

Delphi
实用
编程

Delphi

实用编程

窦万峰 王保保 编著



新 万 变 不 离 其 宗

西北
工业



西北工业大学出版社

实用版

(陕)新登字 009 号

【内容简介】 Delphi 是 Borland 公司在 1995 年推出的可视化的 Windows 应用程序开发系统。它以容易使用的可视化开发方式、高效的编译代码、高性能的客户/服务器数据访问和强有力的异常处理机制等特点,得到了软件开发人员的喜爱,并获得美国最佳可视化 Windows 应用程序开发工具。本书讲述了 Delphi 的基础知识和基本使用方法以及如何使用 Delphi 开发 Windows 应用软件技术。书中列举了大量的开发实例,安排由简单到复杂,讲述由浅入深,旨在让读者快速掌握 Delphi 编程的要领和一些实用开发技术等。

全书共分为十一章,主要包括 Delphi 编程基本知识、Object Pascal 应用、Delphi 菜单编程技术、图形应用编程、多媒体应用以及 Delphi 数据库基本编程技术等内容。另外,书后附有 Delphi 组件列表,以便广大读者查阅。

本书资料丰富,内容翔实,开发实例多,具有易学易用、简洁明了的特点。本书既适合于 Delphi 初学者学习和使用,也可作为大专院校广大师生和软件开发人员的必备参考书。

JS202/12

Delphi 实用编程

察万峰 王保保 编著

责任编辑 何格夫

责任校对 潘玉浩

*

©1998 西北工业大学出版社出版发行

(710072 西安市友谊西路 127 号 电话 8493844)

全国各地新华书店经销

陕西省富平印刷厂印装

ISBN 7-5612-1000-0/TP·138

*

开本:787×1092 毫米 1/16 印张:12.875 字数:306 千字

1998 年 1 月第 1 版 1998 年 1 月第 1 次印刷

印数:1—5 000 册 定价:18.00 元

购买本社出版的图书,如有缺页、错页的,本社发行部负责调换。

前 言

随着可视化技术的发展和可视化编程与软件开发工具的日趋成熟, Borland 公司推出了一个完全导向的窗体编制工具——Delphi, 使许多软件开发者惊喜不已。不断地学习和使用新的应用软件和编程技术, 开发应用程序, 一直是我喜欢做的工作。过去用 C 语言或者 Pascal 语言编写一个窗体程序, 需要花去我许多的时间和精力, 编写上千条语句。然而, 使用 Delphi 开发, 则代码量大大减少, 并能在短短的时间里得到满意的结果。

在稍稍了解了 Delphi 的强大功能后, 我为它那妙不可言的完全导向的编程感到欣喜若狂。过去在编写一个应用软件的界面时, 常常要耗费我许多的时间和精力, 现在 Delphi 却使我的烦恼一扫而光, 不但变得轻松愉快、美不可言, 而且用它能开发出更为丰富、漂亮的用户界面。随着我对 Delphi 的深入了解, 我对 Delphi 提供的强大功能赞叹不已。

Delphi 结合了多方面的技术, 包括可视化技术、面向对象技术以及数据库技术等。它可用于开发各种 Windows 应用软件, 开发的软件运行速度快, 汉字调用容易。Delphi 是在 Object Pascal 语言基础上开发的可视化编程工具软件。Pascal 语言具有语法严格, 结构清晰的特点, 因此得到了十分广泛的应用, 尤其在可视化技术应用方面, 它有着极其重要的优点。

本书是一本 Delphi 从入门到精通的指导书。全书共分为十一章, 主要包括 Delphi 编程基本知识、Object Pascal 应用编程、创建窗体表单、Delphi 菜单编程技术、图形应用编程、文件操作技术、多媒体应用以及 Delphi 数据库基本编程技术等内容。书中使用大量的编程实例, 循序渐进地介绍了 Delphi 编程的基本原理和实际应用, 旨在使读者迅速掌握 Delphi 的使用、编程和它的本质。

本书第一至第七章由窦万峰同志编写, 第八至第十一章由王保保同志编写, 李华同志对本书的编写提出一些宝贵的意见, 在此表示我们最诚挚的谢意。书中所列实例, 由易到难, 并具有代表性, 可以使读者起到举一反三的作用。

本书也是在出版社同仁的共同努力下才得以顺利出版的, 在此对他们表示衷心感谢。通过对本书的阅读, 读者一定能够从中获得很大的裨益, 这也是我们最大的愿望。

本书适用于广大计算机语言爱好者及软件开发人员学习和使用, 也可作为 Delphi 学习的培训教材。

由于时间和水平有限, 不妥之处在所难免, 尚祈广大读者不吝赐教, 不胜感激之至。

作 者

1997 年 9 月

目 录

| | |
|-----------------------------------|----|
| 第一章 Delphi 概述 | 1 |
| 1.1 Delphi 的特点 | 1 |
| 1.2 Delphi 的运行环境和安装 | 3 |
| 1.2.1 安装环境要求 | 3 |
| 1.2.2 安装过程指南 | 3 |
| 1.3 Delphi 的主界面 | 6 |
| 1.4 Delphi 的属性 | 9 |
| 1.5 Delphi 的事件 | 10 |
| 1.6 Delphi 的窗体与表单文件 | 12 |
| 1.7 Delphi 的模板 | 13 |
| 1.8 Delphi 专家 | 15 |
| 1.9 一个 Delphi 编程的例子 | 16 |
| 第二章 Delphi 编辑环境及调试工具 | 19 |
| 2.1 Delphi 主屏幕组成 | 19 |
| 2.2 文件管理菜单 | 20 |
| 2.3 编辑功能 | 22 |
| 2.4 搜索功能 | 24 |
| 2.5 察看功能 | 25 |
| 2.6 选项功能 | 27 |
| 2.7 工具菜单 | 29 |
| 2.8 编译运行及调试 | 30 |
| 第三章 Delphi 窗体设计 | 33 |
| 3.1 建立窗体表单 | 33 |
| 3.2 单元文件结构 | 35 |
| 3.3 Uses 子句 | 36 |
| 3.4 单元设计 | 38 |
| 3.5 窗体表单属性 | 39 |
| 3.6 窗体表单事件 | 43 |

| | |
|----------------------------|-----------|
| 3.7 多窗体调用 | 46 |
| 第四章 Delphi 编程基础 | 48 |
| 4.1 Delphi 语句 | 48 |
| 4.1.1 Delphi 语句类型 | 48 |
| 4.1.2 属性设置 | 49 |
| 4.2 Delphi 的对象 | 50 |
| 4.2.1 面向对象编程基本术语 | 50 |
| 4.2.2 可视组件库对象 | 51 |
| 4.3 Object Browser(对象浏览器) | 53 |
| 4.4 Object Pascal 块和保留字 | 54 |
| 4.4.1 Object Pascal 块 | 54 |
| 4.4.2 保留字 | 56 |
| 4.5 Delphi 变量 | 57 |
| 4.6 Delphi 中数和运算符的使用 | 59 |
| 4.6.1 使用整数和运算符 | 59 |
| 4.6.2 实数类型及函数 | 69 |
| 4.7 函数与过程 | 70 |
| 4.7.1 内建函数 | 71 |
| 4.7.2 Windows API 函数 | 72 |
| 4.7.3 用户定义的函数和过程 | 72 |
| 4.8 数组与记录 | 73 |
| 4.8.1 数组 | 73 |
| 4.8.2 记录 | 75 |
| 第五章 Delphi 控制语句及其编程 | 77 |
| 5.1 循环语句 | 77 |
| 5.1.1 For 循环 | 77 |
| 5.1.2 While 循环和 Repeat 循环 | 78 |
| 5.2 条件语句 | 79 |
| 5.2.1 布尔数据类型 | 80 |
| 5.2.2 if-then 语句 | 80 |
| 5.2.3 if-then-else 格式 | 84 |
| 5.3 Case 语句 | 85 |
| 5.4 用户定义数据类型 | 87 |
| 5.4.1 枚举类型 | 87 |
| 5.4.2 集合类型 | 88 |

| | |
|-------------------------------------|-----|
| 第六章 常用组件使用 | 89 |
| 6.1 文本相关组件 | 89 |
| 6.1.1 标签(Label)组件 | 89 |
| 6.1.2 编辑框(Edit)组件 | 90 |
| 6.1.3 Memo 组件 | 90 |
| 6.2 Button 按钮与 BitBn 位按钮 | 91 |
| 6.2.1 Button 按钮 | 91 |
| 6.2.2 BitBn 位按钮 | 92 |
| 6.3 组件使用例程 | 93 |
| 6.4 ListBox 列表框与 ComboBox 组合框 | 95 |
| 6.5 单选按钮与复选按钮 | 98 |
| 6.6 GroupBox 与 RadioGroup | 98 |
| 6.7 Timer 组件 | 99 |
| 第七章 菜单窗体设计与编程 | 104 |
| 7.1 主菜单设计 | 104 |
| 7.2 弹出式菜单设计 | 106 |
| 7.3 添加面板和加速条 | 107 |
| 7.4 对话框组件介绍 | 108 |
| 7.5 对话框组件使用 | 113 |
| 7.6 一个菜单编程的例子 | 116 |
| 7.7 自动生成菜单 | 122 |
| 第八章 Delphi 图形与图像组件 | 137 |
| 8.1 执行阶段的绘图 | 137 |
| 8.1.1 Canvas 对象的基本属性 | 137 |
| 8.1.2 画笔(Pen) | 138 |
| 8.1.3 画刷(Brush) | 139 |
| 8.1.4 像素(Pixels) | 140 |
| 8.2 使用 Canvas 对象绘图 | 140 |
| 8.3 简单图形的设计 | 144 |
| 8.4 图形组件及使用 | 144 |
| 8.5 图像格式的认识 | 147 |
| 8.5.1 图像对象之间的关系 | 147 |
| 8.5.2 图形对象的属性 | 147 |
| 8.5.3 图像的获取 | 148 |

| | |
|---------------------------------|-----|
| 第九章 文件操作 | 149 |
| 9.1 文件操作命令 | 149 |
| 9.2 使用 System 组件板上的文件相关组件 | 151 |
| 9.2.1 文件相关组件 | 151 |
| 9.2.2 文件系统组件的使用 | 152 |
| 9.3 文本文件的处理 | 153 |
| 9.4 格式化文件的处理 | 155 |
| 9.5 随机文件的处理 | 157 |
| 9.6 字符串的处理 | 158 |
| 9.7 出错处理和异常处理 | 161 |
| 9.7.1 代码块的保护 | 161 |
| 9.7.2 异常的分类 | 162 |
| 9.7.3 用户自定义异常 | 164 |
| 第十章 多媒体及其应用 | 165 |
| 10.1 多媒体概览..... | 165 |
| 10.2 MediaPlayer 组件 | 165 |
| 10.2.1 MediaPlayer 的属性 | 166 |
| 10.2.2 MediaPlayer 的事件 | 167 |
| 10.3 一个简单的例程..... | 168 |
| 第十一章 数据库应用程序设计 | 170 |
| 11.1 关系数据库概述..... | 170 |
| 11.1.1 数据库和数据字典..... | 170 |
| 11.1.2 整体框架功能..... | 170 |
| 11.1.3 Delphi 的数据库工具 | 171 |
| 11.2 数据库配置应用..... | 172 |
| 11.2.1 Driver 页面..... | 173 |
| 11.2.2 Alias 页面 | 174 |
| 11.3 数据库应用程序开发..... | 175 |
| 11.3.1 数据库选择..... | 175 |
| 11.3.2 数据库表格设计..... | 176 |
| 11.3.3 数据库程序设计..... | 176 |
| 11.4 Database Desktop | 176 |
| 11.5 数据库组件和一个简单的例子..... | 180 |
| 11.5.1 数据库组件..... | 180 |
| 11.5.2 一个图片显示..... | 181 |
| 11.6 开发数据库应用程序..... | 183 |

| | |
|-------------------------------|------------|
| 11.6.1 组件 Table 和 Query | 184 |
| 11.6.2 DataSource 组件 | 190 |
| 11.6.3 Data Control 组件板 | 190 |
| 附录 Delphi 组件简介 | 192 |

第一章 Delphi 概述

1.1 Delphi 的特点

Delphi 是 Borland 公司在 1995 年推出的一个完全导向的可视化系统开发环境的工具软件,被评为美国最优秀软件之一。它具有功能强大、运行速度快、易于学习和使用、开发迅速等特点,一经推出就受到了广大用户的喜爱。Delphi 结合了可视化技术、面向对象技术、数据库技术等多种先进的软件编程技术和思想,使其成为创建功能丰富、界面友好的 Windows 应用软件的有力武器之一。

Delphi 不会受限于任何东西,它的许多性能和优点可在以下几个方面体现出来,

1. 优美而强大的可视化集成开发环境

Delphi 是一个运行在 Windows 环境下的可视化编程工具软件,可以用来创建各种各样的 Windows 应用程序。Delphi 提供了易于使用的图形化工具供软件开发人员创建程序中的可视化部分。也就是说,开发者只需单击一下鼠标就可以把选择的按钮和组件粘贴到一个窗体之中,而且开发者可以按照自己的喜好及用户要求安排窗体的外观,使之恰到好处地调整。如果没有提供诸如此类的工具,那么程序的可视化部分只能全部以代码的形式由程序人员编写出来。

如果用户曾经编写过 Windows 应用程序,就会知道一个 Windows 应用程序的可视化设计到其最后的成功是多么地不容易! Delphi 提供了一套可视化设计工具,通过它们,软件开发人员可以很容易地和及时地观察窗体设计中的动态变化。较之传统的编程方式,这个过程类似于桌面设计或叫动态设计。它可容易地反复调整窗体的外观,即设计出窗体表单和外观布局,然后运行一下并观察其外观,随后返回并对不满意的地方作适当的修改即可。

的确,Delphi 的可视化集成开发环境大大地节约了程序开发人员的时间和精力,使得开发人员把精力集中到窗体动态调用技巧和程序算法设计的方面,从而加速了软件开发进程。

关于 Delphi 的可视化集成开发环境将在第二章作详细的讨论。

2. 强大而严格的底层编程语言

Delphi 所采用的底层编程语言是 Borland 公司的 Object Pascal 版本。Borland 公司的 Object Pascal 语言是一种相当简洁的编程语言。其语法严格,结构清晰,阅读性强,并且它还是在软件设计上使人们形成一种良好编程习惯的语言。Object Pascal 的语言风格可能更适合于可视化的程序设计,比如在 Delphi 中,当双击窗体中的某个组件,程序就会马上跳到单元(代码编写)中的代码编写处,设计人员就可以在此写入该组件要完成的动作或状态等,而无须在程序中来回移动和查找定位。Object Pascal 语言的严格形式对于 Delphi 的编程实现是非

常适合和方便的,几乎不受任何限制,而且编译后的程序代码精练,运行速度快。

有关 Object Pascal 语言的 Delphi 编程实现将在第四、五章详述。

3. 能够创建真正独立的可执行文件及动态链接库(DLLs)

Dynamic Link Libraries(动态链接库 DLLs)使设计人员有可能创建经过编译的程序代码库,这种程序代码库可以为许多不同的应用程序所使用。使用动态链接库可以使程序的维护工作变得更加轻松,使各种琐碎程序的实现细节有可能都被放入一个带有过程的库中,而这些过程则可以通过非常一般化的表达来加以调用。

通常设计人员希望能够从外部的动态链接库中调用函数和过程,以利用现有的代码库。这种出于模块化、重用性及维护性的目的,会使设计人员将自己的应用程序中经常要被应用的部分编写成 DLL,而不是可执行文件。此外,由于多个应用程序可以访问共同的目录,将代码编译为 DLL,从而有节约空间的优点。

Delphi 提供了两种方式访问外部的例程:

- 在程序中使用一个外部声明。
- 将 Windows 的 GetProcAddress 和 LoadLibrary API 过程结合起来使用,以便装载指向 DLL 过程的指针。

Delphi 提供的上述两种容易的外部调用方式,既大大地方便了设计人员的操作,又降低了程序设计的难度。Delphi 的这种动态输入功能可节省内存空间,而且程序可以在不知道任何硬件特性的情况下,使用其设备驱动程序的 DLLs。即使程序调用中未找到一个 DLL,它也可以继续运行。

4. 极为方便的汉字加载能力

Delphi 虽然是西文环境下的工具软件,但使用 Delphi 编写中文界面是非常方便的。设计人员只需简单地借助于 Windows 中的汉字库和输入法,就可以直接地向窗体或窗体的组件中加入所需的任意汉字。这种加注汉字的方法既可以在程序设计时通过窗体或组件的属性加入,也可在运行时的程序代码中加入,且不影响程序运行的速度。

5. 一流的代码编辑工具

Delphi 的代码编辑工具是非常友好和清晰的。在代码编辑器中,使用者可以设置关键字的颜色,使得程序代码清晰明快,修改方便。

Delphi 的代码编辑器始终在监视当前窗体的变化,并能及时地反映设计者的修改。这种动态跟踪减少了设计人员因窗体的变化而对程序代码的修改。设计人员只对窗体进行操作,无须对代码进行修改,因为代码编辑器会自动进行修改。

Delphi 的代码编辑器允许使用者打开多个代码文件,并放在不同的页上,这样使使用者很方便地在各单元之间切换,并完成程序的编辑、剪切、复制、修改等工作。

另外,Delphi 的项目管理和资源管理也允许用户在窗体和单元之间切换。

6. 具有创建新的组件和模板来扩展 Delphi 的能力

Delphi 提供了 Visual Component Library(动态组件库,简称为 VCL)和定制并为 Delphi Component Palette(Delphi 组件模板)增加页面功能。VCL 是指构成 Delphi 组件结构基础的对象类层次体系。VCL 包括了可视化组件、非可视化组件、不能作为组件的类以及一些不能视为对象的通用函数。

由于 Delphi 是设计成模块化、可重用的开发工具,利用这些特点可使设计人员受益匪浅。

把自己认为可重用的程序或者一些应用程序,以作模板的形式保存成表单,以动态链接库的形式创建 Delphi 的组件或编译设计人员的应用程序,这将是非常有用的。

7. 响应几乎所有 Windows 消息的能力

Delphi 通过为一个应用程序中的所有组件类型注册一个名为 MainWndproc 的方法作为窗体过程来处理消息流跟踪。任何时候,你的应用程序做任何事情时(或者用户使用你的应用程序做任何事情时),都要启动一个 Windows 消息。移动鼠标或单击一个按键就是生成 Windows 消息的例子。

另一方面,Windows 会把各种消息反映到应用程序中,比如一些警告、提示等。Delphi 提供了对 Windows 消息处理的方法,它为每种不同类型的消息都定义了一种特定的记录类型,从而使得每个参数有了便于记忆的名字。

8. 分类继承及封装概念的实现——面向对象编程

Delphi 是以面向对象的方式使用的,在 Delphi 中一切都被视为对象。编写面向对象的程序能够重复使用开发者自己的工作成果,并使程序维护起来更为容易。程序的扩展也变得更加方便,减少了修改所带来的大量错误。

继承性可使设计人员在已有对象类型的基础上创建新的对象,这个新对象将继承其先前对象的所有事件、属性及方法,并且可添加新过程和方法,具有很大的灵活性。此外,不同的对象,虽然它们有着共同的祖先,但它们在同一过程中却做着不同的事情。

以上 Delphi 的优点说明,使用 Delphi 将会使设计人员更加得心应手。相信使用者在经过一段时间的使用后,会对 Delphi 有更为全面的认识。

1.2 Delphi 的运行环境和安装

Delphi 的安装和启动是非常容易的,有经验的读者可略过之。本节带领初学者成功地安装 Delphi,并能使读者对一些设定和概念有一个基本的了解,并认识 Delphi 环境的全貌。

1.2.1 安装环境要求

在了解了 Delphi 的一些优点后,用户可能很想马上试一试 Delphi,以期有一个更为“真实”的认识。首先要检查自己的设备作了哪些配备。

安装 Delphi 的基本配备需求为:

- 中文或西文 Windows 3.1 以上的操作系统,Delphi 2.0 则须在 Windows 95 或 Windows NT 上工作;
- 至少 30 MB 的硬盘空间;
- 80386 以上微机,建议使用 80486 或奔腾系列微机;
- 至少 8 MB 以上的内存空间;
- 建议配备 CD-ROM。

1.2.2 安装过程指南

Delphi 工具软件大约有 75 MB,故一般安装都是从 CD-ROM 上安装。首先运行 Windows,进入 Windows 的文件管理器中,在文件管理器中选取 CD-ROM 中 Delphi 提供的 set-

up.exe 程序,并运行之,则出现如图 1.1 所示的窗体,其要求 Delphi 选择安装方式:

- 完全安装(Complete:约需 75 MB 硬盘空间)
- 最小安装(Minimum:约需 24 MB 硬盘空间)
- 选择安装(Custom:约需 24~75 MB 硬盘空间)

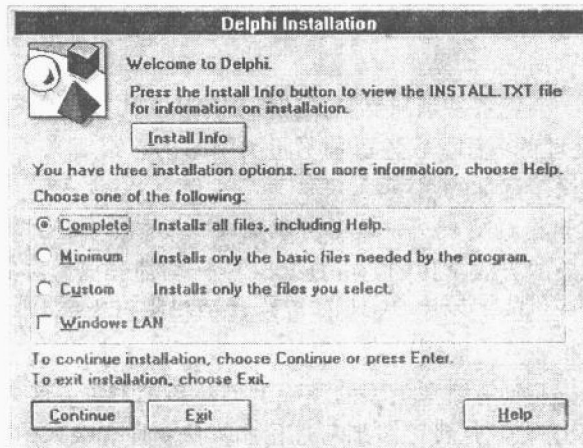


图 1.1 Delphi 安装方式选择

如果选择完全安装,则进入如图 1.2 所示的对话框。该对话框要求设定 Delphi 的安装路径,其五个选项介绍如下:

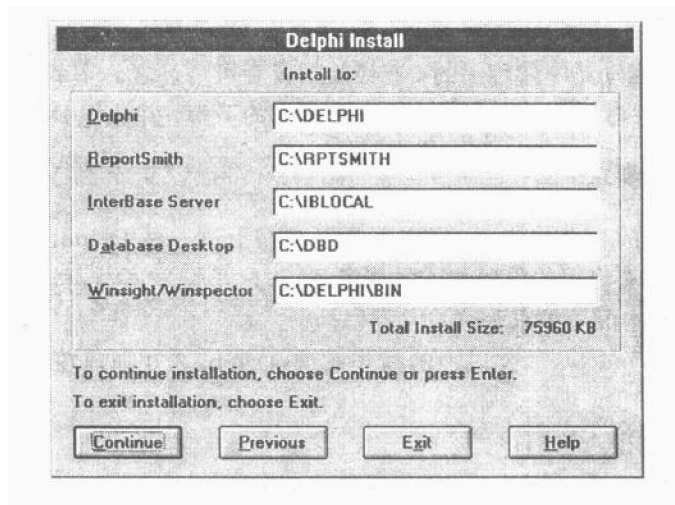


图 1.2 Delphi 安装路径选择

- Delphi——设置 Delphi 安装路径,运行 Delphi 和用户文件存储一般都在该目录下,缺省选择为:C:\Delphi。

- Reportsmith——设置可视化的数据库报表生成器与查询工具路径,可依据数据库文件产生所需要的报表,缺省选择为:C:\Rptsmith。

- InterBase Server——与 ANSI SQL - 92 兼容的关系数据库管理系统,内含 Windows ISQL 与 Server Manager,用以链接进程或网络远程端的数据库,缺省选择为:C:\IBLOCAL (进程)。

- Database Desktop——数据库工具软件,可完成数据表格(Tables)的插入、修改、查询、管理等,缺省选择为:C:\DBD。

- Winsight/Winspector——侦错用工具软件,Winsight 可监看系统的信息传递,Winspector 则在系统发生 UAE(Unrecoverable Application Errors)与 GPE(General Protect Errors)时产生记录报表以供侦错,缺省选择为:C:\Delphi\Bin。

设置完成后,按 Continue 按钮,则出现如图 1.3 所示的对话框,该对话框要求对 Borland 数据库驱动器(BDE)路径进行设定。

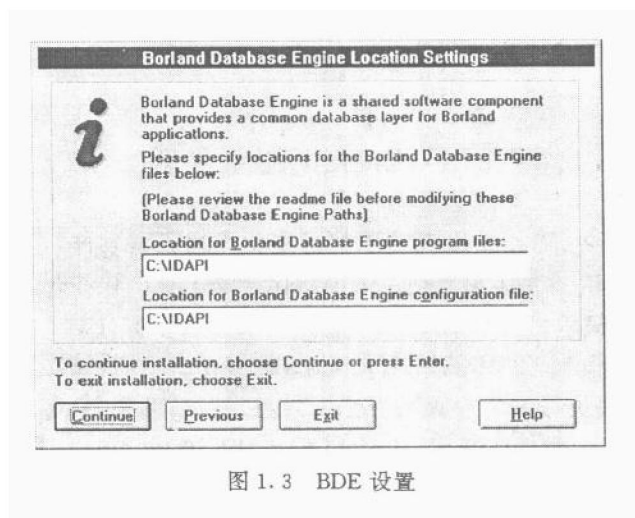


图 1.3 BDE 设置

BDE(Borland Database Engine)是 Borland 公司在 Windows 环境下的高效率数据库引擎,它提供了一个单一且一致的应用程序开发界面(即 IDAPI——Independent Database Application Programming Interface),使设计人员可用同一方式存取各种不同的数据库源(Data Source),如 dBASE,Paradox,Access 等数据库。通过 IDAPI,可以令使用方式不尽相同的各种数据库源提供相同的 API 函数供 dBASE,Paradox,Access 等数据库用户使用。

完成 BDE 设置后,按 Continue 按钮,则出现如图 1.4 所示的安装确认对话框。该对话框要求确认并开始 Delphi 安装,这时屏幕出现一个很漂亮的画面,并显示安装过程。安装完成后,屏幕出现一个对话框,Delphi 提示使用者安装 SHARE.EXE 到 C:\DOS 中,并在 Config.sys 中加入如下行:

```
INSTALL:=C:\DOS\SHARE.EXE/F:4096/L:40
```

SHARE.EXE 是处理文件共享与锁定的 DOS 程序。因为 Windows 是一个多任务的操作系统,同时可以有一个以上的程序一起运行,所以当多个任务对同一个文件进行读、写、改变和删除等操作时,此时就需对该文件的使用作出仲裁,要么共享,要么锁定,这个任务由 SHARE 来完成。

最后,安装程序要求对 Autoexec.bat 文件作修改。

至此,Delphi 的安装全部完毕。注意:重新启动机器后方可进行 Delphi 的工作!



图 1.4 安装确认对话框

1.3 Delphi 的主界面

Delphi 是一个运行在 Windows 环境下的可视化编程工具软件,可以用来创建 Windows 的各种应用程序。Delphi 提供了易于使用的图形化开发工具供软件开发人员创建程序中的可视化部分,使用者只需简单地使用鼠标点取 Delphi 组件页上的任意一个组件,在自己的窗体上任意拖放和设置。Delphi 会根据使用者的要求,能够作出恰到好处的调整。

Delphi 为其开发人员提供了一套可视化工具,通过它们,使用者就可以很容易及时地观察自己的界面设计过程中的任何变化。Delphi 的这种桌面设计方式,能够使使用者快速地设计出符合要求和自己满意的窗体。

图 1.5 所示为 Delphi 的集成开发环境(IDE),其四周由标签形式的控件构成。单击一个标签,就可以得到一个不同的选项页面。在缺省的启动状态下,Delphi 是以四个部分出现的。

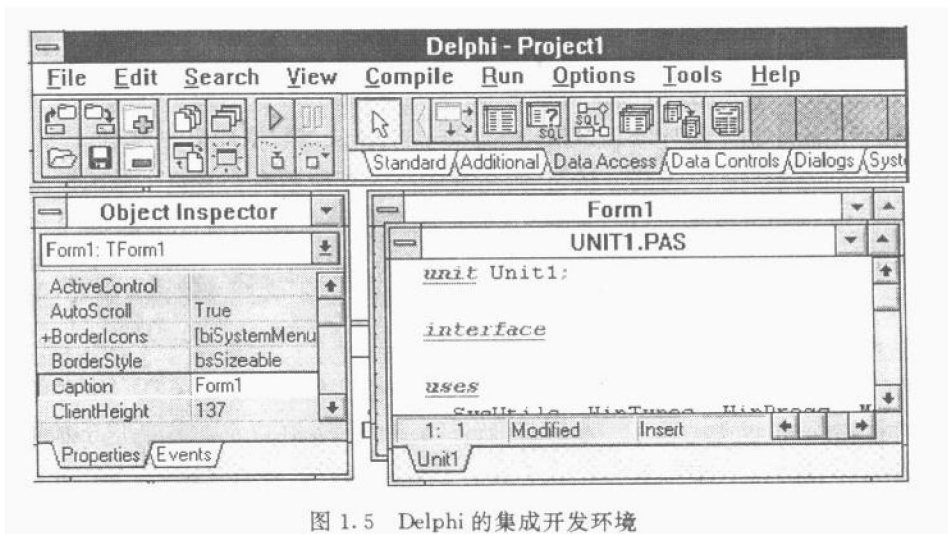


图 1.5 Delphi 的集成开发环境

1. Main Menu(主菜单)

主菜单和一般 Windows 的各种应用程序菜单类似,只是其菜单选项有所不同,功能上有一些差异而已。主菜单主要包括项目文件、单元文件和表单文件等的管理(包括文件打开、存储、关闭、打印等);文件编辑(包括剪切、复制、粘贴和删除等);文本查询(包括查找、替换、移动、查错和符号浏览等);察看(包括工程管理、资源察看、对象察看、对齐操作、断点和堆栈、单元和窗体等);编译、运行与调试;选项设置;在线帮助等。

2. Speedbar(加速条)

加速条是一些菜单命令的快捷按钮,它们可用来快速执行菜单命令。它们包括打开工程项目设计、保存工程项目设计、向工程中加入文件、从列表中选单元、从列表中选择窗体、运行、暂停、打开文件和保存文件、从工程中去掉一个文件、切换窗体/单元、创建新窗体、跟踪以及单步执行等。

3. Object Inspector(对象检查器)

对象检查器如图 1.6 所示。对象检查器有两个页,即属性页和事件页。属性页用于设置窗体和窗体上的组件的属性,比如一个按钮(Button 组件),用对象检查器可设置它的大小、位置、颜色、显示效果、字体、字型、名字隐藏等属性。当选择窗体中的一个组件并在对象检查器中进行其属性设置或修改时,其显示效果立刻会在窗体中反映出来。这即所谓的所见即所得。

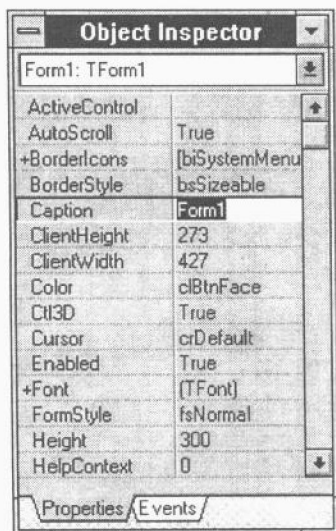


图 1.6 对象检查器的属性页

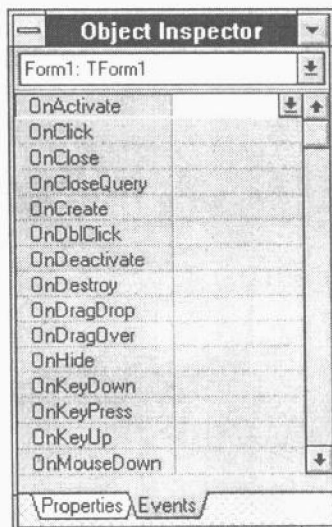


图 1.7 对象检查器的事件页

事件页如图 1.7 所示。窗体和组件的事件用于编写窗体和组件在运行过程中要产生的动作和响应,比如窗体的 OnClose 事件就是用于完成窗体的关闭。

Delphi 的属性和事件将分别在 1.4 和 1.5 节讲述。

4. Component Palette(组件模板)

组件模板在屏幕的右上部,有多个组件页。它们分别为:Standard 页、Additional 页、Data Access 页、Data Controls 页、Dialogs 页、System 页、VBX 页和 Samples 页。每个组件页上都含

有一些不同用途的组件,使用者可根据窗体设计要求进行组件选择和布局。Delphi 提供了非常丰富的组件,几乎可以设计出各种 Windows 应用程序需要的窗体和界面。

若要选取某个页上的组件,首先激活该页,然后加亮要选取的组件,接着用鼠标在窗体合适的位置单击,并放置或拖动组件的位置和大小。如果亲自操作一下,用户就会觉得这是多么地容易啊!

5. 窗体(Form)

在缺省情况下,每次打开 Delphi 时,Delphi 的集成开发环境会自动地放置一个缺省窗体(在屏幕的右下部)。该窗体是一个空白框,上面什么组件也没有,使用者可在上面设计自己的界面。当然,用户可以通过 Delphi 提供的窗体模板来设置不同的缺省窗体。

6. 单元(Unit)

单元是用来编写程序代码的文件。每个窗体都对应着一个单元,用以对窗体和窗体上的组件进行控制或者完成某些动作或处于某种状态。双击一下窗体,就会自动弹出一个单元文件,如图 1.8 所示。单元文件以扩展名 .PAS 结尾,其代码是用 Pascal 语言编写的。学习过 Pascal 语言或其他语言编程的读者,都会看懂单元文件的内容。单元文件主要由 Unit(单元名),Interface(接口),Uses 子句,Type(类型定义),Var(变量定义),Implementation(实现)和 Procedure(过程)等几大部分组成。Delphi 具有自动编写窗体和窗体中组件对象定义和类型及部分变量的代码的功能,从而大大减轻了开发人员的工作量。

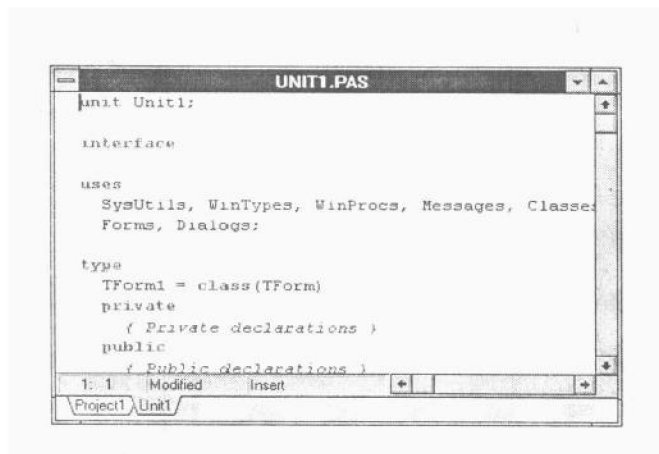


图 1.8 Delphi 的单元文件

Delphi 还提供了工程项目文件(.DPR)和表单文件(.DFM)。工程项目文件用于组织和管理工作多个窗体和单元,这与 C 语言中的工程文件相类似,如图 1.9 所示。

表单文件用于记录窗体和窗体上的组件的各种属性设置,如图 1.10 所示。表单文件是二进制形式的,而非 ASCII 码文件。不过,它们可以以 ASCII 码的形式进行存储,以便修改和重用等。

有关 Delphi 的集成开发环境将在第二章作详细的讨论。

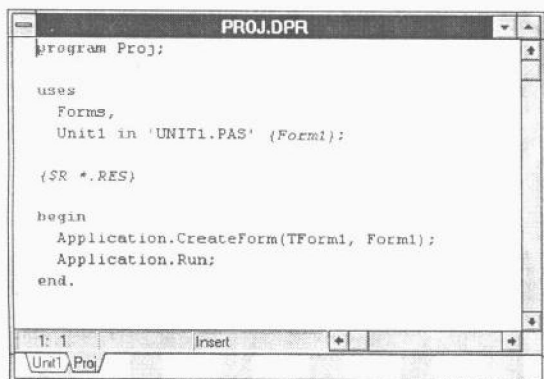


图 1.9 Delphi 的工程项目文件

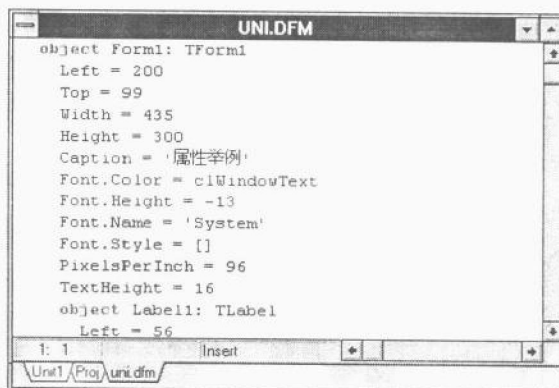


图 1.10 Delphi 的表单文件

1.4 Delphi 的属性

属性由附加到一个对象上的数据构成。使用者既可以在设计阶段利用 Object Inspector 进行对象的属性设置,也可以在程序运行过程中使用代码进行属性设置和改动。这样会使开发的程序功能更加灵活、丰富和强大。一般,一个对象的许多方面都是在设计阶段通过其属性进行设定的。下面用一个简单的例子来说明属性的概念和如何设置属性,并使读者对 Delphi 的环境和编程有一个初步的了解。

首先,在 Delphi 菜单中选择 File | New Form 命令,打开一个新窗体,这时屏幕的右下部出现一个空白窗体。在 Standard 页上选择一个 Label 组件,然后在窗体上单击一下,就有一个 Label 组件出现在窗体上且自动命名为 Label1。照此方法依次选取一个 Edit 组件,另一个 Label 组件和一个 Button 组件。它们在窗体上的布局请参看图 1.11 所示。

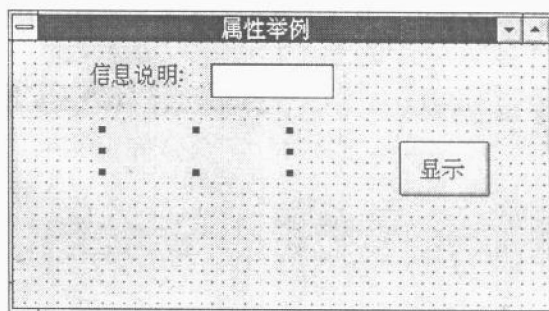


图 1.11 属性设置举例

点取窗体上 Label1 组件,则其上立刻出现暗影,表示选中,同时在 Object Inspector 中出现 Label1 组件的各种属性信息。在 Caption 属性框中键入“输入信息”,注意汉字输入可调用中文 Windows 的汉字输入。其他的组件属性设置同 Label1 组件属性设置方法类似,下面请按表 1.1 所示各组件的属性值设置。