可视化编程丛书

Delphi 4.0

快速入门与提高

谢志诚 王亚平 王峥 杜微 编著

河南科学技术出版社

可视化编程丛书

Delphi 4.0 快速入门与提高

谢志诚　王亚平　王峥　杜微　编著

河南科学技术出版社

内 容 提 要

全书分 16 章。详尽地阐述了以下主要内容：Delphi 可视化程序设计方法，Delphi 可视部件类库中各种部件的使用方法及应用，文件管理和编程图形图像，动态链接库编程，异常处理与程序调试，自定义部件开发。通过本书的学习，读者将掌握 Delphi 的编程方法、Delphi 各种常规部件的综合应用方法和 Windows 95/98 高级特性开发方法等技巧。

本书侧重于介绍如何使用 Delphi 4.0 开发 Windows 95 应用程序，可作为广大计算机编程人员的参考用书。

图书在版编目（CIP）数据

Delphi 4.0 快速入门与提高/谢志诚等编著.—郑州：河南科学技术出版社，1999.6
（可视化编程丛书）
ISBN 7-5349-2392-1

Ⅰ.D… Ⅱ.谢… Ⅲ.Delphi 语言-程序设计 Ⅳ.TP312

中国版本图书馆 CIP 数据核字（1999）第 25881 号

责任编辑 王广照 责任校对 王艳红

河南科学技术出版社出版
郑州市农业路 73 号
邮政编码：450002 电话：(0371)5721450
河南省教委印刷厂印刷
全国新华书店发行
开本：787×1092 1/16 印张：20.5 字数：460 千字
1999 年 11 月第 1 版 1999 年 11 月第 1 次印刷
印数：1—4 000
ISBN 7-5349-2392-1/T·490 定价：27.00 元

前　言

Delphi 4.0 是 Borland(现为 Inprise)公司最新推出的基于 Object Pascal 语言的可视化开发工具。对于开发遵循 Windows 标准的应用程序，Delphi 有许多优越特性。它提供了新颖而直接的可视化设计工具，采用面向对象的方法将 Windows 编程的复杂性封装起来，实现了将可视化设计与 Object Pascal 语言的有机集成，配有 Database Engine，可通过 SQL Links、ODBC 访问多种数据库，并且提供了强大的开发基于客户/服务器模式的数据库应用功能。

今天，Delphi 已经发展到 4.0 版，自 Delphi 2.0 开始，几乎重新改写了 Delphi 的核心，使 Delphi 本身成为符合 Microsoft Windows 95 Logo 的应用程序。它全面支持 Windows 95&NT 的 OLE Automation、OCXs、多线程、Unicode 和 MAPI 等功能。在可视化开发环境方面，Delphi 4.0 换成了 Windows 95 控制元件，增强了 IDE 对象的操作方式（如 Drag-Drop），它还提供了新的储存各种资源的工具——对象仓储（Object Repository）。在数据库前端应用开发方面，Delphi 4.0 使用了 Multi Session 和 Thread Safe 的数据库引擎、数据库过滤器（Filter）、Visual Query Build、查询引擎和新的数据更新模式等。

与以前的版本相比，Delphi 4.0 还大大加强了对 Web 数据库的支持，增加了不少 Internet 应用的基本构件。Delphi 4.0 是更符合 Microsoft Windows 95 Logo 的应用程序。更重要的是使用 Delphi 4.0 可以很容易地编写符合 Windows 95 Logo 的应用程序，这使得 Delphi 4.0 成为目前在 32 位 Windows 环境下最具有吸引力的开发工具之一。

本书共 16 章。主要讲述使用 Delphi 所必须了解的概念，可视化程序设计方法，Delphi 可视部件类库（VCL）中各种部件的使用方法和它们在各类应用程序中的应用，以及关于 OLE、DDE、MDI、图形图像、文件管理、异常与调试和 DLL 等 Windows 高级特性开发方法。读者将掌握 Delphi 的编程方法、Delphi 各种常规部件的综合应用方法和 Delphi 高级应用程序开发技巧。其中第 12～15 章是关于 Delphi 开发数据库应用程序所必须掌握的内容，这几章讲述了 Delphi 数据库开发原理和方法以及数据库访问和控制部件的编程。通过学习，读者将掌握 Delphi 数据库开发原理、各种数据库部件的综合应用方法和高级数据库应用的开发技术。第 16 章讲解用户自定义部件原理和开发方法，通过本章的学习，读者将大大提高自己的 Delphi 高级应用开发能力。

本书中列举的开发实例、开发经验、开发技巧和 Windows 高级特性开发，适合于各个层次的 Delphi 用户。

对初学者来说，可通过对本书的学习迅速加入 Delphi 高级用户的行列；对有一定使用经验的读者，也可通过本书掌握一些 Delphi 深层次的开发方法，学会用更巧妙的办法开发出高水平的 Delphi 应用程序。

参加本书编写的人员还有刘俊霞、郭玉梁、方盛明、刘志、段永红、刘文超、王苏贵等。

作　者

1999 年 3 月

目　录

1 Delphi 快速入门

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先进的数据库技术。与 Delphi 3.0 相比，Delphi 4.0 的集成开发环境界面仍是 Windows 95 控制元件，同时提供对 Windows98 所新提出的用户界面元素的支持，还对菜单作了部分调整。Delphi 4.0 将更有利于程序员查询所使用的函数等，可以建立的对象分类更为详细，并新增了一些部件，尤其加强了对 Internet 和对 Web 数据库的支持。

在本章，考虑到初学者的需要，我们将从 Delphi 的基本概念开始向读者介绍快速入门的基本知识，直至读者能够开发、调试应用程序。

1.1 基本概念

1.1.1 Delphi 的基本形式

Delphi 实际上是 Pascal 语言的一种版本，但它与传统的 Pascal 语言有较大的区别。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。您的工作只是在“骨架”中加入您的程序。缺省的应用程序是一个空白的窗体(Form)，您可以运行它，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，但您却没有编写一行程序。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 已经为您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。您所需要做的，只是在程序中加入完成您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。由于您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，这样您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 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。您不必自己建立对象，只要在提供的程序框架中加入完成功能的代码，其余的都可交给 Delphi 去做。欲生成漂亮的界面和结构良好的程序不必绞尽脑汁，Delphi 将帮助您轻松地完成。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 Object Pascal 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 4.0 的新特性

新推出的 Delphi 4.0 版本对 Object Pascal 语言进行了新的扩展，新增了动态数组、方法重载、缺省参数及其他内容，也增加了很多功能，如新的工程管理器、代码探测器、增强的数据库、客户数据集、新的 RTL(Run Time Library 运行时间库，可很好地解决 2000 年问题)、增强的 ActiveX 和 VCL，还改进和完善了调试功能，使 Delphi 4.0 成为更加强大的可视化编程工具。

1.2.1 对 Object Pascal 语言的新扩展

1.2.1.1 动态数组

以前版本的 Delphi 中，可以将静态数组申明为

A：array[1..100] of string;

现在，可以申明一个动态数组，动态数组仅指定数组的维数和元素的类型等信息，而元素的数目可以是不定的，如：

A：array of integer;

B：array of array of string;

以上申明了两个动态数组，其中 A 是一个一维的整型数组，B 是一个二维的字符串数组。

1.2.1.2 方法重载

方法重载是指对象可以具有多个名字相同的方法。这些具有相同名字的方法通过参数的识别标志来进行区分，而被重载的方法则通过关键字 overload 作出标记，例如，一个对象可以具有以下两个构造函数：

```
constructor Create(Aowner: TComponent);overload;override;
```

constructor Create(Aowner: Tcomponent;Text:string);overload;

但是，在申明第三个构造函数时，如果其参数类型和上述两个参数之一相匹配的话，就不合法了。如下面的申明就是非法的，因为其识别标志不是唯一的：

constructor Create（Aowner：Tcomponent；Name：string）；overload；

另外，全局函数和过程也不能被重载。

1.2.1.3 缺省参数

Delphi 4.0 和 C++语言一样，提供一个缺省的参数，该参数只能出现在参数表的结尾处，格式如下：

参数明：类型=值

在调用一个带参数的函数或过程时，可以将缺省参数的值省略。例如：有一个申明为

procedure FileArray(A:array of Integer;Value:Integer=0);

那么下面的两个语句所完成的工作是相同的：

FileArray(MyArray,0);

FileArray(MyArray);

注意：如果同时使用方法重载和缺省参数，那么在申明中不能包括两个模棱两可的识别标志。例如，您的应用程序中不能包含与下面类似的两个申明：

procedure P(I:Integer; J:Integer = 0);overload;

procedure P(Size:Integer);overload;

应用程序中存在这样的申明是非法的，因为它在运行下面的语句：

P(1);

时就无法知道要调用的究竟是哪个申明的过程了。

1.2.1.4 32 位无符号整数类型

Delphi 4.0 引入了一个真 32 位无符号整数类型——Longword，它的值域是：0～4 294 967 295,以前的 Cardinal 类型等价于 Longword 类型。

1.2.1.5 64 位整数类型

为了使您的管理超出普通的整数类型（32 位 Longint）的整数值，Delphi 4.0 为用户提供了 Int64。Int64 是一个 64 位的整数类型，它的值域是 -2^{63}～2^{63}。

大多数标准的例程都将 64 位的整型参数截断为 32 位，但是像 High、Low、Succ、Pred、Inc、Dec、IntToStr 和 IntToHex 这些例程完全支持 Int64 类型的参数，并且函数 Round 和 Trunc 还可以返回 Int64 类型的值。另外，Delphi 4.0 还增加了支持 Int64 的两个新函数 StrToInt64 和 StrToInt64Def。当然，还有相当一部分例程（包括 Ord）都不支持 Int64。

整数类型的常数只要不超出 Longint 的值域，都可以将类型申明省略，否则它的类型就是 Int64，如下列申明中只有 Number1 是 Longint 类型，其他三个都是 Int64 类型：

```
const Number1 = 10;
const Number2 = Int64(10);
const Number3 : Int64 = 10;
const Number4 = 2 000 000 000;
```

1.2.1.6 对实数(Real)类型的修改

Real 类型以前表示的是一个 48 位的浮点数，现在被修改成与 Double 类型一致的 64 位

浮点数。这个改动提高了应用程序的性能，因为 Double 类型和 Inter CPU 家族是兼容的，而且现在 Real 类型的属性可以是公用的。

为了保持向后兼容，在 Delphi 4.0 中为老的 48 位实数类型增加了一个编译开关，如下列代码所示：

```
{$REALCOMPATIBILITY ON}
```

如果有必要用新的代码表示 48 位的实数类型,用户可以使用 Real48。

1.2.1.7 实现接口的代表形式

Delphi 4.0 增加了一个新的指令 implements，它可以在一个属性中代表一个接口的实现，如：

```
Property
MyInterface:ImyInterface read FmyInterface implements ImyInterface;
```

代码申明了一个 MyInterface 的属性,它实现了接口 ImyInterface 。

1.2.2 新的工程管理器

与 C++ Builder 3.0 一样，Delphi 4.0 引入了工程组的概念，它可以将多个一起工作的工程结合到一个工程组中，这样就可以使用户将一些相互关联的工程组织起来并保持在一起工作，像多级应用程序中的独立级，或者在一起工作的 DLL 以及实现模块等。

1.2.3 新的代码探测器

在新面孔的 Code Explorer 中，包含有类完成器、模块导航以及代码浏览器等。

Code Explorer 使类的创建变得更加容易，用户只要在 interface 部分写入一个方法属性，类完成器就会在 implementation 部分自动为用户生成类的框架。模块导航可以使用户在单元文件以及 interface 和 implememntation 几个部分之间快速切换。符号预测技术(TooLtip Symbol Insight) 则可以浏览任何标识符的申明信息，并使用代码浏览器跳转到该申明中。

1.2.4 新的工具窗口特性——Dockable

Delphi 4.0 的集成环境变得更加容易配置，而且增加了一个非常有吸引力的特性——Dockable。Dockable 特性就是使两个工具窗口合为一个窗口，而且只需将其中一个窗口拖动到另一个窗口中的适当位置释放就行了。如图 1.1 所示即将 Call Stack 窗口与 Watch List 窗口合成为一个成窗口。

如果要重新分解这个合成窗口，用户只要将其中的一个窗口拖出来即可。代码浏览器（Code Explorer）、对象观察器（Object Inspector）和工程管理器（Project Manager）等都可以利用这种方法进行窗口的合成与分解。

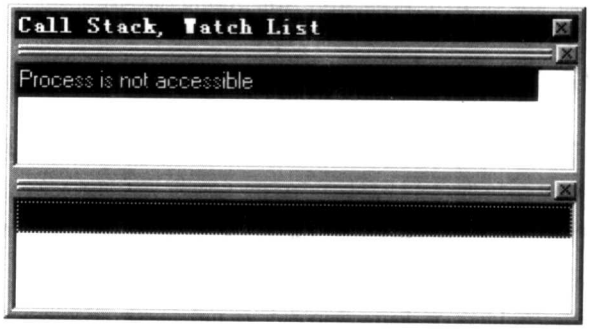

图 1.1 Call Stack 窗口与 Watch List 窗口合成为一个成窗口

1.2.5 改进的调试功能

Delphi 4.0 的合成调试器也增加了许多新的功能，如远程和多进程调试、CPU 窗口、监视器、增强的断点功能、调试器专用的子菜单以及上述窗口的 Dockable 特性等等。其中远程调试支持带调试符号建立的 EXE、DLL 和包。多进程调试支持本地和远程进程，模块浏览窗口和线进程窗口也是多进程敏感性的。它们显示所有与线程和模块一同装入的进程，这些线程和模块属于各自的进程。

CPU 窗口专门用于显示程序的底层信息，如堆栈、寄存器或 CPU 标志、内存转储以及应用程序机器码的反汇编。

Modules 增加了两个新的面板，如图 1.2 所示：一个用于源文件，另一个用于登录点。Module List Windows 还为这些面板增加了新的弹出式菜单。

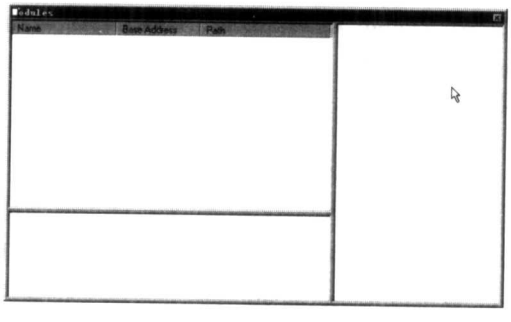

图 1.2 Modules 窗口

如果用户选择了 View|Debug Windows|Event Log 命令，那么会出现一个事件注册(Event Log)窗口，该窗口可以用于显示进程控制信息、断点信息、来自 OutputDebugString 的输出信息以及窗口信息等，用户可以再选择 Tools|Debugger Options 命令，在 Event Log 页面上设定事件注册的选项。

选择 View|Debugger Windows|Local Variables 命令，将会得到一个局部变量（Local Variables）窗口，它们用于显示调试模式下的当前函数的局部变量。

1.2.6 Run 菜单中新增的命令

在 Run|Parameters 命令打开的对话框中增加了一个新的 Remote 页面，用于远程调试。Run|Inspect 命令可以输入一个用户所希望监视的表达式。在 Run|AddBreakpoint 命令中，使用了全新的地址和数据断点对话框，在 Breakpoint List 窗口、CPU 窗口、代码编辑器的弹出

式菜单中的 Properties 命令都使用了新的 Add Breakpoint 对话框。

1.2.7 增强的数据库

Delphi 4.0 在数据库的功能上做了很多的工作。数据库访问控件可以在用户建立窗体或数据模块时调整数据模型。数据访问控件和 BDE（数据库引擎）修改后，可以使用数据库服务器的新类型访问数据，包括 Access 7 和 Oracle 8 对 SQL 的扩展、ADT（抽象数据类型）、数组、引用和嵌套表，支持这些扩展类型的相应对象为 Tfield 类型、TADTField、TreferenceField、TdataSetField 以及 TarrayField。Visual Query Builder 已经被 SQL Builder 取代，它是一个全新的智能化查询创建器。

Table 控件也进行了修改，它允许从窗体编辑中创建、重命名或删除表。

Grid 控件可以支持对 ADT 和嵌套表的浏览。

1.2.8 客户数据集

客户数据集在 Delphi 4.0 中支持更广泛的过滤表达式、合计功能，并可用关联对象的字段类型。在 Delphi 4.0 中，创建 flat 型文件数据库应用程序的数据集变得更加容易。

1.2.9 运行时间库——RTL

在 Delphi 4.0 中增加了两个函数 StrToData 和 StrToDataTime，它们可以调用全局变量 TwoDigitYearCenturyWindows 来控制两位数年份的解释。不过，用户在新开发的应用程序中，年份的输入最好还是使用四位数。

1.2.10 增强的 ActiveX 和 VCL

在 Delphi 4.0 中，增强后的 ActiveX 表现出以下特征：

●可以用 Automation 向导生成支持任何自动服务器对象事件的代码。

●实施了数据帮定的 ActiveX 控制可以和 VCL 数据集进行通讯。

●Delphi 4.0 中新的 COM 对象向导可以引导用户创建简单的 COM 对象。

●整个 COM 构架都具有线程安全性。

●类型库编辑器支持 IDL 和 Object Pascal。

Delphi 4.0 对 VCL 增强了以下几项：

●增加了一个 ActionList 控件,它可以集中响应用户命令,并可以对所有响应用户命令的控制提供集中控制。

●开发 Windows NT 下的应用程序时，程序员可以用 TServiceApplication 和 TService 创建 Service 应用程序，在 New Items 对话框中提供了 Wizards Service 和 Service Application，它们可以管理创建上述应用程序的细节，可省程序员不少时间。

●TContorl 和 TWinContorl 对象中增加了对应用程序窗口“Dockable”功能的支持，使程序员的应用程序窗口可通过拖放操作任意地结合在一起。

●TControl 对象还增加了限制窗口改变大小的功能，可以使程序员的应用程序始终保持理想的外观。

●增加了一个新的 TRegistryIniFile 类，使用它可以用代码来管理注册标以及 INI 文件；原来的一个公共的基类 TCustomIniFile 也派生出一个 TMenIniFile 类，它可以使程序员确保对 INI 文件的修改被缓存到内存中，即使应用程序在 NT 下运行也无妨。

●为了适应 TClientDataSet 中增强的参数支持，Tparams 的单元名从 dbtables 改为 db；增加了新的支持中东地区从右到左的阅读习惯的属性，并且可以使垂直滚动条出现在左边。

●TObject 对象增加了两个 protected 方法：BeforeDestruction 和 AfterConstruction。BeforeDestruction 在一个对象的析构函数之前会直接被自动调用；AfterConstruction 在一个对象的构造函数之后会直接被自动调用。派生的类可以重载这两个方法来实现那些不应该在析构函数和构造函数本身中发生的动作。

1.2.11 支持 CORBA

CORBA(Common Object Request Broker Architecture——通用对象请求中间结构)是一个在应用程序中使用分布式对象的方法，它可以用于多种平台上，所以用该标准开发的应用程序可以在其他非 Windows 系统的机器上运行。

和 COM 对象一样，CORBA 是一个分布式对象结构，这就意味着客户的应用程序可以使用在远程服务器上实现的对象。

Delphi 4.0 的客户/服务器版和企业版都支持开发 CORBA 客户和服务器应用程序，而且向导功能可以使用户很轻松地创建一个 CORBA 服务器，Dynamic Invocation Interface（DII）则可以帮助用户写出已有 CORBA 服务器的客户程序。另外，在多级数据库支持中也内建了 CORBA 兼容性，用户甚至还可以建立一个同时管理 COM 客户和 CORBA 客户的服务器应用程序。

1.2.12 支持多级开发

在开发多级应用程序中，Delphi 4.0 提供了更多的控制，它们包括：

●更多的对数据包中所包含内容和所应用的升级方式的控制。

●明/细方式支持使用嵌套表。

●中断连接的钩子函数。

●增强了客户向一个应用程序服务器传递参数或向数据包中保存定制信息的能力。

●调用服务器接口更加容易。

●增加了一个新类 TdataSetProvider，它可以从一个数据集中提供或者分解数据。

●多种连接控件可以使客户应用程序和应用程序服务器相连接。

●Socket 服务器支持回调，它的新版本 SocktSrvr.exe 可以作为一个 NT 服务器运行,并且支持两个命令选项：install(安装服务器)，uninstall（卸载服务器）。用户还可以在它的图标上按下鼠标右键来配置 SocktSrvr.exe。

1.3　Delphi 4.0 可视化编程环境

在这一节中，我们用一个小程序逐步介绍 Delphi 的主要部件及其操作方法。建议读者按照本书介绍的过程，在您的电脑上直接操作。您将对 Delphi 的可视化编程有一个直观、快捷的了解，必将收到事半功倍的效果。

1.3.1　进入 Delphi 的可视化编程环境

单击“开始 | 程序 | Borland Delphi 4.0 | Delphi 4.0”，可以启动 Delphi 4.0 应用程序（如图 1.3 所示），加载后会出现如图 1.4 所示的窗口。

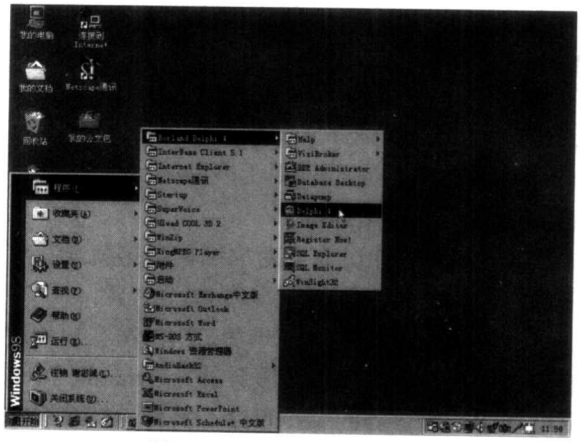

图 1.3　Delphi 4.0 的程序组

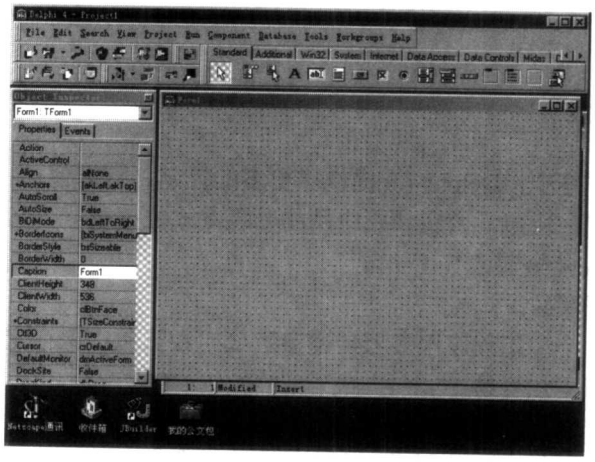

图 1.4　Delphi 4.0 的运行界面

为避免隐藏在 Delphi 后的 Program Manager 和已经运行的其他程序扰乱画面，分散您的注意力，不妨在启动 Delphi 4.0 前关掉或最小化其他应用程序，保持屏幕上只有 Delphi 4.0 应用程序窗口可见。

首次加载 Delphi，屏幕上会出现四个窗口：