

北京科海培训中心

● 开发与编程
实践系列丛书

张威 编著

Delphi 3.0

数据库开发指南



清华大学出版社

• 开发与编程实践系列丛书

Delphi 3 数据库开发指南

张 威 编著

清华大学出版社

(京)新登字 158 号

内 容 提 要

本书从实用的角度,以通俗的语言,由浅入深地介绍如何使用 Delphi 开发数据库应用程序。

全书共分 13 章,内容包括:Delphi 环境、数据库工具、数据维护、菜单、查询、报表等。此外,还介绍了 Windows 通用对话框、工具栏、集成化调试器和异常处理。书中给出了一些在应用实践中具有参考价值的实例,指导用户快速掌握用 Delphi 开发数据库应用程序的方法。

本书适用于 Delphi 初学者及有一定编程经验的 Delphi 程序员。

版权所有,盗版必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得进入各书店。

书 名:Delphi 3 数据库开发指南

作 者:张威

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

印刷者:北京门头沟胶印厂

发 行:新华书店总店北京科技发行所

开 本:16 印张:16.5 字数:401 千字

版 次:1998 年 10 月第 1 版 1999 年 2 月第 2 次印刷

印 数:5001~10000

书 号:ISBN 7-302-03222-X/TP·1724

定 价:22.00 元

前 言

随着 Windows 的普及,图形用户界面越来越受到人们的青睐。Windows 环境界面友好,操作方便,但也向程序员提出了更高的要求,即必须掌握面向 Windows 的应用程序的开发方法。

过去开发面向 Windows 的应用程序,主要借助于诸如 Turbo Pascal for Windows, C++ for Windows 这样的开发工具。这些工具确实提供了强大的编辑和编译功能,但它们主要是针对专业人员的,每一步操作,如创建窗口、建立对话框、添加按钮等都要靠具体的代码来实现。这样仅在界面设计这一项上,就要耗费程序员大量的时间。

最近几年,兴起了一种高效的程序设计方法,即可视化程序设计方法。相应地也出现了诸如 Visual Basic, Visual C++ 等可视化应用程序开发工具。利用这类工具,程序员可以从繁琐的界面设计工作中解放出来,而将主要精力放在应用程序的功能上。

Delphi 也是一种可视化的编程工具。和 Visual Basic 相比,Delphi 具有如下优点:首先,Delphi 是一个真编译系统。用它开发出的可执行文件效率相当高。其次,Delphi 优化了对数据库的操作,提供了数据库引擎,使用户可以方便地操作本地和网络数据库。

本书从实用的角度出发,结合笔者在用 Delphi 做项目开发时积累的经验与体会编写而成。全书由浅入深,语言通俗易懂,从介绍 Delphi 的开发环境入手,使用户从整体上了解用 Delphi 开发应用程序的过程。然后重点介绍了如何开发数据库应用程序。全书共分 13 章,内容包括数据维护,菜单,查询,报表,使用 Windows 通用对话框,使用工具栏和 CoolBar,使用图形文件,文件操作,集成化调试器以及异常处理,并且所有的程序均在 Delphi 3 集成环境下调试通过。全书适用于 Delphi 编程的初学者和有一定经验的程序员。本书的姊妹篇《Delphi 编程与技巧》,也将于近期出版,该书适用于 Delphi 开发人员。由于编者水平所限,难免有疏漏之处,欢迎广大读者批评指正。

作者

1998 年 9 月

目 录

第 1 章 Delphi 3 概述	(1)
1.1 Delphi 的产生与发展	(1)
1.2 Delphi 3 的特点	(1)
1.3 Delphi 3 的安装	(2)
1.4 Delphi 3 集成开发环境的组成	(3)
1.4.1 加速栏	(4)
1.4.2 组件栏	(5)
1.4.3 窗体	(6)
1.4.4 代码编辑器(Code Editor)	(7)
1.4.5 对象检查器	(7)
1.4.6 主菜单	(8)
1.4.7 弹出式菜单	(10)
第 2 章 项目与用户界面	(11)
2.1 项目	(11)
2.1.1 文件类型	(11)
2.1.2 建立项目	(14)
2.2 查看和操作项目的内容	(15)
2.2.1 更新 Project Manager 内容列表	(15)
2.2.2 获得项目路径和单元信息	(15)
2.2.3 使用 Project Manager 命令	(15)
2.2.4 使用 Project Manager 加速菜单	(16)
2.3 项目组件导航	(17)
2.3.1 在窗体图象和单元源代码之间切换	(17)
2.3.2 将窗体前置	(17)
2.3.3 查看窗体和单元	(17)
2.4 保存项目和单个项目文件	(18)
2.4.1 保存项目	(18)
2.4.2 保存项目的副本	(20)
2.4.3 保存单个文件	(20)
2.4.4 备份整个项目	(21)
2.5 项目的编译、连编和运行	(21)
2.5.1 语法检查	(21)
2.5.2 编译项目	(21)

2.5.3 连编项目	(22)
2.5.4 运行项目	(22)
2.6 主窗体	(23)
2.7 窗体设计	(23)
2.7.1 组件分类	(23)
2.7.2 组件选择	(24)
2.7.3 窗体布局和组件属性	(25)
2.8 为退出按钮添加代码	(30)
2.9 为项目添加其他窗体和单元	(30)
2.10 从项目中移去窗体和单元	(32)
第3章 数据库工具	(33)
3.1 Database Desktop	(33)
3.1.1 设置工作目录	(33)
3.1.2 设置工作目录的别名	(34)
3.1.3 建立表	(35)
3.2 BDE	(37)
3.2.1 BDE 的特点	(37)
3.2.2 使用 BDE	(38)
3.3 Database Explorer	(41)
3.3.1 Database Explorer 环境	(41)
3.3.2 使用 Database Explorer 维护数据库别名	(46)
第4章 数据维护	(47)
4.1 用 Database Form Wizard 创建数据维护窗体	(47)
4.2 定制数据维护窗体	(56)
4.2.1 建立与数据库的联系	(57)
4.2.2 以网格形式显示数据	(58)
4.2.3 使用 DBNavigator 控件	(59)
4.3 备注和图形	(61)
4.3.1 使用 DBMemo 控件	(61)
4.3.2 使用 DBImage 控件	(63)
第5章 菜单	(65)
5.1 建立主菜单	(65)
5.1.1 主菜单的建立方法	(65)
5.1.2 为数据维护窗体添加主菜单	(69)
5.1.3 为主菜单添加代码	(71)
5.2 建立弹出式菜单	(73)
第6章 查询	(76)
6.1 SQL 语言	(76)

6.1.1 SQL 语言基础	(76)
6.1.2 SELECT 语句	(78)
6.2 用 Database Form Wizard 创建查询窗体	(82)
6.3 在数据维护窗体中添加查询功能	(91)
6.4 修改菜单项的代码	(94)
6.5 设置查询条件	(96)
6.5.1 单项查询	(96)
6.5.2 组合查询	(101)
第 7 章 报表	(104)
7.1 用 QuickReport Wizard 生成报表	(104)
7.1.1 生成过程	(104)
7.1.2 设计时预览和打印	(110)
7.2 用手工方式制作报表	(112)
7.2.1 制作过程	(112)
7.2.2 代码设计	(116)
第 8 章 使用 Windows 通用对话框	(117)
8.1 为项目添加通用对话框	(117)
8.2 使用打开文件对话框	(117)
8.3 使用保存文件对话框	(119)
8.3.1 保存现有文件	(119)
8.3.2 保存新的未命名文件	(120)
8.3.3 创建一个备份文件	(121)
8.4 打印文件	(122)
8.4.1 设置打印机	(122)
8.4.2 打印文本	(122)
8.4.3 将打印机用做文本设备	(122)
8.4.4 设置打印机字体	(123)
8.5 打印图形	(124)
8.6 使用字体对话框	(124)
第 9 章 使用工具栏和 CoolBar	(126)
9.1 各种工具栏控件	(126)
9.2 使用面板控件添加工具栏	(126)
9.2.1 向面板上添加加速按钮	(127)
9.2.2 可以切换的加速按钮	(129)
9.3 使用工具栏控件添加工具栏	(129)
9.3.1 向工具栏添加工具按钮	(130)
9.3.2 可以切换的工具按钮	(132)
9.4 添加和配置 CoolBar 控件	(132)

9.5 响应单击	(133)
9.5.1 为单击按钮编写事件处理程序	(133)
9.5.2 为工具按钮指定菜单	(133)
9.6 隐藏和显示工具栏	(134)
第 10 章 使用图形文件	(135)
10.1 从文件中加载图片	(135)
10.2 将图片存入文件	(135)
10.3 使用剪贴板处理图形	(136)
10.3.1 将图形复制到剪贴板	(136)
10.3.2 将图形剪切到剪贴板	(137)
10.3.3 从剪贴板粘贴图形	(137)
第 11 章 文件操作	(139)
11.1 启用和禁用文件	(139)
11.2 删除文件	(140)
11.3 更改文件属性	(140)
11.3.1 读取文件属性	(141)
11.3.2 更改文件属性	(141)
11.3.3 设置文件属性	(141)
11.4 移动、复制和重命名文件	(142)
11.5 运行应用程序	(143)
第 12 章 集成化调试器	(145)
12.1 错误类型	(145)
12.1.1 逻辑错误	(145)
12.1.2 运行错误	(145)
12.2 调试方法	(146)
12.2.1 生成调试信息	(146)
12.2.2 执行应用程序	(147)
12.3 控制程序的运行	(147)
12.3.1 执行到光标所在的位置	(148)
12.3.2 单步执行	(148)
12.3.3 执行到断点	(149)
12.3.4 暂停程序	(149)
12.3.5 重新开始执行程序	(150)
12.4 使用断点	(150)
12.4.1 设置断点	(150)
12.4.2 使用断点	(151)
12.4.3 修改断点属性	(152)
12.4.4 自定义断点和执行点的颜色	(154)

12.5 检查程序的数据值	(154)
12.5.1 监视窗口(Watch List)	(154)
12.5.2 计算和修改窗口	(156)
12.6 查看函数调用	(158)
12.7 处理硬件和语言异常	(159)
第 13 章 异常处理	(160)
13.1 保护代码块	(160)
13.1.1 响应异常	(160)
13.1.2 异常和执行流程	(161)
13.2 保护资源分配	(161)
13.2.1 哪几类资源需要保护	(162)
13.2.2 创建资源保护块	(162)
13.3 处理 RTL 异常	(163)
13.3.1 什么是 RTL 异常	(163)
13.3.2 创建异常处理程序	(164)
13.3.3 处理异常的类	(166)
13.3.4 重新引发异常	(167)
13.4 处理组件异常	(168)
13.5 沉默异常	(168)
13.6 自定义异常	(169)
13.6.1 声明异常对象类型	(169)
13.6.2 引发异常	(170)
附录 A 组件及其常用属性、方法和事件	(171)
附录 B Object Pascal 语言	(215)

第1章 Delphi 3 概述

1.1 Delphi 的产生与发展

在 Windows 产生之后,面向 Windows 的应用程序开发变得越来越重要,随之出现了许多运行于 Windows 环境下的开发工具。例如 Turbo Pascal 5 for Windows、Borland、C++ 4.5、Visual C++ 2.0 等,这些开发工具提供了强大的编辑和编译功能,但它们只能供专业人员使用,因为每一步操作都要靠编写代码来实现,既费时,又容易产生错误。

随后出现的可视化程序设计方法,减轻了开发人员的负担,使软件开发效率得到大幅度提高。Visual Basic 就是这样一种可视化开发工具,使用这类工具,使界面设计变得非常容易,因而受到开发人员的欢迎。但它不是一个真编译系统,运行速度受到一定影响,而且用它开发的软件在发行时必须带上相应的.VBX 和.DLL 文件。针对这种情况,Borland 公司推出了 Delphi 1.0,它是一个真编译系统,开发出的可执行文件的效率比较高,相当于用 C/C++开发的软件的运行速度,而且打包发布也非常方便。此外,它的数据库功能非常强大,提供了 Borland Database Engine(数据库引擎),使用户能够快速方便地管理平台和网络数据库。

继 Delphi 1.0 之后,Borland 公司又推出了 Delphi 2.0,该版本的组件更为丰富,数据库功能也有所增强。现在 Borland 公司又推出了 Delphi 3.0 版本(即 Delphi 3,以后在不引起混淆的情况简称 Delphi),这一版本和前期版本相比,在数据库的体系结构与连接、ActiveX 控件、Web 应用程序、包等方面都作了改进,因而功能更强大,使用也更为灵活和方便,大大提高了应用程序的开发效率。

1.2 Delphi 3 的特点

Delphi 3 是面向对象的可视化编程环境,利用该环境既可以开发系统软件,也可以开发包括数据库软件在内的各种应用程序。Delphi 3 的主要特点有:

- (1) 集成化开发环境。可以在一个集成的环境中完成应用程序的开发、测试和维护。
- (2) 优秀的界面设计功能。Delphi 3 提供了大量的用于设计界面的控件,用它们可以设计出各种优秀的用户界面。而且,Delphi 3 还可以自动生成众多的定制界面,极大地简化了界面设计工作。
- (3) 强大的数据库功能。Delphi 3 提供了许多与数据库相关的控件,用它们可以快速开发功能强大的数据库应用程序。Delphi 3 中内置的数据库引擎(BDE)支持访问 Paradox、Access、Foxpro、dBASE、InFORMix、Oracle、Sybase 和 SQL Server 等数据库。

(4) 支持 Internet/Intranet 开发。Delphi 3 的 Client/Server 版本提供了几个用于开发 Internet 和 Intranet 应用程序的组件, 这些组件可以处理与 Web 服务器的所有通信任务。

(5) 支持 ActiveX 控件。Delphi 3 提供了创建标准 ActiveX 控件的向导。创建的 ActiveX 控件可以用于其他开发工具, 如 C++、Java、Visual Basic 或 PowerBuilder。

1.3 Delphi 3 的安装

和其他 Windows 应用程序一样, Delphi 3 软件包也带有安装程序, 其安装步骤如下:

(1) 启动 Windows 95 或 Windows NT, 关闭所有正在运行的 Windows 程序。然后, 通过资源管理器或运行对话框运行 Delphi 3 CD-ROM 根目录下的 Install.exe 程序, 这时会出现如图 1.1 所示的画面。

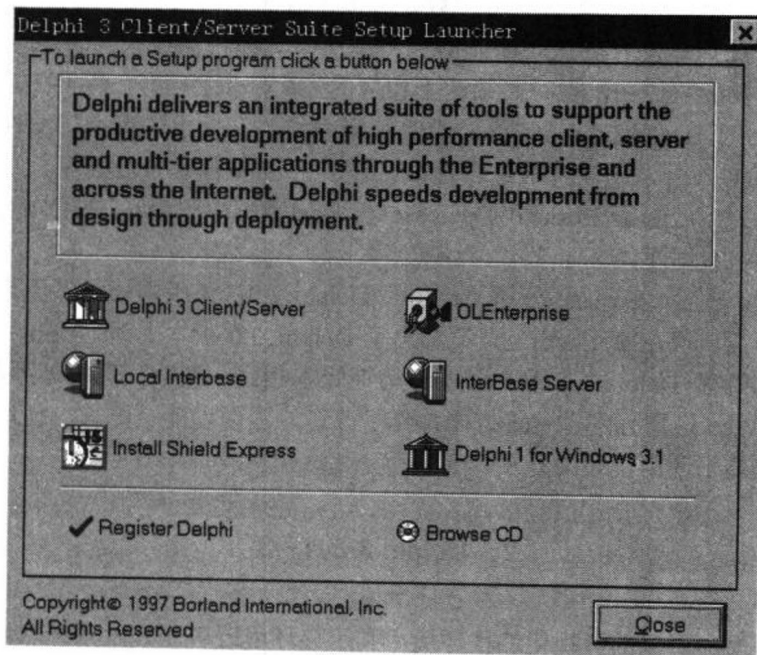


图 1.1 Delphi 3 安装选项

(2) 单击 Delphi 3 Client/Server 之后, 安装程序开始装载安装向导, 引导你完成后面的安装过程。

(3) 安装向导装载完之后, 会出现欢迎屏, 阅读其中的信息, 然后单击 Next 进入协议屏。

(4) 如果你同意协议屏上显示的协议, 则单击 Yes; 否则单击 No 退出安装。

(5) 单击 Yes 之后进入安装信息屏, 其中提示你要卸载以前安装的 Delphi 版本。单击 Next 可出现安装类型屏, 如图 1.2 所示。

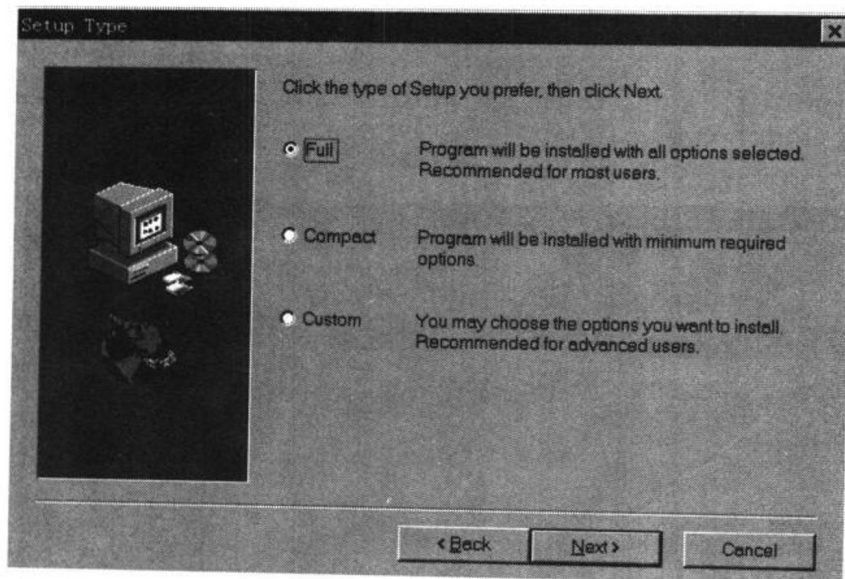


图 1.2 安装类型选择

共有三种可选类型：

- **Full** 完全安装，建议大多数用户用此选项。
- **Compact** 最小安装，适用于硬盘资源紧张的用户。
- **Custom** 可以选择安装需要的内容，仅适用于高级用户。

(6) 选择好安装类型之后，单击 **Next** 进入目录选择屏。在其中选择目录后再次单击 **Next**，然后选择一个存放公共项目文件的目录。

(7) 单击 **Next** 进入组件选择屏，选择 **Sybase CT-Lib Driver** 或 **Sybase DB-Lib Driver**，然后单击 **Next**。

(8) 在出现的程序文件夹选择屏中选择一个文件夹。

(9) 单击 **Next** 进入选择数据集许可屏，阅读其中的内容，然后单击 **Agree**。

(10) 这时屏幕上会列出你的当前设置，如果没有问题，可单击 **Install** 开始拷贝文件。屏幕上会出现进度栏显示安装进度。

(11) 文件拷贝完之后，屏幕上会显示 **Readme** 文件，其中含有一些有关 **Delphi 3** 的重要信息。

(12) 阅读完重要信息之后单击 **Next**，这时屏幕提示是否重新启动计算机，选择重新启动，然后单击 **Finish** 完成安装。

1.4 Delphi 3 集成开发环境的组成

要了解 **Delphi 3** 集成开发环境(IDE: Integrated Development Environment)，首先应启

动 Delphi 3。其启动过程相当简单，只需单击开始菜单程序中的 Borland Delphi 3 文件夹下的 Delphi 3 即可。

启动 Delphi 3 之后，就进入 Delphi 3 集成开发环境，如图 1.3 所示。该环境由加速栏、组件栏、窗体、代码编辑器、对象检查器、主菜单和弹出式菜单组成。

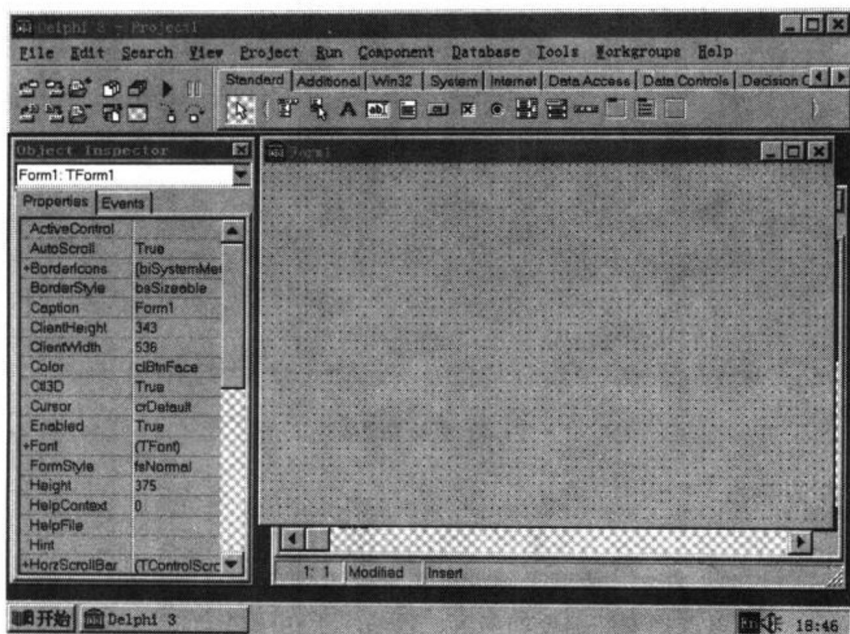


图 1.3 Delphi 3 集成开发环境

1.4.1 加速栏

Delphi 3 加速栏(SpeedBar)位于集成开发环境的左上部，如图 1.4 所示。其中含有执行一般操作和命令的按钮，例如打开项目(Open project)、打开文件(Open file)、保存文件(Save file)、运行(Run)等。

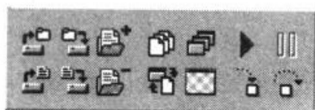


图 1.4 加速栏

要了解某个按钮的功能，可以将鼠标光标移到其上，一会儿就会出现提示。右击加速栏可出现如图 1.5 所示的弹出式菜单。选中 Show Hints 可显示提示，取消选中可关闭提示；选中 Hide 可隐藏加速栏；选中 Help 可显示帮助信息；选中 Properties 可进入 Speedbar Editor 窗口，如图 1.6 所示。要添加命令按钮，可把它们从该窗口拖放到加速栏中；要移去命令按钮，只需把它们拖出加速栏即可。

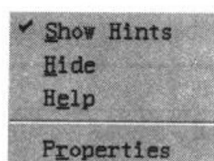


图 1.5 加速栏弹出式菜单

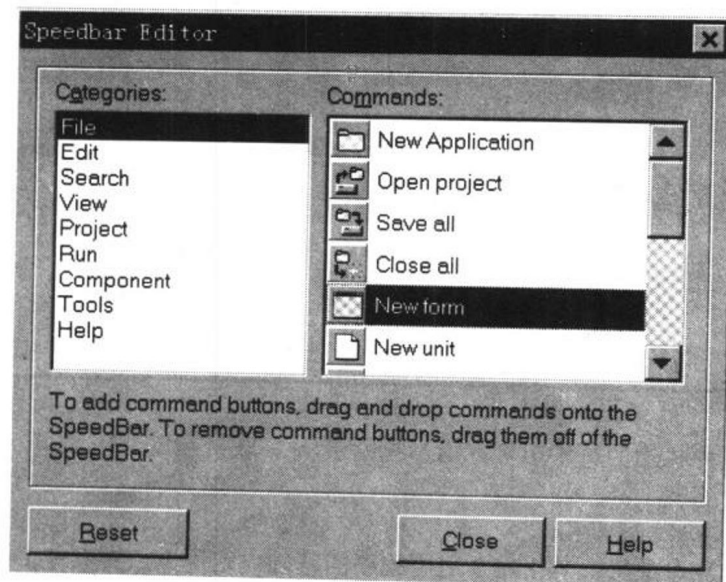


图 1.6 Speedbar Editor 窗口

1.4.2 组件栏

Delphi 3 的组件栏位于集成开发环境的右上部,如图 1.7 所示。其中包含大量的组件,可用于你的应用程序中。你可以根据需要在组件栏上增加、删除和重新安排组件。还可以创建组件模板,并把模板添加到组件栏中。根据功能相似性,组件栏中的组件被分配到多个页面上。表 1.1 列出了缺省页面及其中包含的组件种类。

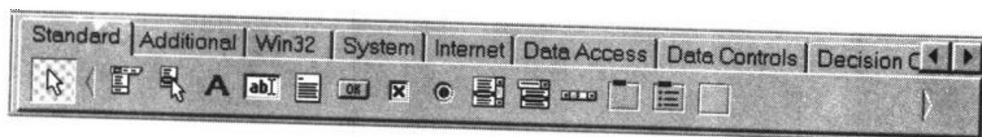


图 1.7 组件栏

表 1.1 Delphi 3 的组件栏页面

页 名	内 容
Standard	标准 Windows 控件、菜单
Additional	附加的常用控件
Win32	较为复杂的控件
System	系统级访问所需的组件和控件，包括定时器、文件系统、多媒体和 DDE
Internet	Internet 组件
Data Access	访问数据库、表的不可见组件
Data Control	可见的数据感知控件
Decision Cube	即时地改变分析焦点的控件，允许你汇总数据库信息，以便从不同角度观察它
QReport	用于轻松创建嵌入式报表的控件
Dialogs	通用对话框
Win3.1	兼容 Delphi1.0 项目的控件
Samples	自定义组件的样本
ActiveX	ActiveX 控件样本

关于缺省组件栏上的每种组件，请参见附录 A 和联机帮助。

1.4.3 窗体

窗体(Form)几乎是每个 Delphi 3 应用程序的基础。它用于放置其他 Delphi 3 组件，通常作为 Windows 应用程序的背景。图 1.8 是典型的 Delphi 3 的空白窗体。

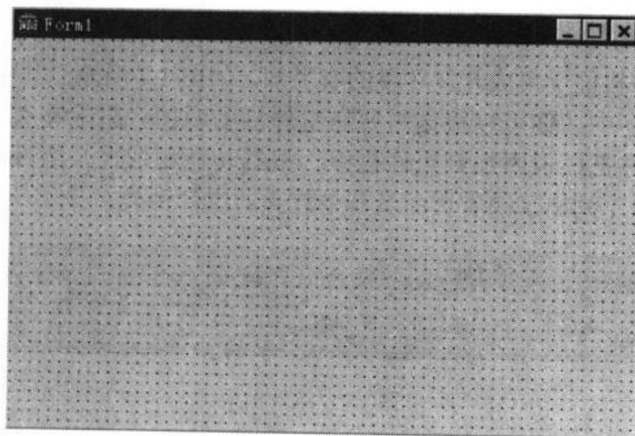


图 1.8 Delphi 3 的空白窗体

Delphi 3 中的窗体具有与任何其他 Windows 95 窗口相同的属性。其左上角有一个控

制菜单；顶部有标题栏；右上角有最小化按钮、最大化按钮和关闭按钮。

1.4.4 代码编辑器(Code Editor)

当你以可视方式设计应用程序的用户界面时，Delphi 3 产生了基础 Pascal 代码。当你向窗体添加组件或修改窗体和组件的属性时，改动会自动出现在源代码中，反之亦然。

你可以使用 Delphi 3 的代码编辑器修改文件，也可以使用任何其他文本编辑器。所做的改动直接反映在可视环境中。Delphi 3 的代码编辑器如图 1.9 所示。

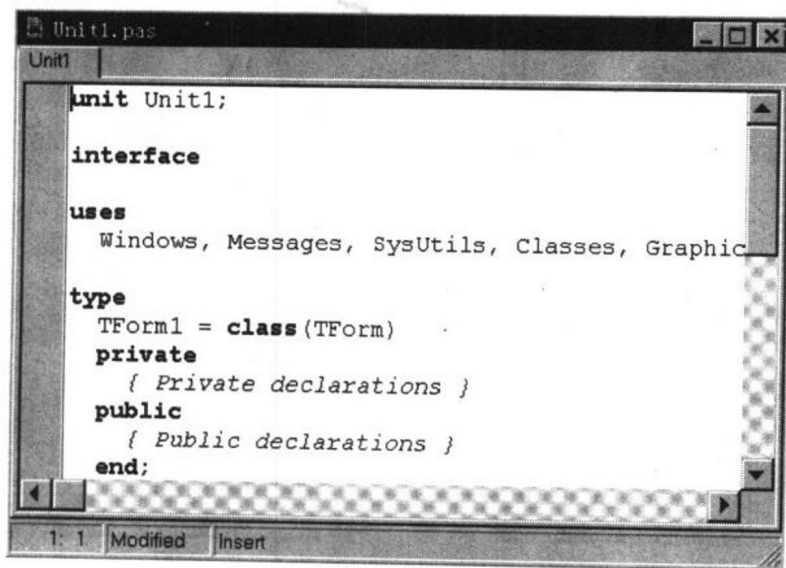


图 1.9 Delphi 3 的代码编辑器

1.4.5 对象检查器

对象检查器(Object Inspector)对你开发 Delphi 3 应用程序是必不可少的。它提供了一种容易使用的界面来改变 Delphi 3 对象的属性，同时也控制对象响应的事件。对象检查器含有以下两个标签。

1. Properties 标签

对象检查器的 Properties 标签如图 1.10 所示，这个标签使我们能够了解和修改一个对象所拥有的属性。单击窗体窗口，然后观察对象检查器的 Properties 标签中出现的属性。其中有些属性旁边带“+”号，表示该属性下面嵌套有子属性。

例如，当选择窗体后，对象检查器中出现的 Font 属性旁边带有“+”号。如果双击 Font 属性，它就被打开并扩展为多个属性，如 Color、Height、Name 和其他属性等。这种格式是改变对象属性的一种简捷而有效的方式。

2. Events 标签

Events 标签是对象检查器的另一组成部分,如图 1.11 所示。这个标签显示所选对象可以响应的所有事件。例如,如果需要在单击某个对象时完成某些特殊的操作,就可以使用该对象的 OnClick 事件来完成这个任务。

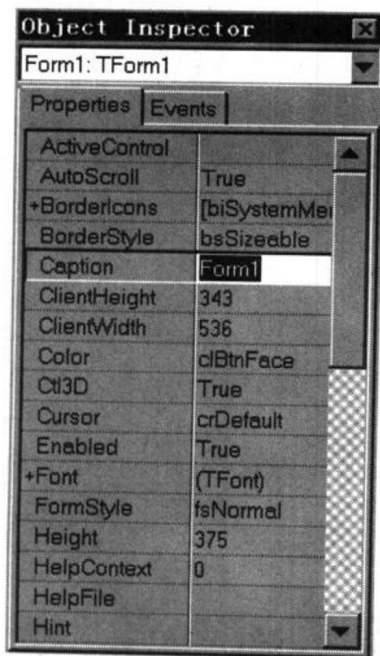


图 1.10 对象检查器的 Properties 标签

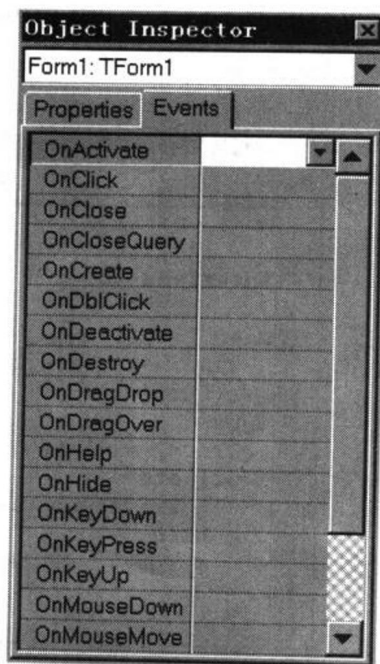


图 1.11 对象检查器的 Events 标签

1.4.6 主菜单

Delphi 3 的主菜单位于集成开发环境的上部,它使我们能够访问丰富的工具集,这些工具集对于应用程序的开发非常有用。主菜单栏如图 1.12 所示,下面我们说明各个菜单的功能。



图 1.12 Delphi 3 的主菜单栏

1. File 菜单

用于打开、保存、关闭和打印项目和文件,以及为打开的项目添加窗体和单元。它含有以下选项: New, New Application, New Form, New Data Module, Open, Reopen, Save, Save As, Save Project As, Save All, Close, Close All, Use Unit, Add to Project, Remove Form Project, Print, Exit。