

实用操作与编程



系列丛书

Borland

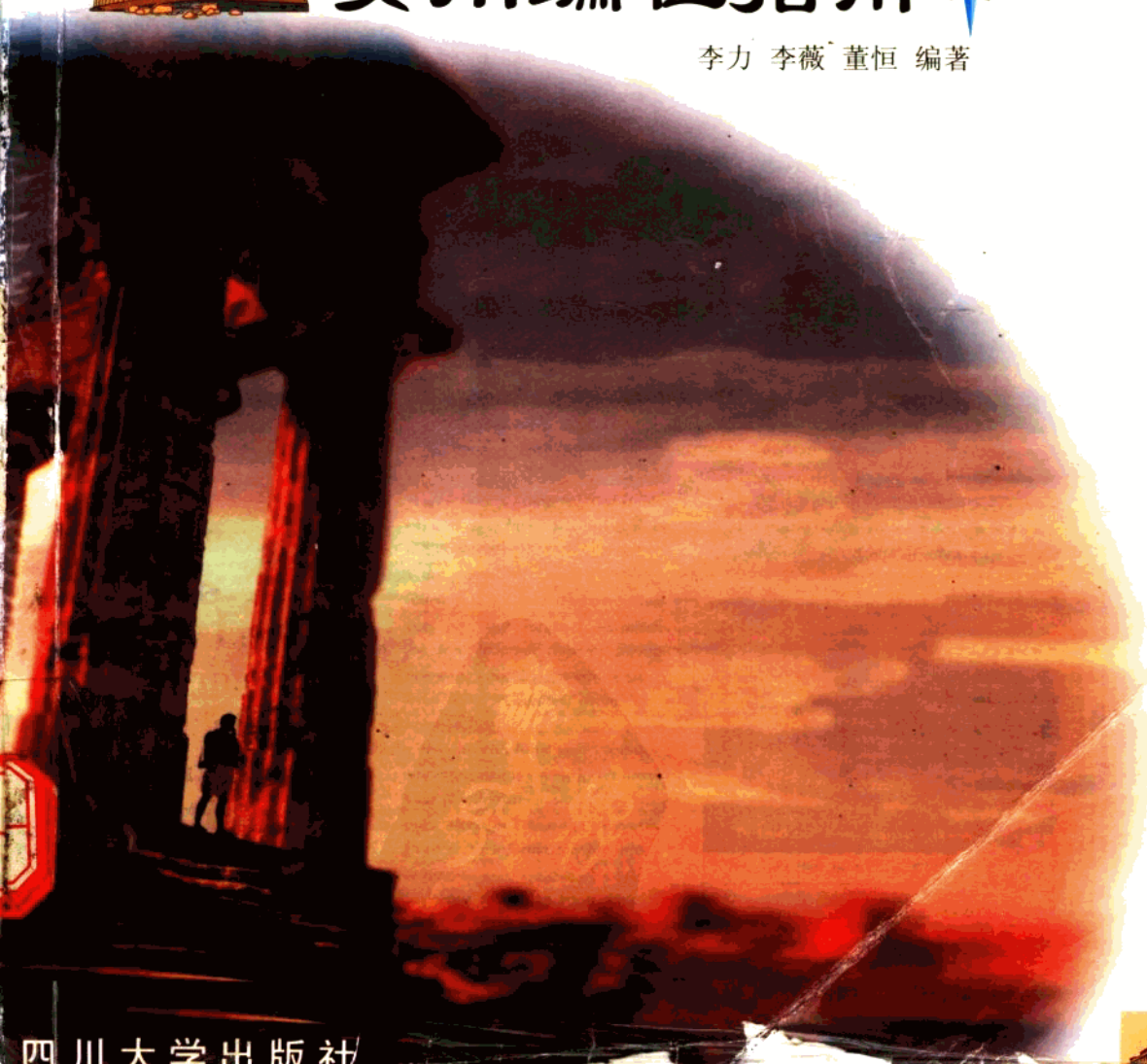
Delphi 4.0



实用编程指南



李力 李薇 董恒 编著



四川大学出版社



目 录

第一章 Delphi 4.0 快速入门	(1)
§1.1 Delphi 4.0 的特点.....	(1)
§1.2 Delphi 4.0 新特性.....	(2)
§1.3 Delphi 4.0 概貌.....	(4)
§1.4 Delphi 4.0 菜单结构.....	(8)
§1.5 制作一个简单的 Delphi 应用程序.....	(32)
第二章 对象 Pascal 语言	(38)
§2.1 对象 Pascal 语言的基本元素.....	(38)
§2.2 基本结构.....	(61)
§2.3 面向对象编程.....	(73)
§2.4 结构化异常处理.....	(79)
第三章 应用程序、文件和工程管理器	(91)
§3.1 Delphi 应用程序的构成.....	(91)
§3.2 组织 Delphi 工程.....	(94)
§3.3 工程管理器.....	(95)
§3.4 工程选项.....	(97)
§3.5 创建工程.....	(103)
第四章 编辑器与调试器	(109)
§4.1 编辑器.....	(109)
§4.2 代码编辑器的类完成与模块导航.....	(113)
§4.3 使用编辑器编辑一个小程序.....	(114)
§4.4 Delphi4.0 的调试工具.....	(117)
§4.5 小 结.....	(129)
第五章 可视化组件库 VCL	(131)
§5.1 VCL 概述.....	(131)
§5.2 组件的属性、事件和方法.....	(132)
§5.3 组件库.....	(134)
§5.4 Standard 标签.....	(135)
§5.5 Additional 标签.....	(136)

§5.6	Dialogs 标签	(137)
§5.7	一个画板演示程序	(137)
§5.8	Win32 标签	(152)
§5.9	System 标签	(153)
§5.10	Internet 标签	(154)
§5.11	Data Access 标签	(156)
§5.12	Data Controls 标签	(157)
§5.13	Midas 标签	(159)
§5.14	Decision Cube 标签	(160)
§5.15	QReport 标签	(160)
§5.16	Win3.1 标签	(162)
§5.17	Samples 标签	(163)
§5.18	Active X 标签	(163)
§5.19	小 结	(164)
第六章	文件管理和打印	(168)
§6.1	文件的属性、过程与函数	(168)
§6.2	文件类型	(176)
§6.3	Delphi 管理文件的一点补充	(188)
§6.4	打 印	(189)
§6.5	DEVMODE 结构与打印机控制码	(199)
§6.6	小 结	(201)
第七章	用 Delphi 开发数据库应用程序	(202)
§7.1	Delphi 数据库系统概述	(202)
§7.2	Delphi 数据库应用程序的开发步骤	(203)
§7.3	BDE 和 SQL Links 简介	(204)
§7.4	学习使用 Form Wizard	(208)
§7.5	学习使用 Database Desktop 创建自己的数据库表	(210)
§7.6	基于表 Cominfo.db 的数据库应用	(213)
§7.7	处理一个以上的数据库表 (主要-明细数据库)	(216)
§7.8	利用代码访问数据库	(219)
§7.9	数据浏览部件的应用及编程	(243)
§7.10	使用 QuickReport 制作报表	(248)
§7.11	在 QuickReport 报表上创建图表	(262)
§7.12	在 Delphi 中使用 SQL 编程	(267)
§7.13	小 结	(279)

第八章 动态链接库 (DLL) 编程	(280)
§8.1 DLL 在 Windows 中的作用	(280)
§8.2 DLL 的访问和编写	(281)
§8.3 在 DLL 中实现数据传输	(292)
§8.4 利用 DLL 实现窗体重用	(306)
§8.5 小 结	(320)
第九章 在 Delphi 中使用 OLE 技术	(321)
§9.1 什么是 OLE	(321)
§9.2 创建 OLE 客户的一般步骤	(323)
§9.3 TOLEContainer 组件的属性与方法	(325)
§9.4 OLE 自动化	(333)
§9.5 小 结	(358)
第十章 动态数据交换 (DDE)	(363)
§10.1 DDE 原理和 Delphi 的 DDE 实现机制	(363)
§10.2 开发 DDE 程序的一般步骤	(365)
§10.3 简单的 DDE 范例	(366)
§10.4 复制与粘贴 DDE 链接	(370)
§10.5 使用计时器的 DDE 连接	(375)
§10.6 小 结	(378)
第十一章 创建 VCL 组件与 Active X 组件	(379)
§11.1 关于创建 VCL 的一些基本概念	(379)
§11.2 创建并安装组件	(380)
§11.3 编写可视组件	(384)
§11.4 修改现有组件——TcomboBar	(394)
§11.5 Clock 组件	(401)
§11.6 Active X 组件概述	(406)
§11.7 将可视组件转化为 Active X 组件	(408)
§11.8 直接给 Active X 组件添加方法	(421)
§11.9 在可改写的 Web 页上使用 Active X 组件	(422)
§11.10 小 结	(424)
第十二章 在 Delphi 中应用多媒体技术	(425)
§12.1 快速开始: 在应用程序中使用声音	(425)
§12.2 使用 Media Player 组件	(433)
§12.3 使用 CD 驱动器	(444)

§12.4 小结.....	(450)
第十三章 利用 Delphi 创建 Internet 应用程序	(451)
§13.1 创建自己主机上的本地服务器 Localhost.....	(451)
§13.2 HTTP 和 HTML 语言.....	(454)
§13.3 利用 Delphi 生成 HTML 文件	(455)
§13.4 在 Delphi 中利用 CGI 技术编程.....	(463)
§13.5 将 CGI 应用程序转换为 ISAPI DLL.....	(468)
§13.6 Web Serve 与数据库的连接.....	(471)
§13.7 使用窗体与客户交互	(474)
§13.8 在 Delphi 中使用 ActiveForm	(481)
§13.9 调试 Web 服务器应用程序.....	(498)
§13.10 小 结.....	(500)
第十四章 应用程序的发行.....	(501)
§14.1 包的概念.....	(501)
§14.2 如何使用包.....	(502)
§14.3 Windows95/98 Logo 要求: Registry	(506)
§14.4 编写一个简单的安装程序	(509)
§14.5 使用商业安装程序生成器	(518)
§14.6 小 结.....	(520)
第十五章 与 C++混合编程	(521)
§15.1 Object Pascal 与 C++共享代码的方式	(521)
§15.2 关于调用上的约定	(522)
§15.3 在 Delphi 中使用 C++的 OBJ 文件	(523)
§15.4 在 C++中链接 Delphi 的 OBJ	(524)
§15.5 在 C++中调用 Delphi 的 DLLs	(525)
§15.6 在 Delphi 中访问 C++中的类	(526)
§15.7 在 Delphi 中访问中 C++中类的 Virtual 方法.....	(528)
§15.8 小 结.....	(530)

第一章 Delphi 4.0 快速入门

欢迎使用 Delphi 4.0 !它是在 Windows 95 和 Windows NT 下的进行快速应用程序 (RAD) 及数据库开发的强有力的优秀工具。它集成了可视化开发环境、优化了的 32 位编译器 and 由紧密集成可调整的数据引擎, 结合了其提供的数据库管理能力。Delphi 4.0 是能够将这三种主要技术无缝地集成起来的开发环境软件包。

Delphi 4.0 针对不同的需要有三个不同的包装, 分别是 Desktop、Developer 和 Client/Server Suite, 应用对象则为初学者、非 Client/Server 专业开发者和 Client/Server 专业开发者。读者可以根据自己的切实需求来选择其中的一个版本。

下面, 本章将对 Delphi 4.0 的特点、新特性、概貌与菜单结构及其使用方法与技巧等作一介绍, 并将通过建立一个简单的应用程序, 来帮助读者对 Delphi 4.0 有一个较为全面而快速的了解。

§ 1.1 Delphi 4.0 的特点

如果你曾经使用过 VB, 你会感到 Delphi 4.0 开发环境与 VB 的开发环境十分接近 (见图 1-1)。

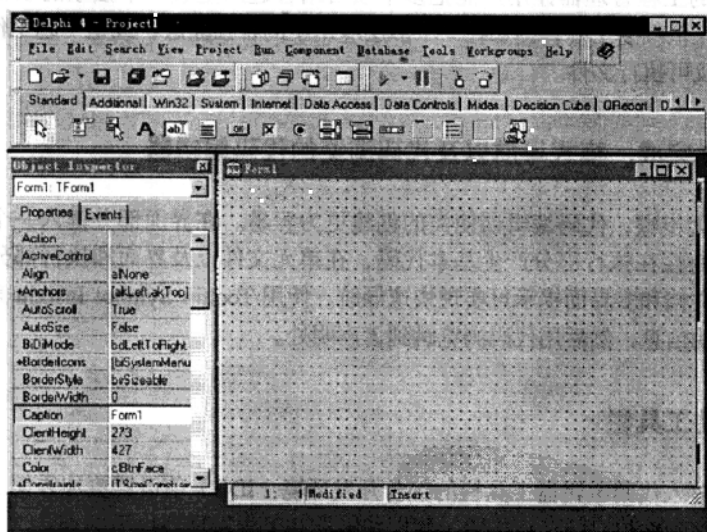


图 1-1 Delphi 4.0 开发环境

但事实上, 尽管 Delphi 4.0 在外观上看起来与 VB 相似, 但与 VB 有着本质的不同。由于 VB 语言本身设计得不够完善, 因此难以创建高度结构化的、紧凑的和精炼的代码; VB

过于注重提高程序快速开发的能力，忽略了培养开发人员的良好开发习惯。与之相反，Delphi 4.0 使用面向对象的 Pascal 作为它的基本语言——Borland Pascal 编译器从出现开始，就一直是最快的编译器之一，Borland 给这种语言添加了基于对象的扩充，使 Pascal 能面向对象，让编程人员能够利用最新的编程技术进行开发。由于 Borland Pascal 的严谨的语法结构，使得编程人员能够保持良好的编程习惯，并能提供更加高效的代码。

§ 1.2 Delphi 4.0 新特性

单从外表来看，Delphi 4.0 与 Delphi 3.0 的区别不是很大，但两者在界面之后，实际上存在着较大的差异。Delphi 4.0 在下列几个方面做出了改进和更新：

1.2.1 新的语言扩展

Delphi 4.0 对 Object Pascal 语言进行了扩展，使之包含了动态数组、方法重载、默认参数等等，其数据类型新增了 64 位长整数类型与 32 位无符号整数类型，并且对固定类型转换进行了增强。

1.2.2 工程管理器

一个全新的工程管理器让用户能把多个项目结合起来生成一个独立的项目组。这使用户可以支配相互依赖的工程，例如在一个多层应用中的各独立层，或是一起工作的动态链接库（DLL）以及可执行文件。

1.2.3 带有类完成、模式导航以及代码浏览的代码编辑器

通过自动化步骤，代码编辑器使类的创建更为容易。在界面部分键入一个方法原型，并使用类完成特性在执行部分产生基本代码，在单元文件以及界面和执行部分产生基本代码。通过一个文档和多界面提示可实现快速导航。使用 Tooltip Symbol Insight 来查看有关任意标识符的声明信息，然后用代码浏览跳到该声明处。

1.2.4 可移动工具栏

带有可被放入代码编辑器工具栏的综合开发环境（IDE）使设置更加方便。你可以把工具栏用鼠标拖动到你想要的任何位置，代码编辑器和工程管理器也不例外。

1.2.5 更先进的调试工具

该调试工具拥有许多新特色，包括远程和多进程调试器、一个 CPU 监视器、检查器、

增加断点、调试器细节子菜单和可移动窗口。

1.2.6 MTS 支持

利用完整的 MTS 以支持多层数据库，提供一个向导，使用户能更方便地创建新的 MTS 服务对象。

1.2.7 Active X 改进

Delphi 4.0 提供了扩展的 Active X 支持，任何新增的 Active X 控件经添加后即可使用。

1.2.8 VCL 改进

Delphi 4.0 对象层拥有一个用于 NT 服务器的新组件。另外，一个新的动作列表（在标准调色板页中）使你能集中精力管理菜单和工具按钮命令。VCL 控件支持拖放行为，以便看到当前被挡住的控件。

1.2.9 用于 2000 年维护的 RTL 支持

其全局变量 TwoDigitYearCentury 被 StrToDate 和 StrToDateTime 函数用来控制两位数年份的日期转换。

1.2.10 CORBA 支持

在 Delphi 4.0 的客户机/服务器（Client/Server）版和企业版中都包含了 CORBA 的客户机端和服务端端的 CORBA 支持。利用导航使得建立 CORBA 服务更为方便，而动态触发界面（DLL）使你能为已有的 CORBA 服务器书写客户端。CORBA 功能也被建立到多层数据库支持中。你甚至可以创建同时控制 COM 客户和 CORBA 客户的一个服务器。

1.2.11 客户数据调整改进

客户数据调整如今能够支持过滤器描述的一个更广阔的类型，提供对维护总计的支持，并允许对象相关的域类型。这些改进使创建用于数据库应用的数据调整更为容易。

1.2.12 多层支持

Delphi 4.0 提供了更多的对多层应用的控制，包括更新/重同步支持，以及对于包括在数据库包中的应用控制与更新方法。调用服务器界面如今变得更为容易。一个新的类——TDataSetProvider 使你向一个数据调整提供结果。Master/Detail 支持使用嵌套表。对客户端数

据调整功能的改进,使得向一个应用程序服务器传递参数或在数据包中保存自定义信息更为容易。

1.2.13 数据库改进

被改进的 Delphi 4.0 组件使你在创建表格或数据模式时,可以调整你的数据模式。对数据访问组件和 Borland 数据库引擎 (BDE) 的改进使你能够在新型数据库服务器中的数据包括 Access 7 和新的 Oracle 8 对 SQL 的扩展,包括抽象数据类型 (ADT)、数组、指代以及嵌套表。原来的 Visual Query Builder 已经被 SQL Builder——一个新的智能查询器所取代。

§ 1.3 Delphi 4.0 概貌

现在,让我们开始熟悉 Delphi 4.0 的开发环境。Delphi 4.0 开发环境是一个标准的 IDE,所谓 IDE (Integrated Development Environment) 是提供设计、运行和测试应用程序所需的所有工具的一种环境,它对这些工具进行了很好的连接,以减少程序开发的难度。在 Delphi 4.0 中,IDE 由代码编辑器、调试器、工具栏、图象编辑器和数据库工具组成,所有这些都以集成方式进行操作,它们协调操作、相互补充,使复杂应用程序的开发速度更快,更不容易出现错误。

了解 IDE 的第一步是启动 Delphi 4.0。Delphi 4.0 的开发环境如图 1-2 所示。

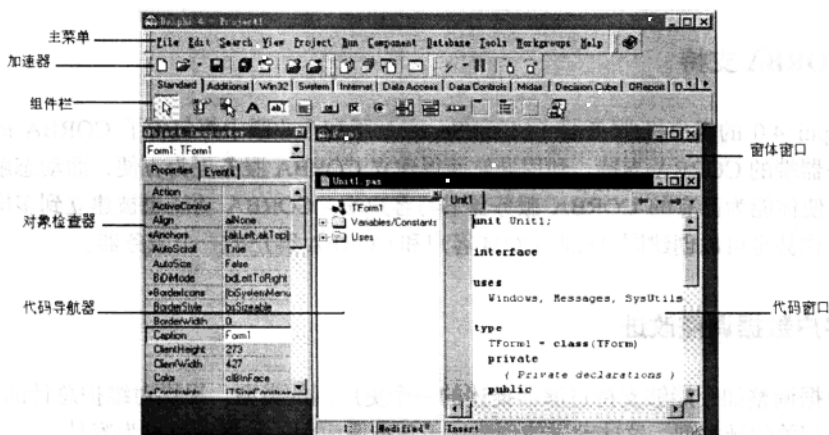


图 1-2 Delphi 4.0 的开发环境 IDE

与 Delphi 3.0 以下版本不同,Delphi 4.0 具有的一项新的特性,即可移动的工具栏。比如菜单栏,读者可以将其拖动到其它任意位置,用户完全可以利用这一新特性,根据编程喜好与习惯,随意设定桌面的布局。

1.3.1 加速器栏

加速器栏如图 1-3 所示。它使用户可以容易而快速地得到最常用的功能。这些功能在 Delphi 4.0 菜单中也可以得到, 用户还可以对加速器栏进行自定义, 以达到更方便地使用为目的。如果不知道加速器栏上某个图标的意义, 可以将鼠标指针在该图栏上停留几秒钟, 就能得到它的文字提示(hint)。对于 Delphi 4.0 默认的图标, 将在下一节中予以详细的描述。

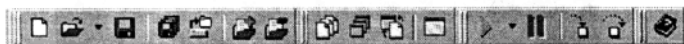


图 1-3 加速器栏

1.3.2 组件栏

组件栏中存放的是 VCL 组件。在缺省状态下, Delphi 4.0 根据各 VCL 组件的功能来进行分类。你也可能根据你自己的使用情况将其更合理地分类。Delphi 4.0 在缺省状态下将其分为 14 个类, 它们分别是:

Standard Internet Decision Cube Samples	Additional DataAccess Qreport Active X	Win32 DataControls Datalogs	System Midas Win3.1
--	---	-----------------------------------	---------------------------

每个标签都含有一个指针按钮图标，不论用户选择哪个类的标签，这个指针按钮总是处于组件栏的左端。当此按钮处于有效（按下）状态时，用户可以在窗体和窗口之间进行移动，以操纵 Delphi 环境。当用户从某个组件页选择了一个组件时，该指针按钮将变为无效，表示 Delphi 认为用户准备在窗体中放置一个组件。当选择好某个组件后，用户只需在窗体上单击鼠标即可完成操作，此时指针按钮将返回有效状态。如果用户选择了一个组件后又改变了主意，在单击窗体将此组件放置之前，可以单击鼠标指针来取消当前选择。

对于每个组件的具体细节，将在第四章中进行阐述。

1.3.3 窗体

图 1-4 给出了一个 Delphi 4.0 的空白窗体。窗体几乎是每个 Delphi 4.0 应用程序的基础。不要认为窗体就是窗口！在 Delphi 4.0 中，窗体是放置其它 Delphi 组件的基础控件，它是 Windows 应用程序的背景。

Delphi 4.0 中的窗体具有与 Windows 95 窗口相同的属性，如它在窗体的左上角有一个控制菜单，标题栏横贯窗体的顶部，右上角同样拥有最小化、最大化和关闭按钮。如果你的应用程序中不需要这些窗体控件，那么，可以将它们隐藏起来，或者可以将每个按钮的属性设置为无效，这使你可以控制用户完成的操作。

注意：尽管大多数 Delphi 4.0 应用程序都以窗体为基础，但当你使用 Delphi 4.0 来编写 Win32 DLL、个人的 Active X 控件和包，都可以不使用或依赖窗体。

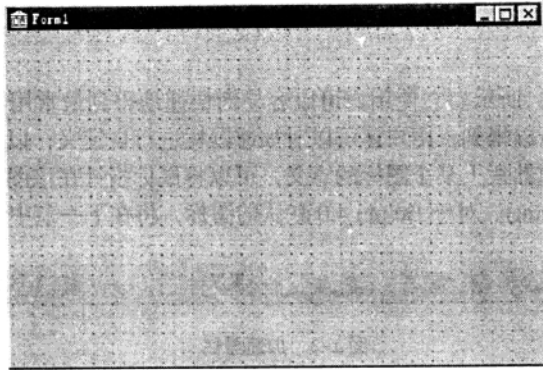


图 1-4 Delphi 4.0 窗体

1.3.4 编辑窗口

Delphi 4.0 环境最基本的一个组成部分就是编辑窗口。编辑窗口是你在开发 Delphi 4.0 应用程序时输入 Delphi 代码的场所。Delphi 代码编辑器是一种高性能的编辑器，如图 1-5 所示。它的彩色语法高亮显示功能使你在阅读和编辑代码时条理清楚，并且有利于你更加快速方便地查找到语法错误。

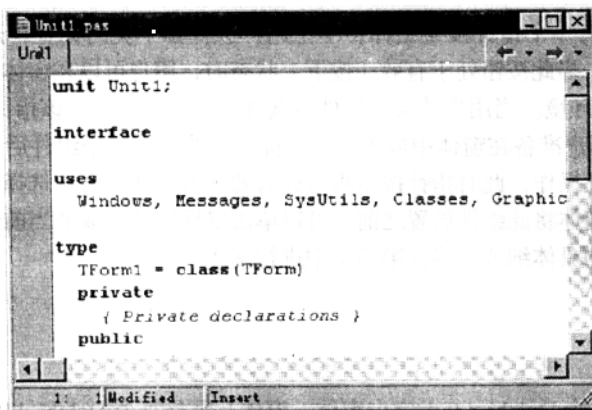


图 1-5 Delphi 4.0 的编辑窗口

1.3.5 对象检查器

对象检查器 (Object Inspector) 是在 Delphi4.0 中非常有用且必不可少的，它可以用来改变 Delphi 4.0 项目的属性，同时也控制对象响应的事件。

写相应的代码。

§ 1.4 Delphi 4.0 菜单结构

Delphi 4.0 的菜单使我们得以使用 Delphi 4.0 丰富的工具，为我们进行大量的应用程序开发提供了帮助。图 1-9 为菜单栏，本节将对所有菜单中的所有选项简要地进行描述。以后在使用到它们的以下各章节中，再作详细的讨论。

注意：你所看到的实际菜单项取决于所使用的 Delphi 版本。本书采用 Client/Server 版，它包含有额外的菜单项。

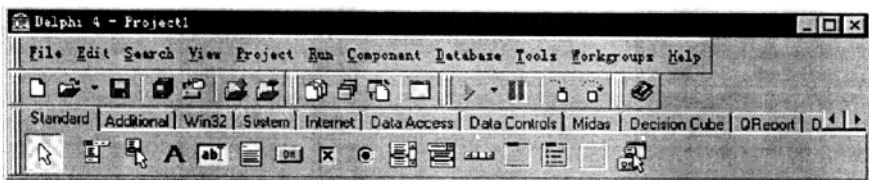


图 1-9 主菜单栏

1.4.1 File 菜单

File 菜单用于打开、保存、关闭和打印新的或现存的工程和文件，以及给打开的工程添加新的窗体和单元。它所包含的菜单项如图 1-10 所示。

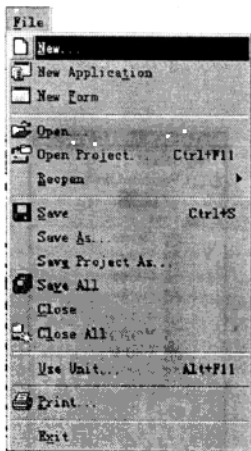


图 1-10 File 菜单栏

1. New 选项

选择 New 告诉 Delphi 4.0 你想创建一个新的对象。此对象可以是对象库内的任何一个成员，包括一个新的工程。选择此项后，将出现 New Items 对话框，它使你能够创建从新窗

口到 Web 服务器的所有对象。

这个对话框中的有些选项将在后面的章节中更详细地讨论。

2. New Application 选项

选择 New Application 选项则告诉 Delphi 4.0 你想创建一个新的工程。Delphi 4.0 的将关闭所有当前的工程，创建一个全新的工程。

3. New Form 选项

选择 New Form 选项则告诉 Delphi 你想创建一个新的窗体。

4. Open 选项

选择 Open 选项则告诉 Delphi 你想打开一个对象。该对象可以是一个工程，也可以是某个代码模块。

5. Open Project 选项

选择 Open Project 选项告诉 Delphi 你想打开一个工程。

6. Reopen 选项

选择 Reopen 选项时将出现一个列表单，其中含有你最后打开的几个工程或文件，你可以从中选择并打开。

7. Save 选项

选择 Save 选项将保存当前正在处理的模块。

8. Save As 选项

选择 Save As 选项将以一个新的名称保存当前模块。

9. Save Project As 选项

选择 Save Project As 选项，表示你可以用一个新的名称保存当前的工程。

10. Save All 选项

选择 Save All 选项将保存当前打开的工程文件和其它所有文件。

11. Close 选项

选择 Close 选项则关闭当前被选中的窗体和与其相联系的代码块。

12. Close All 选项

选择 Close All 选项将关闭当前被打开的工程。

13. Use Unit 选项

选择 Use Unit 选项使你能够将一个 Use 语句置于当前的代码模块中，以得到想使用的单元，而无需你手工插入单元名。

14. Print 选项

选择 Print 选项使你能够打印已在 Delphi 中选中的项目。

如果你选择了一个窗体，然后选择 Print 选项，那么，Delphi 将显示如图 1-11 所示的 Print Form 对话框。在此对话框中，你可指定打印窗体的方式：Proportional（按比例进行打印），Print to fit page（按照纸的尺寸进行打印）、No Scaling（打印时不缩放）。然后按 OK 按钮开始打印。

如果你选择的是代码窗口，然后选择 Print 选项，那么，Delphi 将显示如图 1-12 所示的 Print Selection 对话框。在此对话框中，你可以指定打印代码窗口的方式，如打印行号、只打印所选文本以及其它打印方式。

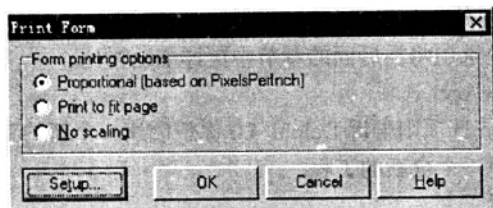


图 1-11 Print Form 对话框

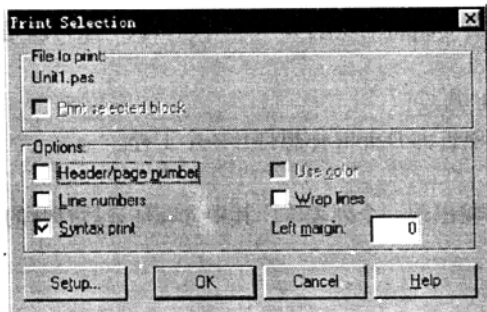


图 1-12 Print Selection 对话框

15. Exit 选项

选择 Exit 选项时，退出 Delphi 环境。

1.4.2 Edit 菜单

当你使用 Delphi 进行设计时，可以使用 Edit 菜单上的命令来操纵文本和组件。它包含的菜单项如图 1-13 所示。

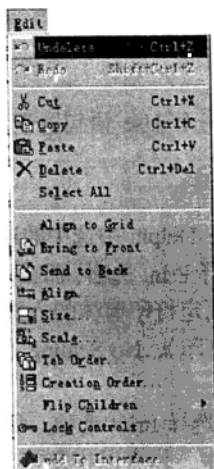


图 1-13 Edit 菜单

1. Undo/Undelete 选项

Undo/Undelete 选项使你能够恢复此次改动前所做的内容。这个菜单选项以 Undo 或 Undelete 的形式出现,这取决于你前面所采取的动作。如果之前你刚刚使用 Delete 键或选择 Delete 选项删除了一个对象或某些代码,那么,此选项就以 Undelete 的形式出现。如果之前你刚刚给工程添加了代码或组件,则此选项以 Undo 的形式出现。

2. Redo 选项

与 Undo 选项的操作相反,Redo 可以取消已完成的任意次数的 Undo 操作。

3. Cut 选项

使用 Cut 选项将使当前被选中的项目被剪贴到剪贴板中,并且所选项目从当前窗体或代码单元中去除。

4. Copy 选项

选择 Copy 选项将当前选中的项目拷贝到剪贴板中。

5. Paste 选项

选择 Paste 选项将剪贴板中的项目拷贝到当前窗体或代码单元中。

6. Delete 选项

选择 Delete 选项将删除当前选中的项目。但它与 Cut 不同,项目将不被剪切到剪贴板,因此,要恢复的话只能使用 Undelete 选项。

7. Select All 选项

选择 Select All 将把当前窗体上的所有组件或当前单元中的所有代码全部选中。

8. Align to Grid 选项

选择 Align to Grid 选项,可以使当前选中的组件对齐到栅格上。

注意:如果在 Tools 菜单中选择了 Environment Options 选项,在选项框的 Preferences 标签页面中选择了 Snap to Grid 选项,那么,此菜单选项就没有必要使用了。

9. Bring to Front 选项

当在窗体中放置了许多组件,且这些组件相互重叠时,这个菜单选项就非常有用。它可以将当前选中的组件移到所有其它组件的顶部。

10. Send to Back 选项

该选项与 Bring to Front 选项产生的效果相反,它可以使当前被选中的组件置于所有组件的后面。

注意:窗口控件和非窗口控件应该分开考虑。所有非窗口控件都位于所有窗口控件的后面。

11. Align 选项

选择 Align 选项将出现如图 1-14 所示的 Alignment 对话框。你可以使用其中的对齐方式,将你在之前选中想对齐的组件进行水平或垂直对齐。

12. Size 选项

该选项使你能够设置所选组件具体的高度和宽度。选择该项将出现如图 1-15 所示的 Size 对话框。如果你选择了多个组件,那么,可以将所有组件的大小增加到页上所选最大组件的大小,或者可以将所有组件的大小缩小到页上所选最小组件的大小。

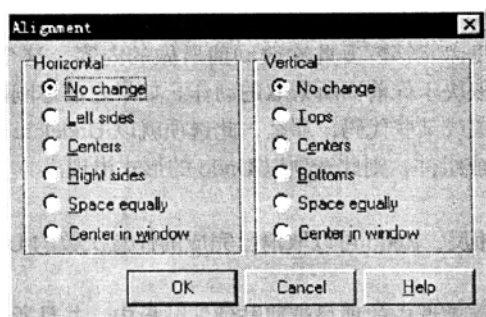


图 1-14 Alignment 对话框

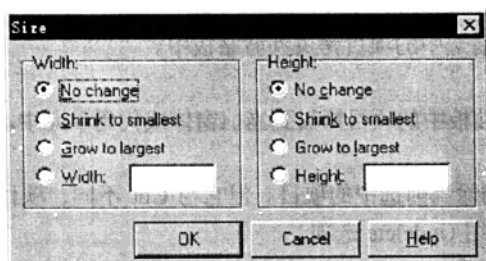


图 1-15 Size 对话框

13. Scale 选项

Scale 选项可以成比例地缩放窗体上的内容。只需在出现的 Scale 对话框中输入缩放比例，再单击 OK 即可，如图 1-16 所示。

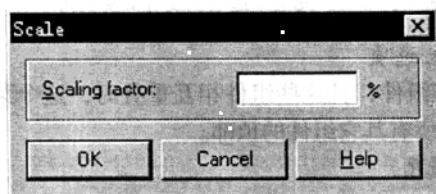


图 1-16 Scale 对话框

14. Tab Order 选项

Delphi 使我们能够调整当前所选窗体中项目的切换顺序。这是用户按动 Tab 键在窗体中的域之间移动时各个项目之间得到聚焦的顺序。这使我们能够控制项目被聚焦的顺序，从而在屏幕上“引导”用户进行操作。

选择 Tab Order 选项时，显示如图 1-17 所示的 Edit Tab Order 对话框，这个对话框在列表框内含有有窗体上所有组件的名称。可以通过选中想重新定位的项目并单击上下箭头，以将其移动到新的位置，从而可视地来设置它们的顺序。

15. Creation Order 选项

选择 Creation Order 选项使你能够控制创建非可视组件的顺序。因为某些非可视组件依