

王忠华 魏志强 杨志刚 乔林 编著

高级编程篇

程序设计 Delphi 5.0



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

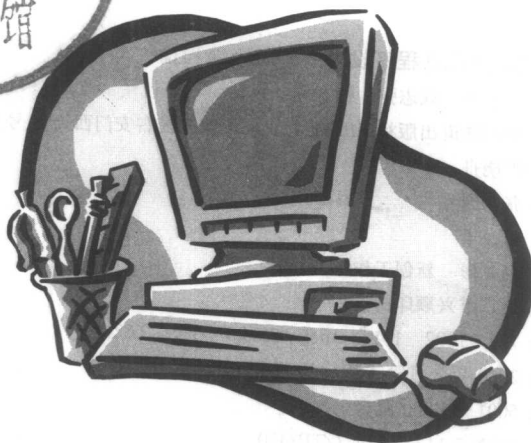
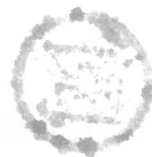
00013349

TP311.56
114
V2

Delphi 5.0 程序设计

——高级编程篇

王忠华 魏志强 杨志刚 乔林 编著



中国铁道出版社

2000·北京



C0489556

JS506/03

(京)新登字 063 号

内 容 简 介

本书中详细介绍了组件的高级用法、自定义组件、DLL 和 ActiveX 控件,以及 COM 应用;本书还对顺应 Windows98、Windows NT 的多线程技术、在 Delphi 中的汇编嵌入问题进行了介绍。在本书的最后还系统地讲解了 Delphi 中应用程序帮助系统和安装问题。

本书面向 Delphi5.0 的高级读者。

图书在版编目(CIP)数据

Delphi5.0 程序设计.高级编程篇/王忠华编著. —北京:中国铁道出版社,2000.5(重印)

ISBN 7-113-03614-7

I. D… II. 王… III. Delphi 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(1999)第 56865 号

书 名: Delphi5.0 程序设计——高级编程篇

作 者: 王忠华 魏志强 杨志刚 乔林

出版发行: 中国铁道出版社(100054,北京市宣武区右安门西街 8 号)

策划编辑: 严晓舟

责任编辑: 张永国

特邀编辑: 田 丰

封面设计: 冯龙彬 新创工作室

印 刷: 北京市兴顺印刷厂

开 本: 787×1092 1/16 印张: 28.25 字数: 682 千

版 本: 2000 年 2 月第 1 版 2000 年 5 月第 2 次印刷

印 数: 5001~10000 册

书 号: ISBN 7-113-03614-7/TP·420

定 价: 45.00 元

版权所有 盗印必究

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。

前 言

Delphi 5.0 一个功能强大的可视化编程环境，它为我们提供了一种方便、快捷的 Windows 应用程序开发工具。它使用了 Microsoft Windows 图形用户界面的许多先进特性和设计思想，采用了弹性的可重用的面向对象 Pascal 程序语言。对于广大的程序开发人员来讲，使用 Delphi 5.0 开发应用软件和数据库应用程序，无疑会大大地提高编程效率，而且随着应用的深入，我们将会发现编程不再是枯燥无味的工作。

Delphi 5.0 的程序首先是一个应用程序框架，而这一框架正是应用程序的“骨架”。在骨架上即使没有附着任何东西，也仍然可以严格地按照设计运行。程序员的工作只是在“骨架”中加入自己的程序。缺省的应用程序是一个空白的窗体 (Form)，我们可以运行它，结果得到一个空白的窗口。这个窗口具有 Windows 窗口的全部性质：可以放大缩小、移动、最大化和最小化等，但我们却没有编写一行程序。因此，可以说应用程序框架通过提供所有应用程序共有的东西，为用户应用程序的开发打下了良好的基础。Delphi 5.0 已经为读者做好了一切基础工作——程序框架就是一个已经完成的可运行应用程序，只是不处理任何事情。我们所需要做的，只是在程序中加入完成所需功能的代码而已。Delphi 5.0 把 Windows 编程的回调、句柄处理等繁复过程都放在一个不可见的覆盖物下面，这样读者可以不为它们所困扰，轻松从容地对可视化部件进行编程。

本丛书共分四个专题，涵盖了 Delphi 5.0 编程的方方面面：

《Delphi5.0 程序设计——基础教学篇》：该书面向 Delphi5.0 的初、中级读者。全书分 3 个部分介绍了 Delphi5.0 的基础知识和基本的编程方法。前 3 章介绍了 Delphi5.0 的新特性、集成开发环境和各种辅助工具的用法以及 Delphi 程序可视化开发方法；第 4、5、6 章介绍了对象 Pascal 语言的基础知识和各种语法实体的用法；第 7、8、9、10 章为编程实践部分，介绍了菜单的设计方法、文本编辑器、文件管理和图形图像应用程序的设计方法，这 4 章中结合实例程序的设计，介绍了 VCL 中菜单、文件、对话框、工具条、图形图像等众多组件的用法。全书由浅入深，循序渐进。在各部分内容的介绍中均配备了实例程序，力求通过这些实例加深理解，掌握编程方法和编程技巧。

《Delphi5.0 程序设计——数据库应用实务篇》：该书面向 Delphi 数据库程序设计编程人员。书中详细介绍了数据库访问部件、数据库控制部件、Delphi 环境中的 SQL 编程、客户机/服务器环境的构造和编程方法、数据仓库技术、InterBase 数据存取部件、综合数据报表制作、Delphi5.0 与 ADO 编程、DCOM、MTS 和 CORBA 等分布式数据库应用程序开发、多层数据库应用程序开发、Delphi 与 Word97、Excel97 等 Office 组件的集成等技术。同时，本书提供了通用数据查询、图形化数据报表分析等实用化范例。本书最后，详细地分析了一个集成化财务管理系统并提供了核心模块的全部源代码，该财务系统已经开始应用于商务环境中。

《Delphi5.0 程序设计——高级编程篇》：该书面向 Delphi5.0 的高级读者。Delphi 这种可视化的 RAD 工具，其核心思想可以认为是基于组件的。步入 Delphi 的世界并不难，但是要在里面建设更为高级的应用却要仔细理会 Delphi 的真谛。本书中详细介绍了组件的高级

用法、自定义组件、DLL 和 ActiveX 控件，以及 COM 应用；本书还对顺应 Windows 98、Windows NT 的多线程技术、在 Delphi 中的汇编嵌入问题进行了介绍。在本书的最后还系统地讲解了 Delphi 中应用程序帮助系统和安装问题。

《Delphi5.0 程序设计——Internet 应用实务篇》：本书讨论如何使用 Delphi 5.0 进行 Internet 编程。我们将注意力主要集中在如何使用 Delphi 5.0 的部件进行客户端编程上，这部分内容是在 Internet 上冲浪时会频繁遇到的。作为本书的结尾，我们专辟一章详细讨论如何使用 Delphi 5.0 开发 Web 服务器应用程序。该书面向 Delphi5.0 的中、高级读者。

本套丛书由杨志刚、魏志强、王忠华、乔林组织策划。《基础教学篇》由杨志刚、何志成、魏志强、王忠华主编，乔林、何敏、谢军、段军、刘文杰、张维平等参编；《数据库应用实务篇》由魏志强、王忠华、何敏、乔林主编，杨志刚、房桂英、魏青、张积中、黄建东、陈爱华等参编；《高级编程篇》由王忠华、魏志强、杨志刚、乔林主编，高群、黄建东、姜涛、林杜、费广正等参编；《Internet 应用实务篇》由乔林、杨志刚、黄建东、魏志强主编，王忠华、何隽、林杜、费广正、金传恩、王诚铭等参编。杨志刚、乔林审阅了全套丛书的内容。限于时间仓促，作者水平有限，书中如有错误，恳请读者谅解。

“边学边做”是学习一种编程工具的最有效方法，我们希望读者在阅读本丛书的过程中能够上机实践。每学完一个例子，尝试着改变一点点，或者添加一点东西，并改变一些代码将帮助读者体验进步和成功的乐趣。

编者
1999 年 10 月

目 录

第 1 章 深入认识 Delphi	1
1.1 Delphi 的特点	1
1.2 Delphi 5.0 的适用范围	1
1.2.1 Delphi 5.0 的开发工具及特征	1
1.2.2 Delphi 5.0 应用概述	3
1.3 应用程序中的窗体	5
1.4 创建窗体	8
1.4.1 Delphi 应用程序的 TApplication 类	8
1.4.2 窗体的创建	9
1.4.3 Delphi 5.0 的窗体创建事件	10
1.4.4 关于 Splash	13
1.4.5 创建全局公用模块	15
1.5 工程管理	23
1.5.1 项目管理器	23
1.5.2 创建 To-do lists	24
1.5.3 创建项目组	25
1.5.4 模板和对象库	26
1.6 Delphi 5.0 环境定制	27
1.6.1 组织工作区	27
1.6.2 定制桌面	29
1.6.3 定制帮助系统	32
1.7 小结	32
第 2 章 使用组件	33
2.1 Delphi 可视化组件概述	33
2.1.1 与文本相关组件	33
2.1.2 按钮与选项组件	36
2.1.3 状态条组件	36
2.1.4 窗体分割组件	36
2.1.5 列表组件	37
2.1.6 滚动组件	38
2.1.7 网格、表格组件	39
2.1.8 图形、图像组件	39
2.1.9 多媒体(MultiMedia)和 OLE 组件	39

2.1.10 标准对话框组件.....	40
2.2 框架 (Frames) 用法.....	41
2.2.1 创建框架.....	41
2.2.2 将框架添加到组件模板.....	42
2.2.3 使用和修改框架.....	42
2.2.4 框架的共享.....	44
2.3 菜单组件及其用法.....	44
2.3.1 菜单项的功能分类.....	45
2.3.2 动态建立菜单项.....	48
2.3.3 在 Delphi 5.0 中创建图像菜单.....	50
2.3.4 Owner-Draw 菜单项.....	51
2.4 ListView 组件与 TreeView 组件.....	53
2.4.1 ListView 组件的用法.....	54
2.4.2 TreeView 组件的用法.....	57
2.5 工具栏组件的用法.....	61
2.5.1 Toolbar 组件.....	62
2.5.2 Coolbar 组件.....	62
2.5.3 Controlbar 组件.....	66
2.5.4 PageScroller 组件.....	69
2.6 小结.....	69
第3章 Delphi 自定义组件.....	70
3.1 组件原理.....	70
3.1.1 什么是组件?.....	70
3.1.2 建立组件过程概述.....	71
3.1.3 建立组件的途径.....	71
3.1.4 组件 Delphi 中的消息处理机制.....	72
3.2 由现存组件派生组件.....	77
3.2.1 创建简单的组件.....	77
3.2.2 测试新建组件.....	80
3.2.3 改变组件的缺省属性.....	81
3.2.4 将组件放入组件面板中.....	82
3.3 多组件组合.....	87
3.4 创建新组件.....	95
3.4.1 创建属性.....	95
3.4.2 创建事件.....	98
3.4.3 处理消息.....	103
3.4.4 增加 Component Palette 上的位图.....	107
3.4.5 提供有关属性和事件的帮助.....	108

3.5 非可视化组件.....	113
3.5.1 创建非可视化组件.....	113
3.5.2 使用非可视化组件.....	118
3.5.3 定义客户操作.....	119
3.6 小结.....	119
第4章 多线程 (THREAD)	120
4.1 线程的基本知识.....	120
4.1.1 进程和线程.....	120
4.1.2 多任务.....	121
4.1.3 何时创建线程.....	121
4.1.4 何时不用创建线程.....	122
4.1.5 线程的调度.....	122
4.2 用 Delphi 5.0 创建多线程应用程序.....	124
4.2.1 Delphi 5.0 的多线程工作模型	124
4.2.2 线程构件.....	124
4.2.3 创建线程类.....	126
4.2.4 多线程同步.....	131
4.2.5 线程的控制.....	139
4.3 多线程控制应用范例.....	139
4.4 小结.....	144
第5章 动态链接库 (DLLS) 编程.....	145
5.1 动态链接库原理.....	145
5.1.1 动态链接库的工作原理.....	146
5.1.2 Windows 系统的动态链接库	146
5.1.3 为什么使用 DLL?	147
5.1.4 系统 DLL 的含义.....	148
5.1.5 DLL 与 EXE 文件之间的区别	148
5.1.6 DLL 编写规则	149
5.2 在 Delphi 中 DLLs 的编写和调用.....	149
5.2.1 编写一般 DLLs 的步骤.....	151
5.2.2 动态链接库中的标准指示.....	152
5.2.3 DLLs 中的变量和段	153
5.2.4 DLLs 中的运行时间错和处理	153
5.2.5 库初始化代码的编写.....	154
5.2.6 编写一般 DLLs 的应用举例.....	155
5.2.7 DLL 中的重载函数问题	156
5.2.8 从 DLL 中输出字符串.....	156

5.2.9 调用 DLLs	158
5.3 利用 DLLs 实现窗体重用	166
5.3.1 利用 DLLs 实现窗体重用的一般步骤	166
5.3.2 使用 DLL 模态窗体	166
5.3.3 DLL 中的非模态窗体	168
5.3.4 窗体重用实例	168
5.4 利用 DLLs 实现数据传输	174
5.4.1 DLLs 中的全局内存	174
5.4.2 利用 DLLs 实现应用程序间的数据传输	175
5.5 DLL 与 Delphi 组件包	178
5.6 小结	179
第 6 章 COM 对象	180
6.1 COM 的基本概念	180
6.2 Delphi 实现 COM 的机理	180
6.3 创建 COM 应用	182
6.4 COM 的注册与调用	186
6.5 利用 COM 实现系统外壳扩展	189
6.5.1 上下文菜单扩展 COM 的编写	189
6.5.2 程序说明	195
6.6 小结	198
第 7 章 对象链接与嵌入	199
7.1 OLE 简介	199
7.1.1 OLE1.0 和 OLE2.0	199
7.1.2 链接与嵌入	199
7.1.3 设计状态 OLE 对象的创建	200
7.1.4 OLE 类、文件、项目	200
7.1.5 OLE 自动化	201
7.1.6 类型库	201
7.1.7 OLE 数据类型	202
7.1.8 TAutoObject 对象	203
7.2 创建 OLE 自动化服务器	205
7.2.1 编写 OLE 自动化服务器	205
7.2.2 Delphi 5 类型库编辑器	205
7.2.3 服务器程序	208
7.2.4 注册自动化服务器	212
7.2.5 编写客户程序	213
7.3 TOLEContainer 组件	214

7.3.1	OLE 对象创建的步骤	214
7.3.2	OLE 应用程序的开发	215
7.4	OLE 与 Office97.....	219
7.4.1	早期的控制方式.....	220
7.4.2	利用 Delphi 5.0 的新特性向 Word 发送数据.....	232
7.5	小结.....	237
第 8 章 ActiveX 控件		238
8.1	ActiveX 控件原理	238
8.1.1	何谓 ActiveX 控件?.....	238
8.1.2	ActiveX 控件与 Delphi 组件	239
8.2	在 Delphi 中使用 ActiveX 控件.....	240
8.2.1	ActiveX 控件的安装方法	240
8.2.2	TOLeControl 类	245
8.2.3	使用 Microsoft ActiveMovie Control 控件.....	245
8.2.4	使用 Delphi 的 ActiveX 控件.....	248
8.3	ActiveX 控件的编写方法	259
8.3.1	建立 ActiveX ChiCharEdit	259
8.3.2	添加新属性.....	264
8.3.3	添加属性页.....	265
8.3.4	ActiveX Form.....	267
8.4	小结.....	271
第 9 章 多媒体系统.....		272
9.1	认识多媒体.....	272
9.1.1	什么是多媒体 (Multimedia) ?.....	272
9.1.2	多媒体计算机系统.....	273
9.1.3	计算机多媒体系统的应用.....	273
9.1.4	多媒体制作软硬件工具.....	274
9.1.5	多媒体编辑系统.....	276
9.1.6	多媒体的相关名词.....	278
9.2	个人计算机的多媒体功能.....	280
9.2.1	Windows 98 的娱乐功能	280
9.2.2	MCI 接口	281
9.2.3	Microsoft DirectX 技术	283
9.3	Delphi 的多媒体控件	283
9.3.1	多媒体 MediaPlayer 控件.....	284
9.3.2	Delphi 的 MCI 控制播放原理及其时间格式.....	289
9.3.3	多媒体在 Delphi 中的应用.....	290

9.4 卡通构件.....	300
9.4.1 卡通控件简介.....	300
9.4.2 卡通构件.....	300
9.4.3 卡通构件的使用.....	302
9.5 小结.....	303
第10章 在 Delphi 5.0 中使用汇编.....	304
10.1 Delphi 5.0 的内置汇编器.....	304
10.1.1 Asm 语句.....	304
10.1.2 寄存器的使用.....	304
10.1.3 汇编程序语句的语法.....	305
10.1.4 标签.....	305
10.1.5 指令操作码.....	305
10.1.6 汇编指令.....	307
10.1.7 操作数.....	308
10.1.8 表达式.....	309
10.1.9 汇编程序过程和函数.....	318
10.2 应用汇编程序实例.....	319
10.2.1 访问接口及定时器——乐曲演奏程序.....	319
10.2.2 访问寄存器——CPUID.....	322
10.3 小结.....	332
第11章 异常处理与程序调试.....	333
11.1 Delphi 异常处理机制与异常类.....	333
11.1.1 运行库异常类(RTL Exception).....	334
11.1.2 对象异常类.....	338
11.1.3 部件异常类.....	340
11.2 异常处理.....	341
11.2.1 异常响应与 try-except 语句.....	341
11.2.2 异常保护与 try-finally 语句.....	342
11.2.3 使用异常实例.....	345
11.2.4 提供缺省响应.....	345
11.2.5 响应一族异常.....	346
11.2.6 异常的重引发和处理嵌套.....	346
11.2.7 自定义异常类的应用.....	347
11.2.8 利用异常响应编程.....	351
11.3 程序调试.....	353
11.3.1 调试工具的基本设置.....	353
11.3.2 程序运行的控制.....	355

11.3.3 使用断点.....	356
11.3.4 检查变量和表达式的值.....	359
11.3.5 检查函数和过程调用.....	360
11.3.6 监视线程状态.....	361
11.3.7 监视微处理器状态.....	362
11.3.8 局部变量监视窗口.....	363
11.3.9 模块窗口.....	363
11.3.10 事件日志窗口.....	365
11.3.11 FPU 窗口.....	367
11.4 使用 Browser	368
11.4.1 使用加速按钮.....	369
11.4.2 Details 区的显示页	369
11.4.3 使用 Browser	370
11.5 小结.....	371
第 12 章 发布软件.....	372
12.1 Delphi 应用程序的 Help 系统	372
12.1.1 Help 文件的建立	372
12.1.2 应用 hcw.exe 创建 Help 的实例.....	380
12.1.3 Delphi 应用程序的 Help 编程	387
12.1.4 Delphi 帮助提示(Hint)的应用	390
12.2 支持文件.....	394
12.2.1 使用 Windows INI 文件.....	394
12.3 Windows 注册表	397
12.4 Installshield Express 工具	402
12.5 小结.....	414
附录 Delphi 错误信息表	415

第 1 章 深入认识 Delphi



如果阅读了本系列图书的基础篇，那么现在就能够使用 Delphi 的基本组件了。本章将要讨论 Delphi 应用程序开发中的一般问题。介绍 Delphi 的特点、使用窗体、窗口、TApplication 类和应用程序。还介绍了应用程序开发的工程管理、途径以及环境定制。

1.1 Delphi 的特点

Delphi 自 1.0 版本推出以来，已经历了 5 个版本。新推出的 Delphi 5.0 可以全面支持 Microsoft Windows 95、Windows 98 和 Windows NT 的高效率应用程序开发，并使手工编写的代码降至最少。Delphi 有众多优越特性。它提供了新颖的可视化设计工具，采用面向对象的方法将 Windows 编程的复杂性封装起来，实现了将可视化设计与 Object Pascal 语言的有机集成，配有 Database Engine，可通过 SQL Links、ODBC 访问多种数据库，并且提供了强大的开发基于客户/服务器模式的数据库应用的能力。Delphi 全面支持 Windows 95、98 和 NT 的 OLE Automation、OCXs、多线程、Unicode 和 MAPI、多媒体，以及 COM、Midas、CORBA、Internet 应用等功能。作为一种 RAD (Rapid Application Development) 可视化编程环境，它包含了大量的部件库、设计工具、应用程序与窗体模板以及编程向导，为编程者提供了充分得力的开发和调试工具，有利地缩减了程序开发周期。Delphi 使用的基础语言是 Object Pascal，由于 Object Pascal 是一种强类语言，与其他语言相比，它提供了一种快速的编译器。优化的编译模式在很大程度上提高了代码的效率，因此 Delphi 自推出以来一直受到了人的青睐。

1.2 Delphi 5.0 的适用范围

1.2.1 Delphi 5.0 的开发工具及特征

Delphi 5.0 的集成开发环境 (IDE) 中集成了窗体设计器 (Form Designer)、对象监视器 (Object Inspector)、部件模板 (Component palette)、项目管理器 (Project Manager)、代码浏览器 (Code Explorer)、代码编辑器 (Code editor)、数据模块设计器 (Data Module Designer)、软件区域工具 (software localization tools)、调试等许多工具。

Delphi 所有版本都支持真正的 32 位的 Windows 编程、多线程、COM 和自动化 (Automation) 控件，以及多进程调试。某些版本还增加了对服务器应用程序开发的支持，例如 COM 服务器、Web 应用程序，还可以为多种后端的数据库管理系统开发前端的数据库应用程序，支持 SQL 数据库服务器 (例如 Oracle 8 和 InterBase)、Microsoft Transaction Server (MTS)、多层数据库应用程序，以及 CORBA 和决策支持系统。

1 使用 VCL

Delphi 来自部件，而部件又是可视化部件库（VCL）类层次的一部分。VCL 包含了一些运行时可见的对象，例如编辑控制、按钮等用户界面元素，还包括了一些运行时不可见的控件，例如数据库控件和定时器等。图 1-1 所示为组成 VCL 的基类。

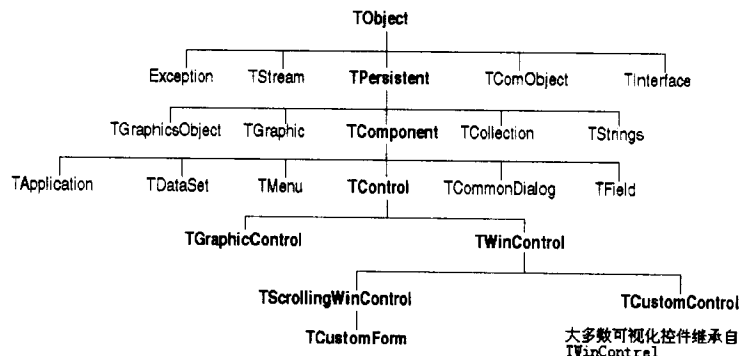


图 1-1 组成 VCL 的基类

由 TComponent 派生的对象所具有的属性和方法使之可以安装在部件模板中或增加在 Delphi 的窗体中。而由于 VCL 部件集成在 IDE 中，所以可以利用诸如窗体设计器之类的工具快速开发应用程序。

部件是高度浓缩封装的，例如按钮被预编为可以响应鼠标的单击事件，这样在使用 VCL 中的按钮控件时就不必编写处理 Windows 消息代码，编程者只需按照应用程序中一定的逻辑编写响应事件代码。

Delphi 的某些版本还提供了部分 VCL 的源代码，还补充了联机帮助文档，VCL 源代码提供了价值无限的 Object Pascal 编程技术的实例。

2 异常处理

Delphi 的错误处理机制是创建在异常处理之上的，异常处理是一个特殊的对象，用于响应不期的输入或程序运行错误。在设计状态和运行状态都可以产生异常，VCL 包含了许多异常类与特定的错误条件关联。在开发应用程序过程中，可以编写异常处理代码以处理运行错误，异常还是一个很有价值的调试工具，因为异常往往能够提供引起错误的线索。

3 数据库的连接和应用

Delphi 和 VCL 提供了大量的简化数据库应用程序的数据库连接工具。Borland Database Engine (BDE) 就是一个支持大量常见数据库格式的驱动程序集，包括 dBASE、Paradox、FoxPro、Access 和许多 ODBC 数据源、SQL 连结驱动程序，有些 Delphi 的版本还支持象 Oracle、Sybase、Informix、DB2、SQL Server 和 InterBase 这样的数据库服务器。

Delphi 中含有通过 InterBase Express (IBX) 访问数据的部件，而 IBX 提供了对 InterBase 高级特征的访问，以及为 InterBase 5.5 及其以前版本提供的最高性能的部件接口。

IBX 基于对 Delphi 数据存取部件结构的定制，并且集成了 Data Module Designer，IBX 与 Delphi 的 data-aware 部件库兼容，因此不需要 BDE 的支持。

在 Delphi 中可以利用窗体设计器在设计状态下创建数据表，首先利用对象监视器创建字段定义，然后在表部件上单击鼠标右键，选择创建表。

还有些 Delphi 的版本包括了应用 ActiveX Data Objects (ADO) 连结数据库的部件，ADO 是 Microsoft 的连结各种数据源的高级接口，包括关系型数据库、非关系型数据库、email 和文件系统、文本和图形，以及自定义交易对象。

此外，Delphi 5.0 还提供了表 1-1 中的一些用于数据库应用程序开发的管理工具。

表 1-1 Delphi 5.0 提供的数据库管理工具

工 具	说 明
BDE Administrator	使用 BDE Administrator (BDEAdmin.exe) 可以配置 BDE 驱动程序和设置别名，而别名由 VCL 中的数据装置控件使用，用于连结数据库
SQL Explorer (Database Explorer)	SQL Explorer (DBExplor.exe) 用于浏览和编辑数据库。可以通过它创建数据库别名、查看信息、执行 SQL 查询，以及维护数据字典和属性设置
Database Desktop	Database Desktop (DBD32.exe) 用来创建、查看、编辑多种格式的 Paradox 和 dBase 数据表
Data Dictionary	Data Dictionary 提供了可定制的独立于应用程序的存储区域，可以创建描述数据内容和外观的扩展字段属性。Data Dictionary 可存在于远程服务器中作为附加信息共享

1.2.2 Delphi 5.0 应用概述

利用 Delphi 开发平台可以编写 Windows GUI 应用程序、控制台应用、服务应用、动态链接库 (DLLs) 以及包 (packages——Delphi 使用的特殊 DLL) 等一些程序。

1 应用程序和服务

使用 Delphi 可以容易的编写分布式的应用程序，包括客户/服务器 (client/server) 系统、多层 (multi-tiered) 系统以及基于 Web 的系统。除了支持诸如 COM 在内的一些标准及一系列的 Internet 部件之外，Delphi 的某些版本还提供了 CORBA 开发的一些扩展工具。

2 DLLs

动态链接库 (DLLs) 已编译的程序模块，由其他应用程序或 DLL 调用。DLL 包含了可共享的代码或资源，可以为多个应用程序使用。

3 自定义部件和包

包是由 Delphi 应用程序或 IDE 使用的特殊动态链接库。包可以有多种使用方式，其最通常的目的就是包装 Delphi 的部件，实际上 IDE 中安装的各种部件都必须编译为包。

Delphi 中的部件都预装在 IDE 中，并且提供了一系列的功能，足以满足大多数应用程序开发的需要。在开发过程中，可能很长的事件不需要安装新的部件，但是，为了处理应用

程序中的特殊问题，或者封装特定类型的行为，需要经常定制部件。定制部件补充 VCL 有利于提高代码的再利用和应用程序的一致性。可以得到许多第三方提供的控件，Delphi 也提供了一个 New Component 向导，使开发者自己可以很容易的创建和安装部件。

4 框架 (Frames)

框架(TFrame 类)，例如窗体，是其它部件的容器，从某种意义上而言，框架更象一个自定义的部件而非一个窗体。框架可以保存在窗体模板中以便再利用，并且可以嵌套在窗体、框架或其它容器对象中。

在创建和保存了框架之后，仍然可以具有对应的单元的功能，并且继承了对所包含的部件的改动。当一个框架嵌入另一个框架或窗体之中时，仍然会继承这些改动。

5 ActiveX 技术与 COM、OLE 和 ActiveX 控件

Delphi 5.0 支持 Microsoft 的 COM(Component Object Model)标准，并且提供便捷的 ActiveX 控件创建向导。ActiveX 控件的样板安装在部件模板的 ActiveX 标签页中，而在部件模板的 Servers 标签页中有大量的 COM 服务器部件，编程者可以象使用 VCL 部件一样方便的使用这些样板。例如，可以在窗体中放置一个 Microsoft Word 部件，在应用程序中调出 Microsoft Word 实例。

6 类型库 (Type libraries)

类型库是一些包含数据类型、接口、成员函数和由 ActiveX 控件或服务输出的对象类的文件。在 COM 应用程序或 ActiveX 库中通过包含类型库，将其中的实体信息提供给其它应用程序或编程工具。Delphi 5.0 提供了一个 Type Library 编辑器用于创建和维护类型库。

7 散发应用程序

当散发应用程序时，首先应当确保为用户提供了所有必需的文件，包括可执行文件、DLLs、包和 BDE 驱动程序。为了简化这一过程，Delphi 5.0 自带了一个 InstallShield Express 的特殊版本，可以作为安装程序开发的工具。

8 应用程序的国际化

Delphi 5.0 提供了许多使应用程序国际化或本地化的功能，贯穿全部 VCL 都提供了对输入法编辑器(IMEs)和扩展字符集的支持，还有 Resource DLL 向导可以轻松地使项目本地化。为了能充分从这些由 Delphi 5.0 提供的特征中受益，建议开发者从项目的开发之初，首先考虑国际化问题。

Delphi 的某些版本中有一个集成翻译环境 (Integrated Translation Environment, 简称 ITE)，是一套软件本地化、同时支持多区域开发的工具，它集成在 IDE 之中，利用它，可以用一个项目文件管理应用程序不同区域的多个版本。

ITE 包括表 1-2 中所列的三个工具。

表 1-2 ITE 包括的工具

工 具	说 明
翻译管理器 (Translation Manager):	查看和编辑翻译的资源
翻译仓库 (Translation Repository):	共享的翻译数据库
资源 DLL 向导 (Resource DLL wizard):	创建和管理资源 DLLs 的向导

1.3 应用程序中的窗体

在 Windows 中, 用户界面的大多数元素都是窗口。因此, 在 Delphi 中, 大多数组件也基于窗口, 但不是全部。当然, 这不是用户可以察觉到的, 区别不很明显, 所以读者应该仔细研究下面的定义, 然后进行深入的讨论。

从用户的角度来看, 窗口是屏幕上被一个边框围住的部分, 它有标题而且通常有系统菜单, 可以在屏幕上移动、关闭以及最大化或最小化。窗口可在屏幕上或在其它窗口中移动, 如在 MDI 应用程序中。这些用户窗口基本上分为两类: 主窗口与对话框。

从技术上讲, 窗口是 Windows 系统内存区的入口, 在屏幕上通常对应着一个可视化元素, 并有一些相关代码。Windows 系统库中, 有一个库包含着每个应用程序所创建的所有窗口的列表, 并赋予每个窗口一个唯一值(通常叫句柄)。其中一些窗口就象用户看到的一样(见上面的第一个定义), 有的作用与控件或可视化组件相同, 有些由系统临时创建(例如, 显示一个下拉式菜单), 而甚至有些由应用程序创建但保持隐藏状态。

所有窗口的一般特点是, Windows 系统可以识别它们, 并为它们的行为引用函数; 每当在系统中有事发生, 通知消息就会发往相应的窗口, 该窗口通过执行一些代码做出响应。实际上, 系统的每个窗口都有一个相应的函数(通常叫作窗口过程), 它用来处理窗口收到的各种消息。

在 Delphi 应用程序中, 这些低级消息被系统转换为事件。但有时, 在一些范例中, 可直接在窗体内处理低级消息。Delphi 允许用户在高于系统的级别上工作, 使得应用程序的开发变得更为简单了。

用来存储所有已建窗口列表的 Windows 系统的内存区是有限的, 所以创建太多的窗口会减少系统资源。Windows 3.1 对系统中可使用的窗口数目有严格的限制, 在 Windows 95 与 98 中, 该限制已经大大地放宽了。在 Windows NT 中甚至不存在限制。一旦在系统中存放了太多的窗口(也包括所有的控件与隐藏窗口), 用户就不能再建一个窗口了。有时会妨碍大多数应用程序的正常运行。这就是为什么在 Delphi 中有大量非窗口化组件包括标签的原因。该方法使用户节省了大量系统内存, 而不用担心甚至不用考虑内存资源的问题。

带着这些一般的定义, 让我们返回到 Delphi 来理解窗体的作用。从用户的角度来看, 窗体可以代表窗口, 可用来构建主窗口, MDI 窗口, 以及对话框。它们的行为主要依赖于它们的代码, 但也依赖于两个非常重要的属性, BroderStyle 和 FormStyle。其它很多组件都基于窗口, 但只是用户眼中定义窗口的窗体。其它窗口化的组件或控件, 根据窗口的第二个