2000 年 7 月北京第一次印刷
印 数	0001—5000 册
定 价	40.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

前 言

Inprise 公司的 C++ Builder 是一种先进的可视化程序开发工具。在平台兼容性方面, C++ Builder 与 Windows 操作系统紧密结合, 使用了包括 Windows GUI 在内的许多先进特性和设计思想。在编程语言的选择上, C++ Builder 采取了主流的面向对象的程序设计语言 C++。一直以来, 每个版本的 C++ Builder 都以其简单易用和高效快捷而受到广大程序开发人员的青睐。

2000 年 1 月 26 日, Inprise 公司发布了其最新版的 C++ 开发工具 C++ Builder 5。作为研究开发工具先锋的 Inprise 公司, 其在面向对象的程序设计方面一直处于领先地位。新世纪之初, Inprise 公司全力推出的这一新款产品——C++ Builder 5, 一面市就引起了业界的强烈关注。在 C++ Builder 5.0 中, Inprise 公司对软件作了如下改进:

- 引进了 ADO、XML 等新的组件技术。
- 重新设计了编译器, 使得应用程序的性能有了很大的提高。
- 把编译置成后台运行, 大大提高了开发效率。
- 优化了开发环境, 允许用户定制并保存自己的环境设置, 有效地解决了多人共享的问题。

经常能听到广大程序员的强烈呼声: 要是有一款既能保持程序中 C++ 语言强大的程序设计能力, 又能很容易地设计出美观界面的开发工具就好了。

出于对计算机事业的热爱, 也是出于对开发效率和开发质量的追求, Inprise 公司推出了其得意产品——C++ Builder 5, 其为业界掀起快速开发应用无疑起到了推波助澜的作用。如今, 掌握一款类似 C++ Builder 5 这样的快速开发工具的使用已经成为对广大程序设计人员的一个基本要求。

为了推广 C++ Builder 5 的应用, 让更多的用户了解 C++ Builder 5、应用 C++ Builder 5, 我们特意编写了本书。

本书主要介绍了 C++ Builder 5 中较为高级的应用开发。在计算机软件业中, 多媒体、数据库和网络一直是应用开发的热点, 因此在第一章中对 C++ Builder 5 应用开发的基本步骤及技术要点作了总结性的介绍后, 还对 C++ Builder 5 在多媒体、数据库和网络等方面的应用作了比较深入的介绍。其中, 本书第二和第三章分别介绍了在 C++ Builder 中使用 OpenGL 和 DirectX 技术开发多媒体应用程序的方法; 本书的四、五、六、七章主要介绍 C++ Builder 在数据库应用程序方面的应用; 第九和第十章详细介绍了 C++ Builder 在开发网络程序上的应用。

对以上内容的介绍, 本书力求深入浅出, 除了对每种方法作了详细的说明外, 还给出了大量的程序范例供读者参考。本书所附带的实例都经作者严格挑选和测试, 大部分例子给出了详细的分析说明。

本书适合对 C++ 语法及 C++ Builder 5 的开发环境比较熟悉或有兴趣的读者，尤其适合那些希望使用 C++ Builder 开发多媒体、数据库和网络应用的读者朋友们。

本书主要由李冬、王宏负责编写，另外参加编写的还有朱晓华、陈春生、周皓晖、梁鹏、左建武、白庆中、段雪明、谢敏、孙友峰、杨国庆、张雨、黄伟、张强、齐超龙、郭烈、吴天力、陈风、杨敏中、应崎、王小雨、李明盛等同志。全书由魏红统稿。在此一并表示衷心的感谢。

鉴于作者水平有限，错误和疏漏之处在所难免，欢迎广大读者朋友批评指正。

编者

2000 年 4 月于北京

目 录

前言

第一章 C++ Builder 应用程序开发综述	1
1.1 应用程序开发方式的演变	1
1.1.1 基于 DOS 的应用程序开发	1
1.1.2 基于 Windows API 的应用开发	2
1.1.3 基于可视组件库 (VCL) 的应用开发	3
1.2 C++ Builder 的特色	3
1.2.1 C++ Builder 的集成开发环境	4
1.2.2 C++ Builder 中的程序调试手段	8
1.3 多文档应用程序	11
1.3.1 多文档应用程序设计的技术要点	11
1.3.2 多文档应用程序设计实例	12
1.4 多窗体应用程序	23
1.4.1 多窗体应用程序设计的技术要点	23
1.4.2 多窗体应用程序设计实例	23
1.5 多页面应用程序	28
1.5.1 多页面应用程序设计的技术要点	28
1.5.2 多页面应用程序设计实例	28
1.6 应用程序向导的应用	32
1.7 小结	34
第二章 图形图像处理	36
2.1 基本的图形图像类	36
2.1.1 TScreen 类	36
2.1.2 TCanvas 类	38
2.1.3 TPen 类	41
2.1.4 TBrush 类	43
2.1.5 基本类在绘图程序中的应用	44
2.2 基本图形实体的绘制	49
2.2.1 绘制基本图形实体的函数	49
2.2.2 基本图形实体绘图程序	52
2.3 图像显示	58
2.3.1 位图的基础知识	58
2.3.2 位图浏览程序	59
2.4 OpenGL 简介	65

2.4.1	OpenGL 技术概要	65
2.4.2	OpenGL 程序设计	66
2.4.3	OpenGL 的基本绘图函数	68
2.4.4	使用 OpenGL 的绘图实例	69
2.5	小结	76
第三章	DirectX 技术	77
3.1	DirectX 技术基础	77
3.1.1	DirectX 的特点	77
3.1.2	DirectX 的组成	77
3.1.3	DirectX 的结构	79
3.1.4	DirectX 的属性	80
3.2	DirectDraw 技术	82
3.2.1	显示技术的发展	82
3.2.2	DirectDraw 技术基础	84
3.2.3	DirectDraw 的实际应用	94
3.3	DirectSound 技术	103
3.3.1	声音缓冲区	103
3.3.2	DirectSound 声音格式	104
3.3.3	DirectSound 设置	104
3.3.4	DirectSound 回放	110
3.4	DirectInput 技术	114
3.4.1	DirectInput 技术基础	114
3.4.2	DirectInput 的实际应用	119
3.5	DirectX 的网络支持	126
3.6	小结	127
第四章	C++ Builder 的数据库开发环境	128
4.1	数据库基础知识	128
4.1.1	数据库技术的演变和发展	128
4.1.2	数据库的基本概念和相关术语	130
4.1.3	SQL 语言基础	132
4.2	C++ Builder 的数据库开发工具简介	151
4.2.1	BDE Administrator 工具	151
4.2.2	Database Desktop 的使用	152
4.2.3	ODBC Administrator	159
4.3	小结	161
第五章	C++ Builder 的基本数据库组件	162
5.1	数据库操纵组件	162

5.1.1	TDataSet 类和 DataSet 组件	162
5.1.2	Table 组件	164
5.1.3	Query 组件	177
5.1.4	StoredProc 组件	182
5.1.5	DataSource 组件	185
5.1.6	Session 组件	186
5.1.7	Database 组件	193
5.2	数据库控制组件	195
5.2.1	DBGrid 组件	197
5.2.2	DBNavigator 组件	200
5.2.3	DBText 组件, DBEdit 组件和 DBMemo 组件	201
5.2.4	其他组件数据库操纵组件	202
5.2.5	数据库访问和控制组件应用实例	202
5.3	快速报表组件	210
5.3.1	QuickRep 组件	210
5.3.2	TQRBand 组件	214
5.3.3	QRGroup 组件	215
5.3.4	TQRLabel 组件	216
5.3.5	TQRDBText 组件	216
5.3.6	TQRExpr 组件	216
5.3.7	TQRSysData 组件	219
5.4	小结	220
第六章	BDE 和 ODBC 数据库引擎	221
6.1	BDE 综述	221
6.2	BDE API 中的重要函数	222
6.2.1	打开数据库	222
6.2.2	打开数据表	223
6.2.3	定位游标	224
6.2.4	获取字段值	225
6.2.5	释放资源	227
6.2.6	BDE 程序实例	227
6.3	ODBC 概论	230
6.3.1	ODBC 的结构	230
6.3.2	ODBC 会话	234
6.4	ODBC 对数据库的操作	234
6.4.1	ODBC 句柄	234
6.4.2	建立和数据源的连接	239

6.4.3	SQL 语句的执行	241
6.4.4	获取结果集的信息	242
6.4.5	错误捕获	246
6.5	ODBC 与 SQL 语句	248
6.5.1	配置 SQL 语句属性	248
6.5.2	ODBC 与数据定义语言	250
6.5.3	插入、更新和删除记录	253
6.5.4	维护数据完整性	255
6.5.5	查询和结果集	258
6.5.6	语句预处理	261
6.5.7	语句句柄复用	263
6.6	游标	263
6.6.1	游标库和游标	264
6.6.2	利用游标处理结果集	264
6.6.3	设置游标位置	265
6.6.4	提取多行数据	265
6.6.5	书签	266
6.7	字典和统计函数	266
6.7.1	获取数据库中的表和列	266
6.7.2	获取和表相关的索引	270
6.7.3	获取用户权限	271
6.8	小结	272
第七章	ADO 技术和 ADO 组件	273
7.1	ADO 技术体系	273
7.1.1	ADO 技术的发展历程	273
7.1.2	ADO 技术的体系结构	274
7.2	ADO 组件概述	276
7.3	ADODbConnection 组件	277
7.3.1	ADODbConnection 组件的主要属性	277
7.3.2	ADODbConnection 组件的主要方法	284
7.3.3	ADODbConnection 组件的主要事件	286
7.4	ADODbCommand 组件	287
7.4.1	ADODbCommand 组件的主要属性	287
7.4.2	ADODbCommand 组件的主要方法	288
7.5	ADODbDataSet 组件	289
7.5.1	ADODbDataSet 组件的主要属性	289
7.5.2	ADODbDataSet 组件的主要方法	291

7.5.3	ADO 组件的简单应用	293
7.6	ADOTable 组件	302
7.6.1	ADOTable 组件的主要属性和方法	302
7.6.2	ADOTable 组件的应用	303
7.7	ADOQuery 组件	309
7.7.1	ADOQuery 组件的主要属性和方法	309
7.7.2	ADOQuery 组件应用	310
7.8	RDSCConnection 组件	314
7.9	小结	315
第八章	组件开发	316
8.1	面向对象的程序设计基础	316
8.1.1	面向对象程序设计的基本概念	316
8.1.2	C++ Builder 中的面向对象程序设计	317
8.2	自定义日历组件	320
8.2.1	创建并注册组件	320
8.2.2	发布继承的属性和事件	322
8.2.3	改变初始值	322
8.2.4	调整单元格大小	323
8.2.5	填充单元格	324
8.2.6	跟踪日期	324
8.2.7	日期导航	330
8.2.8	清除空白单元格	331
8.3	创建一个数据浏览组件	332
8.3.1	创建并注册组件	332
8.3.2	添加组件的只读属性	333
8.3.3	添加数据链接	335
8.3.4	定义数据源	336
8.3.5	初始化数据链接	337
8.3.6	响应数据的变化	337
8.4	创建数据编辑组件	339
8.4.1	改变 FReadOnly 的值为 false	339
8.4.2	处理鼠标及键盘消息	339
8.4.3	更新数据链接	341
8.4.4	修改 DataChange 方法	341
8.4.5	更新数据集	342
8.5	小结	343

第九章 开发网络应用程序	344
9.1 计算机网络概述	344
9.1.1 网络技术的发展历程	344
9.1.2 网络协议	344
9.2 Windows Socket 应用开发	345
9.2.1 ClientSocket 组件	346
9.2.2 TServerSocket 组件	349
9.2.3 PowerSocket 组件	351
9.2.4 Socket 的综合运用	358
9.3 NMFTP 组件	364
9.3.1 NMFTP 组件的主要属性	364
9.3.2 NMFTP 组件的主要方法	365
9.3.3 NMFTP 组件的主要事件	372
9.3.4 NMFTP 组件的应用	380
9.4 NMPOP3 组件	392
9.4.1 NMPOP3 组件的主要属性	392
9.4.2 NMPOP3 组件的主要方法	393
9.4.3 NMPOP3 组件的主要事件	396
9.5 小结	400
第十章 WWW 平台下的应用开发	401
10.1 WWW 概述	401
10.1.1 WWW 基本概念	401
10.1.2 HTML 简介	402
10.2 CGI/ISAPI 程序设计	403
10.2.1 CGI 的工作原理	403
10.2.2 CGI 的运行环境	404
10.2.3 一个简单的 CGI 应用程序	404
10.2.4 CGI 和客户的交互	406
10.2.5 ISAPI 简介	407
10.3 WebModule	408
10.3.1 创建 WebModule	408
10.3.2 TWebResponse 类	410
10.3.3 TWebRequest 类	411
10.4 PageProducer 组件	412
10.4.1 PageProducer 组件的主要属性	412
10.4.2 PageProducer 组件的主要方法	413
10.5 DataSetPageProducer 组件	414

10.6 QueryTableProducer 组件.....	414
10.7 WebModule 和 WebBroker 组件的使用	415
10.8 小结	427

第一章 C++ Builder 应用程序开发综述

程序开发并不是一件神秘的工作，简单而言，就是将简单的应用程序组合起来，完成复杂的功能。应用可视化的编程工具编程，更是事半功倍，只需学会简单的程序语言就可以编制出功能强大的应用程序。而一般的从事商业目的的程序开发必须具有坚实的计算机软件和硬件基础。

本章将结合几个典型的应用程序来介绍 C++Builder 编程的基础知识。

1.1 应用程序开发方式的演变

随着计算机硬件和软件技术的不断发展，特别是 CPU 运算速度的不断加快和操作系统功能的日益完善，应用程序设计的方法和开发环境也逐渐演变和发展。总体来说，程序开发的方式经历了以下三个阶段：

- 基于 DOS 的应用程序开发
- 基于 Windows API 的应用开发
- 基于可视组件库 (VCL) 的应用开发

1.1.1 基于 DOS 的应用程序开发

在 MS-DOS 中无论用何种语言进行程序设计都有一个明显的特点，就是程序主要使用顺序的过程驱动的方法。例如，一个程序由 A、B、C、D 四个功能块组成，其中 C 为判断的功能块。则程序的执行过程一共有两个执行顺序：A→B→C→D 或 A→B→C→B（循环）→D。程序执行的流程图如图 1-1 所示。

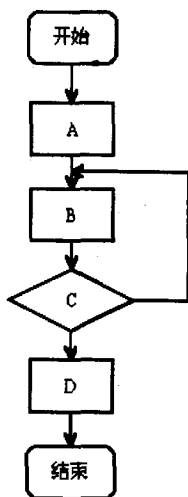


图 1-1 DOS 程序执行流程图

而且 DOS 下的程序开发工具一般只由字符编辑器、程序编译器和调试工具组成。所有的程序功能都必须由编程者写入源程序中, 实现程序界面和菜单等功能需要编写大量的代码。这样, 程序员就必须将大量的精力投入到程序界面的实现上, 而程序的核心功能就相对得不到足够的资源投入。

所以, 相比目前计算机技术状况而言, DOS 下的程序开发是一项效率十分低的工作, 程序员的大量时间和精力被耗费在开发程序界面上了。

1.1.2 基于 Windows API 的应用开发

Windows 程序设计的方法是以事件来驱动程序的。事件驱动程序的开发是以一种“并行”的方法, 一种非顺序的方式处理事件, 从而避开了顺序的过程驱动的方法。

Windows 事件驱动程序设计是围绕着消息的产生与处理而展开的。一条消息 (Message) 标识着一个事件的发生, 例如当键盘上的某个键被按下时, 程序可以收到一条消息; 当该键被释放时, 程序又收到另一条消息。

Windows 下程序执行的流程图如图 1-2 所示。

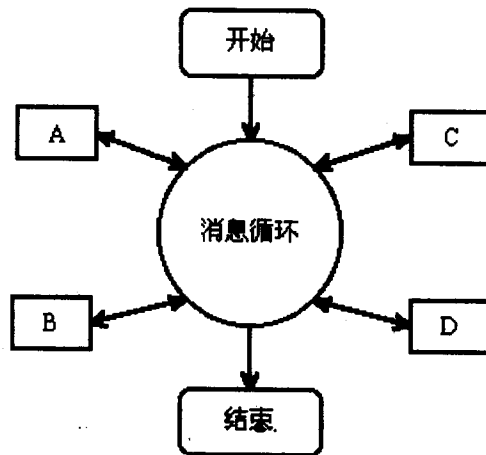


图 1-2 Windows 程序执行流程图

由于消息是由事件随机产生的, 所以程序不会以任何预先定义的顺序执行, 而是根据收到的不同消息执行不同的功能块。这样就比 DOS 下的程序大大前进了一步, 使程序的功能和利用率得到很大的提高。

Windows 环境是通过应用程序接口 (API, Application Programme Interface) 被应用程序访问的。API 由许多可以被用户程序调用的函数组成, 通过 API 函数可以调用 Windows 支持的所有系统服务。API 的出现为程序员在程序中充分利用系统资源带来了很大的便利, 有利于写出高效的程序代码。

Windows 3.x 系列使用的是 16 位 API, 仅支持 16 位的分段内存 (segment-memory) 模式。而 Windows 9x 系列使用的是 32 位 API, 支持 32 位寻址。实际上, Windows 9x 的 API 是将 Windows 3.x 的 API 扩展而成的, 是 Windows 3.x API 的一个超集。

由于这个原因, Windows 9x API 中的许多函数都被扩展为可以接收 32 位参数和返回 32 位函数值的函数。而且, 函数的功能也为了使用 32 位的结构体系而作出了相应的修改, API 增加了支持新的多任务方式、新界面元素和 Window 9x 的其他增强特性。所以, 读者在阅读较早的有关 Windows 编程的书籍时要注意这一点。

1.1.3 基于可视组件库 (VCL) 的应用开发

在 Windows 操作系统 (特别是 Windows 95 操作系统) 取得巨大成功的同时, 各大软件厂商都推出了许多基于 Windows 的集成开发工具包。比较流行的有 Microsoft 的 Develop Studio (包括 VB、VC 和 VJ)、IBM 的 PowerBuilder 和 Borland (Inprise) 公司的 Delphi、C++ Builder、JBuilder 等。

这些集成开发工具包除了提供传统开发工具的编译器、调试工具、代码编辑器等必备的工具外, 还具有以下的优点:

- 全图形界面、友善易用的集成开发环境。
- 具有功能强大的项目管理器 (Project Manager), 便于编程者开发复杂的应用程序。
- 提供功能强大的可视组件库 (VCL)。

其中可视组件库是程序开发工具最具革命性的突破之一。

有了可视组件库, 程序界面的设计和实现就成了轻而易举的事情。每一种工具都提供了多至上百种的标准组件, 供用户使用, 包括 Windows 下的所有窗体元素、对话框、选项框和菜单等等。通过向窗体中添加各种组件, 可以轻易设计出整齐规范的界面。

组件本身提供了完备的功能, 例如向窗体中添加一个打开文件对话框, 不需要进行任何编程, 在程序运行时对话框中已经能够正确显示本地微机中各个驱动器的内容。这样的设计使得没有任何编程知识的人都可以通过简单的学习编制出功能强大的应用程序, 使得编程首次成为“平易近人”的游戏。

程序的各项功能被封装在各组件当中, 可以在组件的事件中处理 Windows 的各种消息, 并通过改变组件的属性, 执行组件的方法来实现程序的各项功能。所以, 在编程中应用组件不会削弱和限制程序的功能。

组件库都支持强大的扩展功能, 表现在:

- 可以自己设计新的组件并添加到程序的组件库中。
- 可以利用他人的组件, 将他人设计好的组件添加到自己的组件库中。
- 可以购买第三方软件公司开发的商业组件。

这一扩展可能使得程序代码得到最大程度的有效利用, 将重复劳动量降到最低。并使得程序开发的效率和质量直接与组件的功能密切相关。这一特点同时促进了二级开发的发展, 开发新组件、功能更强大的组件成为程序开发的重要组成部分。

1.2 C++ Builder 的特色

Borland 公司在 DOS 下的 C 语言编译器 Turbo C 及 Turbo C++, 获得了巨大的成功,

成为第二个在 PC 上推出商业化 C++ 的编译器厂商（第一个是 Zortech C++，由 C++ 之父 Bjarne Stroustrup 亲自设计，现在已被 Symantec 收购）。这几个成功的产品，将 Borland 的声势带上高峰，于是为了更加打响 Borland 的名号，将旗下的所有产品，都冠以 Borland 之名，于是 Borland 的后续产品以 Borland 命名，如 Borland C++、Borland Pascal 等等。

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 操作系统中开发 Windows 程序的开发工具，当时 Borland 还以此为号召，大力抨击当时的 MSC (Microsoft C) 只能在 Dos 下编写 Windows 程序。然而随着 Microsoft 推出 Visual C++ 之后，形势逆转，在视觉发展工具的优胜劣汰下，Borland 慢慢地丧失它原有的领先地位。

Borland 为了扭转不利形势，在经过了几年的卧薪尝胆之后，终于推出了一个令全世界程序员为之瞩目的开发工具，这就是 Borland Delphi。Delphi 以快速程序发展工具 (RAD, Rapid Application Development) 为卖点，推出之后立即成为程序员关注的焦点，成为挽救当时 Borland 公司的最大功臣。

在 Delphi 推出之后，奠定了 Borland 在 RAD 领域的不朽地位，而 Delphi 也成为在 PC 上唯一可以和工作站级的 Nextstep IB (Interface Builder) 等量齐观的重量级程序开发工具。Borland 的公司随后乘胜追击，又推出了基于 VCL (Visual Component Library) 的 Windows 下 C++ 编辑器，即本书要介绍的 Borland C++ Builder，简称 BCB。

Borland C++ Builder 是一种极优秀的程序开发工具，与目前市面上多种开发工具相比，它具有以下特点：

- 用 C++ Builder 开发的程序，无论从程序的大小还是程序的执行效率来说都足以与基于 MFC (Microsoft Function Class) 的 Visual C++ 和基于 OWL (Object Window Library) 的 Borland C++ 所开发的程序相媲美。

- C++ Builder 没有 VB 和 PowerBuilder 的功能局限，所有用 MSVC 或 Borland C++ 5 能够完成的工作都可以用 C++ Builder 来完成。

由于 C++ Builder 既保持了 Borland C++ 的优点，又具有完善的 RAD 工具特性，它越来越受到程序员的喜爱，成为足以和 MSVC 一争天下的优秀开发工具。目前，C++ Builder 的最高版本为 5.0。

1.2.1 C++ Builder 的集成开发环境

C++ Builder 的集成开发环境主要包括以下几部分：

- 窗体 (Form)。
- 组件面板 (Component Palette)。
- 对象检视器 (Object Inspector)。
- 程序编辑器 (Code Editor)。
- 加速条 (SpeedBar)。

1. 窗体 (Form)

刚进入 C++ Builder 时，会自动生成一个空白的窗体，如图 1-3 所示。

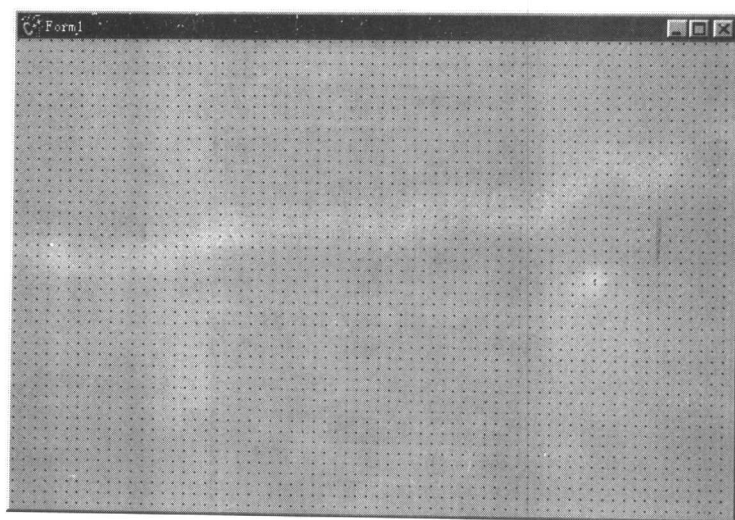


图 1-3 窗体

窗体是 C++ 程序设计的焦点，我们可以将窗体视为一张画布，在程序设计时，我们主要做的工作只是将程序组件放置在窗体上而已。因此窗体其实可视为一个包含其他组件的组件。

2. 组件面板(Component Palette)

组件是用以架构 C++ Builder 程序的基本元素。在 C++ Builder 中组件依其组件特性分门别类地放置在活页式的组件面板中，在 C++ Builder 中提供了超过了百种以上的基本组件，程序设计时，只要将组件箱内的组件拖放至表格中即可，图 1-4 所示即为组件面板。

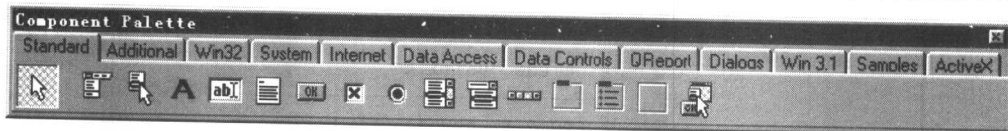


图 1-4 组件面板

3. 对象检视器(Object Inspector)

对象检视器是用以改变对象的内容，改变该对象的行为模式以符合程序需要。对象的内容包含了属性 (Property) 及其事件处理程序 (Event Handler)。编程者可以根据需要改变指定对象的状态及行为模式。图 1-5 所示为对象检视器。

4. 程序编辑器(Code Editor)

程序编辑器如图 1-6 所示。

C++ Builder 的程序编辑器是一个功能齐全的编辑器，它使得用户可以存取及编辑本项目内包含的所有程序文件。

在程序编辑器中还可以同时打开多个文件，程序编辑器会以活页的形式显示每一个文件的内容。