



程序员指南丛书



Delphi 7 编程基础

- Delphi 7 采用面向对象的程序设计、组件化的编程方式、快速 Pascal 编译器、OLE 自动化、ActiveX 编程、数据库以及 Internet/Intranet 编程支持。
- Delphi 7 最激动人心、也是最大的发展在于对 Linux 平台下应用程序开发的更强大支持，以及对网络程序开发如 Web Service 的强大支持。
- 本书结合大量实例，对 Delphi 7 的几乎所有方面都进行了介绍，适合初、中级读者学习 Delphi 编程。

肖建 杨新臣 等编著



清华大学出版社

<http://www.tup.tsinghua.edu.cn>

Delphi 7 编程基础

肖 建 杨新臣 等 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

Delphi 7 是 Inprise(原 Borland)公司推出的 Delphi 编程系列软件的最新版本。本书结合大量实例,循序渐进地介绍了 Delphi 7 的使用与编程。首先介绍的是 Delphi 7 几种最基本组件的使用;然后以此为基础进行程序开发,并进一步介绍各种组件的应用;最后则以专题的形式介绍了 Delphi 7 在多媒体、网络、数据库等方面的应用。

本书各章节相对独立又紧密联系,内容由浅入深,讲解详细,重点突出,适合初、中级读者学习 Delphi 编程,也可作为各类培训班的培训教材。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Delphi 7 编程基础/肖建等编著,一北京:清华大学出版社,2003.1

(程序员指南丛书)

ISBN 7-302-06139-4

I . D . . . II . 肖 . . . III . 软件工具 - 程序设计 IV . TP311.56

中国版本图书馆 CIP 数据核字(2002)第 097393 号

出 版 者:清华大学出版社(北京清华大学学研大厦,邮编:100084)

<http://www.tup.tsinghua.edu.cn>

<http://www.tup.com.cn>

责任编辑:胡先福

印 刷 者:北京市清华园胶印厂

发 行 者:新华书店总店北京发行所

开 本:787×1092 1/16 **印张:**24 **字数:**600 千字

版 次:2003 年 1 月第 1 版 2003 年 1 月第 1 次印刷

书 号:ISBN 7-302-06139-4/TP·3670

印 数:0001 ~ 4000

定 价:36.00 元

前 言

随着计算机技术的发展,计算机软件产业也随之迅速崛起,自从面向对象的编程概念提出之后,可视化编程的技术应运而生,相继出现了 Visual C++、Visual Basic、C++ Builder 和 Delphi 等可视化应用程序开发工具。与此同时,人们的编程思想也发生了根本的变化,为了提高应用程序的高效性和统一性,可以尽量地应用别人的编程经验,而把主要精力放在应用程序的功能设计上。

经过 Borland 公司的软件设计师们多年的改进,如今 Delphi 程序开发工具在各个方面都有了不同程度的提高,如数据库体系结构、VCL 控件开发、Web 应用程序和安全线程等,从而使得 Delphi 的功能更加强大,使用起来也更加方便灵活,提高了应用程序的开发效率。总之,Delphi 是开发 Windows 应用程序最为强大的工具之一,无论是 Windows 程序设计的初学者,还是有经验的 Windows 程序员,利用 Delphi 都可以迅速开发出自己满意的应用程序。

Delphi 7 是 Borland 公司推出的 Delphi 编程系列软件的最新版本。它采用面向对象的程序设计、组件化的编程方式、快速 Pascal 编译器、OLE 自动化、ActiveX 编程、数据库以及 Internet/Intranet 编程支持,从而使得 Delphi 7 开发功能更加强大,同时也更加易于学习和使用。

Delphi 从一开始出现在市场上便受到了广大用户的喜爱。Delphi 7 更是在以前版本的基础上有了长足的发展:一方面继续保持着原来的优势,例如提供数量众多、功能强大、使用方便的各种组件,又如非常优秀的数据库开发支持;另一方面弥补了以前版本的不足,大力开拓新的功能和应用,例如提供了更完备更优秀的网络功能组件,使 Delphi 有了更加强大的网络功能,适应了当今网络迅猛发展的趋势,同时对各组件页面中所包含组件的归类也更加趋于合理。

在所有新增功能中,Delphi 7 最激动人心、也是最大的发展在于对 Linux 平台下应用程序开发的更强大支持,以及对网络程序开发如 Web Service 的强大支持。由于本书介绍的是如何使用 Delphi 7 开发 Windows 应用程序,因此对于第一个功能就只能略过不予介绍,希望读者能够谅解。

当然,对于一些用户来讲,可能 Delphi 7 会让他们有一点点失望,因为它并没有像传闻中那样能够支持 .NET 应用程序的开发。但是,Delphi 7 已经为支持 .NET 迈出了第一步,首先表现在增加了一些关于 unsafe 类型的编译警告。另外,在 Delphi 7 中附带了 Delphi .NET Preview,因为 IDE 还没开发出来,VCL .NET 还没有包括进去,因此尚无 RAD 环境,只是带了编译器、.NET 链接器,但可以写所有基于 .NET 的程序了(因为没有 RAD 环境,所以只能用代码生成窗体和组件)。Borland 基于 .NET 的全新 IDE 开发代号叫做 Galileo,2003 年才能推出,据称可在 IDE 中使用所有基于 .NET 的语言,包括 C#。

可以说,Delphi 7 已经具备了作为最优秀的编程开发软件的一切条件:它与 Microsoft 公司的 VC/C++ 相比,其优势是具有简单易懂的 IDE 开发环境;与 VB 相比,它的代码更加规范,增强了程序的可移植性,并且开发效率更高。

因此,对于一名编程爱好者来说,Delphi 7 应该是不容错过的。正是为了满足广大读者

的愿望和需求——迅速掌握 Windows 环境下的 Delphi 编程方法,我们编写了本书。本书主要面向 Delphi 的初、中级用户,因此书中会有一些必要的理论介绍。但是作者认为,优秀的编程指导书籍应该是在实例中体现编程方法和技巧,通过自己动手编程来理解整个编程体系,因此书中采用了大量鲜活实用的例程,紧密结合各部分的知识进行示范,引导读者亲自动手,在实际应用和开发中激发兴趣,从而迅速加入 Delphi 高级用户的行列。

作者在写作时以 Delphi 7 在各方面的应用为准则,制定了全书的布局,各章节都对应着 Delphi 在某方面的应用,具有相对的独立性,以方便读者分章节给自己制定学习计划,这一点读者可以从目录上看出来。同时各章节又是紧密联系在一起,秉承着另外一条准则——由浅入深。首先在前面几章介绍 Delphi 7 几种最基本组件的使用;随后几章则以此为基础进行程序开发,并进一步介绍各种组件的应用;最后几章则以专题的形式介绍了 Delphi 7 在多媒体、网络、数据库等方面的应用。这两条准则便是这本书的经纬,希望读者在阅读本书之前要了解这一点,以便更有效地使用本书。

除封面署名外,参与本书编写、校对等工作的还有段爱华、岳剑平、冯之明、刘洋、浦汉清、白树先、高志刚、杜云鹏、杨丹、宋永春、汤兴宝、章文智、邱西良、康建生、李红源等,在此一并表示感谢。限于作者水平,书中不足和纰漏之处在所难免,恳请广大读者批评指正。

编 者
2002 年 11 月

目 录

第 1 章 Delphi 7 入门	1
1.1 Delphi 基本概念介绍	1
1.1.1 Delphi 的基本形式	1
1.1.2 面向对象编程的概念	1
1.2 Delphi 7 新特性	2
1.2.1 Delphi 7 的新特点和优点	2
1.2.2 增强的网络应用功能	3
1.2.3 新增组件和组件新特性	4
1.3 第一个 Delphi 程序	5
1.4 本章小结	7
第 2 章 Delphi 7 开发环境	9
2.1 Delphi 7 主窗口	10
2.1.1 Menu(菜单栏)	10
2.1.2 Speed Bar(快捷工具条)	31
2.1.3 Component Palette(组件模板)	32
2.2 代码编辑窗口	33
2.2.1 快捷菜单	33
2.2.2 代码浏览	36
2.3 窗体设计窗口	36
2.4 对象查看窗口	38
2.5 项目管理窗口	39
2.5.1 增加及删除文件	40
2.5.2 编译项目	40
2.6 本章小结	40
第 3 章 Delphi 语言	41
3.1 编写 Delphi 程序代码	41
3.1.1 编写赋值语句	41
3.1.2 标识符的说明与使用	42
3.1.3 过程与函数	44
3.1.4 跳转语句	47
3.1.5 循环语句	48
3.1.6 程序模块	49
3.1.7 关于作用范围	50
3.1.8 编写一个过程或函数	51

3.1.9	定义新的数据类型	55
3.1.10	Delphi 的库单元 unil	60
3.2	用 Delphi 的对象进行编程	62
3.2.1	什么是对象	62
3.2.2	从一个对象中继承数据和方法	65
3.2.3	对象的范围	66
3.2.4	对象公有域和私有域的说明	67
3.2.5	访问对象的域和方法	67
3.2.6	对象变量的赋值	68
3.2.7	建立非可视化对象	70
3.3	Delphi 异常处理机制与异常类	71
3.3.1	运行时间库异常类(RTL Exception)	72
3.3.2	对象异常类	75
3.3.3	部件异常类	77
3.4	异常保护	78
3.4.1	需要保护的资源	78
3.4.2	产生一个资源保护块	79
3.5	异常响应	80
3.5.1	使用异常实例	81
3.5.2	提供默认响应	82
3.5.3	响应一组异常	82
3.5.4	异常的重引发和处理嵌套	83
3.5.5	自定义异常类的应用	83
3.5.6	利用异常响应编程	88
3.6	程序调试简介	89
3.6.1	调试的准备和开始	90
3.6.2	程序运行的控制	91
3.6.3	断点的使用	91
3.6.4	监视数据的值	94
3.7	其他调试工具	96
3.7.1	Turbo Debugger	96
3.7.2	WinSight	96
3.8	使用 Browser	98
3.8.1	设置过滤器	99
3.8.2	Details 区的显示页	99
3.9	本章小结	101
第 4 章	常用组件	102
4.1	组件及对象	102
4.1.1	组件	102
4.1.2	组件的分类	104

4.1.3 Delphi 的标准组件	104
4.1.4 对象	105
4.2 按钮类组件	105
4.3 标签和编辑组件	107
4.3.1 用标签和编辑框设计窗体	108
4.3.2 掩码编辑框	110
4.3.3 Memo 组件和 RichEdit 组件	111
4.4 单选按钮和复选框	116
4.4.1 单选按钮和复选框	116
4.4.2 单选按钮组	118
4.5 列表框和组合列表框	120
4.5.1 列表框	120
4.5.2 自绘列表框	124
4.5.3 组合列表框	127
4.6 本章小结	128
第 5 章 菜单设计	129
5.1 创建主菜单	129
5.1.1 使用 MainMenu Designer 建立菜单	129
5.2 动态改变菜单	133
5.2.1 菜单项的有效和无效	134
5.2.2 复选标记	135
5.2.3 隐藏菜单	136
5.2.4 单选菜单项	137
5.2.5 动态创建菜单项	138
5.3 打开和存储文件的菜单项	142
5.4 设计快捷菜单	144
5.4.1 快捷菜单设计	144
5.5 使用菜单模板	146
5.6 图形菜单项	148
5.6.1 定制图形复选标记	148
5.6.2 位图菜单项	150
5.7 本章小结	152
第 6 章 窗体设计	153
6.1 窗体属性	153
6.2 窗体设计	155
6.2.1 创建窗体	155
6.2.2 添加标题	155
6.2.3 选择和放置组件	156
6.2.4 编辑组件	157

6.2.5	控制 Tab 顺序和 Create 顺序	158
6.2.6	改变窗体设计环境	158
6.2.7	保存窗体	159
6.2.8	窗体的文本描述	159
6.3	利用模板生成窗体	161
6.4	多页窗体	163
6.5	建立分割窗体	177
6.6	本章小结	178
第 7 章	多文档界面设计	179
7.1	MDI 的组成	179
7.1.1	父窗口	179
7.1.2	子窗口	179
7.2	MDI 应用程序设计	181
7.3	MDI 菜单管理	189
7.3.1	菜单合并	190
7.3.2	菜单覆盖	190
7.4	使用 MDI 模板	191
7.5	本章小结	195
第 8 章	图形程序设计	196
8.1	图形设计综述	196
8.2	基本图形绘制	197
8.2.1	直线的绘制	197
8.2.2	椭圆的绘制	199
8.2.3	多边形绘制	201
8.2.4	矩形的绘制	203
8.3	复杂图形绘制	205
8.4	简单动画	209
8.5	图像处理	213
8.5.1	图像动态显示	213
8.5.2	图像动画效果	217
8.6	图像特技	219
8.6.1	镜像效果	219
8.6.2	推进、交错、百叶窗等效果	225
8.7	本章小结	232
第 9 章	对话框设计	233
9.1	模式对话框	233
9.2	无模式对话框	236
9.3	消息框	239
9.4	通用对话框组件	241

9.4.1	文件对话框组件	242
9.4.2	字体对话框组件	244
9.4.3	颜色对话框组件	246
9.4.4	打印机打印设置对话框组件	247
9.4.5	查找及替换对话框组件	248
9.5	本章小结	252
第 10 章	数据库应用程序设计	253
10.1	数据库管理系统	253
10.2	Delphi 7 数据库应用程序的体系结构	254
10.3	基本数据库组件	254
10.4	使用 Table 组件编程	270
10.4.1	第一个例子	271
10.4.2	第二个例子	277
10.5	使用 Query 组件编程	284
10.5.1	SQL 属性	285
10.5.2	动态 SQL 表达式	288
10.6	设计 InterBase 管理器	291
10.6.1	InterBase 6.5 简介	292
10.6.2	程序外观设计	294
10.6.3	程序功能实现	297
10.7	本章小结	305
第 11 章	多媒体编程	306
11.1	多媒体组件	306
11.1.1	TmediaPlayer 组件	306
11.1.2	使用 TmediaPlayer 组件的实例	312
11.2	CD 播放器	313
11.3	使用 Win32 的 MCI 函数	317
11.3.1	MCI 简介	318
11.3.2	使用 MCI 编程实例	318
11.4	本章小结	330
第 12 章	多线程编程	331
12.1	关于线程	331
12.1.1	线程的优先级	332
12.1.2	线程的同步	333
12.1.3	线程的局部存储(TLS)	333
12.2	定义线程对象	333
12.2.1	创建线程对象	333
12.2.2	初始化线程对象	335
12.2.3	编写线程函数	336

12.2.4	编写线程的清除代码	338
12.3	使用线程对象	338
12.3.1	线程的同步	338
12.3.2	执行线程对象	339
12.4	多线程编程实例	340
12.5	本章小结	349
第 13 章	网络编程	350
13.1	聊天程序	350
13.1.1	程序功能分析	350
13.1.2	程序外观设计	351
13.1.3	程序功能实现	352
13.2	开发 ASP 应用系统	356
13.2.1	程序功能分析	356
13.2.2	程序外观设计	360
13.2.3	程序功能实现	364
13.3	Web 服务	365
13.3.1	程序功能分析	365
13.3.2	程序外观设计	368
13.3.3	程序功能实现	371
13.4	本章小结	374

第 1 章 Delphi 7 入门

Delphi 是全新的可视化编程环境,为我们提供了一种方便、快捷的 Windows 应用程序开发工具。它使用了 Microsoft Windows 图形用户界面的许多先进特性和设计思想,采用了弹性可重复利用的完整的面向对象程序语言(Object Oriented Language)、当今世界上最快的编辑器和最为领先的数据库技术。对于广大的程序开发人员来讲,使用 Delphi 开发应用软件,无疑会大大地提高编程效率,而且随着应用的深入,将会发现编程不再是枯燥无味的工作——Delphi 的每一个设计细节,都将带来一份欣喜。

Delphi 7 的出现,更是给广大的 Delphi 用户带来了喜讯,因为它在性能上更加完善,组件更加丰富,特别是在 Web 服务功能方面,更是在原来优势的基础上加以提高,给网络编程带来了更大的便利。下面我们将从 Delphi 的最基本概念讲起,带领 Delphi 爱好者进入 Delphi 7 的殿堂。

1.1 Delphi 基本概念介绍

1.1.1 Delphi 的基本形式

在 Delphi 7 中所使用的编程语言已经正式被命名为 Delphi 语言,Delphi 语言实际上是 Pascal 语言的一种版本,但它与传统的 Pascal 语言有天壤之别。一个 Delphi 程序首先是应用程序框架,而这一框架正是应用程序的“骨架”。在骨架上即使没有附着任何东西,仍可以严格地按照设计运行。你的工作只是在“骨架”中加入程序。默认的应用程序是一个空白的窗体(Form),你可以运行它,结果得到一个空白的窗口。这个窗口具有 Windows 窗口的全部性质,可以被放大缩小、移动、最大最小化等,但你却没有编写一程序。因此,可以说应用程序框架通过提供所有应用程序共有的东西,为用户应用程序的开发打下了良好的基础。Delphi 已经做好了一切基础工作——程序框架就是一个已经完成的可运行应用程序,只是不处理任何事情。你所需要做的,只是在程序中加入完成你所需功能的代码而已。

在空白窗口的背后,应用程序的框架正在等待用户的输入。由于你并未告诉它接收到用户输入后作何反应,窗口除了响应 Windows 的基本操作(移动、缩放等)外,它只是接受用户的输入,然后再忽略。Delphi 把 Windows 编程的回调、句柄处理等繁复过程都放在一个不可见的 Romulam 覆盖物下面,这样你可以不为它们所困扰,轻松从容地对可视部件进行编程。

1.1.2 面向对象编程的概念

面向对象的程序设计(Object Oriented Programming, OOP)是 Delphi 诞生的基础。OOP 立

意于创建软件重用代码,具备更好地模拟现实世界环境的能力,这使它被公认为是自上而下编程的优胜者。它通过给程序中加入扩展语句,把函数“封装”进 Windows 编程所必需的“对象”中。面向对象的编程语言使得复杂的工作条理清晰、编写容易。说它是一场革命,不是对对象本身而言,而是对它们处理工作的能力而言。对象并不与传统程序设计和编程方法兼容,只是部分面向对象反而会使情形更糟。除非整个开发环境都是面向对象的,否则对象产生的好处还没有带来的麻烦多。而 Delphi 是完全面向对象的,这就使得 Delphi 成为一种触手可及的促进软件重用的开发工具,从而具有强大的吸引力。

一些早期的具有 OOP 性能的程序语言,如 C++、Pascal、Smalltalk 等,虽然具有面向对象的特征,但不能轻松地画出可视化对象,与用户交互能力较差,程序员仍然要编写大量的代码。Delphi 的推出,填补了这项空白。你不必自己建立对象,只要在提供的程序框架中加入完成功能的代码,其余的都交给 Delphi 去做。欲生成漂亮的界面和结构良好的程序丝毫不必绞尽脑汁,Delphi 将帮助你轻松地完成。它允许在一个具有真正 OOP 扩展的可视化编程环境中,使用它的 Delphi 语言。这种革命性的组合,使得可视化编程与面向对象的开发框架紧密地结合起来。

1.2 Delphi 7 新特性

今年 8 月份, Borland 公司推出了 Delphi 的最新版本——Delphi 7。Borland 公司首次通过 Delphi 7 推出一条独立开发 .NET 应用的道路。Delphi 7 是 Windows 环境下进行交叉平台 (cross-platform) 快速应用开发 (Rapid Application Development, RAD) 的软件。Delphi 7 在企业应用设计和开发上起到重要作用。开发者能够通过使用先进的统一建模语言 (Unified Modeling Language, UML) 和模型驱动体系 (Model Driven Architecture, MDA) 技术来加快从概念到产品的开发过程。

相对于 Delphi 6, Delphi 7 在 VCL 上面可能没有很大的改进,但是它在支持 .NET 上迈出了第一步,也在网络编程方面有所加强。同时, Delphi 7 可以让企业按照自己的时间表,使用开发者已经熟悉的技巧和资源,实现向 .NET 的平滑过渡。Delphi 7 新增的高品质应用生命周期开发路线,诸如建模、MDA、报告和交叉平台开发等,开创了一个新型和激动人心的应用开发时期,并走上持续革新的应用开发新台阶。

Delphi 7 Studio 是从设计到开发,模型驱动体系 (Model Driven Architecture, MDA) 方案的大全,它集成了建模、开发,和基于 Windows 平台的电子商务开发等功能。它扩展了 Delphi 7 的电子商务和 Web 服务框架,它的全集成技术增加了开发者的开发能力,它的 .NET 移植工具包让开发者能够向 .NET 应用过渡。Delphi 7 Studio 将先进的应用开发技术集成为一个易于使用的开发包,缩短了应用开发周期,加速了市场投入进程。

1.2.1 Delphi 7 的新特点和优点

相对于 Delphi 6, Delphi 7 在下面这几个方面有比较大的进步:

- 企业应用 MDA 开发:让开发者在根本上减少从设计到开发所需的编码数量和时间,

从而加速应用开发过程。

- RAD 可视 Web 开发:在 Delphi 7 环境下,开发者能够进行可视 Web 应用开发,它的应用模式(Application Mode)框架能够在切断与服务器连接情况下进行透明的对话(session)管理。
- 内建 Linux 跨平台支持器:Delphi 7 装配 Delphi 语言版的 Borland Kylix 3。它提供了第一个高性能的可视集成开发环境(IDE),可迅速创建数据库、图形用户接口(GUI)、Web 和 Linux 操作系统的 Web 服务应用。
- 企业型报告:可创建跨平台报告,帮助分析应用执行的效率。
- 免版税的 DataSnap(前称多层应用开发(Multi-tier Application Deployment, MIDAS)):新版的 Delphi 7 DataSnap 许可证允许免除从单层和客户/服务模式无缝过渡到多层应用模式的附加费。
- Windows XP 应用:Delphi 7 支持 Windows XP 主题(Theme),开发者可以利用 Windows XP 用户接口主题(User Interface Themes)来开发应用。

1.2.2 增强的网络应用功能

Delphi 7 从 AtoZed 软件公司引入了 IntraWeb 技术,现在用户就可以用 Delphi 7 提供的窗体工具来使用 IntraWeb 技术开发网络服务程序,当然也可以使用 IntraWeb 来开发 Web Broker 和 WebSnap 应用程序的页面。

除此之外,Delphi 7 在网络应用上的拓展还包括加强了对 Web Services 的支持。

新添加的组件和编译器上的一些变化使用户可以轻松地使用 SOAP(Simple Object Access Protocol)组件来编写网络服务(Web Services)程序以及访问这些网络服务的客户端应用程序。网络服务(Web Services)程序是独立的模块程序,可以在网上进行发表和调用。网络服务程序为描述网络所能提供的服务制定了良好的界面。

Delphi 7 的 SOAP 技术已经可以成功地应用于 Windows 操作系统和 Linux 操作系统中,这样就为跨平台的分布式应用程序提供很好的基础。

使用 Delphi 7,用户既可以编写执行这些网络服务的服务器端程序,也可以很轻松地编写访问这些服务的客户端程序。如果用户同时编写服务器端和客户端程序,那么就可以在两个程序中共用一个单元文件,只要该单元文件定义了网络服务的所有接口。

对于 Web Services 的支持加强方面,具体表现在如下几点:

新的 UDDI 浏览器

Delphi 7 提供了新的 UDDI 浏览器,使用户可以通过 UDDI 接口方便地从 Web Services 中定位或者导出 WSDL 文档。

SOAP headers

SOAP 记录了用户在客户端和服务端传递数据时的一些信息,Delphi 7 提供了一些新的类和接口,使用户能够方便地从中提取和插入头节点(header nodes)。

Attachments

Web Services 应用程序,包括客户端和服务端,现在可以处理 attachments 了,attachments(TSOAPAttachment 的子类)作为多窗体的一部分采用 SOAP 编码的信息进行传递,当一个应

用程序接收到传递过来的 attachment 之后,会自动将其保存为一个临时文件,然后就可以在应用程序中使用。

1.2.3 新增组件和组件新特性

支持 Windows XP 主题

Delphi 7 支持 Windows XP 主题(Theme),开发者可以利用 Windows XP 用户接口主题(User Interface Themes)来开发应用。在 Delphi 7 环境下开发的应用程序会自动使用它在 Windows XP 系统下能检测到的新的 Windows 控件。

新增单元文件

Delphi 7 中新增了一个名为 DBClientActns 的单元文件,在该单元文件中包含了 3 个新的动作组件,这 3 个动作组件主要帮助用户开发与客户端数据集相关的数据库应用程序。这 3 个动作组件分别为: TClientDataSetApply、TClientDataSetUndo 和 TClientDataSetRevert。

新增组件

在组件面板的 dbExpress 组件页中,新增了一个 TSimpleDataSet 组件,该组件替代了原来的 TSQLClientDataSet 组件,用于开发简单的 two-tier 数据库应用程序。

在组件面板的 Dialogs 组件页中,新增了一个 TPageSetupDialog 组件,使用该组件可以为应用程序提供一个 Windows 标准页面设置对话框。

在组件面板的 Additional 组件页中,新增了 3 个组件: TXPColorMap、TStandardColorMap 和 TTWilightColorMap,这 3 个组件可以为菜单和工具条设置颜色。

另外,Indy 组件页里面的组件也都升级到了 9.0 版本。

组件新增特性

CLX 版本的 TOpenDialog 和 TSaveDialog 组件经过扩展之后,可以支持新增的特性例如文件的预览。

VCL 版本的 TCustomForm 组件新增了两个属性: ScreenSnap 和 SnapBuffer 属性,这两个属性用来控制一个窗体在拖动时,窗体中的内容是否跟着同步变化,使用过一些播放器的读者应该不难理解该功能的作用。

TCustomComboBoxEx 组件新增了一个 AutoCompleteOptions 属性,当该属性被设置为 true 时,该组合框就可以响应用户的按键动作。

CLX 版本的所有继承自 TOpenDialog 和 TQtDialog 的对话框组件新增了一个 UseNativeDialog 属性,该属性的默认值为 true,此时应用程序使用 Windows 的标准对话框;而如果将该属性设置为 false,应用程序中的对话框将仍采用原来的 Qt 对话框。

Delphi 7 在组件方面主要的变动差不多如此,其他微小细节方面的变化只有在实际编程中才能体会出来,因此我们将在后面章节中继续具体介绍。后面我们将用一个简单的例子来演示开发 Delphi 程序的一般过程。

1.3 第一个 Delphi 程序

本节通过一个简单的例子来说明 Delphi 是如何实现可视化和面向对象的。

我们假设你已经在自己电脑上安装了 Delphi 7, 单击“开始”按钮, 在弹出菜单中选择“程序”选项, 然后依次从级联菜单中选择 Borland Delphi 7 和 Delphi 7, 如图 1-1 所示, 就进入了 Delphi 7 的界面。

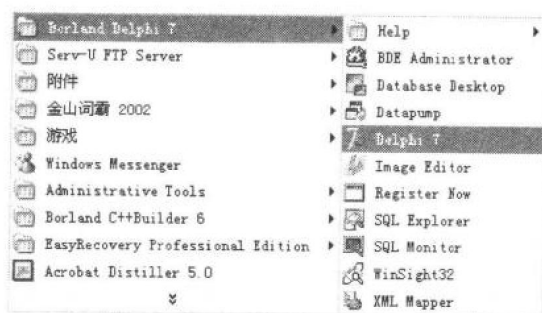


图 1-1 启动 Delphi 7

初次进入 Delphi 7, 其界面如图 1-2 所示。

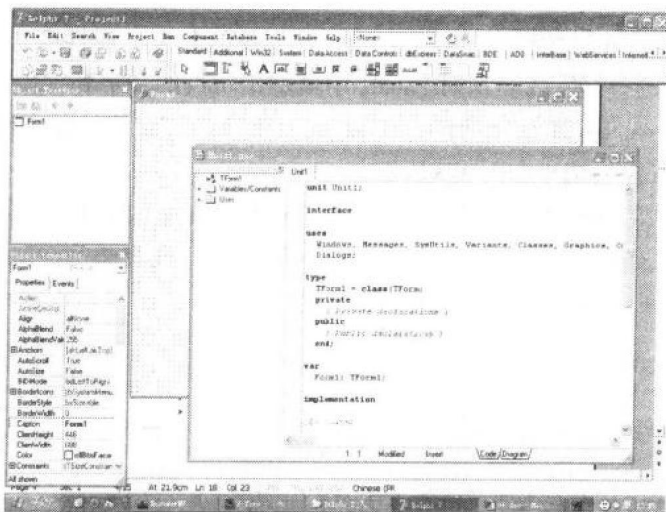


图 1-2 Delphi 7 界面

下面举一个简单的例子来说明 Delphi 是如何实现可视化和面向对象的。

(1) 利用刚开始 Delphi 7 时 Delphi 自动提供的界面就可以开始进行设计。如果用户不小心关掉了窗体窗口, 那么可以通过用 File 菜单下的 New Application 命令创建一个新的应用文件。

(2) 如图 1-3 所示设计好窗体的界面。这里用到以下几个组件和过程:


```
type
  TForm1 = class(TForm)
    Label1: TLabel;
    Button1: TButton;
    Button2: TButton;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
  end;
```



图 1-3 设计界面

然后将窗体的 Caption 属性设为“第一个应用程序”，这时窗体的标题变为如图 1-3 中所示的文字。

再将标签的 Caption 属性设为“你好, Delphi 7!”,并单击 Font 属性旁边的“...”按钮,在弹出的字体编辑窗口调整好字体。这时 Label1 中显示出所设的文字和字体。

将两个按钮的 caption 属性分别设为“Welcome”和“Exit”,直接显示在组件上。这时完成了窗体界面的设计过程,结果如图 1-3 所示。

(3) 接下来就要编写事件响应代码。首先选中 Button1 组件,选择对象查看器中的 Events 选项卡,双击 Onclick 事件右侧,进入代码编辑窗口。或者通过在窗体窗口上双击 Button1 组件来完成。在代码编辑窗口中编写如下代码:

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  ShowMessage('欢迎进入 Delphi 的世界!');
  Button1.Caption:= 'hello';
  Label1.Caption:= '这是我的第一个 Delphi 程序!';
end;
```

(4) 然后选中 Button2 组件,双击它进入代码编辑窗口,编写如下代码:

```
procedure TForm1.Button2Click(Sender: TObject);
begin
  close;
end;
```

(5) 在完成这些工作之后,编程工作可以说已经结束了。选择 Run 菜单下的 Run 命令项或按 F9 运行程序。如果有错误,会在代码编辑窗口下方显示错误信息。