

黄文钰 编著

# Visual Basic .NET

## 网络数据库程序设计



### 本书重点:

- .NET程序的设计基础
- .NET程序的深入剖析
- 数据库的概念及分析
- 数据库的程序设计
- 网络数据库的制作实务
- 数据库的查询及报表



清华大学出版社

# Visual Basic .NET 网络数据库程序设计

黄文钰 编著

清华大学出版社

北 京

## 内 容 简 介

Visual Basic .NET 属于 .NET 中的语言, 是 Visual Basic 6 的升级版本。本书从 Visual Basic .NET 在数据库中常用的一般程序开始说明, 然后以实例的方式指导读者在 Microsoft Access 和 Microsoft SQL Server 中访问数据库, 主要内容包括 .NET 深入浅出、.NET 基础、Visual Basic 实战应用、Visual Basic 高级应用、访问数据库、单机数据库应用、OLE DB 数据库设计、客户端/服务器 (Microsoft SQL Sever) 介绍、客户端/服务器数据库设计、数据查询及报表。

本书适合 .NET 初学者和对 Visual Basic 已有基础的读者, 同时它还可以供广大教师和计算机专业编程人员使用。

本书繁体字版名为《Visual Basic .NET 网络资料库程序设计》, 由文魁资讯股份有限公司出版, 版权属黄文钰所有。本书简体字中文版由文魁资讯股份有限公司授权清华大学出版社独家出版。未经本书原版出版者和本书出版者书面许可, 任何单位和个人均不得以任何形式或任何手段复制或传播本书的部分或全部。

北京市版权局著作权合同登记号 图字: 01-2003-7375

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

Visual Basic .NET 网络数据库程序设计/黄文钰编著. —北京: 清华大学出版社, 2003  
ISBN 7-302-07870-X

I.V… II.黄… III.BASIC 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字(2003)第 122103 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社 总 机: 010-62770175 客户服务: 010-62776969

文稿编辑: 杨作梅

封面设计: 陈刘源

印 装 者: 清华大学印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 185 × 260 印张: 26 字数: 624 千字

版 次: 2004 年 2 月第 1 版 2004 年 2 月第 1 次印刷

书 号: ISBN 7-302-07870-X/TP · 5717

印 数: 1 ~ 5000

定 价: 42.00 元(含 1 张光盘)

# 前 言

.NET 是 Microsoft 公司 2002 年的重头戏，为什么这么说呢，因为近年来 Microsoft 公司已经将操作系统 Windows 系列、文字处理软件 Office、浏览器 Internet Explorer 等各方面的软件趋于完善，并且已经占据了市场主导地位，但在程序开发语言上却不尽人意，虽然 Visual Basic 在世界上使用的人数远远高于其他开发语言，但并不代表其技术是最领先的。

.NET 整合了 Microsoft 公司多个强大的开发工具，无非就是要增强其竞争优势，将 Delphi、Java、PHP 和 Perl 等开发语言一举击溃。.NET 适合在最新的操作系统 2000/XP 上开发并兼容旧有的程序代码，由此可以看出 Microsoft 公司想要整合旗下各种语言的决心，或许语言的整合就是未来的趋势。

Visual Basic .NET 是 .NET 的语言之一，就是 Visual Basic 6 的升级版本。而 .NET 同时还包含 ASP.NET 和 C# .NET，也就说明这些语言是可以相互支持的。

本书从 Visual Basic .NET 在数据库常用的一般程序开始说明，包含文件的存取和注册表(Registry)的控制，让读者在设计数据库前，能对数据库相关的程序有所了解；再利用实例的方式，指导读者在 Microsoft Access 和 Microsoft SQL Server 中存取数据库。

本书共分 10 章：

- |                              |                                |
|------------------------------|--------------------------------|
| ● 第 1 章 .NET 深入浅出            | 了解 Visual Basic .NET 的基本常识     |
| ● 第 2 章 .NET 基础              | 学习 Visual Basic .NET 的基本程序设计   |
| ● 第 3 章 Visual Basic 实战应用    | 学习 Visual Basic .NET 的常见程序设计   |
| ● 第 4 章 Visual Basic 高级应用    | 学习 Visual Basic .NET 的文件及注册表存取 |
| ● 第 5 章 访问数据库                | 学习数据库的基本概念                     |
| ● 第 6 章 单机数据库应用              | 学习 Access 的操作方法                |
| ● 第 7 章 OLE DB 数据库设计         | 学习 Access 数据库的存取设计             |
| ● 第 8 章 Microsoft SQL Server | 学习 SQL Server 的操作方法            |
| ● 第 9 章 客户端/服务器数据库设计         | 学习 SQL Server 数据库的存取设计         |
| ● 第 10 章 数据查询及报表             | 学习制作数据表格的查询及报表打印               |

本书适合 .NET 初学者和熟悉 Visual Basic 语言的读者使用，如果读者无数据库概念或完全无程序设计经验，最好先学习一些相关知识，以便更好地阅读本书的内容，因为本书非常实用，只针对重点来说明，可能无法照顾到初学者。

在本书的编写过程中，得到许多朋友的帮助，让本书有机会展现在各位面前。要特别感谢黄群弼老师的技术指导，并感谢文魁资讯的黄雪秋和萧丽娟小姐，还要感谢赵紫君小姐细心的编排设计使本书更加完美。

作者 黄文钰  
2002 年 8 月

# 目 录

|                                            |    |                                        |    |
|--------------------------------------------|----|----------------------------------------|----|
| 第 1 章 .NET 深入浅出 .....                      | 1  | 属性及方法 .....                            | 40 |
| 1.1 Visual Studio .NET 简介 .....            | 1  | 2.4.1 设计方式 .....                       | 42 |
| 1.1.1 .NET 的影响 .....                       | 2  | 2.4.2 程序说明 .....                       | 43 |
| 1.1.2 .NET 的结构 .....                       | 3  | 2.5 Visual Basic .NET 的数组 .....        | 46 |
| 1.2 .NET Framework .....                   | 3  | 2.5.1 设计方式 .....                       | 48 |
| 1.2.1 何谓 .NET Framework .....              | 4  | 2.5.2 程序说明 .....                       | 48 |
| 1.2.2 从 Windows API 到 .NET Framework ..... | 5  | 2.6 Visual Basic .NET 的子程序及自定义函数 ..... | 52 |
| 1.2.3 从编程语言谈起 .....                        | 8  | 2.6.1 设计方式 .....                       | 54 |
| 1.2.4 不同编程语言的互通 .....                      | 9  | 2.6.2 程序说明 .....                       | 54 |
| 1.3 .NET 简介 .....                          | 9  | 第 3 章 Visual Basic 实战应用 .....          | 58 |
| 1.4 XML 支持 .....                           | 11 | 3.1 键盘事件的控制 .....                      | 58 |
| 1.5 ASP.NET .....                          | 12 | 3.1.1 设计方式 .....                       | 59 |
| 1.6 安装 .NET .....                          | 13 | 3.1.2 程序说明 .....                       | 61 |
| 1.6.1 硬件需求 .....                           | 13 | 3.2 利用 Timer 和键盘事件做游戏 .....            | 64 |
| 1.6.2 软件需求 .....                           | 13 | 3.2.1 设计方式 .....                       | 65 |
| 第 2 章 .NET 基础 .....                        | 15 | 3.2.2 程序说明 .....                       | 66 |
| 2.1 Visual Basic .NET 的基本操作 .....          | 15 | 3.2.3 执行结果 .....                       | 72 |
| 2.1.1 打开新项目 .....                          | 15 | 3.3 鼠标事件的控制 .....                      | 72 |
| 2.1.2 窗口介绍 .....                           | 18 | 3.3.1 设计方式 .....                       | 74 |
| 2.2 执行第 1 个 Visual Basic .NET 的程序 .....    | 25 | 3.3.2 程序说明 .....                       | 75 |
| 2.2.1 设计方式 .....                           | 27 | 3.3.3 执行结果 .....                       | 77 |
| 2.2.2 程序说明 .....                           | 28 | 3.4 菜单制作与设计 .....                      | 79 |
| 2.2.3 执行结果 .....                           | 28 | 3.4.1 设计方式 .....                       | 79 |
| 2.3 Visual Basic .NET 常用的循环控制 .....        | 30 | 3.4.2 程序说明 .....                       | 84 |
| 2.3.1 设计范例 .....                           | 34 | 3.5 MDI 窗口控制与设计 .....                  | 89 |
| 2.3.2 程序说明 .....                           | 34 | 3.5.1 设计方式 .....                       | 89 |
| 2.3.3 执行结果 .....                           | 37 | 3.5.2 程序说明 .....                       | 90 |
| 2.4 Visual Basic .NET 的对象、                 |    | 3.5.3 执行结果 .....                       | 91 |
|                                            |    | 3.6 对话框设计 .....                        | 93 |
|                                            |    | 3.6.1 设计方式 .....                       | 94 |
|                                            |    | 3.6.2 程序说明 .....                       | 97 |

|                                         |                                           |
|-----------------------------------------|-------------------------------------------|
| <b>第 4 章 Visual Basic 高级应用</b> .....103 | 5.3.5 SQL..... 154                        |
| 4.1 注册表控制设计.....103                     | 5.4 分析数据库..... 155                        |
| 4.1.1 设计方式.....107                      | 5.4.1 数据库设计前初步分析..... 155                 |
| 4.1.2 程序说明.....107                      | 5.4.2 程序设计和用户的<br>关系架构..... 156           |
| 4.2 用注册表保存文件.....109                    | 5.5 设计数据库表格..... 158                      |
| 4.2.1 设计方式.....110                      | <b>第 6 章 单机数据库应用</b> ..... 165            |
| 4.2.2 程序说明.....111                      | 6.1 Microsoft Access 应用..... 165          |
| 4.3 使用 RegistryKey 存取注册表.....114        | 6.1.1 Microsoft Access 简介..... 165        |
| 4.3.1 设计方式.....115                      | 6.1.2 Microsoft Access 的<br>数据类型..... 168 |
| 4.3.2 程序说明.....116                      | 6.1.3 新增修改一个数据表格..... 169                 |
| 4.4 存取文本文件.....120                      | 6.2 数据表格的使用..... 171                      |
| 4.4.1 设计方式.....121                      | 6.2.1 导入数据表格..... 171                     |
| 4.4.2 程序说明.....121                      | 6.2.2 复制数据表格..... 173                     |
| 4.5 图片文件预览应用.....124                    | 6.2.3 删除数据表格..... 174                     |
| 4.5.1 设计方式.....127                      | 6.2.4 更改数据表格名称..... 174                   |
| 4.5.2 程序说明.....129                      | 6.3 制作查询..... 175                         |
| <b>第 5 章 访问数据库</b> .....132             | 6.4 制作窗体..... 178                         |
| 5.1 客户端/服务器概念.....132                   | <b>第 7 章 OLE DB 数据库设计</b> ..... 182       |
| 5.1.1 单机数据库的架构.....132                  | 7.1 用户接口设计..... 182                       |
| 5.1.2 客户端/服务器的架构.....133                | 7.2 使用 OLE DB 连接数据库..... 190              |
| 5.1.3 客户端/服务器的启动<br>过程.....135          | 7.2.1 设计方式..... 190                       |
| 5.1.4 服务器的处理过程.....136                  | 7.2.2 程序说明..... 192                       |
| 5.1.5 客户端的处理过程.....136                  | 7.3 查询(Querying)数据库..... 193              |
| 5.1.6 两层式客户端/服务器<br>架构.....137          | 7.3.1 设计方式..... 198                       |
| 5.1.7 三层式客户端/服务器<br>架构.....137          | 7.3.2 程序说明..... 199                       |
| 5.1.8 客户端/服务器的存取<br>方式.....139          | 7.4 基本文件制作第一部分..... 203                   |
| 5.2 数据库的使用及管理.....140                   | 7.4.1 设计方式..... 204                       |
| 5.2.1 网络的安装及使用.....140                  | 7.4.2 程序说明..... 206                       |
| 5.2.2 ODBC 的使用方式.....142                | 7.5 基本文件制作第二部分..... 209                   |
| 5.3 数据库基础.....149                       | 7.5.1 设计方式..... 210                       |
| 5.3.1 何谓数据库.....150                     | 7.5.2 程序说明..... 211                       |
| 5.3.2 数据库能帮你做什么.....151                 | 7.6 AB 文件制作——单窗体..... 215                 |
| 5.3.3 选择数据库.....152                     | 7.6.1 设计方式..... 215                       |
| 5.3.4 关系型数据库.....153                    | 7.6.2 程序说明..... 218                       |
|                                         | 7.7 AB 文件制作——双窗体..... 222                 |
|                                         | 7.7.1 设计方式..... 223                       |

|                                              |     |                                          |     |
|----------------------------------------------|-----|------------------------------------------|-----|
| 7.7.2 程序说明——Form1<br>(显示 SCLASS 表格).....     | 228 | 9.4.1 设计方式 .....                         | 306 |
| 7.7.3 程序说明——Form2<br>(显示 STUDENT 表格).....    | 234 | 9.4.2 程序说明 .....                         | 307 |
| <b>第 8 章 Microsoft SQL Server</b> .....      | 238 | 9.5 AB 文件制作——单窗口 .....                   | 311 |
| 8.1 SQL Server 2000 概述.....                  | 238 | 9.5.1 设计方式 .....                         | 311 |
| 8.2 SQL Server 数据库安装与启动.....                 | 239 | 9.5.2 程序说明 .....                         | 313 |
| 8.2.1 安装 Microsoft SQL<br>Server.....        | 239 | 9.6 AB 文件制作——双窗口 .....                   | 318 |
| 8.2.2 启动 Microsoft SQL<br>Server.....        | 240 | 9.6.1 设计方式 .....                         | 319 |
| 8.2.3 启动 Microsoft SQL Server<br>服务管理器 ..... | 244 | 9.6.2 程序说明——Form1<br>(显示 BSUP 表格).....   | 322 |
| 8.3 基本的数据库管理.....                            | 244 | 9.6.3 程序说明——Form2<br>(显示 BGDS 表格) .....  | 328 |
| 8.3.1 建立一个数据库 .....                          | 244 | 9.7 数据访问的事务 Transaction .....            | 331 |
| 8.3.2 数据库的导出与导入 .....                        | 247 | 9.7.1 Querying Transaction<br>设计方式 ..... | 332 |
| 8.3.3 备份及还原数据库 .....                         | 252 | 9.7.2 Querying Transaction<br>程序说明 ..... | 333 |
| 8.3.4 数据库的定时维护计划.....                        | 255 | 9.7.3 Update Transaction<br>设计方式 .....   | 336 |
| 8.3.5 删除数据库 .....                            | 260 | 9.7.4 Update Transaction<br>程序说明 .....   | 337 |
| 8.4 表的基本概念 .....                             | 261 | 9.8 Parameter 参数传递.....                  | 343 |
| 8.4.1 表的基本概念 .....                           | 261 | 9.8.1 设计方式 .....                         | 344 |
| 8.4.2 利用企业管理器建立表.....                        | 262 | 9.8.2 程序说明 .....                         | 345 |
| 8.4.3 修改表格 .....                             | 264 | 9.9 导出 Html .....                        | 352 |
| 8.4.4 数据类型的种类 .....                          | 266 | 9.9.1 设计方式 .....                         | 353 |
| 8.5 设计及运行存储过程.....                           | 267 | 9.9.2 程序说明 .....                         | 354 |
| 8.5.1 存储过程的概念 .....                          | 267 | 9.10 导出 Xml .....                        | 356 |
| 8.5.2 建立及管理存储过程 .....                        | 268 | 9.10.1 设计方式 .....                        | 357 |
| 8.5.3 运行存储过程 .....                           | 274 | 9.10.2 程序说明 .....                        | 358 |
| 8.6 使用 SQL 查询分析器 .....                       | 278 | 9.11 存储过程的应用 .....                       | 361 |
| <b>第 9 章 客户端/服务器数据库设计</b> .....              | 283 | 9.11.1 设计方式 .....                        | 362 |
| 9.1 SQL Server 组件的方法及属性.....                 | 283 | 9.11.2 程序说明 .....                        | 362 |
| 9.2 使用 SQLClient 来连接数据库.....                 | 290 | <b>第 10 章 数据查询及报表</b> .....              | 367 |
| 9.2.1 设计方式 .....                             | 293 | 10.1 数据库表格基本查询.....                      | 367 |
| 9.2.2 程序说明 .....                             | 295 | 10.1.1 设计方式 .....                        | 368 |
| 9.3 查询数据库 .....                              | 297 | 10.1.2 程序说明 .....                        | 369 |
| 9.3.1 设计方式 .....                             | 298 | 10.2 使用参数来查询数据 .....                     | 374 |
| 9.3.2 程序说明 .....                             | 298 | 10.2.1 设计方式 .....                        | 374 |
| 9.4 基本文件制作 .....                             | 305 |                                          |     |

|                                   |     |                   |     |
|-----------------------------------|-----|-------------------|-----|
| 10.2.2 程序说明 .....                 | 375 | 10.4.2 程序说明 ..... | 391 |
| 10.3 使用 Crystal Report 设计报表 ..... | 380 | 10.5 直接输出报表 ..... | 395 |
| 10.3.1 设计方式 .....                 | 383 | 10.6 制作图形报表 ..... | 398 |
| 10.3.2 程序说明 .....                 | 387 | 10.7 制作组报表 .....  | 401 |
| 10.4 导出各种类型的报表 .....              | 390 | 10.8 直接输出报表 ..... | 404 |
| 10.4.1 设计方式 .....                 | 390 |                   |     |

# 第 1 章 .NET 深入浅出

## 本章概要

Visual Studio .NET 是 Microsoft 公司在 2002 年推出的产品,它是一个全新的开发工具,是一个将 Microsoft 公司所有编程语言开发工具整合在一起的工具的集合。如果开发人员只会使用一套语言开发,在和别的语言整合时,就容易出现一些问题。

Visual Studio .NET 除了可以开发 Windows 最基本的窗口应用程序外,还可以开发移动通信、ASP、XML 等 Web 服务。其中包含了 Visual Basic .NET、Visual C++ .NET 及 Visual C# .NET,同时可以实现语言和开发环境的共享,可以很容易让程序员开发 Web 应用程序。

本书重点讲解的是 Visual Basic .NET 和客户端/服务器数据库,利用 Visual Basic .NET 强大的功能来编写程序,并用 SQL Server 来设计数据库,两者都是 Microsoft 公司的产品,因此在搭配上更是天衣无缝。

Visual Basic .NET 是 Visual Basic 6 的升级版,但是两个版本在功能上有很大的不同,除了外观做了一些改变,其核心也有较大更新,Visual Basic .NET 不但加入许多全新的和改进后的语言功能,在内部设计上也增加了许多 Visual Basic 6 不具有的功能,如多线程处理、继承、接口及过载化(Overloading)、结构化异常处理(Structured Exception Handling) 和自定义属性等。

Visual Basic .NET 是一个新的、强大的面向对象的编程语言。Visual Basic 6 在设计过程中,本身能提供的功能有限,它升级为 Visual Basic .NET 用了很长的时间,这是因为除了 Visual Basic .NET 增加了许多功能外,其内部的设计逻辑已经和原来的逻辑大不相同。例如继承,Visual Basic 6 在继承时并没有让用户可以很清楚地了解设计时的因果,所以用户可以设计出许多功能强大的程序,但是却无法弄清楚为什么会有这样的结果。Visual Basic .NET 在继承时,已经比 Visual Basic 6 有更清楚的逻辑关联,用户在设计时,就可以很清楚地了解整个程序的来龙去脉了。

## 1.1 Visual Studio .NET 简介

Visual Studio 是 Microsoft 公司很早就发布的一套系统,其中包含了 Visual 系列的语言,有 Microsoft Visual Basic、Microsoft Visual C++、Microsoft C#和 Microsoft Jscript,虽然有些语言没有加上 Visual,但也都是使用所见即所得的设计思想(What you see is what you get)。

Microsoft 公司在很早以前就已经将许多语言的帮助放在 MSDN Library 中了,用户只要装上 MSDN Library,就可以马上查询到所有语言的帮助,但是这样的帮助文件似乎过于庞大,而且并不是每个人都全部使用 Microsoft 公司同一系列的产品。但在 Visual

Studio .NET 中, Microsoft 公司不仅整合了所有的帮助文件, 而且将所有重点的编程语言全部整合在 Visual Studio .NET 中。

### 1.1.1 .NET 的影响

Microsoft 公司在 Windows 上的开发产品相当多, 几年前, 他们已经将原来各自独立的 Word、Excel、Access、Outlook 和 PowerPoint 整合成 Office, 并且风格也都统一了, 每个文件之间也可以相互支持和沟通。

Microsoft 公司在 Visual Basic 6 发布后, 一直不见 Visual Basic 7 的身影, 原来他们将整合一套 .NET, 而 .NET 本身将会包含 Microsoft 公司已有的开发语言软件, 例如包含 Visual Basic、C#、C++、ASP 和 J# 等, 每一套各有所长的开发工具, 将合成一个全能的开发环境。

.NET 延伸了来自 Internet 及 OS 的观念, 让程序开发成为操作系统的基础, 从而使开发人员可以很容易地进行程序开发, 并可通过应用程序与 Internet 连接。因此 .NET 对开发环境而言是一个相当重要的新的里程碑。

例如: 突然有个项目要在 Windows 上让用户输入数据, 并能够在网络的接口显示, 最后要在特定的硬件中输出, 如果同时要写出几个不同关联的程序, 以及要使用几个不同性质的工具, 这时只要用一套整合的、可开发出多种功能的程序, 就可以很快地解决问题。这在早期的 Office 中也有所体现, 因为早期在 Word 中的资料无法在 Excel 中操作, 更无法最后变成演示文稿, 后来 Office 整合了这些功能, 就可以达到相互支持的目的。

.NET 本身要面对的对象相当广, 所以它从最简单的整合性开发环境 IDE, 到各种深入开发的技术支持, 都有非常详尽的设计。如果不懂编程语言, 可以从 Visual Basic 开始, 因为它很容易上手, 如果已经掌握编程语言, 也可以向 .NET 转变, 因为它的支持程度, 可以让你有无限发挥的空间。

.NET 将是未来相当重要的程序之一, 因为它让原来对计算机程序开发相当惧怕的人, 由于它的易用性变得不再惧怕。它原来只能设计基于 Windows 平台的程序, 也因应各种新硬件的出现, 而能在各种接口上使用(例如 PDA、手机和笔记本电脑), 可以在相互之间轻松地完成连接, 而不用再为不同的硬件而大伤脑筋。而网络的发展, 也让 .NET 在单机上的存取延伸到网络中, 从而使用户可以随时随地存取所需要的数据。

.NET 将是新的银色子弹, 为什么? .NET 会是新的开发主角, 除了 Microsoft 公司在目前的 Windows 平台仍是相当受到欢迎, 而搭配的开发软件全部整合在 .NET 之中, 可以看出 Microsoft 公司对于 .NET 的期望。 .NET 不但可以在 Windows 开发出全新的应用程序, 网页服务(Web Services)也将是新的整合服务。

高速的开发环境将来临。早期的语言都是给予相当多的指令, 但程序员必须要将这些指令先写成常用的函数, 然后才能开始有比较好的开发环境, 如果重新开发了一套新的程序, 也许这些旧有的函数还可以用; 也许这些函数又要根据不同的环境, 必须重新开发。 .NET 在一开始就针对设计人员常用的需求预先准备好了许多的资源, 无论是不是原来在 .NET 上开发的程序员, 都不用浪费时间再设计这些重复的函数, 并且能将这些函数组件整合成更新的功能来使用。

这是一个新的程序开发时代, 因为从早期的文字接口跨入图形接口, 本身就是一个相

当大的革命, 如果不使用图形和鼠标就是过时落伍的程序, 而随着网络的发展, 使程序开发又面临了另一场革命, 不但原来 Windows 的程序要能上网, 而且文件还要能输出成 HTML 或 XML 格式, 软件之间还能用 Internet 来作沟通, 这种新的开发趋势, 已经是势在必行了。

软件网络化是程序开发的新方向, 除了在网页上存取数据后展现出的型态外(例如 ASP、PHP)等, 网络服务的功能也是新的方向, 例如, 开发网络数据库、交互式软件等, 只要是能应用在网络上的, 将是新的开发重点。

就用户而言, 语言只要是能达到目的, 就是好语言, 因为用户不管程序是用什么完成的; 但是对程序员而言, 可以有很多方式, 而 .NET 整合了许多功能, 程序员可以利用其完整的结构平台来开发, 不但整合性高, 具有完整的支持, 而且对于相同开发环境的程序员, 也将更容易上手。

### 1.1.2 .NET 的结构

.NET 本身区分出很多功能(如图 1.1), 从另一个角度来看, 它则是一个有许多强大功能的组合体, Microsoft 公司将 ASP 和 ADO 纳入 .NET Framework, 然后利用 Visual Studio 原来就有的 Visual Basic、Visual C++ 和 J# 加上新的 C#, 并涵盖了单机和网络的所有设计范围。

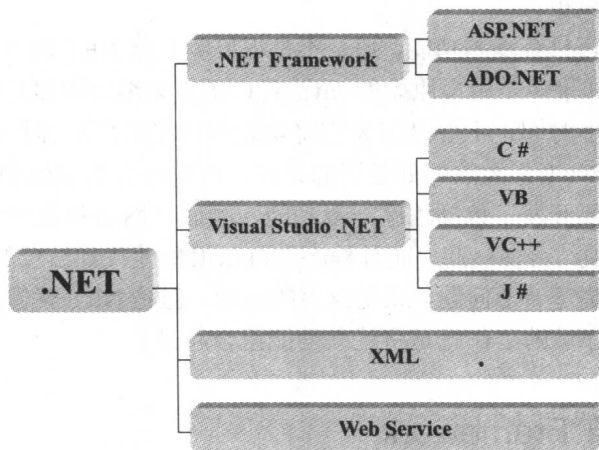


图 1.1

## 1.2 .NET Framework

大部分的程序员, 在经过一段时间的程序设计后, 常常会遇到一些令人痛苦的事: 程序设计好了, 要设计下一个程序时, 发觉上一个程序也写过同样的功能, 但是将程序代码复制过来后, 还要改许多地方, 若要写成模块, 又有点小题大作, 不写成模块, 又深怕下次还要重改。

当我们在遇到需要写重复使用的程序代码时, 最需要的就是将这些程序代码写成一堆函数(Function), 最好这些是自己的资源, 而别人的也能复制使用, 最好还有源代码, 修改

后，又可完成新的功能。但是如果这些功能是早就已经有了，那是不是可以节省程序员的设计时间，可以让程序员在最短的时间内就能找到并且学会使用呢？

早期的编程语言都只提供程序指令，所有功能都要程序员自己来写，但是现在语言的组件化、模块化，使得可以重用的观念开始流行，大家只要一直调用别人已经设计好的功能，就能完成整个程序。例如，本来要制作报表，只要将数据导入 Microsoft Word，用户就能轻易修改成自己所需的样式，而程序员就不用因为不同的需求，而将一个报表修改成很多格式。

在 Microsoft Word 导出时，就用到 OLE 的功能，使它能导入 Microsoft Excel 等其他程序，而 OLE 也是最早使用的组件模块，OLE 是最早的 COM 模块，它结构在 COM 之上，能够使用户在文件内嵌入其他文件。虽然在使用 OLE 时，只是将其他应用程序的功能和文件在另一个应用程序上显示，但是这对于必须要在同一个文件上使用两种应用程序所产生的结果而言，是相当不容易的事。

OLE 可以让你在不同应用程序的对象模型(Object Model)中使用对象(Object)；在整合不同的应用程序时使用 OLE，就可以使整合的效果更加完美。而在利用程序写作时，如果这些对象的属性(Property)与方法(Method)，能够像 OLE 之间那样相互使用，并且在继承时可以使用，就可以使设计的程序更加容易使用，而且因为大部分的设计方式相同，从而大大减轻程序员的负担。

如果一个编程语言只有基本功能，那是不是很好呢？就.NET 而言，至少它提供了许多可以用的附加功能，如果你是用最原始的 Java 或 C 语言来编写程序，可能写个功能要很长时间，修改也要花很多时间，主要是因为它将功能都放在用户不容易找到的地方，或是根本要自己来编写；而.NET 最大的好处是资源丰富，可以结合的资源相当多。

程序有许多层面要开发，包括程序一般功能、画面、网络和数据库等，如果每一个层面是相关的，那就必须结合在一起，而且程序员可以很快地找到结合的方式，如果是以前，可能要自己想办法解决每一个问题，或是兼容的问题，但是.NET 就像是个大家庭，想要的功能都已经存在，程序员的工作就是将这些功能组合起来。

### 1.2.1 何谓.NET Framework

.NET Framework 是一个多语言、多组件的开发结构及环境，而可以执行的环境包含了下列 3 个主要部分：

- 公共语言运行时(Common Language Runtime, CLR)
- 统一编程类
- Active Server Pages+ (ASP+)

#### 1. 公共语言运行时

公共语言运行时，就是程序在设计完成后，要先执行组件，公共语言运行时就负责内存管理、执行线程激活及关闭和组件安全防护等，通过这样的动作，在执行程序时，可以防止无效程序的发生。所有的程序都有开发阶段和执行阶段，但是.NET 的执行阶段提供的是一个多语言同时执行的结构，可以包含所有的.NET 语言共同使用。

## 2. 统一编程类

.NET 中的所有对象类主要提供了统一、面向对象、可扩充和多层次的结构,使这些类程序库可以让开发人员使用。原本 Microsoft 中所有的编程语言的程序员,都要开发或使用自己语言的函数库,而 .NET 就是要将这些原本分散且重复的功能整合起来,让程序员不用再学习多种开发结构。如果所有程序员使用任何 .NET 的语言,却不用重新学习开发的结构,那就意味着任何语言都只是和程序员沟通的表层,而内在都是相同的,那程序员就可以任意地选择自己想要的语言,而不必因为开发环境而困扰。

## 3. Active Server Pages+ (ASP+)

ASP 在开发时是相当简单的,通过程序和 HTML 相互结合,可以轻易地构建出网页应用程序服务。而现在 ASP 有许多的模块可以使用,不但有原来 HTML 可以作基本的控制选项等,还有 .NET 提供的模块,虽然大部分的功能在服务器端就可以执行了,但通过会话状态管理(session state management)及进程循环(process recycling),可以使程序开发所要编写的程序代码减少,并可增加应用程序的可靠性,程序员只要针对商业逻辑来设计程序即可很快完成项目。

## 1.2.2 从 Windows API 到 .NET Framework

以往程序在设计时,除了语言本身所提供的组件和功能外,程序员必须要自己开发函数和对象,但是现在的结构可以让程序员开发程序时,使用别人的组件,这样就可以节省许多时间。调用或使用别人的组件,除了在编程语言中安装组件,也可以用调用 Windows API 的方式来使用外部的组件,如 DLL 文件,这样可以许多程序之间沟通更加容易,并且更有扩充性。

### 1. Windows API

在 Windows 系统中,有许多 API 可以使用,因为要控制或使用 Windows 系统的资源,可以通过 API 的支持来完成,正因为有了这些 API,才让程序员可以节省许多重新开发的时间。但是 Windows API 本身仍然还不够,主要是因为 API 太多太复杂,不容易调用,而且要找到适合我们使用的 API 资源也要花费很多时间,所以有时自行设计会比较快,如图 1.2 所示。

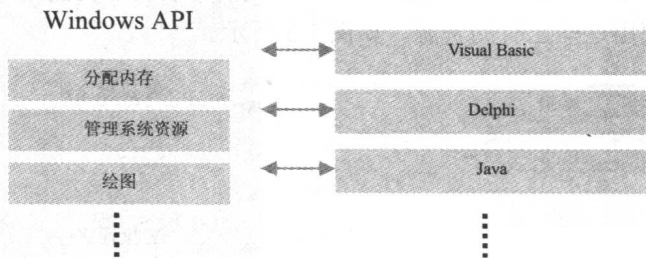


图 1.2

## 2. DLL(Dynamic Linking Library)

如果 Windows 所提供的 API 不够, 可以亲自来写专用的函数, 而 DLL 文件就是其中的一种, 虽然有的 Windows API 也是存在 DLL 中, 但是有些可以公开使用, 有些则不行。当我们写好了 DLL, 各种的编程语言就可根据调用的格式来使用此函数, 而我们只要知道使用的格式, 就能使用, 这样对于不同语言之间的沟通十分有益, 如图 1.3 所示。

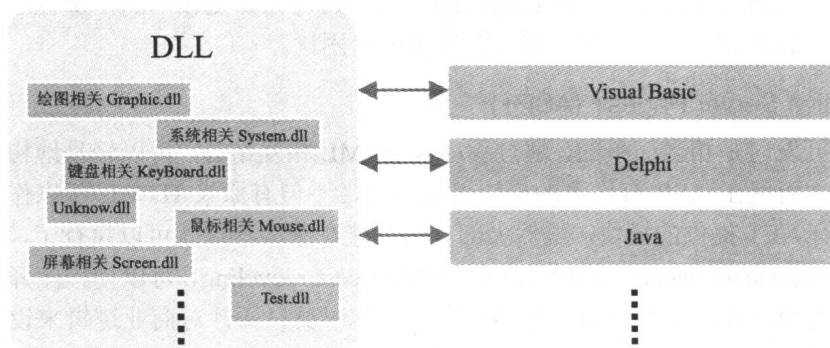


图 1.3

例如图 1.4, 同一个功能的 DLL 文件, 可以使用许多程序来产生, 也就是大多数的语言都能编写 DLL 文件, 使用的语言语法都各不相同, 但编译之后的 DLL 文件是一样的。所以我们用其他语言来调用已经编译好的 DLL 文件, 就不用在乎是使用什么程序来做编译了, 因为此时的 DLL 已经是共享格式的了。

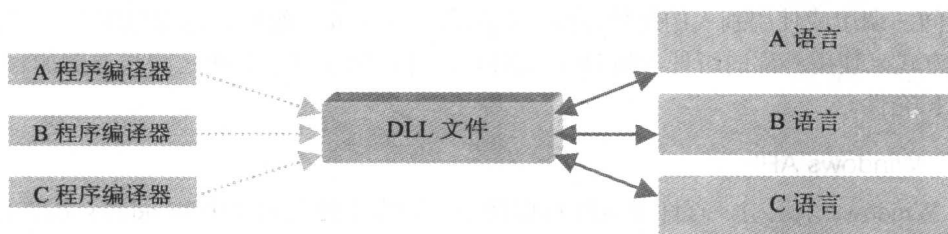


图 1.4

DLL 文件是单独编译的, 而执行文件也是自行编译完成的, DLL 内的函数可以共享, 但是执行文件自己内部的函数是不能共享的, 所以 X 和 Y 可以同时通过相同的 DLL 来完成功能, 节省不少工作、时间和资源, 所以有时我们可以将必要的常用函数做成 DLL, 这样就可以共同来使用这些已有的资源, 如图 1.5 所示。

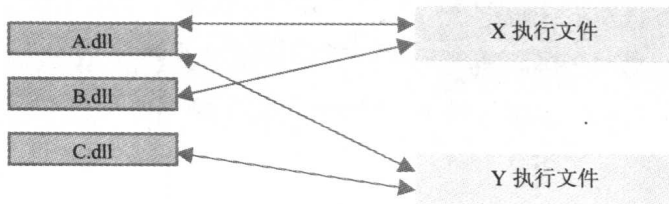


图 1.5

### 3. DLL Hell

原来设计的 DLL 是为了方便大家来使用共同的程序,进而节省空间和内存,但是如果共享的 DLL 文件,一再地被更新,那旧版本 DLL 的应用程序,就有可能无法继续使用,如图 1.6 所示。

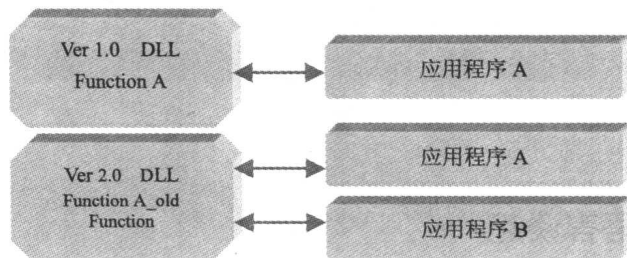


图 1.6

原来 1.0 版的 DLL 有函数 A 可用,但到了 2.0 版,可能将原来函数 A 改成了新功能,原来的功能就不能使用了,而原来的应用程序 A 可能是很早之前的程序,但是又无法修改,所以应用程序 A 就变成垃圾了,因此不断更新的功能,也可能使旧的功能无法使用,进而使 DLL 不断产生新的,又要保留旧的,不断产生大量的 DLL 垃圾或不能使用的旧应用程序。而这种情形却会一直地恶性循环下去,因为开发出新程序是必定会发生的事,而新版的 DLL 也就会一直被更新。

### 4. ActiveX 对象

ActiveX 有 3 种类型:

- Control(控制组件): 如 Visual Basic 所使用的控制组件,此类对象通常以 .ocx 为扩展名。
- ActiveX EXE: 兼具执行程序并提供对象功能,例如 Word、Excel、Outlook 都属于 ActiveX EXE,此类对象通常以 .exe 为扩展名。
- ActiveX DLL: 只提供对象功能,此类对象通常以 .dll 为扩展名。

这样看起来好像 ActiveX 和 DLL 等组件功能差不多,而事实也是如此,但是它比较方便,因为它是通过登录文件来存取其中的功能,所以如果将 ActiveX 的目录改变,也不会影响其执行。而 Microsoft 公司之所以要安排一个共享的组件,主要也是为了方便在 Windows 平台内所有的应用程序都能够利用最少的程序来执行最多的功能,从而减少重复性,使内存更能有效运用。

### 5. .NET Framework 对象类库

发布 .NET Framework 的目的就是要提供大量的对象,供 Windows 平台上的应用程序使用,因为 .NET 将许多开发语言给整合了起来,而共同的组件也就要同时可以使用,这样才会有效率,如图 1.7 所示。

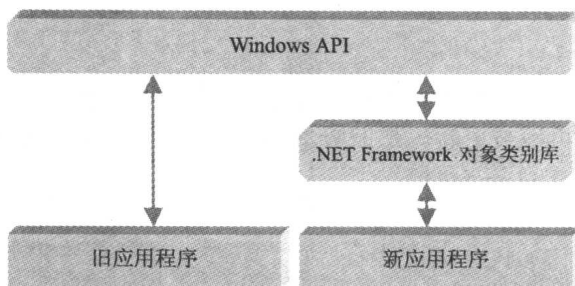


图 1.7

### 1.2.3 从编程语言谈起

在全新的 .NET 下编写程序，确实可以让人非常兴奋，但是原来要使用一种的编程语言，一下变成许多种语言整合在一起；原来只要学习一种语言，现在要学好多种语言；好像还没享受到大餐的美味，就先付出了许多学习上的代价，虽然 Microsoft 公司声称升级不需要花太多功夫，但是如果要完全掌握及操作整个 .NET，还真是要下点功夫。

大家最早的印象是 Basic 本身就没有执行文件，它必须要有编译器执行程序，也由于每次都要重新编译，所以执行速度很慢，而 Visual Basic 虽然可以先编译成执行文件，但是也要有可执行的 DLL 文件才能执行，所以 Visual Basic 的执行文件虽然小，但是执行效率并不快。

现在 Microsoft 在编译成执行文件后，还会将执行文件使用 JIT(Just In Time)编译器来执行，如图 1.8 所示。

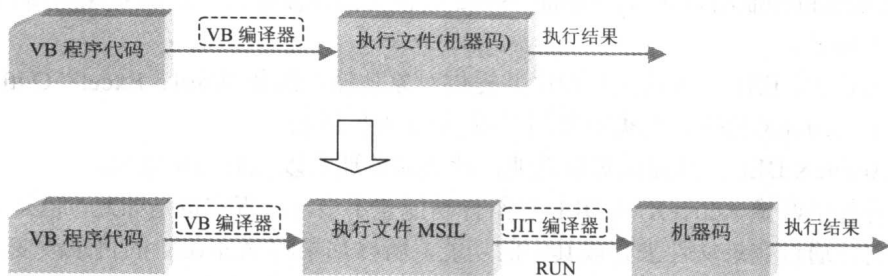


图 1.8

而 JIT 是新的构想，不过它和 Java 的 VM 似乎有点相像，因为有了这个编译器，使得在 .NET 开发出来的应用程序能够在任何平台上执行，而多了这个编译器，多少会对执行的速度上有点影响，不过也要看程序应用的范围，但这个影响是有限的。

通过 JIT 编译器，可以在任何计算机上执行开发出来的 .NET 程序，JIT 编译器就像是应用程序的通信协议一样，我们只要将程序做成执行文件。此外，通过 JIT 编译器，还可以让其他不同硬件或操作平台的用户来使用，如图 1.9 所示。

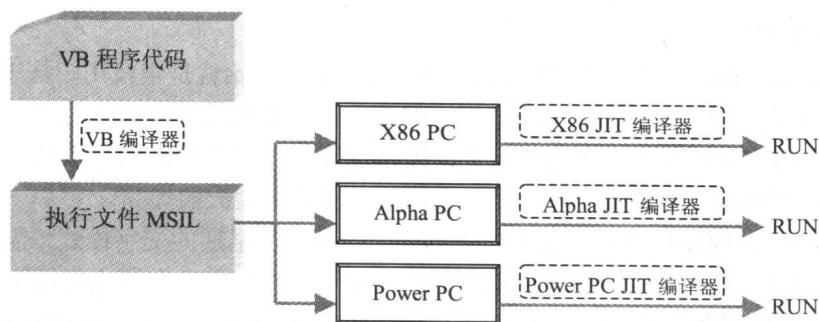


图 1.9

可以发现 Visual Basic 的程序代码可以通过不同的 JIT 编译器生成可以在各种平台上执行的程序，这样对于程序的延伸很有益处，不过这个感觉有点像 Java 的 VM(virtual machine)，不过这也代表这个方向是没有问题的。

### 1.2.4 不同编程语言的互通

设计 Visual Studio .NET 的目的就是要解决虽然是不同编程语言却可以互通共同程序代码的问题，因为大部分程序员都知道，有些程序设计的某一方面的软件非常好用，但在另一方面可能就不太好用，如果能将每个编程语言的特性结合起来，让每种语言都发挥自己特有的特性，那对于软件开发将很有成效，如图 1.10 所示。

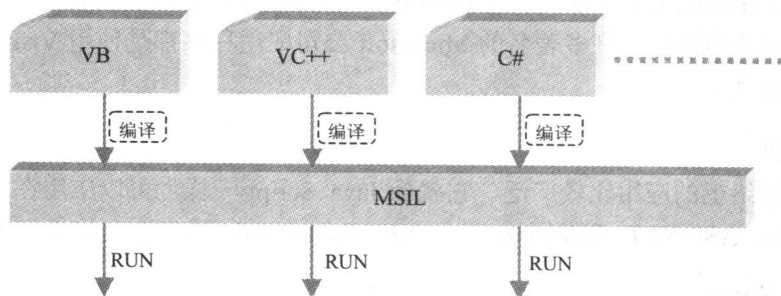


图 1.10

虽然在 Visual Studio .NET 内的编程语言是可以互通的，但是这不代表可以和 .NET 以外的语言相通。程序要能共通，还是有相当的困难，因为毕竟是由不同的公司开发出来的，如果能让所有的语言都能在共通的平台开发，那将会是编程语言的另一个新纪元。

## 1.3 .NET 简介

### 1. Visual Basic

Visual Basic 是从 Basic 就一直流行至今的语言，让许多人第一次写程序就能马上学会的语言，现在在 Microsoft 公司的完善下，仍在继续壮大。而借着 .NET 的强大功能，其影