



全面掌握 Delphi 4

计算机实用教程



◆ 郭迅华 冯铁军 肖通波 冯锦锋 编



人民邮电出版社
PEOPLE'S POSTS &
TELECOMMUNICATIONS
PUBLISHING HOUSE



计算机实用教程

全面掌握 Delphi 4

郭迅华 冯铁军 肖通波 冯锦锋 编

JS 26

人民邮电出版社

内 容 提 要

Delphi 4 是 Inprise 公司于 1998 年 5 月推出的最新的 RAD 开发工具, 它将强大的 Delphi 的开发能力又提高了一个档次。本书力图从实用的角度, 全面地介绍 Delphi 4 的使用和开发。我们从 Delphi 4 的背景和开发环境入手, 大体介绍对象 Pascal 的语言要点, 其后的大部分篇幅用于讲述使用 Delphi 中的 VCL 组件库进行 GUI 开发的方法, 内容涉及多数常用的组件以及数据库和网络应用程序。介绍中力求简单明了、深入浅出, 使初学者能够迅速入手; 同时兼顾全面性, 使它对有一定经验的使用者也能够有所作用。

计算机实用教程 全面掌握 Delphi 4

-
- ◆ 编 郭迅华 冯铁军 肖通波 冯锦锋
责任编辑 潘春燕
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
北京鸿佳印刷厂印刷
新华书店总店北京发行所经销
 - ◆ 开本: 787 × 1092 1/16
印张: 20
字数: 485 千字 1998 年 12 月第 1 版
印数: 1 - 5 000 册 1998 年 12 月北京第 1 次印刷
ISBN 7-115-07568-9/TP·952
-

定价: 28.00 元

出版者的话

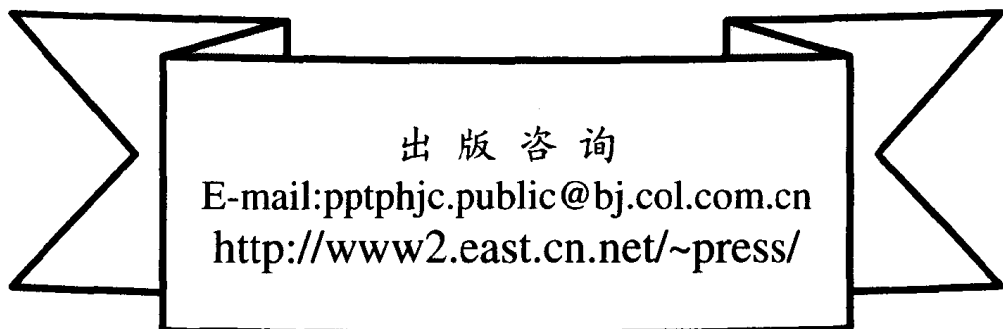
随着计算机技术的飞速发展,计算机应用的迅速推广,广大计算机开发者及使用者急切地需要了解计算机新技术、新软件及新知识。为进一步向全社会普及计算机知识,提高计算机使用人员的技术水平,使计算机在各个领域发挥更大作用,我们组织编写了这套既具有实用性,又适合培训和自学的《计算机实用教程》丛书。

本套丛书在一定程度上反映了计算机技术的发展趋势,并将社会上较为成功的操作技巧、操作方法吸收过来,适当加入一些服务于操作的原理,使读者不仅知道怎么做,还知道为什么这么做,从而达到举一反三、触类旁通的目的。

这套丛书重点突出、深浅适度、图文并茂、实用性强,每章都附有习题或思考题。以供读者自学和复习之用。

本套丛书首次推出的13种,受到了广大读者的欢迎和好评,为了更好地满足计算机爱好者的需求,我们还将不断充实与更新,愿它能为读者开辟一个崭新的天地,成为读者的良师益友。

1998年1月



编者的话

对 Delphi 的溢美之词已是汗牛充栋，然而对这样一个光彩夺目的开发工具来说，任何的褒奖都不会显得太过分。目前，在国外，有调查表明，使用 Delphi 的开发人员数量已经超过了 Visual Basic 的使用者，Delphi 开始取得了快速开发工具领域的领导地位。在国内，Delphi 也以其独有的魅力而为数以万计的程序员所情有独衷。Delphi 是一个通用的图形用户界面（GUI）快速开发工具，它是基于可视化和组件的概念而设计出来的，可以方便快捷地开发出精彩纷呈的程序来。Delphi 第一个版本的推出就风卷残云般令无数人倾倒，短期之内，Delphi 便红极一时。它也接二连三地给开发者 Borland 公司带来了一项项令人艳羡的产品桂冠。

正如许多人所知道的，组件对象是微软公司在 Visual Basic 中率先引入的，从最初问世开始它就明白无误地告诉人们，可视化快速开发代表着未来软件开发工作的潮流。因为这种设计第一次使软件开发在整体上超脱了专业人员的范围，使普通的用户也能够以一种十分直观而又简洁的方式开发出所需的应用程序来；同时，对专业程序员来说，可视化快速开发也逐步帮助他们摆脱了烦琐代码的纠缠，他们再也不必将大量的精力花费在图形用户界面代码的编写中，也不必深入到操作系统中去使用诸如 API 之类的艰深工具，因为可视化开发环境已经将这些包装了起来，自动解决了那些用户其实并不关心的问题，于是程序员就能够集中力量投入到软件功能及性能目标的实现中去。

尽管 Visual Basic 是第一个 RAD（应用程序快速开发）环境，但它的缺点从一开始就表现的和它的优点一样明显。首先它采用的程序语言原型是 BASIC，而 BASIC 是一种结构性极差的语言，虽然屡有改进，但其巨大随意性仍然存在，对于用户尤其是初学者来说，几乎是不经意之中，就会草率地写出结构松散的代码来，从而给后续的开发带来潜在的危险。这一缺陷在大型软件工程中将会更显突出，而开发中的严谨性对大工程却是至关重要的（也正是由于这个原因，Visual C++ 和 Borland C++ 至今还具有强大的生命力）。同时，当今的开发工作已经完全与面向对象的方法融为一体，组件本身就是面向对象的产物。我们知道，面向对象的方法，是从结构化程序设计的理论中发展而来的。遗憾的是，从前 BASIC 语言本身并不是一种结构化语言，因而 VB 中的语言尽管经过改造已经能够基本适应面向对象的要求，却也显得捉襟见肘。对于已经熟练掌握了 C++ 面向对象方法的程序员来说，VB 中的对象更是漏洞百出。此外，VB 最初采用的是伪编译的方式，这也使得程序的运行效率大大降低；其对数据库开发的处理方式也显得不够直接。

正是在这样的情况下，Borland 公司推出了苦心培育的 Delphi。它在事实上具备了 VB 的绝大多数优点，同时用对象 Pascal 语言克服了 BASIC 的顽症。VCL 组件库更给人以耳目一新的感受。同 Delphi 一同问世的，还有 Borland 独具匠心的开发理念以及一整套的先进的网络及数据库开发体系，从而使 Delphi 具有广泛的适应性以及前所未有的效率。天生丽质，自然引人注目。

对于 90 年代初以前的程序员来说，他们对 Borland 公司总是具有一种独特的感情

(Borland 公司现已改名为 Inprise, 这似乎表明, 它将要逐步地将航标转移到企业的集成解决方案中来, 但其在开发工具领域的巨大影响仍将会经久不衰)。许多人记忆犹新, 当初的 Turbo 系列语言编译器曾经怎样在 DOS 时代里为我们创造了一个完美的开发环境。Delphi 又一次以事实证明了 Borland 的地位和价值。其实, 也正是由于 Borland 曾经拥有 Turbo Pascal、Turbo C 以及 Borland C++ 这样的顶尖产品, 才能有今天这一脉相承的 Delphi。

在 Delphi 捷报频传的同时, C++Builder 成为 Inprise 公司的又一记重拳。它史无前例地将 C++ 语言与可视化快速开发结合起来, 作出了一次大胆而又精彩的开拓 (参阅郭迅华著:《Borland C++Builder 3 快速应用程序开发》, 北京大学出版社)。其目标是要将 C++ 的强大开发能力和 Delphi 的高速开发效率相综合, 为开发人员提供更为广阔的施展身手的舞台。C++Builder 已经取得了初步的成功, 并且将在不久的将来与 Delphi 一起成为 Inprise 公司 RAD 工具的“双璧”。二者相辅相成, 将更有利于全方位的完善。

1998 年 5 月, Delphi 4 的推出, 使其在网络和分布式数据库方面的开发能力又上了一个新台阶, 对 ActiveX、CORBA 以及 Midas 支持也使 Delphi 进一步完善了开发工作的平台兼容性。它不像 PowerBuilder 那样与数据库紧密结合, 但也因此具有通用开发方面的更大优势。Inprise 又一次以豪言重申了其在 Turbo Pascal 中的宣言: “Nothing is impossible!”。虽然我们仍然看到, Delphi 从前的某些缺陷, 如帮助系统的粗糙, 依然没有得到很大的改观, 但 Delphi 4 仍不失为目前最有生命力的开发工具和 Visual Basic 的最强硬的对手。VB 的王牌, 是它是微软的产品, 因而具有先天优势, 能够得天独厚地应用 Windows 系统的一切最新进展, 但 Delphi 凭借卓越的综合实力, 一步一步的占据了上风。

本书力图从实用的角度, 全面地介绍 Delphi 的使用和开发。我们从 Delphi 的背景和开发环境入手, 接着大体介绍对象 Pascal 的语言要点, 其后的大部分篇幅用于讲述使用 Delphi 中的 VCL 组件库进行 GUI 开发的方法, 内容涉及多数常用的组件以及数据库和网络应用程序。介绍中力求简单明了、深入浅出, 使初学者能够迅速入手; 同时兼顾全面性, 使它对有一定经验的使用者也能够有所作用。

此外需说明的是, 由于 Borland 公司已更名为 Inprise, 其产品却仍以 Borland 命名, 因而本书中总是尽量在二者中选择较为贴切的一个使用, 故它们可能交替出现。事实上, 这两个标识基本可视为是等价的。本书主要参阅的是郭迅华编写的《Borland C++Builder 3 快速应用程序开发》, 借鉴了该书较为完整明了的编写风格。本书第一章至第五章由冯铁军编写, 第六章至第十一章由肖勇波编写, 第十二章至第十七章由冯锦锋编写, 全书由郭迅华统稿。

最后谨向每一位大力协作帮助的人们致以深深的谢意。这些人包括: 张林鹏、石宪杰、李志军、刘理晖、陈其三、周刚兵。

由于时间仓促, 自觉错漏颇多, 在此先行致歉。若有任何批评意见, 还望不吝赐教, 以期日后能有所弥补。

编者

1998. 12

目 录

第 1 章 Delphi 4 概述.....	1
1.1 软件开发工具的发展.....	1
1.1.1 从 DOS 到 Windows 95: 开发理念的变迁.....	1
1.1.2 RAD 的产生.....	2
1.1.3 VB 和 Delphi.....	2
1.1.4 拖放技术 (Drag-and-Drop)	3
1.2 应用程序类库.....	3
1.2.1 程序库、类库和组件库	3
1.2.2 VCL.....	4
1.3 数据库应用程序开发.....	4
1.3.1 数据库引擎.....	5
1.3.2 数据库工具.....	5
1.3.3 数据库组件.....	5
1.4 开发组件.....	5
1.5 Internet 应用程序开发及其他.....	5
1.6 Delphi 4 的新特性.....	6
1.6.1 改进的集成开发环境	6
1.6.2 增强的 VCL.....	6
1.6.3 先进的 MIDAS 和 CORBA 技术与分布式对象计算	7
1.6.4 全面支持 Oracle 8 的对象关联扩展功能	7
1.6.5 其它特性.....	7
第 2 章 Delphi 4 的集成开发环境 (IDE)	8
2.1 IDE 概貌.....	8
2.2 IDE 基本结构.....	8
2.2.1 菜单和快捷工具栏	8
2.2.2 窗体 (Form) 和窗体设计器 (Form Designer)	9
2.2.3 组件调色板 (Component Palette)	10
2.2.4 代码编辑器 (Code Editor)	10
2.2.5 对象监视器 (Object Inspector)	11
2.3 菜单体系.....	11
2.3.1 “File”菜单.....	11
2.3.2 “Edit”菜单.....	13
2.3.3 “Search”菜单.....	15
2.3.4 “View”菜单.....	16
2.3.5 “Project”菜单.....	18
2.3.6 “Run”菜单.....	19
2.3.7 “Component”菜单.....	21

2.3.8 “Database”菜单	22
2.3.9 “Tools”菜单	23
2.3.10 “WorkGroup”菜单	23
2.3.11 “Help”菜单	24
2.3.12 右键快速菜单	24
2.3.13 快捷键	25
2.4 编辑器的使用	25
2.4.1 激活编辑器	25
2.4.2 使用编辑器	25
2.4.3 定制编辑器	25
2.4.4 代码洞察 (Code Insight)	25
2.5 IDE 的定制	26
2.5.1 组件调色板定制	26
2.5.2 快捷工具栏定制	27
第 3 章 Delphi 4 的程序组成	28
3.1 文件类型	28
3.1.1 工程文件	28
3.1.2 窗体文件	29
3.1.3 包文件	31
3.1.4 代码单元文件	31
3.1.5 桌面配置文件	31
3.1.6 工程目标文件	31
3.2 变量、函数、对象和组件	31
3.3 窗体设计和代码编写、GUI 设计风格	32
3.3.1 窗体设计和代码编写	32
3.3.2 窗体设计和代码编写的关系	32
3.3.3 GUI 设计风格	32
3.4 让程序运行起来	33
3.4.1 “Hello World” (一)	33
第 4 章 Object Pascal 基本程序设计	34
4.1 常量和变量	34
4.1.1 标识符 (Identifier)	34
4.1.2 常量	35
4.1.3 变量	36
4.1.4 基本 (Fundamental) 数据类型	36
4.1.5 常变量	38
4.2 数据结构	38
4.2.1 枚举类型 (Enumerated types)	39
4.2.2 子界类型 (Subrange types)	39

4.2.3 数组类型 (Array types).....	39
4.2.4 集合类型 (Set types).....	40
4.2.5 记录类型 (Record types).....	40
4.2.6 变体类型 (Variant types)	41
4.3 运算符 (Operators) 和表达式 (Expressions)	42
4.3.1 算术 (Arithmetic) 运算符.....	42
4.3.2 逻辑运算符 (Logical operators)	42
4.3.3 关系运算符 (Relational operators)	43
4.3.4 @ 运算符.....	43
4.3.5 运算符优先级	43
4.4 流程控制.....	44
4.4.1 条件语句.....	44
4.4.2 情况语句.....	46
4.4.3 for 循环语句	47
4.4.4 while 循环语句	47
4.4.5 repeat 循环语句	48
4.5 分支.....	48
4.5.1 goto.....	48
4.5.2 break.....	49
4.5.3 continue	49
4.5.4 exit.....	50
4.5.5 halt.....	50
4.6 程序 (Program)	50
4.6.1 单元(Unit).....	51
4.6.2 过程与函数(Procedure and Function).....	52
第 5 章 面向对象的程序设计	54
5.1 关于软件工程.....	54
5.1.1 软件危机.....	54
5.1.2 软件生命周期	54
5.1.3 软件工程的目标和原则	55
5.1.4 从结构化到 OOP.....	55
5.1.5 再谈面向对象	56
5.2 对象的引入.....	57
5.2.1 抽象和封装.....	57
5.2.2 函数和过程、模块、单元.....	57
5.2.3 对象.....	58
5.2.4 构造函数(constructor)和析构函数(destructor)	60
5.3 多态和重载.....	62
5.4 继承和派生.....	63

5.4.1 派生类.....	63
5.5 多态和虚拟.....	65
5.5.1 虚拟函数.....	65
第6章 创建应用程序.....	67
6.1 对象仓库 (Object Repository)	67
6.1.1 “New Items”中的标签	68
6.1.2 三种应用方式.....	69
6.1.3 浏览对象仓库.....	69
6.1.4 对象仓库的管理.....	70
6.1.5 向对象仓库内添加对象.....	71
6.2 向导 (Wizards)	72
6.2.1 用向导创建应用程序.....	72
6.2.2 用向导创建对话框.....	75
6.3 应用程序开发的一般模式.....	76
6.3.1 建立应用程序工程.....	76
6.3.2 在窗体上添加组件.....	76
6.3.3 设定对象的属性.....	77
6.3.4 编写事件处理 (Event Handler)	79
6.3.5 编译运行.....	79
6.3.6 添加成员.....	80
6.3.7 “Hello World” (三)	81
6.4 设计时的技巧.....	82
6.4.1 连续建立同一种组件.....	82
6.4.2 选取多个对象.....	82
6.4.3 使用剪贴板.....	83
6.4.4 在成组组件上添加组件.....	83
6.4.5 对象对齐.....	86
6.5 窗体对象.....	86
6.5.1 外形及状态设定.....	87
6.5.2 控制管理.....	88
6.5.3 显示与关闭.....	89
6.5.4 “Hello World” (四)	90
6.5.5 动态创建窗体.....	92
第7章 可视组件库 (VCL)	94
7.1 关于组件.....	94
7.1.1 组件和组件类.....	94
7.1.2 可见组件和不可见组件.....	94
7.1.3 属性 (Properties)	94
7.1.4 方法 (Methods)	95

7.1.5 事件 (Events)	95
7.1.6 事件句柄 (Event Handler)	95
7.2 组件说明.....	96
7.2.1 标准 (Standard) 组件	96
7.2.2 附加 (Additional) 组件	97
7.2.3 32 位 Windows (Win32) 组件.....	97
7.2.4 系统 (System) 组件	98
7.2.5 互联网络 (Internet) 组件.....	99
7.2.6 数据存取 (Data Access) 组件.....	101
7.2.7 数据控制 (Data Controls) 组件	102
7.2.8 Midas 组件.....	103
7.2.9 多维数据 (Decision Cube) 组件.....	104
7.2.10 快速报表 (QReport) 组件.....	105
7.2.11 对话框 (Dialogs) 组件	106
7.2.12 Windows 3.1 (Win 3.1) 组件.....	107
7.2.13 范例 (Samples) 组件.....	108
7.2.14 ActiveX (ActiveX) 组件	109
7.3 补充说明.....	110
7.3.1 重要的共同属性	110
7.3.2 基本的事件和方法	113
第 8 章 输入、输出和打印	115
8.1 文字的输入输出.....	115
8.1.1 Label 组件 (Standard)	115
8.1.2 Edit 组件 (Standard)	115
8.1.3 Memo 组件 (Standard)	117
8.1.4 MaskEdit 组件 (Additional)	117
8.1.5 RichEdit 组件 (Win32)	119
8.1.6 StaticText 组件 (Additional)	121
8.2 信息对话框.....	122
8.2.1 InputBox 和 InputQuery 函数.....	122
8.2.2 消息对话框.....	123
8.3 OpenFileDialog 和 SaveDialog 组件 (Dialogs)	127
8.4 打印.....	131
8.4.1 使用 Printer 组件	132
8.4.2 打印窗体.....	133
8.4.3 PrintDialog 组件 (Dialogs)	133
8.4.4 PrinterSetupDialog 组件 (Dialogs)	135
8.4.5 范例.....	136
第 9 章 按钮、成组组件与选择表	138

9.1 按钮.....	138
9.1.1 OnClick 事件响应.....	138
9.1.2 关于 ModalResult	138
9.1.3 默认 (Default) 按钮和取消 (Cancel) 按钮.....	139
9.1.4 按钮上的图标.....	139
9.1.5 BitBtn 的 Kind 属性	141
9.1.6 快速按钮 SpeedButton 组件.....	142
9.2 单选钮和复选框.....	146
9.2.1 选项值的设定	147
9.2.2 单选钮 RadioButton (Standard)	147
9.2.3 复选框 CheckBox (Standard)	147
9.3 成组组件.....	148
9.3.1 GroupBox 组件 (Standard)	148
9.3.2 RadioGroup 组件 (Standard)	149
9.3.3 Panel 组件 (Standard)	152
9.3.4 ScrollBox 组件 (Additional)	153
9.3.5 成组组件的其他问题	154
9.4 Bevel 组件和 Splitter 组件	154
9.5 菜单设计器.....	155
9.5.1 菜单设计器的基本操作	156
9.5.2 菜单命令的响应	157
9.5.3 菜单设计器的右键快速菜单	157
9.5.4 预建菜单模板及其汉化	158
9.6 菜单组件.....	159
9.6.1 指定使用的 MainMenu	159
9.6.2 合并主菜单、MDI 简介	159
9.6.3 使用 PopupMenu 组件.....	161
9.7 列表框和组合框.....	161
9.7.1 ListBox 列表框组件 (Standard)	161
9.7.2 ComboBox 组合框组件 (Standard)	163
9.7.3 CheckListBox 组件 (Additional)	163
9.7.4 FileListBox 及其他.....	163
第 10 章 图形、图像和多媒体.....	165
10.1 执行阶段的绘图、Canvas 对象	165
10.1.1 Canvas 的基本属性	165
10.1.2 绘图方法.....	167
10.1.3 关于重画.....	169
10.2 其他绘图组件.....	169
10.2.1 PaintBox 组件 (System)	169

10.2.2 Shape 组件 (Additional)	170
10.3 图像文件处理	171
10.3.1 重要属性	171
10.3.2 基本操作	172
10.3.3 Image 组件 (Additional)	174
10.3.4 范例	174
10.4 无声视频播放 (动画)	181
10.4.1 Animate 组件 (Win32)	181
10.4.2 动画窗体	182
10.5 媒体播放	184
10.5.1 媒体文件类型	184
10.5.2 MediaPlayer 组件 (System)	184
10.5.3 MediaPlayer 组件的重要属性和方法	186
10.5.4 关于 DirectX	187
第 11 章 其他重要组件	188
11.1 系统对象	188
11.1.1 Application 对象	188
11.1.2 提示信息的显示	189
11.1.3 Screen 对象	190
11.1.4 关于剪贴板的补充说明	190
11.1.5 Timer 计时器组件 (System)	191
11.2 通用对话框	192
11.2.1 FontDialog 对话框组件 (Dialogs)	192
11.2.2 ColorDialog 对话框组件 (Dialogs)	194
11.2.3 FindDialog 和 ReplaceDialog 组件 (Dialogs)	195
11.3 多页组件	195
11.3.1 Notebook、TabSet 和 TabbedNotebook (Win 3.1)	196
11.3.2 TabControl 和 PageControl (Win32)	196
11.4 杂项	197
11.4.1 ScrollBar 组件 (Standard)	197
11.4.2 TrackBar 组件 (Win32)	201
11.4.3 ProgressBar (Win32) 和 Gauge (Samples)	201
11.4.4 StringGrid 组件 (Additional)	203
11.4.5 HeaderControl 组件 (Win32)	205
11.4.6 TreeView 和 ListView 组件 (Win32)	206
第 12 章 工程管理与程序调试	208
12.1 工程组织的基本问题	208
12.1.1 目录管理	208
12.1.2 文件命名	209

12.1.3 对象命名	209
12.2 工程管理器 (Project Manager)	209
12.2.1 工程管理器的使用	209
12.2.2 工程选项设置	210
12.2.3 版本控制	211
12.3 关于调试	211
12.4 调试选项设置	212
12.4.1 编译器中的调试选项	212
12.5 调试器使用	212
12.5.1 程序运行方式	213
12.5.2 使用断点	213
12.5.3 变量监视	214
12.6 WinSight32	215
第 13 章 数据库应用程序开发	217
13.1 Delphi 的数据库体系	218
13.1.1 关系数据模型	218
13.1.2 数据库管理模式	218
13.1.3 BDE 概述	219
13.1.4 BDE 管理	219
13.1.5 其他数据库工具	220
13.2 用 Database Desktop 创建数据库表	222
13.2.1 数据库应用程序的一般开发过程	222
13.2.2 创建数据库表	222
13.2.3 索引	224
13.2.4 补充说明	225
13.2.5 数据输入	226
13.2.6 Database Desktop 的其他功能	227
13.3 使用 Form Wizard	227
13.3.1 建立数据库应用窗体	227
13.3.2 运行	229
13.4 数据库访问机制	230
13.4.1 数据的基本流程	231
13.4.2 Table 和 Query 组件 (Data Access)	232
13.4.3 数据访问	234
13.4.4 DataSource 组件 (Data Access)	236
13.4.5 数据集中的字段	236
13.4.6 字段的使用	237
13.5 数据控件 (Data Controls)	238
13.5.1 一般性质	238

13.5.2 导航组件 DBNavigator.....	238
13.5.3 范例.....	239
13.6 数据查询.....	243
13.6.1 数据过滤.....	243
13.6.2 范围设定.....	244
13.6.3 使用索引 (Indexes)	245
13.6.4 数据集的状态 (States)	246
13.7 计算出 (Calculated) 的字段.....	247
13.8 使用多个数据库表.....	248
13.8.1 使用查找 (Lookup) 字段.....	249
13.8.2 主从复合表.....	251
13.8.3 用 Form Wizard 创建主从复合表窗体.....	251
13.8.4 有关主从复合结构的属性.....	253
第 14 章 报表和图表.....	256
14.1 Quick Report	256
14.1.1 报表模板.....	256
14.1.2 报表向导 (QuickReport Wizard)	257
14.1.3 快速报表组件.....	258
14.1.4 报表数据控件.....	260
14.1.5 报表的预览和打印.....	261
14.1.6 定制预览窗体.....	261
14.2 Chart 图表.....	262
14.2.1 图表向导 (Chart Wizard)	262
14.2.2 使用 Chart 组件.....	267
14.3 Decision Cube 简介.....	268
第 15 章 Internet 开发.....	270
15.1 实例.....	270
15.1.1 搭建应用程序框架.....	271
15.2 应用程序实现.....	272
15.2.1 初始化.....	272
15.2.2 定位与寻址.....	273
15.2.3 导航操作.....	273
第 16 章 应用程序发行.....	275
16.1 帮助系统.....	275
16.1.1 软件说明文档.....	275
16.1.2 联机帮助系统.....	276
16.2 使用注册表.....	278
16.2.1 Registry 对象.....	278
16.3 再谈包.....	279

16.4 创建安装程序	279
第 17 章 从 VB 到 Delphi	284
17.1 开发环境	284
17.1.1 重要功能键比较	284
17.2 代码	286
17.2.1 变量及数据类型	286
17.2.2 赋值语句	286
17.2.3 字符串	286
17.2.4 注释代码	287
17.3 组件	287
17.3.1 窗体(Forms)	288
17.3.2 按钮 (CommandButtons)	290
17.3.3 编辑框 (Text Boxes)	290
17.3.4 下拉框(ListBoxes)	291
17.3.5 文件控制	292
17.3.6 单元(Units)	293
17.3.7 变量范围	293
17.3.8 条件语句	294
17.3.9 循环语句	296
17.3.10 过程和函数	298
17.3.11 控制数组(Control Arrays)	299
17.3.12 对象变量 (Object Variables)	299
17.4 图像 (Graphics)	300
17.4.1 画布 (Canvas)	301
17.4.2 图象处理实现	301

第1章 Delphi 4 概述

Delphi 4 是以一个先驱者的身份出现的, 她为应用程序开发工具指明了一条走向成功的必经之路。如果你是编程新手, 想走进纷繁绚丽的 Windows 图形界面编程世界, 想享受迅速快捷地编制出功能强大外表迷人的软件, 想用一种舒适美妙的方式解决自己的计算机软件需求, 那么 Delphi 将会给你前所未有的新鲜和快意; 如果你是一个熟练的程序员, 在为程序开发工作寻找一种更为完善的工具, 以使自己繁重的工作得到轻松的改进, 正在寻找一种能够方便地创建数据库、Web 以及多线程应用程序的开发工具, Delphi 4 就是你的最佳选择。她集完美的界面、快速的可视化编程、完善的窗口系统的底层接口于一身, 把强大的对象 Pascal 语言和快捷便利的 RAD 相结合, 为程序开发开创了一个崭新的视野。

在这个最新的版本中, Delphi 保留了以往一切强大的特性, 同时在目前的种种热门的发展方向上进行了成功的扩展。Delphi 4 依旧是从前那个雅俗共赏、轻松上手的 Delphi, 同时也已经成长为代表着当今软件开发潮流的先锋派开发工具。在这一章中, 我们将粗略地巡视这一令人心醉神迷的软件的出现背景和发展历程, 并大致地窥视她的风貌。

1.1 软件开发工具的发展

计算机发展的历史同时也是软件发展的历史。从 1946 年的 ENIAC 出现开始, 就有了计算机程序。然而, 软件产业的真正兴起却是在高级语言的广泛应用之后。PC 和 DOS 的出现为计算机软件的开发打开了全新的天地。回顾软件开发工具的发展, 真可谓是白驹过隙、沧海桑田。

1.1.1 从 DOS 到 Windows 95: 开发理念的变迁

众所周知, DOS 曾经是最为成功的个人计算机操作系统, 几乎可以说是一个时代的标志。不过它采用的是命令行的用户界面, 这足以让许多使用者望而生畏。百余条的操作命令掌管了计算机的运作, 计算机的一切行动几乎都在使用者一字一句的催促下进行。那时, 计算机程序必须相对直接地对系统进行管理和指派。因此, 那一度是软件屈从于硬件的时代。最初的程序设计差不多成了一门艺术, 人们所关心的, 是如何用最少的资源实现最多的功能。随着软硬件的发展, 软件开发的效率开始受到广泛的关注, 然而, 程序的立足点, 仍然是直接对机器的控制。

后来, Windows 出现了, 并随着 3.0 版的推出风靡了全球。令人耳目一新的 GUI (图形用户接口, Graphic User Interface) 设计擦去了阻隔在用户和机器之间的纱帘, 使计算机以一种更为平易亲切的面貌出现在使用者面前。从此, Windows 成为 Microsoft 倾尽心血培育的一代天骄, 她的目标不仅仅是用户界面上的革新, 而是要推动一种计算机软件开发