



光盘内附书中范例

以实例介绍常用组件与函数应用程序的编写与设计方法

Object-Pascal 程序语法的必备宝典

充分领略 Delphi 的强大火力



康吉成 编著

Delphi 6

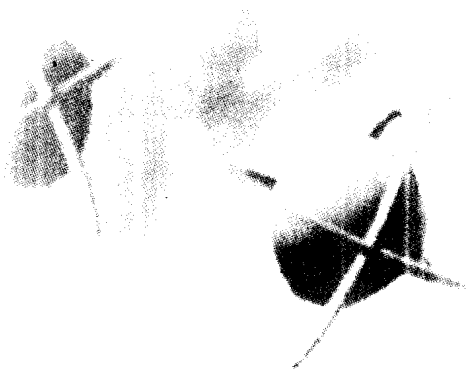
程序设计



中国铁道出版社

CHINA RAILWAY PUBLISHING HOUSE

Delphi6 程序设计



康吉成 编著

中国铁道出版社

2002年·北京

(京)新登字 063 号

北京市版权局著作合同登记号: 01-2002-1087 号

版 权 声 明

本书为台湾碁峯资讯股份有限公司独家授权的中文简化字版本。本书专有出版权属中国铁道出版社所有。在没有得到本书原版出版者和本书出版者书面许可时任何单位和个人不得擅自摘抄、复制本书的一部分或全部以任何方式包括(资料和出版物)进行传播。本书原版版权属碁峯资讯股份有限公司。版权所有,侵权必究。

图书在版编目(CIP)数据

Delphi 6 程序设计/康吉成著. —北京:中国铁道出版社, 2002. 4

ISBN 7-113-04651-7

I. D… II. 康… III. DELPHI 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2002)第 023091 号

书 名: Delphi 6 程序设计

作 者: 康吉成

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 郭毅鹏

责任编辑: 苏 茜 袁秀珍

封面设计: 王 晖

印 刷: 北京市彩桥印刷厂

开 本: 787×1092 1/16 印张: 24.5 字数: 595 千

版 本: 2002 年 5 月第 1 版 2002 年 5 月第 1 次印刷

印 数: 1~5000 册

书 号: ISBN7-113-04651-7/TP·699

定 价: 40.00 元

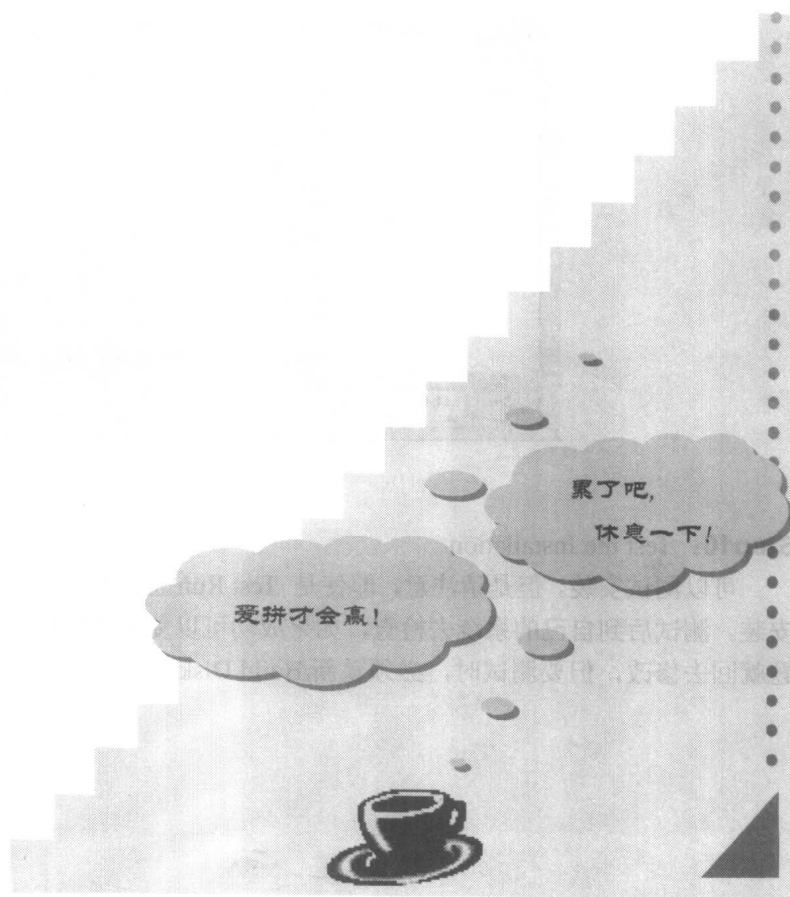
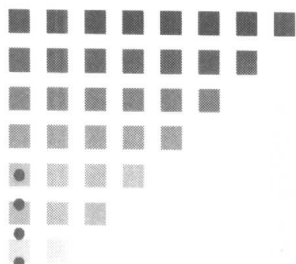
版权所有 盗印必究

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。

出版说明

本书由基本的 Delphi 6.0 SDK(软件开发工具)的使用界面开始讲解，并完整说明 Object-Pascal 的语法与变量类型，然后进一步地说明一般窗口应用程序所具备的各项功能及其安装程序的设计与编写方法。为了方便读者学习，本书所有范例程序的源代码收录于随书所附光盘的 **Sample** 文件夹中，让您的学习能事半功倍。

本书由台湾碁峯资讯股份有限公司提供版权，中国铁道出版社计算机图书项目中心审选；张翰文、罗怡、葛兰、彭涛、肖志军、廖康良、裘伟力、陈贤淑等同志完成了本书的整稿及排版工作。

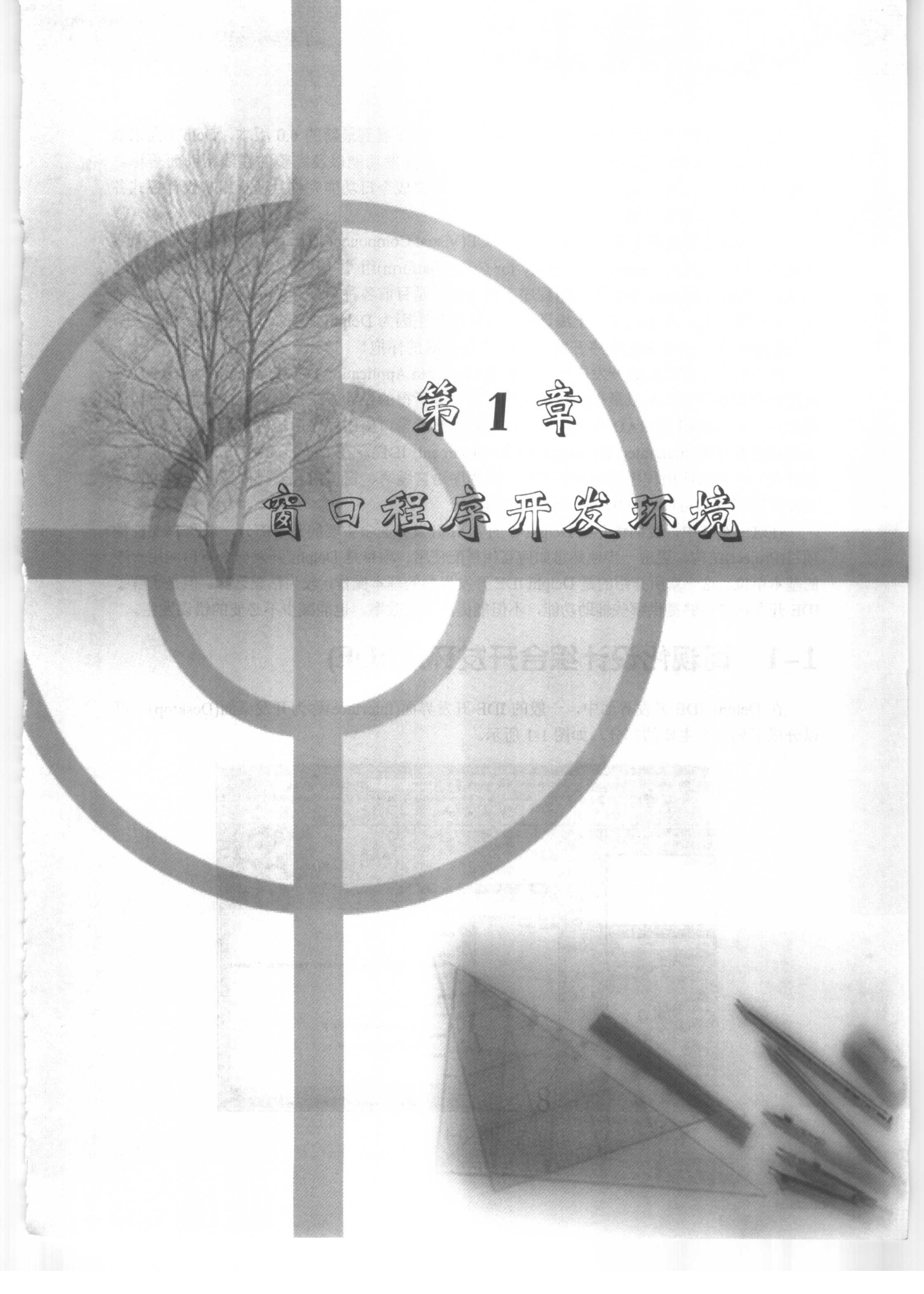


目 录

第 1 章 Delphi 6 窗口程序开发环境	1
1-1 可视化设计综合开发环境(IDE)	2
1-1-1 菜单(Menus)	3
1-1-2 快捷工具栏(Speed Toolbars)	18
1-1-3 组件面板(Component Palette)	18
1-1-4 对象树状图(Object TreeView)	19
1-1-5 对象查看器(Object Inspector)	20
1-1-6 窗口窗体(Form)	21
1-1-7 程序代码浏览/编辑器(Code Explorer/Editor)	21
1-2 Delphi 项目结构	24
1-2-1 Pascal-Source Files	25
1-2-2 Other-Enhance Files	25
1-2-3 Compiler-Generated Files	26
1-2-4 常见的项目结构	26
1-3 开发第一个窗口应用程序	28
第 2 章 Delphi 6 Object Pascal 程序语法	33
2-1 程序(Program)和单元(Unit)	34
2-1-1 Program 的文件结构	34
2-1-2 Unit 的文件结构	35
2-1-3 uses 语句	36
2-2 数据类型(Data Type)	37
2-2-1 简单类型(Simple type)	38
2-2-2 字符串类型(String type)	45
2-2-3 结构化类型(Structured type)	46
2-2-4 指针类型(Pointer type)	54
2-2-5 过程类型(Procedural type)	55
2-2-6 变体类型(Variant type)	57
2-3 基本运算符(Operator)	58
2-3-1 算术运算符(Arithmetic operators)	58
2-3-2 布尔运算符(Boolean operators)	59
2-3-3 逻辑运算符(Logical operators)	60
2-3-4 指针运算符(Pointer operators)	60
2-3-5 集合运算符(Set operators)	61

2-3-6	关系运算符(Relational operators)	61
2-3-7	其他运算符	61
2-4	变量与常数声明(Declaration)	62
2-4-1	类型声明(Declaring type)	62
2-4-2	变量声明(Declaring variable)	63
2-4-3	常数声明(Declaring constant)	65
2-5	基本语法与流程控制	67
2-5-1	基本语法	67
2-5-2	流程控制	69
2-6	程序(Procedure)与函数(Function)	73
2-6-1	程序与函数 Directive 声明指令	76
2-6-2	参数传递方式与种类	79
第 3 章	Delphi 6 程序开发设计基础	83
3-1	认识基本容器组件——窗体(Form)	84
3-1-1	TForm 的属性	84
3-1-2	TForm 的方法	93
3-1-3	TForm 的事件	100
3-2	认识基本组件	109
3-2-1	Standard 组件	109
3-2-2	Additional 组件	133
3-2-3	Win32 组件	158
3-3	基础应用程序开发	177
3-3-1	应用程序开发流程	177
3-3-2	认识各种窗口	178
第 4 章	Delphi 6 应用实务：数据库程序开发设计	203
4-1	自定义数据库文件——纯文本文件与随机文件	204
4-1-1	纯文本文件	204
4-1-2	随机文件	210
4-1-3	其他重要文件处理函数	211
4-2	数据库系统与数据库组件的使用	223
4-2-1	数据库的建立与维护	223
4-2-2	数据库链接的设置	227
4-2-3	常用数据库组件	231
4-3	报表的设计与打印机的控制	238
4-3-1	使用报表组件	238
4-3-2	打印机的控制	247
4-4	工时数据库程序	254
4-5	光盘整理数据库程序	284

第 5 章 Delphi 6 应用实务：多媒体程序开发设计	295
5-1 制作不规则窗体.....	296
5-1-1 不规则画面的呈现.....	296
5-1-2 窗体拖曳功能.....	305
5-2 影音 API 函数与 Flash 动画.....	308
5-2-1 图形处理以及播放声音的 API 函数.....	308
5-2-2 播放 Flash 动画.....	312
5-3 认识多媒体组件.....	313
5-4 多媒体播放程序.....	318
5-5 屏幕破坏程序.....	328
5-6 屏幕小可爱程序.....	333
第 6 章 Delphi 6 高级应用实务：线路程序开发设计	339
6-1 认识基本网络组件.....	340
6-2 Client-Server 程序开发.....	341
6-3 FTP Client 端程序开发.....	353
6-4 互联网 CGI 程序开发.....	357
第 7 章 Delphi 6 高级应用实务	363
7-1 资源文件类与使用.....	364
7-2 常驻程序的开发.....	367
附录 A Delphi 6 InstallShield Express	371



第 1 章

窗口程序开发环境

Delphi 是窗口应用程序的主流开发环境,目前已发展到最新的 6.0 版本,Delphi 向来以其方便的操作界面、简易的程序语法、强大完备的开发功能以及完整的在线帮助而著称。Delphi 也由早期 1.0 只适用于开发数据库程序,演变成今日功能完整且适合用来设计各式各样的窗口程序的开发工具。

Delphi 6 比 Delphi 5 提供更多更炫的 VCL(Visual Component Library)组件,甚至提供兼容于 Kylix 1.0 的 CLX(Component Library for Cross-platform)组件,从而让程序开发人员轻松建构跨越 Windows/Linux 平台的应用程序。Delphi 6 是目前各种窗口开发系统中唯一一个能使用相同程序代码,在不同平台上编译的开发环境!正因为 Delphi 具有如此完整强大的能力,所以能够吸引许许多多的程序开发人员投入 Delphi 的怀抱!

Delphi 提供可以快速开发应用程序的 RAD(Rapid Application Development)环境。RAD 环境能够让程序开发人员以最少的程序代码与所见即所得的设计方式,来建立高效率的窗口应用程序。在 Delphi 的 RAD 开发环境中,综合了开发窗口应用程序所需的工具及组件,称之为集成开发环境(Integrated Development Environment, IDE)。在 Delphi 的 IDE 开发界面中包含开发一个窗口应用程序所需的各部分,如程序语言编写、窗口内容安排以及编译成可执行文件和调试等都可以在 IDE 操作环境中搞定。

在这一章节中,我们将介绍 Delphi 6 的 IDE 开发界面中主要的几个部分,以及 Delphi 的项目(Project)结构,更进一步地熟悉如何互相搭配使用。项目是 Delphi 开发一个窗口应用程序的基本单位。这一章所介绍的是 Delphi IDE 开发界面的基本操作,较为枯燥乏味。但是了解了 IDE 开发环境能够提供哪些辅助功能,不但能提高开发效率,也能减少不必要的错误发生。

1-1 可视化设计综合开发环境(IDE)

在 Delphi IDE 开发界面中,一般的 IDE 开发界面(Interface)称为开发桌面(Desktop),可以分成下列几个主要的部分,如图 1-1 所示。

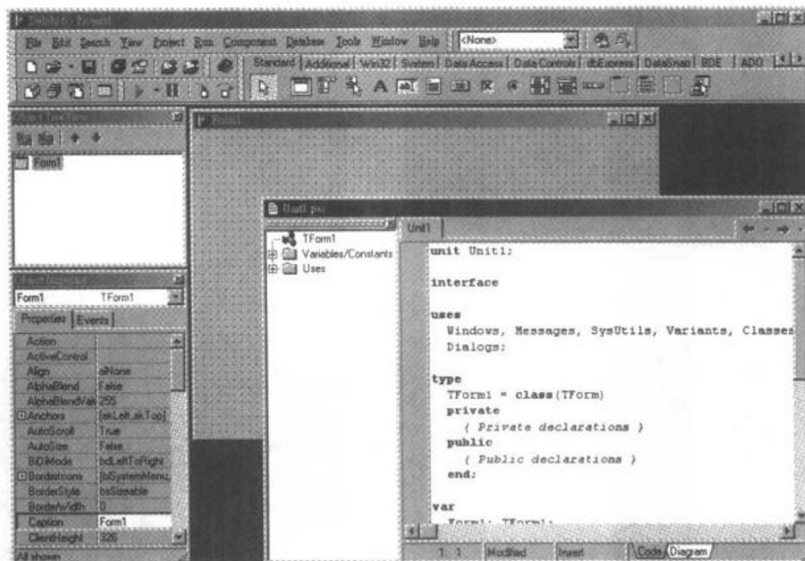


图 1-1

1. 菜单(Menu): 许多常用的选项已经在快捷工具栏(Speed Toolbars)当中显示了。此外还有几个重要方便的工具窗口需要我们手动选取, 如: Project Manager、Alignment Palette 等。
2. 快捷工具栏(Speed Toolbars): 由数个菜单中常用的功能选项所组成的按钮群, 分为 Standard、View、Debug 等群组。
3. 组件面板(Component Palette): 分类列出 Delphi 中所提供的组件, 包含可视与不可视的 VCL 组件以及 CLX 组件。
4. 对象树状图(Object TreeView): 将一个窗体(Form)中所使用到的对象(Object)以及子对象以树状图方式列出。
5. 对象查看器(Object Inspector): 显示出所选择的对象(Object)或窗体(Form)所具有的属性(Properties)与事件(Events)。
6. 窗口窗体(Form): 让我们安排窗口程序显示的畫面, 以所见即所得方式设置所选用的对象(Object)。
7. 程序代码浏览/编辑器(Code Explorer/Editor): 具备单元程序代码浏览, 以及完整的编辑功能的工具窗口。

1-1-1 菜单(Menu)

菜单包含所有 IDE 开发界面的指令与功能。除了前面列出的几个主要工作窗口外, 还有几个常用的工具窗口是必须知道的。以下将一一说明这几个常用的工具窗口。

项目管理器(Project Manager)

项目管理器是让程序开发人员管理建立一个应用程序所需的窗体(Form)与单元(Unit), 单元即是代码(Source Code)。一个项目(Project)往往就是一个可执行的应用程序, 而一个应用程序中会包含许多个窗口, 如: 主执行窗口、对话窗口以及设置窗口等。每一个窗口设计时均会包含一个窗体与一个单元, 每个窗体与单元可能存放在不同路径中, 因此过多的窗口会造成应用程序开发的困难。项目管理器可以辅助程序开发人员快速有条理地显示所有项目中使用的窗体或单元。项目管理器是开发具备多个窗口画面程序时必不可少的重要角色, 在许多开发实例中都会常常看到项目管理器的踪影。

通常在开发大型应用程序时, 一个项目可能无法一次涵盖, 需要开发团队(Team)一同开发。此时, 往往需要将数个不同的项目合并至同一个群组中, 形成项目群组(Project Group)。在项目管理器中, 可以利用建立一个新增项目或加入一个现有项目的方式来建立项目群组, 方便开发团队的综合, 如图 1-2 所示。



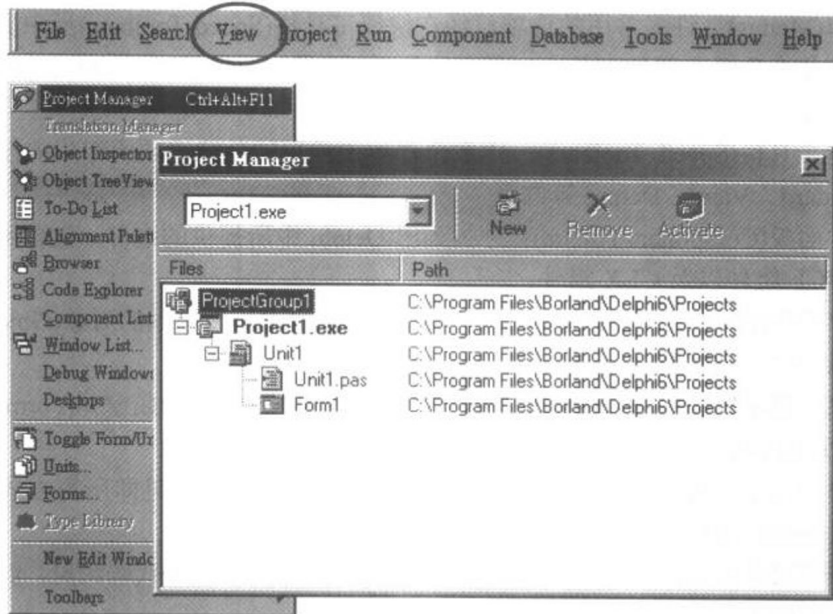


图 1-2

对齐工具栏(Alignment Palette)

在窗体上配置组件位置时，通常可以利用窗体上的栅格(Grid)或修改对象的 Top 和 Left 属性值来达成对齐处理。但是这都只是针对单一对象所做的对齐处理，如果同时处理许多对象的对齐工作则显得耗时费力。对齐工具栏主要目的就是辅助处理窗体上所有选取对象的整体对齐工作。如图 1-3 所示。

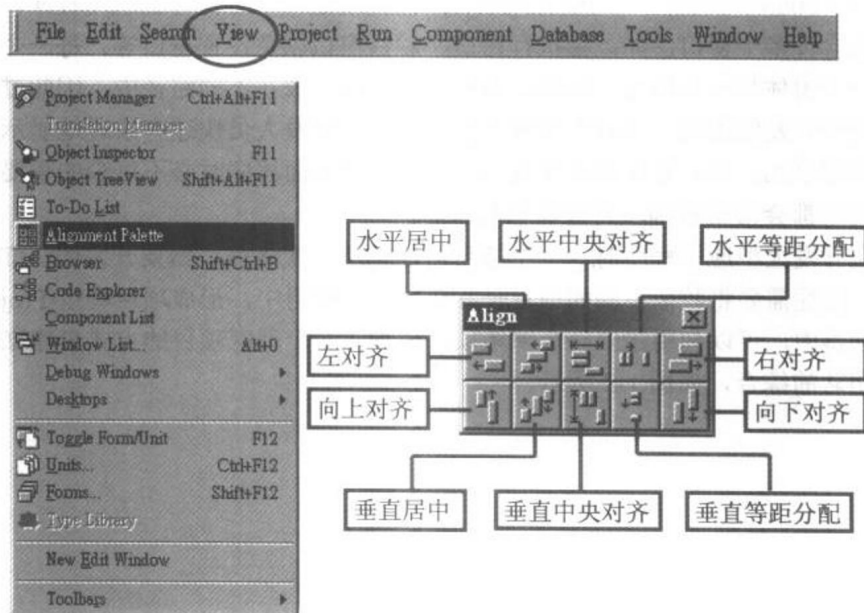


图 1-3

在窗体上同时选取多个对象，首先按住键盘上的【Shift】键不放，接着——点选想要选取的对象即可同时选取多个对象。或直接在窗体上按住鼠标左键不放，拖曳出一个涵盖想要选取对象的虚线选取区，也可以达到同时选取多个对象的目的。

项目设置(Project Options)

项目设置窗口主要设置内容包含以此项目开发的应用程序激活时窗体建立情形、项目相关信息、编译连接设置、输出路径以及版本信息等，共有七个选项卡的参数可以进行设置。项目文件设置窗口可以通过选择菜单【Project】的子菜单中的【Options】选项调用出来。以下会对每个选项卡的内容进行详细的说明。

◆ Forms 选项卡

当一个项目中包含多个窗体(Form)，则编译此项目为可执行的应用程序时就面临了一些问题，例如哪个窗体才是主窗口画面，哪些窗体要直接建立加载内存，哪些不用等。每当在项目中新建一个窗体时，Delphi 会默认此窗体为自动建立窗体，如图 1-4 所示。

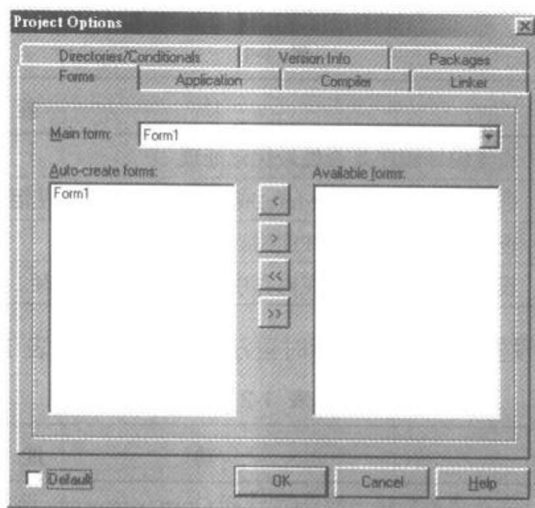


图 1-4

Forms 选项卡主要目的就是用来设置当应用程序(由项目所编译产生的)执行时，哪一个窗体是应用程序的主窗体(主窗口)；哪些窗体是在程序激活时就应该自动建立的自动建立窗体(Auto-create forms)；哪些窗体是不需程序激活即自动建立，而是当程序需要使用时才以程序代码建立的可用窗体(Available forms)。如果全部自动建立加载内存，当窗体很多时则会浪费内存，因此需取得一个适当的平衡点。

有一点需要注意的是在一般项目中至少需要一个窗体设为自动建立窗体，而且主窗体必须是自动建立窗体中的第一个。

◆ Application 选项卡

Application 选项卡主要设置项目编译的应用程序所具备的一般属性。主要分为四个属性设置：



1. Title: 设置应用程序名称, 即该应用程序在 Windows 操作系统中的名称, 必须是小
于 255 个字符的字符串。
2. Help file: 设置应用程序的帮助文件, 即一般程序中按【F1】键后会出现的画面。需
使用*.HLP 文件。
3. Icon: 设置应用程序图标, 即该应用程序在 Windows 操作系统中的图标。需使用
ICON(*.ICO)文件。如果不设置则 Delphi 会使用默认图标。
4. Target file extension: 设置应用程序编译输出时的扩展名, 一般空着即使用默认
的.EXE 为扩展名。

Application 选项卡所设置的属性即是每个应用程序 Application 对象(TApplication 类别的
应用程序内建对象)的属性, 通常会呈现在编译出来的可执行文件上。

◆ Compiler 选项卡

Compiler 选项卡主要设置项目文件(.dpr)编译成可执行文件(.exe)时的相关属性, 共分成
五个属性群组:

1. Code generation: 设置程序代码生成模式, 包含四个选项。如表 1-1 所示。

表 1-1

选 项	功 能
Optimization	设置编译程序是否做最优化处理
Stack frames	强迫所有程序(Procedure)或函数(Function)上使用堆栈结构
Pentium-safe FDIV	设置检测 Pentium CPU 错误的浮点运算
Record field alignment	设置数据结构字段在内存中基本计算单位, 有 1、2、4、8 Bytes 四种可选择

2. Syntax options: 设置程序代码编译时语法检查设置。如表 1-2 所示。

表 1-2

选 项	功 能
Strict var-strings	字符串参数错误检查(如选择 Open parameters 选项, 而此选项不运行)
Complete boolean eval	完整的计算每个布尔运算
Extended syntax	是否允许建立无返回值的函数
Typed @ operator	控制利用@运算符返回指针的类型
Open parameters	允许在程序或函数中声明开放式字符串变量
Huge strings	设置关键词 String 声明的是否为 AnsiString
Assignable typed constant	使用 Delphi 1.0 语法编译程序

3. Runtime errors: 设置应用程序执行时期(Run-time)错误检测, 将调试判断码加入编译
的可执行的应用程序中。Range checking 检查数组或字符串变量内容值是否在范围
内、I/O checking 检查输入输出装置是否错误、Overflow checking 检查整数运算是否
超出允许值。
4. Debugging: 设置应用程序编译时的调试参数。如表 1-3 所示。



表 1-3

选 项	功 能
Debug information	是否将调试码加入 Delphi compiled unit (*.DCU)
Local symbols	是否产生区域符号信息
Reference info	是否产生符号参考信息, 如果确定则可进一步设置是否只有定义信息(Definitions only)
Assertions	是否将程序代码的需求置入
Use Debug DCUs	使用包含调试信息的 DCU 文件

5. Messages: 设置是否在编译过程中提供警告信息以及提示信息的调试信息, 分别以 Show warnings 和 Show hints 选项来设置。如图 1-5 所示。

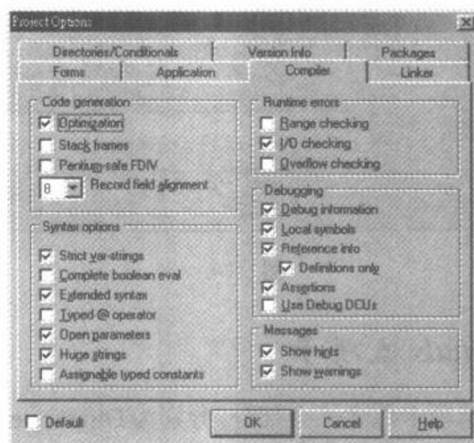


图 1-5

◆ Linker 选项卡

设置链接器(Linker)在链接文件时的相关设置, 共可分成五个属性群组。

1. Map file: 设置是否产生链接对照文件。可选择不产生对照文件或选择产生的对照文件是包含 segments 部分、publics 部分或 detailed(详细对照信息)。
2. Linker output: 设置链接器输出的链接文件种类, Pascal 的 DCU 文件或 C/C++的 OBJ 文件。
3. EXE and DLL options: 设置产生的 EXE(执行文件)或 DLL(动态链接文件)时的属性设置。

如表 1-4 所示。

表 1-4

选 项	功 能
Generate console application	是否产生包含控制装置的应用程序
Include TD32 debug Info	是否包含调试信息。如果确定, 则 EXE 或 DLL 文件会较大
Include remote debug symbols	如果使用远程调试, 是否包含远程调试符号



4. Memory size: 设置内存位置最大以及最小的堆栈(stack)空间。
5. EXE Description: 设置应用程序的文字说明数据。如图 1-6 所示。

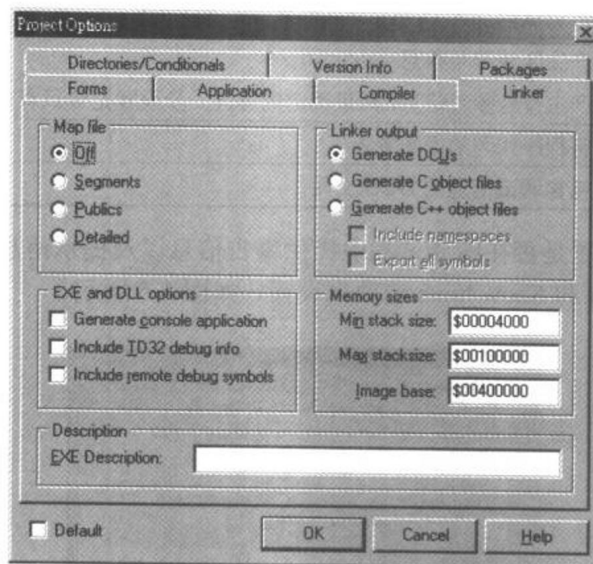


图 1-6

◆ Directories/Conditionals 选项卡

Directories/Conditional 选项卡包含三个部分,分别为 Directories、Conditionals 以及 Aliases。其中 Directories 是用来设置编译或连接结果输出的路径。

1. Output directory: 设置编译的 DCU 文件及 EXE/DLL 文件输出的路径。
2. Unit output directory: 指定另一个路径来放置 DCU 文件。
3. Search directory: 设置源程序文件或程序数据库等的放置路径。
4. Debug directory: 当调试器需要使用某些特定文件来辅助调试时,此参数用来设置这些文件放置的路径。
5. BPL output directory: 设置放置编译产生的可执行文件(*.BPL)的路径。
6. DCP output directory: 设置放置编译的 DCP 文件放置的路径。

此外 Conditionals 包含 Conditionals Defines 用来同时设置数个路径给编译器。而 Aliases 包含 Unit aliases 用来设置 VCL、操作系统 API 或程序使用到的程序代码单元 Source code(.pas)。可以设置多个程序代码单元文件为同一个别名(Alias),之后程序代码中只要使用 uses 语句引用此别名,即可在程序中引用到所设置的程序代码单元文件,关于 uses 语句用法参考 2-1-3 说明。

◆ Version Info 选项卡

Version Info 选项卡设置项目编译为可执行应用程序时,是否要加入版本信息。如果选中 Include version information in project 选项,则表示要在应用程序中加入版本信息。版本信息包含几个语句部分。如图 1-7 所示。



第1章 视口程序开发环境

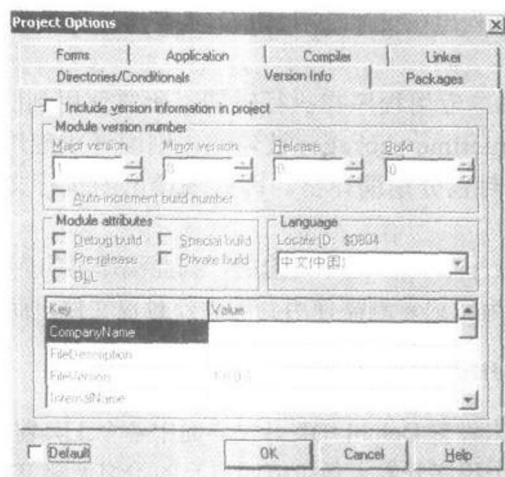


图 1-7

1. **Module Version Number:** 用来设置应用程序的版本数值，版本数值包含四个字段，可以手动调整或勾选 **Auto-increment build number** 自动更新版本数值。
2. **Module attributes:** 设置项目及应用程序的建立模式，版本增加的依据。
3. **Language ID:** 设置项目及应用程序所属的国别语言 ID。
4. **Key/Value:** 设置版本厂商的详细信息，如公司名称、应用程序描述、版权声明、产品名称等。

编译出的可执行文件(.exe)可以在 Windows 资源管理器中点选应用程序单击鼠标右键，弹出窗口画面，选择[属性]选项，即出现[属性]对话框，切换至[版本]选项卡即可看到在这个选项卡所设置的版本信息。

◆ Packages 选项卡

Packages 选项卡主要是选择设计时期(Design-time)，在 IDE 界面中使用的封包和执行时期(Run-time)项目编译出的应用程序所需的封包，主要有两个部分。如图 1-8 所示。

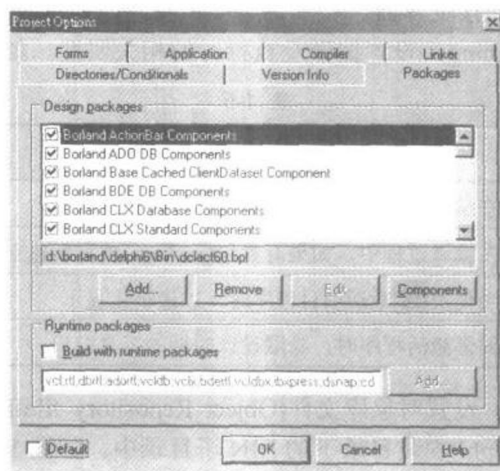


图 1-8