

万水计算机编程技术与应用系列

Visual 程序设计 Basic.NET

——ADO.NET详解

何明国 唐耀 等编著



中国水利水电出版社

www.waterpub.com.cn

内 容 提 要

本书按照计算机技术教育的基本规律和要求,在内容编排上由浅入深,分为:基础篇、提高篇和应用篇,系统地介绍了 ADO.NET 数据库应用程序开发的方方面面。首先简要介绍了 Visual Basic.NET 环境与编程基本知识、程序语言基础和面向对象技术;然后,详细剖析了 ADO.NET 数据访问模型的概念和方法,包括数据连接、数据命令、适配器、数据集、数据视图、数据表和数据绑定等,充分突出了面向对象的思想和 ADO.NET 的特点;最后,运用丰富的实例揭示了一些数据应用开发的技巧。

本书内容翔实、表述准确、结构清晰、可操作性强,适合具有一定的 Visual Basic.NET 使用经验的读者学习使用,也可供从事计算机应用和开发的各类人员参考。

图书在版编目(CIP)数据

Visual Basic.NET 程序设计——ADO.NET 详解 / 何明国等编著. —北京:中国水利水电出版社, 2004.6

(万水计算机编程技术与应用系列)

ISBN 7-5084-2174-4

I. V… II. 何… III. ①BASIC 语言—程序设计②数据库—接口—程序设计 IV. ①TP312②TP311.11

中国版本图书馆 CIP 数据核字(2004)第 054902 号

书 名	Visual Basic.NET 程序设计——ADO.NET 详解
作 者	何明国 唐耀 等编著
出版 发行	中国水利水电出版社(北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 63202266 (总机) 68331835 (营销中心) 82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京北医印刷厂
规 格	787mm×1092mm 16 开本 19 印张 431 千字
版 次	2004 年 6 月第 1 版 2004 年 6 月第 1 次印刷
印 数	0001—5000 册
定 价	30.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

前 言

广大的数据开发人员对于微软的 ADO2.X 数据访问技术一定都很熟悉, ADO2.X 模型带给数据开发人员巨大的便利和乐趣。然而, 随着微软 .NET 战略的推出, 作为 .NET 框架重要组成部分的 ADO.NET 也逐渐成为数据访问的技术现实。

微软在 ADO 如日中天的时候推出 ADO.NET 是一件颇受争议的事, 但这种争议现在看来是没有多少意义的。 .NET 平台是一个以整个 Internet 为背景的全新的开发平台, 同样 ADO.NET 也是一个崭新的数据模型, 它与 ADO 之间并不存在继承关系。但是, 熟悉 ADO 编程模型对于掌握新的 ADO.NET 技术无疑很有帮助。 ADO.NET 秉承了 ADO 模型简单易用的特点, 而结构的复杂和功能的强大却是 ADO 无法比拟的。

尽管颇受争议, 但 ADO.NET 还是正在取代 ADO。人们虽然可以在 Visual Basic.NET 中继续使用自己熟悉的 ADO, 但是 ADO 在开发基于 Internet 的三层甚至多层分布式结构程序方面的局限性将迫使你不得不转向 ADO.NET。

本书详细讲述了 ADO.NET 数据访问模型的结构和组件, 重点突出了各个组件之间的关联和使用, 对于需要深入研究 ADO.NET 的开发人员有很强的指导意义。需要指出的是, 尽管本书中所有的源代码和示例都以 Visual Basic.NET 作为基础语言, 但是 ADO.NET 并不是 Visual Basic.NET 语言的固有成分, 而是 .NET 框架的重要组成部分。如果你习惯于使用受 .NET 框架平台支持的其他语言 (例如 C#、托管 C++ 和 J#) 来编程, 则除了语言书写有差异外, 对 ADO.NET 组件的访问和使用没有区别。

全书内容分为三个部分: 基础篇、提高篇和应用篇。

基础篇, 包括第 1 章~第 3 章, 主要讲解了 Visual Basic.NET 环境与项目、程序语言基础、面向对象技术和数据访问模型, 描述了一般的数据库访问方法。

提高篇, 包括第 4 章~第 11 章, 详细地剖析了 ADO.NET 的对象模型、组件特性 (数据连接、数据命令、适配器、数据集、数据视图、数据表和数据绑定等) 以及相互作用和应用。

应用篇, 包括第 12 章和第 13 章, 每一章或小节讲述一个专题, 以实例的形式引导读者解决数据库开发中的一些问题, 介绍一些开发技巧以拓宽读者思路。

本书在内容的编排上, 考虑到学习的方便, 采取循序渐进的方式, 将程序设计语言、面向对象技术、数据模型对象和编程例程有机地结合为一体, 所有的编程例程都上机调试通过。

参加本书编写的作者都是从事计算机相关专业教学和研发的教师, 具有丰富的教学经验和教材编写经验。其中, 第 1 章~第 3 章由段宇编写, 第 4 章~第 6 章由唐耀编写, 第 7 章~第 9 章由王文田编写, 第 10 章和第 11 章由李岐旭编写, 第 12 章和第 13 章由何明国、孙珠峰编写, 全书由何明国、唐耀统稿。

由于时间仓促及作者水平有限, 书中错误和疏漏之处在所难免, 恳请广大读者批评指正。

编者

2004 年 3 月

目 录

前言

基础篇

第 1 章 接触 Visual Basic.NET	2
1.1 Visual Basic.NET 的新特点	2
1.2 Visual Basic.NET 集成开发环境	3
1.2.1 界面布局	3
1.2.2 自定义开发环境	7
1.3 Visual Basic.NET 项目组织	8
1.4 Visual Basic.NET 编程思想	10
1.5 获取帮助	11
1.6 典型的 Visual Basic.NET 程序结构	12
1.6.1 程序开发的一般步骤	12
1.6.2 程序语句的基本结构	14
第 2 章 Visual Basic.NET 语言基础	15
2.1 基本标识符	15
2.1.1 标识符定义规则	15
2.1.2 变量和常量	16
2.1.3 基本数据类型	18
2.1.4 字符型和字符串	19
2.1.5 用户定义数据类型	20
2.1.6 值类型与引用类型	23
2.1.7 运算符与表达式	23
2.2 基本语句	25
2.2.1 赋值语句	25
2.2.2 条件语句	25
2.2.3 循环语句	27
2.2.4 跳转语句	29
2.3 过程	31
2.3.1 Sub 过程	31
2.3.2 Function 过程	33
2.3.3 递归调用	34
2.4 类技术	35

2.4.1	类和对象	35
2.4.2	类成员	36
2.4.3	构造函数和析构函数	40
2.4.4	共享成员	41
2.4.5	类的事件	42
2.4.6	类继承	43
2.5	接口	45
2.6	特定对象的引用	47
第3章	Visual Basic.NET 数据库编程概论	49
3.1	数据库系统概述	49
3.1.1	数据库管理系统 (DBMS)	49
3.1.2	关系数据库	49
3.1.3	数据库应用程序	51
3.2	SQL 语言简介	52
3.2.1	SQL 的优点	52
3.2.2	SQL 常用语句	53
3.3	ADO.NET 数据访问特征	56
3.4	ADO.NET 模型简介	58
3.5	数据库程序初步体验	59

提高篇

第4章	数据连接	63
4.1	创建连接	63
4.2	Connection 属性	66
4.3	Connection 方法	67
4.4	Connection 事件	68
第5章	数据命令	70
5.1	关于数据命令	70
5.2	创建数据命令	71
5.3	命令属性	72
5.4	命令参数设置	75
5.5	Command 方法	80
5.6	数据阅读器 (DataReader)	82
第6章	数据适配器	88
6.1	创建数据适配器	88
6.2	数据适配器属性	93
6.2.1	xxxCommand 属性	93

6.2.2	适配器表映射	96
6.3	数据适配器方法	99
6.4	数据适配器事件	105
第 7 章	数据集	110
7.1	类型化数据集与非类型化数据集	110
7.2	数据集结构	111
7.3	创建数据集	112
7.4	数据集属性	115
7.5	数据集方法	119
7.6	数据表	123
7.6.1	创建数据表	124
7.6.2	DataTable 的属性	125
7.6.3	DataTable 中数据的访问	126
7.6.4	Constraints 集合	132
7.6.5	数据表方法	135
7.6.6	数据表的事件	138
第 8 章	数据视图	142
8.1	创建数据视图	142
8.2	数据视图属性	145
8.3	数据视图方法	149
第 9 章	数据操作	151
9.1	编辑和更新数据	151
9.2	数据行的状态和版本	151
9.3	操作数据集中的数据	155
9.3.1	读取单元格数据	156
9.3.2	查找记录	157
9.3.3	添加记录	159
9.3.4	更改记录	160
9.3.5	删除记录	161
9.3.6	挂起数据值修改	161
9.4	合并数据集	163
9.5	更新数据源	165
9.5.1	更新数据集	166
9.5.2	更新数据源	168
第 10 章	Windows 窗体中的数据绑定	172
10.1	了解 Windows 窗体中的数据绑定	172
10.2	实现控件绑定到数据源	174

10.2.1	简单数据绑定	174
10.2.2	复杂数据绑定	178
10.3	BindingContext 对象	181
10.4	CurrencyManager 对象	182
10.5	Binding 对象	188
10.6	数据同步	193
第 11 章	Web 窗体中的数据绑定	198
11.1	了解 Web 程序	198
11.2	创建一个 Web 应用	200
11.3	Web 数据绑定的特点	205
11.4	简单绑定	207
11.5	使用 DataBinder 类绑定	212
11.6	复杂绑定	214
11.7	维护 ADO.NET 对象状态	216
11.8	Web 窗体中更新数据	219

应用篇

第 12 章	Crystal Reports 报表制作	226
12.1	报表使用基本概念	226
12.2	Windows 程序中的报表制作	228
12.3	Web 程序中的报表制作	233
12.3.1	基本概念和实例	234
12.3.2	Pull 模型报表演示	238
12.3.3	Push 模型报表演示	241
第 13 章	ADO.NET 技巧实践	246
13.1	数据集中多表关联和浏览	246
13.1.1	实例创建	246
13.1.2	数据合计	255
13.2	面向对象特征在 MIS 中的应用	256
13.3	Visual Basic.NET 中使用 ADO 访问数据库	272
13.4	数据的图形表示	281
13.5	ADO.NET 创建三层结构应用程序	285
13.5.1	创建中间层业务对象	286
13.5.2	Windows 用户层	289
13.5.3	Web 用户层	292
13.5.4	小结	295

基础篇

本篇介绍了 Visual Basic.NET 的主要新特点、集成开发环境、项目的管理、程序一般结构和 Visual Basic.NET 语言基础等,为初学者提供一个快速进入 Visual Basic.NET 世界的入口。主要包括以下知识点:

- Visual Basic.NET 新特点
- 集成开发环境简介
- 项目组织与管理
- Visual Basic.NET 程序结构
- 基本数据类型
- 变量和流控语句
- 类与接口技术
- ADO.NET 数据访问模型简介

第 1 章 接触 Visual Basic.NET

本章主要向读者介绍 Visual Basic.NET（以下缩写为 Visual Basic.NET）的集成开发环境（缩写为 IDE）、编程思想、项目组织和典型的程序结构。通过本章的学习，可以帮助读者快速地熟悉 Visual Studio IDE（以下缩写为 VS.NET）的同时，领略 Visual Basic.NET 程序开发的独特魅力。

1.1 Visual Basic.NET 的新特点

从 Visual Basic 1.0 开始，Visual Basic 就备受广大程序员的喜爱，同时，作为 Windows 平台的快速应用开发（缩写为 RAD）工具 Visual Basic 已经风靡世界，出尽了风头。当发展到 Visual Basic 6.0 版本时，Visual Basic 似乎已经尽善尽美。然而微软公司为了实施其 .NET 战略，在 2002 年 2 月 13 日再出重拳，充满蕴义地推出了 VS.NET 开发套件，Visual Basic.NET 作为旗舰产品理所当然地又一次充当了开路先锋，成为了最新一代的 Visual Basic 语言，而不能再被看成一种玩具语言，因为它已经与 C#、C++ 并驾齐驱，成为创建各种 .NET 应用程序（包括 XML Web Services 和 ASP.NET Web 应用程序）的三架马车之一。

Visual Basic.NET 的重要革新主要体现在两个方面：Visual Basic 语言本身和 .NET 框架。

首先，微软公司对 Visual Basic 语言本身增加了许多新功能，特别是支持许多新的面向对象语言特征，如继承、重载、OverRides 关键字、接口、共享成员和构造函数等，这使 Visual Basic.NET 成为功能强大的完全面向对象的编程语言。其他的新语言功能还包括结构化异常处理、自定义属性和符合公共语言规范（Common Language Specification，缩写为 CLS）等。

CLS 是对诸如数据类型之类的东西以及对象的公开和交互操作方式进行标准化的一组规则。任何符合 CLS 的语言都可以用在 Visual Basic.NET 中创建的类、对象和组件中；同样，Visual Basic 用户也可以使用其他符合 CLS 的编程语言创建的类、组件和对象（例如，访问 C#.NET 和 C++.NET 中创建的对象），而不用担心诸如数据类型转换之类带来的差异。因此，CLS 是提供 VS.NET 套件中语言互操作性的基础。

Visual Basic.NET 相对 Visual Basic 旧版本最大的差别在于该软件引入了全新的 .NET 框架。

.NET 框架主要组件包括公共语言运行时（Common Language Runtime，CLR）和 .NET 框架类库。

CLR 的任务是管理和执行为 .NET 框架编写的代码。用 Visual Basic.NET 编译器编译的代码必须在 CLR 的帮助下才能运行，这样的代码称为托管代码。CLR 负责监视代码执行

的方方面面：管理内存、线程执行、代码执行、代码安全验证、编译以及其他系统服务。

.NET 框架类库是一个与 CLR 紧密集成的可重用的类型集合。该类库是面向对象的，全面封装了操作系统的 API，使之更易于使用。

.NET 框架是以支持多种语言编译器为目标的，这些语言用来创建小到手持设备组件，大到跨 Web Farms 的 ASP.NET 程序。.NET 框架并不是 Visual Basic 语言的一部分，而是负责提供一个可以为这些应用程序所共享的基础平台。记住：框架是独立于语言的，如果离开了类库的支持，Visual Basic.NET 语言将一事无成。

1.2 Visual Basic.NET 集成开发环境

熟练使用一个新的开发工具首先要熟悉其操作环境。Visual Basic.NET 完全集成到 Visual Studio.NET(简称 VS.NET)集成开发环境(IDE)中，这种集成开发环境与 Visual Basic 6.0 IDE 有很大的差异。

1.2.1 界面布局

VS.NET 集成开发环境提供了大量功能强大的工具，帮助程序员尽量地简化工作。第一次打开 VS.NET 时，界面如图 1-1 所示。

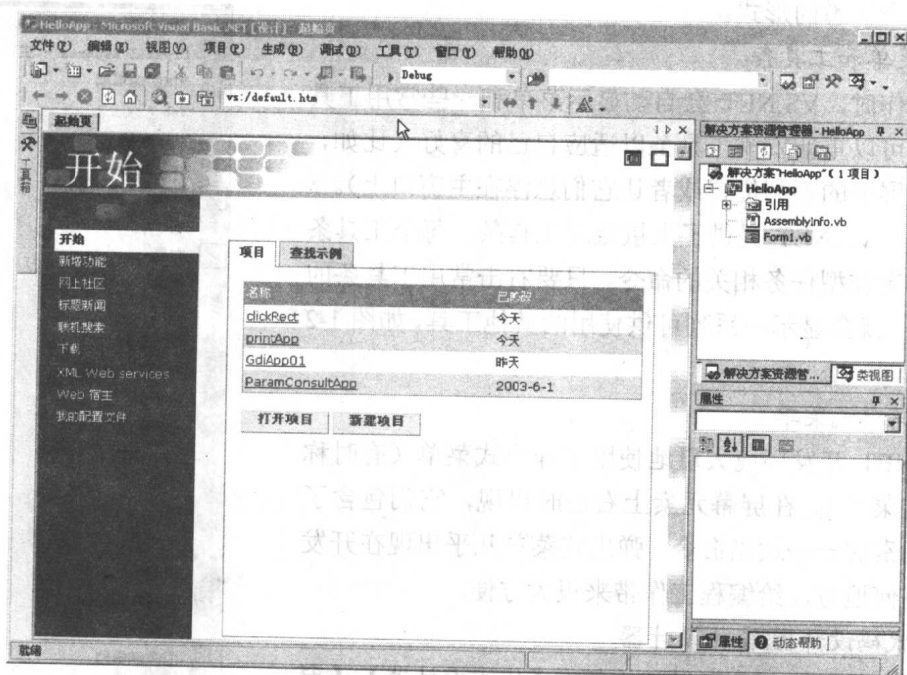


图 1-1 初始界面

下面分别介绍初始界面的各项内容。

1. 起始页

中间的【起始页】是获取外部信息和服务的地方。【起始页】上有许多选项，这些选

项提供了关于 VS.NET 的最新信息，有些选项功能需要 Internet 连接。

- 开始：启动之后显示的就是【开始】选项，它包含有一个近期处理过的解决方案的列表。第一次启动时，这个解决方案列表是空的。
- 新增功能：是一个 Web 页面，其中包含有最新的有关 VS.NET 的更新和信息。如果连接到 Internet，还可获得从 Microsoft 网站下载的最新内容。
- 网上社区：为 VS.NET 开发者提供了很多新闻组和站点以供选择。
- 标题新闻：包含有来自 MSDN 网站的最新特性、技术资料、新闻和到相关知识库的链接。使用该选项需要连接到 Internet。
- 联机搜索：提供了 MSDN 数据库的联机搜索。使用该选项需要 Internet 连接。
- 下载：提供了与使用 Visual Basic.NET 和 VS.NET 进行的开发相关的下载和例程代码。使用该选项需要连接 Internet。
- XML Web Services：查找当前项目需要的 XML Web 服务。使用该选项需要 Internet 连接。
- Web 宿主：提供了与 Web 宿主选项相关的信息，使用该选项需要 Internet 连接。
- 我的配置文件：用于配置 VS.NET 的用户界面。可以选择一种屏幕布局和键盘方案以及默认的帮助筛选器，把自己标识为熟悉 Visual C++、Visual Basic 6.0 或者其他工具的开发人员，VS.NET 将根据配置信息把工具栏和开发环境窗口调整为开发人员熟悉的形式。

2. 菜单和工具条

在工作时，VS.NET 会自动显示菜单和一些常用工具条。用户可以重新安排工具条以适应自己的喜好（比如，将其移到屏幕的不同位置或者让它们悬浮在主窗口上）。

另外，VS.NET 提供了大量专用工具条，每个工具条包含与特定类型任务相关的命令。只要右击常用工具条的空白区域，就会显示一系列可供使用的其他工具，如图 1-2 所示。

3. 弹出式菜单

VS.NET 开发环境大量地使用了弹出式菜单（有时称为上下文菜单）。在屏幕元素上右击时出现，它们包含了该屏幕元素的一些适当命令。弹出式菜单几乎出现在开发环境的任何地方，给编程工作带来很大方便。

4. 代码设计器/窗体设计器

开发人员的大部分开发工作都在【代码设计器】|【窗体设计器】窗口中进行。【窗体设计器】主要用来设计项目的可视化界面；【代码设计器】则用来编辑项目的源代码。另外，VS.NET 界面的这部分还可用来显示在线帮助信息，如图 1-3 所示。

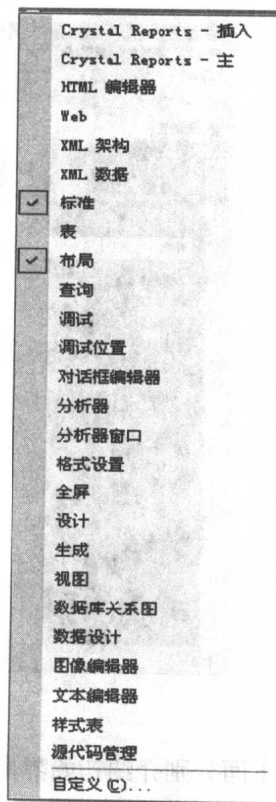


图 1-2 快捷菜单

5. 工具箱

【工具箱】窗口中放置了开发时可能会用到的各种常用控件，控件按功能分组在不同的选项卡中，如图 1-4 所示。根据当前的项目性质，某些选项卡会自动显示或隐藏。例如，开发一个 Windows 应用程序时，【Windows 窗体】选项卡会显示，而【Web 窗体】选项卡是隐藏的。如果要查看隐藏的选项卡，可以在选项卡上右击，选择快捷菜单中的【显示所有选项卡】选项。若要进行选项卡之间的切换，只需单击【工具箱】窗口中相应的选项卡标题即可。

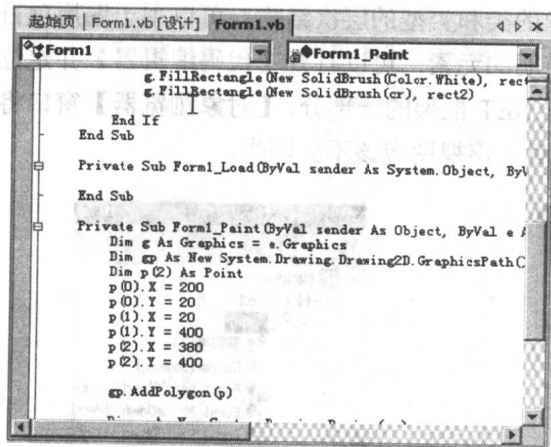


图 1-3 代码设计器/窗体设计器

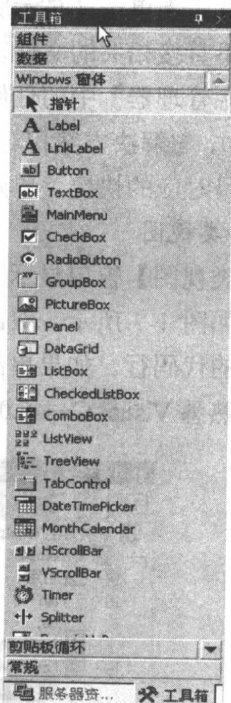


图 1-4 工具箱

6. 服务器资源管理器

通过【服务器资源管理器】窗口可以访问本地机上或网络上的可用系统服务。通常情况下，【服务器资源管理器】窗口停靠在 VS.NET 窗口边缘，当鼠标指针移到【服务器资源管理器】窗口的图标上，【服务器资源管理器】窗口就会展开，并显示一个当前网络的可用服务列表，如图 1-5 所示。

通过【服务器资源管理器】窗口可以方便地访问事件日志、数据库、性能计数器以及其他系统服务。同时，还可以直接从【服务器资源管理器】窗口中将对象拖到项目中去。Visual Basic.NET 会自动生成代码和对象。例如，要向一个 Windows 应用程序中添加数据库连接，可

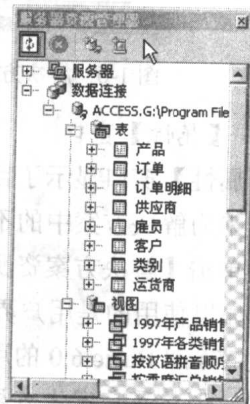


图 1-5 服务器资源管理器

以直接从【服务器资源管理器】窗口中把数据库的图标拖到一个在 VS.NET【Windows 窗体设计器】中打开的窗体中。

7. 解决方案资源管理器

【解决方案资源管理器】窗口以树状结构显示 VS.NET 中当前的解决方案，并以文件的形式组织项目，如图 1-6 所示。通过【解决方案资源管理器】窗口，用户可以浏览组成当前解决方案的全部项目以及分属于各个项目的文件。

双击项目文件即可打开源文件进行编辑。可以在【代码编辑器】或【窗体设计器】中浏览文件。当单击【显示所有文件】按钮时，它会更新显示所有的文件，包括在整个开发过程中不直接相关的文件以及 VS.NET 自动产生的项目文件和项目的输出文件。在【解决方案资源管理器】中的任何项上右击都会出现一个快捷菜单，列出了可以对该项进行的操作。例如，与解决方案相对应的快捷菜单允许进行诸如向解决方案中添加新项之类的操作，而与项目对应的快捷菜单则允许向项目中添加新项或进行其他一些与项目相关的操作。

8. 类视图

【类视图】窗口显示了当前项目中的类和类型的层次结构，可以用于遍历项目的类型层次，如图 1-7 所示。双击在项目中的元素，可以打开【代码编辑器】并定位到声明该元素的代码行。如果单击的元素属于 .NET 框架的一部分，【对象浏览器】窗口将为之而打开。熟悉 Visual C++ 6.0 的开发人员对于该视图应该不会陌生。

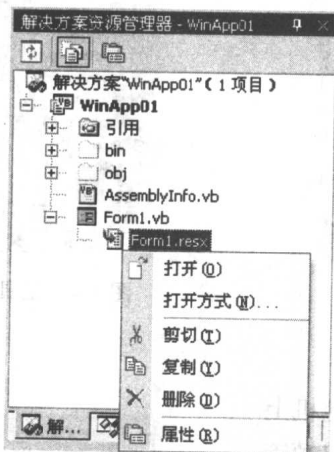


图 1-6 解决方案资源管理器



图 1-7 【类视图】窗口

9. 【属性】窗口

【属性】窗口显示了目前在 VS.NET 中所选择对象的属性，如图 1-8 所示，在程序设计时用来为解决方案中的不同元素设置属性。该窗口的内容随当前使用的项类型的不同而变化。单击【解决方案资源管理器】窗口中的“项目”，将显示对应于所选项目的【属性】窗口。如果使用的是用户界面控件和窗体，那么控件和窗体的属性就显示在【属性】窗口。使用过 Visual Basic 6.0 的用户是很熟悉的。

10. 调整界面窗体布局

如果对于默认的窗体布局不满意，可以在【初始页】显示时选择【我的配置文件】项，

从中选择自己喜爱的窗口布局。

另外，VS.NET 中的各种窗口在屏幕上的布置非常灵活，大大方便了程序员的使用并且节省了界面有效空间。只要在某一窗口的标题条上右击，就会弹出一个快捷菜单，选择需要的窗体表现方式即可。单击“指针”标题条，弹出有关指针的快捷菜单，如图 1-9 所示。



图 1-8 【属性】窗口

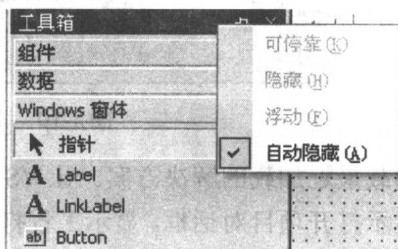


图 1-9 窗口布局快捷菜单

- 可停靠：在默认情况下，界面上的各种子窗口是停靠的（Dock），也就是说，窗口根据相互的大小和位置关系充满主窗口而不是相互重叠。
- 隐藏：隐藏一个窗口，只要单击右上角的“×”按钮，或者右击窗口的标题条，从弹出菜单中选择【隐藏】选项。要显示一个隐藏的窗口，从【视图】菜单中选择合适的命令即可。
- 浮动：窗口还可以悬浮。要悬浮一个窗口，右击标题条，选择【浮动】即可。一个悬浮窗口通过拖拽标题条能够在屏幕上移动，通过拖拽边界也可以设置大小。
- 自动隐藏：VS.NET 的窗口有自动隐藏特性，当窗口不再使用时，它会隐藏自己并以长方形图标显示出来。例如【工具箱】窗口默认就是自动隐藏的。

1.2.2 自定义开发环境

如果对 VS.NET 提供的默认开发环境不满意，可以通过单击【工具】|【选项】菜单来打开【选项】对话框，设置自己喜欢的开发环境，如图 1-10 所示。

在对话框左边的选项框中列出一些选项类，如环境、源代码管理等。单击每一个类又可以展开一系列子类。选择某个子类，则有一个箭头显示在名称的旁边。对话框的右侧则显示了可以进行配置的具体项目。下面介绍常用的一些配置选项。

1. 常规

这个选项指定了 VS.NET 开始时的状态，可以进行的设置将出现在【启动时】下拉列表里，它包括以下几个方面：

- 显示起始页。显示 VS.NET 开始页，其功能是打开一个最近的方案、产生一个工

程或者浏览 Visual Basic.NET 开发人员感兴趣的几个网站。

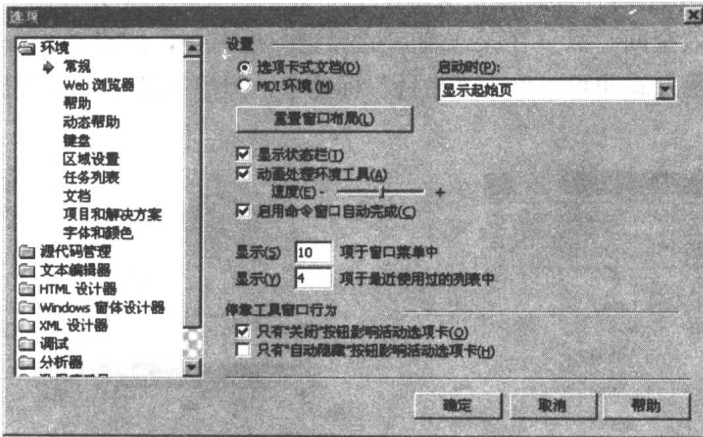


图 1-10 【选项】对话框

- 加载最近加载的解决方案。让 VS.NET 自动装载和显示最近正在进行的方案。
- 显示打开项目对话框。显示工程打开对话框。功能是定位和打开已经存在的工程。
- 显示新建项目对话框。显示新建项目对话框。功能是创建一个新项目。
- 显示空环境。打开开发环境，不加载任何方案。

注意：单击【重置窗口布局】按钮，即可恢复原始默认屏幕环境布局。

2. 项目和解决方案

在【环境】|【项目和解决方案】选项类中包含了大量可更改的选项。【项目位置】设置决定了项目和方案文件存储时的默认位置。【生存和运行】选项影响编译和运行工程时的文件存储。可以指明所有的文件自动存储，这样就不会提示文件是存储还是不存储。

3. 关于 Visual Basic

在【文本编辑器】|【Basic】|【VB 专用】选项中，有三个选项是针对 Visual Basic 语言的，应该将它们都选中。

1.3 Visual Basic.NET 项目组织

为了有效地管理开发工作所需要的各项，如引用、数据连接、文件夹和文件，VS.NET 提供了两个容器：解决方案和项目。查看和管理这些容器及其关联项的界面是【解决方案资源管理器】窗口。

1. 解决方案

从概念上讲，解决方案是应用程序中的项目和解决方案项的容器。解决方案通常包含一个或多个相关的项目，如图 1-11 所示。

解决方案的定义存储在.sln 和.suo 两个文件中。

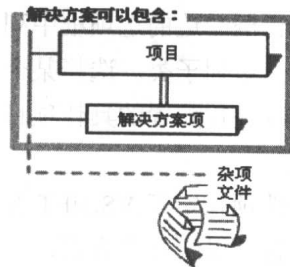


图 1-11 解决方案

解决方案定义文件 (.sln) 存储定义解决方案的元数据，而 .suo 文件则根据所选选项存储对 IDE 进行自定义的元数据。例如，【解决方案资源管理器】窗口提供“杂项文件”文件夹，该文件夹仅在启用显示它的选项后才显示。

2. 项目

项目是在 VS.NET 中创建应用程序、组件和服务的起点，是用来管理源代码、数据连接和引用的容器。

项目的范畴小于解决方案，一个解决方案可以包含一个或多个项目，创建项目时，VS.NET 会自动生成一个解决方案。项目通常输出为可执行程序 (.exe)、动态链接库 (.dll) 文件或模块等，如图 1-12 所示。

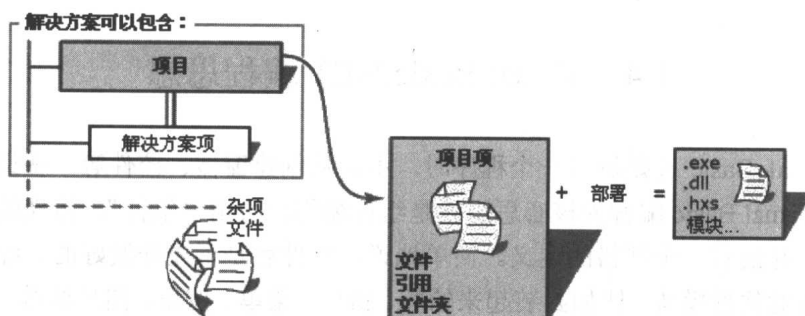


图 1-12 项目概念示意

项目可以简单，也可以复杂。一个简单的项目可能由一个窗体、源代码文件和一个项目文件组成。复杂的项目则可能还包含数据库脚本、存储过程和对现有 Web 服务的引用等。这些包含在项目中的项可以分成两大类：一类表示位于硬盘上的具体文件和文件夹；另一类表示在项目中采用，但又不是项目一部分的那些项。

例如，在图 1-13 中【解决方案资源管理器】所展示的项目组件中，可以看到以下三个主项。

- 引用：这部分包括对 5 个项目的连接或引用：System、System.Data、System.Drawing、System.Windows.Forms 和 System.XML。这些项目可以为开发人员所用，但并不是项目的一部分。
- AssemblyInfo.VisualBasic：这个文件包含了项目集合的信息。
- Form1.VisualBasic：这是项目的源代码文件。

Visual Basic.NET 支持很多项目类型。当创建一个新项目时，可以选择一个合适项目的模板。模板会自动生成框架代码并添加相应的辅助文件。

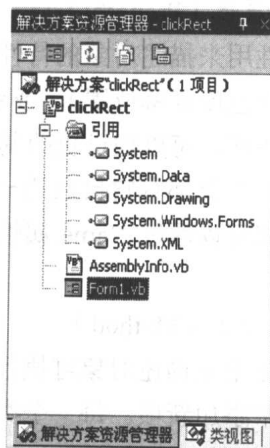


图 1-13 项目组件

3. 项目类型和项目包含的文件

使用 Visual Basic.NET 可以创建许多不同类型的项目，如表 1-1 所示。

表 1-1 Visual Basic 6.0 和 Visual Basic.NET 的项目类型比较

Visual Basic 6.0 项目	Visual Basic.NET 项目
标准 EXE	Windows 应用程序
ActiveX DLL	类库
ActiveX EXE	类库
ActiveX 控件	Windows 控件库
ActiveX 文档	无等效项。Visual Basic.NET 可以与 ActiveX 文档交互操作
DHTML 应用程序	无等效项。使用 ASP.NET Web 应用程序
IIS 应用程序 (Web 类)	无等效项。使用 ASP.NET Web 应用程序

1.4 Visual Basic.NET 编程思想

使用 Visual Basic 开始编写一个程序时，不必从创建窗体、控件等一些最底层的工作做起，因为 Visual Basic 编程的核心思想就是组件编程。所谓“组件”，体现的是一种重复使用的思想，并没有一个严格的定义。简单地说，组件就是事先开发好的、经过测试的、可以重复使用的软件模块，比如封装起来的类、窗口、菜单、按钮、图片框等。Visual Basic 编程人员的主要工作就是“装配”这些组件以实现所需要的功能。用什么来装配呢？就是用 Visual Basic 代码。

大部分的 Visual Basic 组件都是以类的形式实现和提供给用户的。在面向对象程序设计中，类是一个很重要的概念，类描述对象的“结构”，而对象则是类的可用“实例”。由于对象是类的一个“实例”，所以创建对象的动作就称为“实例化”。

例如，在窗体上放置两个命令按钮，实际上就是 Visual Basic 提供的 Button 类的两个实例，这两个按钮对象相互之间是完全独立的，它们的大小、颜色、显示文本都可以不同。

1. 属性 (Property)

属性用来描述对象的特性，是存储在对象中的一小段信息。它们定义对象的特征（如大小、颜色或屏幕位置），或者定义对象行为的某一方面（如是否启用或可见）。若要更改对象的特征，可以通过更改其相应属性的值来完成。

每一个类都有它自己的一套属性，有些属性是许多不同类所共有的。例如，任何一个对象，都可以通过 Name 属性来在程序中惟一确定它。而有些属性则仅是一个类或一些类特有的。

2. 方法 (Method)

方法用来描述对象可执行的操作。例如，Add () 是 ComboBox 对象的一个方法，它向组合框中添加新项。每一个类根据使用目的的不同拥有自己一套独特的方法。习惯上，在方法名后面加一对括号，以区分属性名和方法名。

3. 事件 (Event)

事件是由对象识别的操作（如单击鼠标或按下某个键），可以为其编写代码以进行响