



Delphi 3.0

开发技术实例

窦万峰 王保保 魏天功 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

URL: <http://www.phei.com.cn>

Delphi 3.0 开发技术实例

窦万峰 王保保 魏天功 编著

電子工業出版社
Publishing House of Electronics Industry

内 容 提 要

Delphi 以其非常容易使用的可视化开发方式、高效的编译代码、高性能的客户/服务器数据访问、完全支持图形图像以及 Internet 技术和强有力的异常处理机制等特点,深受广大软件开发人员的喜爱。本书讲述了如何使用 Delphi 来开发实用的 Windows 应用软件,重点放在 Delphi 开发的实用技术和高级编程技巧方面,旨在让读者掌握 Delphi 开发的实用技术和如何编写高效的程序代码等等。

全书共分为十六章,主要包括 Delphi 编程基本知识、Object Pascal 语言及应用编程、Delphi 菜单编程技术、Delphi 多窗口应用、文件操作技术、图形图像技术、多媒体开发技术、Delphi 数据库技术、Internet 编程技术、报表快速制作技术和 Delphi 的调试工具及异常处理等内容。

本书资料丰富,内容翔实,列举了大量的开发实例,具有易学易用、简洁明了的特点。本书既适合于软件开发人员使用,又能够使 Delphi 初学者在学习和使用中受益匪浅,也可作为高等院校广大师生和科研人员进行软件开发的必备参考书。

图书在版编目(CIP)数据

Delphi 3.0 开发技术实例/窦万峰等编著。-北京:电子工业出版社,1998.4

ISBN 7-5053-4685-7

I.D… II. 窦… III. 软件工具, Delphi3.0 IV. TP311.56

中国版本图书馆 CIP 数据核字(98)第 03003 号

书 名: Delphi 3.0 开发技术实例

编 著 者: 窦万峰 王保保 魏天功

责任编辑: 张 琛

印 刷 者: 北京天宇星印刷厂

出版发行: 电子工业出版社出版、发行 URL: <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036 发行部电话 68214070

经 销: 各地新华书店经销

开 本: 787×1092 1/16 印张: 25.5 字数: 653 千字

版 次: 1998 年 8 月第 1 版 1998 年 8 月第 1 次印刷

书 号: ISBN 7-5053-4685-7
TP·2246

定 价: 32.00 元

凡购买电子工业出版社的图书,如有缺页、倒页、脱页者,本社发行部负责调换

版权所有·翻印必究

前 言

Delphi 软件开发工具是 Borland 公司推出的一个完全导向的可视化系统开发环境的工具。它具有功能强大、运行速度快、易于学习和使用以及开发迅速等特点,一经推出就受到广大用户的喜爱。Delphi 结合了可视化技术、面向对象技术、数据库技术以及网络技术等多种先进的软件编程技术和思想,并使用了世界上最快的编译器(Compiler),使其成为创建功能丰富、界面友好的 Windows 应用程序的工具之一。

一、Delphi 特点

Delphi 是一个运行在 Windows 环境下的可视化应用程序开发工具软件,它可以使你用最少的程序量来得到最高的开发效率,从而快速地开发一个 Windows 的应用程序。Delphi 提供了易于使用的图形化工具供软件开发人员创建程序中的可视化部份,开发者可以按照自己的喜好及用户要求,随心所欲地安排窗口的外观,使之调整得能恰到好处。使用 Delphi 提供的可视化工具,开发人员可以很容易地观察窗口设计中的动态变化,并及时地作出修改。

Delphi 所采用的底层编程语言是 Borland 公司的 Object Pascal。Object Pascal 语言是一种相当简炼的编程语言,其语法严格、结构清晰、阅读性强,并且它还是在软件设计上使人们形成一种良好编程习惯的语言。Object Pascal 的语言风格可能更适合于可视化的程序设计语言。

Delphi 是以面向对象的方式设计程序代码的。在 Delphi 中,一切都被视为对象。编写面向对象的程序能够重复使用开发者自己的工作成果,并使程序维护起来更为容易。程序的扩展也变得更为方便,并减少了修改所带来的大量错误。继承性可使设计人员在已有对象类型的基础上创建新的对象,这个新对象将继承其先前对象的所有事件、属性及方法,并且可添加新过程和方法,具有很大的灵活性。

Delphi 的代码编辑器是非常友好和清晰的。在代码编辑器中,使用者可以设置关键字的颜色,使得程序代码清晰明快且修改方便。Delphi 的代码编辑器始终在监视当前窗体的变化,并能及时地响应当前的修改。这种动态跟踪手段减少了设计人员因窗体的变化而对程序代码的修改所花费的大量时间。

Delphi 提供可视化组件库和定制并为 Delphi 的组件模板增加页面组件功能。动态链接库(DLLs)使设计人员有可能创建经过编译的程序代码库,这种程序代码库可以为许多不同的应用程序所使用。使用动态链接库可以使程序的维护工作变得更加轻松自如。

Delphi 是设计成模块化、可重用的开发工具。把自己认为可重用的程序,或者一些应用程序以模板的形式保存成窗体表单,或以动态链接库的形式编译设计人员的应用程序,这将是非常有用的。

Delphi 的数据库应用程序的前端开发工具是一个易用且高效的开发工具,它支持开发强大的桌面或主从结构的 Client/Server(客户/服务器)的数据库应用程序和报表。开发人员可以使用 Delphi 为一个数据库管理系统建立一个前端,而不必过多地考虑数据库本身是存放在本地硬盘上,还是位于一个远程的 SQL 数据库服务器中。

Delphi 还提供了支持 Internet 编程技术,从而提供网络方面的服务功能。

二、本书的内容

本书共分为十六章,具体介绍:

第一章——Delphi 各版本的特点,编程的基本概念和术语。

第二章——Delphi 编程的步骤和方法,常用组件的功能及使用。

第三章——Object Pascal 语言和一些编程实例。

第四章——多窗体应用编程技术,涉及多窗体调用的概念和编程方法。

第五章——使用 Delphi 编写 Windows 标准菜单技术,Delphi 提供的标准文件对话框组件和实用编程等。

第六章——图形编程技术,涉及图形对象的属性、事件和方法,并给出几个应用于鼠标的绘图方法。

第七章——文件操作技术,讲述了文件组件及编程,文本文件、格式化文件、随机文件、初始化文件的处理以及 DLL 创建和调用等。

第八章——图像编程技术,主要介绍了图像编程的几个组件功能,图像文件格式及获取等,最后给出几个实例。

第九章——多媒体应用编程技术,包括多媒体组件的属性、事件和方法以及几个例程。

第十章——Delphi2.0 新增的重要组件及基本使用。

第十一章——Delphi3.0 提供的 Internet 编程技术,包括各个协议组件的介绍和使用例程。

第十二章——Delphi 的集成调试器使用和异常处理,介绍各种调试方法和异常保护措施。

第十三章——Delphi 的数据库基本知识,包括 Delphi 数据库基本概念、数据库工具及使用、数据库组件介绍和基本使用以及数据库开发的步骤。

第十四章——数据库应用程序设计,包括数据库应用程序的组成、数据库组件使用和数据库程序设计等。

第十五章——数据库报表制作工具 ReportSmith,包括报表制作、样式处理和报表打印等技术。

第十六章——Delphi3.0 提供的 QuickReport2.0 报表快速制作及编程技术,详细讲述了 QuickReport2.0 的特点、条框的概念以及 QuickReport2.0 提供的报表组件功能和编程。

三、本书的特点

本书是一本介绍使用 Delphi 快速开发 Windows 应用程序的指导书。书中使用大量的编程实例,循序渐进地介绍 Delphi 编程的基本原理和实际应用,旨在使读者迅速掌握 Delphi 的使用、编程及其本质。本书中所列举的大量例子由易到难,并具有代表性,起到举一反三的作用。

本书也是在出版社同仁共同努力下得以顺利出版,在此对他们表示衷心感谢。通过对本书的阅读,读者一定能够从中获得很大的裨益,这也是我们最大的愿望。

本书适用于广大计算软件开发人员和计算机爱好者学习和使用。

由于时间和水平有限,不妥之处在所难免,恳请广大读者不吝赐教。

编 者

目 录

第一章 Delphi 基础知识	(1)
1.1 Delphi 各版本功能及特点	(1)
1.1.1 Delphi1.0 特性	(1)
1.1.2 Delphi2.0 特性	(2)
1.1.3 Delphi3.0 特性	(2)
1.1.4 Delphi 的特点	(3)
1.2 Delphi 开发基础知识	(5)
1.2.1 主屏幕菜单	(6)
1.2.2 加速条	(6)
1.2.3 对象监视器	(6)
1.2.4 组件模板	(7)
1.2.5 窗体	(7)
1.2.6 单元	(7)
1.2.7 工程项目和窗体表单	(8)
1.2.8 属性	(8)
1.2.9 对象方法	(9)
1.2.10 事件	(9)
1.2.11 事件处理	(9)
1.3 Delphi 窗体工具	(9)
1.3.1 Delphi 模板	(9)
1.3.2 Delphi 专家	(10)
第二章 Delphi 基本编程	(12)
2.1 Delphi 程序设计步骤	(12)
2.2 Delphi 属性	(14)
2.2.1 属性设置	(14)
2.2.2 窗体属性	(15)
2.3 Delphi 事件	(19)
2.3.1 建立一个事件过程	(19)
2.3.2 窗体表单事件	(20)
2.3.3 一个使用拖放事件的例程	(22)
2.4 Delphi 的窗体与表单文件	(24)
2.5 常用组件介绍	(26)
2.5.1 文本相关组件	(26)

2.5.2	Button 按钮和 BitBtn 位按钮	(28)
2.5.3	组件使用例程	(30)
2.5.4	列表框与组合框	(31)
2.5.5	单选按钮与复选按钮	(34)
2.5.6	GroupBox 和 RadioGroup	(35)
2.5.7	Timer 组件	(35)
第三章 Object Pascal 语言及其编程		(41)
3.1	Delphi 语句	(41)
3.1.1	Delphi 语句类型	(41)
3.1.2	使用代码设置属性	(42)
3.2	Delphi 对象	(43)
3.2.1	面向对象编程基本术语	(43)
3.2.2	可视组件库对象	(43)
3.3	Object Pascal 块和保留字	(46)
3.3.1	Object Pascal 块	(46)
3.3.2	保留字	(48)
3.4	Delphi 变量	(48)
3.5	数和运算符	(51)
3.5.1	整数和运算符	(51)
3.5.2	实数	(62)
3.6	函数与过程	(62)
3.6.1	Delphi 内部函数和过程	(63)
3.6.2	Windows API 函数	(64)
3.6.3	用户定义的函数和过程	(64)
3.7	数组	(65)
3.8	Delphi 控制语句	(67)
3.8.1	循环语句	(67)
3.8.2	条件语句	(69)
3.8.3	Case 语句	(75)
3.9	用户定义数据类型	(77)
3.9.1	枚举类型	(77)
3.9.2	记录类型	(78)
3.9.3	集合类型	(78)
第四章 Delphi 多窗体应用编程		(80)
4.1	多窗体应用的概念	(80)
4.2	建立一个多窗体应用程序	(82)
4.3	多窗体的属性与事件及方法	(83)
4.3.1	多窗体的属性	(83)

4.3.2 多窗体的事件	(84)
4.3.3 多窗体的方法	(85)
4.4 一个多窗体应用的例子	(85)
4.5 创建顺序跟踪	(91)
4.6 工具栏设置	(92)
4.7 菜单运作合并	(93)
4.8 使用 MDI 模板	(95)
4.9 建立 MDI 框架的进一步讨论	(96)
4.10 几个重要的对象	(97)
4.10.1 Clipboard 对象	(97)
4.10.2 Printer 对象	(99)
4.10.3 Screen 对象	(100)
4.10.4 Application 对象	(100)
4.11 一个多窗体应用实例	(101)
第五章 窗体菜单设计	(110)
5.1 下拉式菜单设计	(110)
5.1.1 MainMenu 组件对象	(110)
5.1.2 下拉式菜单设计	(110)
5.2 弹出式菜单设计	(113)
5.3 添加面板和快捷按钮	(114)
5.4 对话框组件	(115)
5.4.1 文件打开对话框	(115)
5.4.2 文件保存对话框	(116)
5.4.3 打印对话框	(116)
5.4.4 打印机设置对话框	(116)
5.4.5 颜色设置对话框	(117)
5.4.6 字体设置对话框	(117)
5.4.7 查找对话框	(117)
5.4.8 替换对话框	(117)
5.5 对话框组件使用	(117)
5.6 菜单编程举例	(120)
5.7 从菜单模板自动生成菜单窗体	(126)
5.8 Delphi 3.0 的图像文件存取对话框	(128)
5.8.1 Open Picture Dialog 和 Save Picture Dialog	(128)
5.8.2 使用 Open Picture Dialog 和 Save Picture Dialog 组例的例程	(129)
第六章 图形编程技术	(132)
6.1 画布(Canvas)对象	(132)
6.1.1 像素数组	(132)

6.1.2	画笔	(133)
6.1.3	画刷	(135)
6.1.4	画布对象的综合应用	(136)
6.2	绘制图形	(144)
6.2.1	绘制直线和折线	(144)
6.2.2	绘制几何图形	(144)
6.3	动态演示	(145)
6.3.1	随机绘图程序	(145)
6.3.2	一个动态演示程序	(147)
6.4	手工绘制线条	(153)
6.4.1	响应鼠标事件	(153)
6.4.2	手工绘制线条	(155)
6.4.3	绘制直线	(156)
6.5	绘图快捷按钮设计	(157)
6.6	自定义画笔和画刷	(160)
6.6.1	自定义画笔	(160)
6.6.2	自定义画刷	(162)
6.7	位图对象	(163)
6.8	图形文件处理	(163)
6.9	Delphi 多线程编程与图形处理	(176)
6.9.1	多线程概念	(176)
6.9.2	线程的属性、方法和事件	(177)
6.9.3	使用线程并行处理图形数据举例	(179)
第七章	文件操作	(186)
7.1	文件操作行命令	(186)
7.2	文件相关组件	(188)
7.2.1	文件相关组件	(188)
7.2.2	文件组件使用	(189)
7.3	文本文件的处理	(190)
7.4	格式化文件的处理	(192)
7.5	随机文件的处理	(193)
7.6	初始化文件的处理	(194)
7.7	创建 DLL	(197)
7.7.1	调用 Windows API	(197)
7.7.2	DLL 的调用	(198)
7.7.3	动态加载 DLL	(199)
第八章	图像处理编程	(201)
8.1	图像组件	(201)

8.1.1 Image 组件	(201)
8.1.2 Paintbox 组件	(201)
8.1.3 Shape 组件	(202)
8.1.4 Panel 组件	(203)
8.1.5 Bevel 组件	(204)
8.2 Bitmap 对象及使用	(204)
8.2.1 Bitmap 对象	(204)
8.2.2 使用 Bitmap 对象	(205)
8.3 图像格式的认识	(206)
8.3.1 图像格式	(206)
8.3.2 图像对象属性	(206)
8.3.3 图像的获取	(207)
8.4 图像演示例程	(207)
8.5 自动生成版本窗体	(218)
第九章 多媒体应用编程	(220)
9.1 多媒体简介	(220)
9.2 MediaPlayer 组件	(220)
9.2.1 MediaPlayer 属性	(221)
9.2.2 MediaPlayer 事件	(223)
9.3 一个简单的例程	(223)
9.4 动画控制组件介绍及使用	(225)
9.4.1 TAnimate 组件对象	(225)
9.4.2 使用 TAnimate 组件	(228)
第十章 Delphi3.0 新增常用组件	(230)
10.1 滚动组件对象	(230)
10.1.1 ProgressBar 组件	(230)
10.1.2 UpDown 组件	(230)
10.1.3 TrackBar 组件	(231)
10.1.4 滚动组件使用举例	(232)
10.2 StatusBar 组件	(233)
10.3 RichEdit 组件	(234)
10.4 Outline 组件	(235)
10.5 TreeView 和 ListView 组件	(239)
10.5.1 TreeView 组件	(239)
10.5.2 ListView 组件	(240)
10.6 Imagelist 组件	(241)
10.7 TabbedNotebook 组件	(242)
10.8 TabControl 组件	(243)

10.9 PageControl 组件	(244)
第十一章 Delphi 的 Internet/Intranet 技术	(245)
11.1 Delphi 与 Internet/Intranet 概览	(245)
11.2 Internet 组件板	(248)
11.3 WWW 浏览	(249)
11.4 TCP 组件使用	(255)
11.5 电子邮件	(260)
第十二章 Delphi 调试器和异常处理	(277)
12.1 Delphi 集成调试器	(277)
12.1.1 常见错误种类	(277)
12.1.2 集成调试器	(277)
12.2 使用断点	(279)
12.2.1 设置断点	(279)
12.2.2 断点列表窗口	(280)
12.2.3 断点条件设定	(280)
12.3 使用监视功能	(280)
12.3.1 Watch List 窗口	(280)
12.3.2 使用 Evaluate/Modify 对话框	(281)
12.4 程序运行方式	(282)
12.4.1 执行到光标位置	(282)
12.4.2 单步执行	(282)
12.4.3 程序暂停和复位	(283)
12.5 对象浏览器	(283)
12.6 异常处理	(284)
12.6.1 创建保护块	(284)
12.6.2 异常处理	(286)
12.6.3 引发异常	(288)
12.6.4 创建一个定制的异常	(289)
第十三章 Delphi 数据库功能概述	(291)
13.1 Delphi 关系数据库概述	(291)
13.1.1 数据库概念	(291)
13.1.2 Delphi 数据库框架	(292)
13.1.3 数据字典	(293)
13.1.4 Delphi 数据库工具	(294)
13.2 数据库配置应用程序	(294)
13.2.1 Driver 页面	(295)
13.2.2 Aliases 页面	(296)

13.3 数据库桌面	(297)
13.4 数据库组件对象	(301)
13.4.1 数据库组件	(301)
13.4.2 显示一个图片	(302)
13.5 数据库应用程序开发步骤	(304)
13.5.1 数据库选择	(304)
13.5.2 数据库表格设计	(304)
13.5.3 数据库程序设计	(304)
13.6 数据库应用程序开发全过程概述	(305)
第十四章 数据库应用程序开发	(306)
14.1 Delphi 数据库组件功能	(306)
14.1.1 Table 和 Query 相应属性和方法	(306)
14.1.2 Data Source 组件	(312)
14.1.3 Data Control 组件板	(312)
14.2 创建数据库显示程序	(313)
14.2.1 自动创建数据库显示程序	(314)
14.2.2 用户定义数据库显示窗体	(317)
14.3 使用数据表中的字段	(319)
14.3.1 读写字段	(319)
14.3.2 设置字段值	(320)
14.4 计算字段	(321)
14.4.1 数据库表之间的关系	(321)
14.4.2 计算字段	(322)
14.5 主细表设计	(326)
14.5.1 主要索引与次级索引	(326)
14.5.2 链接表	(326)
14.6 查找数据	(328)
14.7 数据输入	(329)
14.7.1 制作数据输入窗体	(329)
14.7.2 数据排序	(331)
14.8 数据查询和浏览	(332)
14.8.1 使用 Tquery 组件	(332)
14.8.2 建立 SQL 查询陈述	(332)
14.8.3 动态查询	(333)
14.9 使用多线程的数据库查询	(334)
14.9.1 调用汇编程序	(334)
14.9.2 TSession 组件和 Date base 组件	(335)
14.9.3 使用线程进行数据库查询	(337)
14.9.4 数据库查询主窗体	(343)

第十五章 报表制作工具——ReportSmith	(352)
15.1 关于 ReportSmith	(352)
15.2 TReport 组件	(353)
15.2.1 开始 RepotSmith	(353)
15.2.2 定制报表样式	(356)
15.3 配置 ReportSmith	(357)
15.4 新建和制作报表	(358)
15.4.1 选择报表样式	(358)
15.4.2 创建自己的报表样式	(359)
15.4.3 链接字段数据	(359)
15.5 观察打印报表	(364)
第十六章 QuickReport 快速报表	(365)
16.1 QuickReport2.0 特点	(365)
16.2 创建一个基本报表	(366)
16.3 TQuickRep 组件	(366)
16.4 QuickReport 文本组件	(369)
附录 使用 SQL 语言	(375)

第一章 Delphi 基础知识

1.1 Delphi 各版本功能及特点

Delphi 是 Borland 公司推出的一个完全导向的可视化系统开发环境的工具,并被评为美国最优秀软件之一。它具有功能强大、运行速度快、易于学习使用以及开发迅速等特点,一经推出就受到广大用户的喜爱。Delphi 结合了可视化技术、面向对象技术、数据库技术、Internet 技术等多种先进的软件编程技术和思想,并使用了世界上最快的编译器(Compiler),使其成为创建功能丰富、界面友好的 Windows 应用软件工具之一。

Delphi1.0 是在 1995 年 7 月推出,其丰富的功能引起人们的注意,并纷纷转向 Delphi 的学习和使用。不到一年的时间,Delphi2.0 已经上市,引起广大计算机用户的青睐。Delphi3.0 在 1997 年 5 月正式推出,其优良、稳定的性能,强大的数据库支持,更快的速度以及全面支持 Internet 等特征,更成为万众瞩目的焦点。

下面我们具体介绍 Delphi 各版本的主要特性。

1.1.1 Delphi1.0 特性

1. 安装要求

- 中/西文 Windows 3.1 及以上的操作系统
- 完全安装需要 78MB 的硬盘空间
- 使用 80386 以上微机
- 至少 4MB 以上的内存空间
- 建议配备 CD-ROM

2. 功能

- 提供 75 种以上的界面组件来构造窗体
- 制作单窗体制作(SDI)
- 提供制作多窗体(MDI)功能
- 支持下拉式菜单和弹出式菜单的制作
- 支持鼠标及键盘的动作
- 多页的笔记本、页标组、多页文件界面

3. 数据库方面

- 内建的数据库发动机(BDE)让程序直接存取 Paradox 和 dBase 的数据
- 提供一系列数据库存取组件,可完全地控制数据库,并具有 Query 及 Stored Procedure 控制对象
- 提供一系列数据库控制组件,可将取得的数据以窗体、文本框、图片、图像等方式显示,并可存取控制
- 提供 Report Smith3.0 报表制作工具

- 提供 Database Desktop 建立数据表

4. 效能方面

- 16 位原始码编译器
- 执行代码小,速度快
- 调试功能强大

1.1.2 Delphi2.0 特性

1. 安装要求

- 须 Windows 95 或 Windows NT3.05 版本或 100% 兼容的操作系统
- 完全安装需要 112MB 的硬盘空间
- 使用 80486 以上微机
- 至少 8MB 以上的内存空间

2. 功能

除 Delphi1.0 的功能外,还增加如下功能:

- 提供 90 种以上的界面组件来构造窗体
- 支持 OLE 自动化控制器、OLE 服务器、OLE 控制组件
- 提供 Screen 对象,可直接存取屏幕信息
- 提供 Printer 对象,可控制打印质量
- 提供 Application 对象,存取应用程序本身的信息
- 提供 Clipboard 对象,制作剪切板

3. 数据库方面

除 Delphi2.0 的功能外,还增加如下功能:

- 完全支持 ODBC,可与所有开放式数据库结构链接
- 提供 Report Smith7.0 报表制作工具
- Quick Report1.0 报表制作

4. 效能方面

- 最佳化 32 位原始码编译器
- 完全编译出执行代码,速度大大提高
- 建立可直接发行的完成执行文件

1.1.3 Delphi3.0 特性

1. 安装要求

- 须 Windows 95 或 Windows NT4.0 版本或 100% 兼容的操作系统
- 完全安装需要 170MB 的硬盘空间
- 使用 80586 以上微机
- 至少 16MB 以上的内存空间

2. 功能

除 Delphi2.0 的功能外,还增加如下功能:

- 提供 120 种以上的界面组件来构造窗体
- 全面支持 ActiveX,可以创建 ActiveX 构件和 ActiveX 文档

- 引入程序包概念
- 全面支持 Internet, 提供 15 个 Internet 相关组件, 同时支持 Internet Explorer 的 ISAPI 和 NatesapeD 的 NSAPI 动态链接扩充协议, 支持 FTP、HTTP、NNTP、POP、SMTP、TCP 和 UDP 等协议
- 提供自动支持 COM(Component Object Model)
- 支持 JPEG 和 PIF 图像格式文件

3. 数据库方面

除 Delphi2.0 的功能外, 还增加如下功能:

- 新增 Quick Report2.0 报表编写组件
- Decision cube 决策图组件
- TeeChart 图形显示组件
- Delphi3.0 的 Client/Server 版提供强大的服务器端纠错程序

4. 效能方面

- 32 位原始码编译器
- 更小的执行代码, 更快的速度(据评测, Delphi3.0 所生成的执行代码比 Visual Basic 快 3 ~ 6 倍)

1.1.4 Delphi 的特点

综合起来讲, Delphi 具有以下的特点:

1. 可视化集成的环境

Delphi 是一个运行在 Windows 环境下和基于组件(Component - Based)的可视化应用程序开发工具软件, 它可以使你用最少的程序量来得到最高的开发效率, 从而快速地开发一个 Windows 的应用程序。Delphi 提供了易于使用的图形化工具供软件开发人员创建程序中的可视化部份。也就是说, 开发者只需单击一下鼠标就可以把选择的按钮和组件粘贴到一个窗口之中, 而且开发者可以按照自己的喜好及用户要求, 安排窗口的外观, 使之恰到好处地调整。如果没有提供诸如此类的工具, 程序的可视化部份只能全部以代码的形式由程序人员编写出来。如果你曾经编写过 Windows 应用程序, 就会知道其可视化部份的设计, 到其最后成功是多么的不容易!

Delphi 提供了一套可视化工具, 通过它们, 软件开发人员可以很容易、及时地观察窗口设计中的动态变化。较之传统的编程方式, 这个过程类似于桌面设计或叫动态设计。它可容易地反复调整窗口的外观, 即先设计出窗体表单和布局, 然后运行一下, 观察其外观随后返回, 并对其不满意的地方作适当的修改。Delphi 的可视化集成开发环境和提供的许多设计工具, 如应用程序和窗体样板等, 将大大地节约了程序开发人员的时间和精力, 从而使开发人员集中精力于窗口动态调用技巧和程序算法方面, 加速了软件开发进程。

2. 强大而严格的底层编程语言

Delphi 所采用的底层编程语言是 Object Pascal。Object Pascal 语言是一种相当简洁的编程语言。其语法严格、结构清晰、阅读性强, 并且它还能在软件设计上使人们形成一种良好编程习惯。Object Pascal 的语言风格可能更适合于可视化的程序设计语言, 比如在 Delphi 中, 当双击窗口中的某个组件时, 程序就会马上跳到单元(代码存储的地方)中的代码编写处, 设计人员就可以在此写入该组件要完成的动作或状态等, 而无需在程序中来回移动和查找定位。

Object Pascal 语言的严格形式对于 Delphi 的编程实现是非常适合和方便的,几乎不受任何的限制,而且编译后的程序代码精练,运行速度非常地快。

3.能够创建真正独立的可执行文件及动态链接库(DLLs)

Dynamic Link Libraries(动态链接库——DLLs)使设计人员有可能创建经过编译的程序代码库,这种程序代码库可以为许多不同的应用程序所使用。使用动态链接库可以使程序的维护工作变得更加轻松自如,使各种相互联系和相互调用程序的实现细节有可能都被放入一个包含许多过程的库中,而这些过程则可以通过非常一般化的表达方法来加以调用。

通常设计人员希望能够从外部的动态链接库中调用函数、过程及利用现有的程序代码库。出于模块化、可重用性及维护性的目的,会使设计人员将自己的应用程序中,经常要被应用的部份编写成 DLL 文件,而不是可执行文件。此外,由于多个应用程序可以访问共同的目录特点,因而,将代码编译成为 DLLs,还有节约大量存储空间的优点。

Delphi 提供了两种方式访问外部的例程:

(1)在程序中使用一个外部声明。

(2)将 Windows 的 GetProcAddress 和 LoadLibrary API 过程结合起来使用,以便装载指向 DLLs 过程的指针。

上述两种外部调用方式,既大大地方便了设计人员的操作,又降低了程序设计的难度。Delphi 的这种动态调用功能可节省内存空间,而且程序可以在不知道任何硬件特性的情况下,使用其设备驱动程序的 DLLs。即使程序调用中未找到一个 DLL,它也可以继续运行。

4.极为方便的汉字加载能力

Delphi 虽然是西文环境下的工具软件,但使用 Delphi 编写中文应用程序界面也是非常方便的。设计人员只需简单地借助于 Windows 的中文字库和输入法,就可以直接地向窗体或窗体的组件以及程序代码中加入所需的任意汉字。这种加注汉字的方法既可以在程序设计时,通过窗体或组件的属性加入,也可在运行时的程序代码中加入,且不影响程序运行的速度。

5.目前世界上最快的代码编辑器

Delphi 的代码编辑器是非常友好和清晰的。在代码编辑器中,使用者可以设置关键字的颜色,使得程序代码清晰明快且修改方便。

Delphi 的代码编辑器始终在监视当前窗体的变化,并能及时地响应当前的修改。这种动态跟踪手段减少了设计人员因窗体的变化,而修改程序代码所花费的大量时间。设计人员只对窗体进行操作,无需对代码进行修改,因为代码编辑器已对修改做过更新了。

Delphi 的代码编辑器允许使用者打开多个代码文件,并放在不同的页上,这样使用者可很方便地在各单元之间来回切换,并完成编辑、剪切、复制、修改等工作。

另外,Delphi 的项目管理和资源管理也允许用户在窗体和单元之间切换。

6.具有创建新的组件和模板来扩展 Delphi 的能力

Delphi 提供了 Visual Component Library(可视化组件库,简称 VCL),定制并为 Delphi 的 Component Palette(组件模板)增加页面功能。VCL 是指构成 Delphi 组件结构基础的对象类层次体系。它包括了可视化部件、非可视化部件、不能作为组件的类及不能视为对象的通用函数。

由于 Delphi 是设计成模块化、可重用的开发工具,利用这些特点可使设计人员受益匪浅。把自己认为可重用的程序,或者一些应用程序以模板的形式保存成窗体表单,或以动态链接库(DLL 文件)的形式编译设计人员的应用程序,这将是非常有用的。

7.响应几乎所有 Windows 消息的能力