



Visual Basic 和 .NET 应用开发

Application Development Using
Visual Basic and .NET

Robert J. Oberg
[美] Peter Thorsteinson 著
Dana L. Wyatt

潘 英 陈 君 等译



电子工业出版社
Publishing House of Electronics Industry
<http://www.phei.com.cn>

Visual Basic 和 .NET 应用开发

Application Development Using Visual Basic and .NET

Robert J. Oberg

[美] Peter Thorsteinson 著

Dana L. Wyatt

潘 英 陈 君 等译

电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书是 Visual Basic.NET 中、高级开发人员的优秀参考书，共分为六部分：第一部分介绍 Visual Basic.NET 的综述内容，回答什么是微软的 .NET，并给出了 .NET 框架的编程模型；第二部分介绍 VB.NET 编程语言，讲述 VB.NET 在数据类型方面的变化和新的面向对象的功能，还分析了 VB.NET 和 .NET 框架之间交互的问题。第三部分介绍 Windows 窗体的基础知识。第四部分介绍 .NET 框架中重要的基础性内容，讨论了程序集、部署和 .NET 框架类，并介绍了元数据、序列化、线程、属性、异步编程、远程化以及内存管理等内容。第五部分介绍用 VB.NET 进行 UI 编程的高级内容。第六部分介绍 .NET 框架中几个重要的部分，如 ADO.NET, ASP.NET, SOAP 和网络服务及安全问题，.NET 与以前的 COM 和 Win32 应用程序的交互性问题等。

本书可作为 VB.NET 开发人员和相关专业学生的学习指南和参考手册。

Authorized translation from the English language edition, entitled Application Development Using Visual Basic and .NET, ISBN: 0130933821 by Robert J. Oberg, Peter Thorsteinson, Dana L. Wyatt, published by Pearson Education, Inc, publishing as Prentice Hall PTR, Copyright © 2003.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

Simplified Chinese language edition published by Publishing House of Electronics Industry, Copyright © 2003.

This edition is authorized for sale only in the People's Republic of China excluding Hong Kong, Macau and Taiwan.

本书中文简体专有翻译出版版权由 Pearson 教育集团所属的 Prentice Hall PTR 授予电子工业出版社。其原文版权及中文翻译出版版权受法律保护。未经许可，不得以任何形式或手段复制或抄袭本书内容。

此版本仅限在中华人民共和国境内（不包括香港、澳门特别行政区以及台湾地区）发行与销售。

版权贸易合同登记号：图字：01-2003-0369

图书在版编目 (CIP) 数据

Visual Basic 和 .NET 应用开发 / (美) 奥伯格 (Oberg, R. J.) 等著; 潘英等译.

-北京: 电子工业出版社, 2003.10

书名原文: Application Development Using Visual Basic and .NET

ISBN 7-5053-9186-0

I. V… II. ①奥… ②潘… III. BASIC 语言 - 程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2003) 第 085227 号

责任编辑: 赵宏英

印刷者: 北京东光印刷厂

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

北京市海淀区万寿路 173 信箱 邮编: 100036

经 销: 各地新华书店

开 本: 787 × 1092 1/16 印张: 38.5 字数: 986 千字

版 次: 2003 年 10 月第 1 版 2003 年 10 月第 1 次印刷

定 价: 59.00 元

凡购买电子工业出版社的图书, 如有缺损问题, 请向购买书店调换; 若书店售缺, 请与本社发行部联系。

联系电话: (010) 68279077。质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

译者序

全球最大的软件公司微软大力推出了 .NET 系列,这不仅是该公司的战略重点,而且也代表了未来 Internet 技术的发展方向。

为了顺应市场的需求,各家出版社也在鼎力推出与 .NET 有关的各类书籍。现在,市面上这类书比比皆是,但大多集中在基本概念的叙述上,是一些入门的参考教程,而针对实际应用,尤其针对项目开发的书很少,而这又的确是中、高级开发人员最为需要的。

Visual Basic 一直被当做 Windows 环境应用软件的一个快速开发工具。它的易用性推进了 Windows 编程的发展,一系列成功的 Visual Basic 版本使它不断具有更加强大的功能。但是,很多人只是用 Visual Basic 来学习编程,而不依靠它做任何事,还是主要使用例如 C++ 这样的底层编程语言作为开发工具。但是,微软的 .NET 给软件的开发带来了一个巨大的变革,强大的通用语言运行时和丰富的 .NET 框架的类库为所有 .NET 编程语言提供了一个兼容的软件平台,使得 Visual Basic .NET 拥有同其他编程语言相同的功能,可以同刚推出的 Visual C# 以及曾经是主流的 Visual C++ 抗衡,而且还保留着 Visual Basic 最基本的易用性特点和强大的 RAD 功能,这些优势将使 Visual Basic 成为广泛使用的开发工具。通用语言规范和通用类型系统使得不同 .NET 编程语言间的协同几乎是无缝的,而其中的每一种语言的功能都可以足够强大,对于大多数项目,都不必使用混合语言编程,除非是原有的代码或编程习惯等因素迫使你这样做。所以,为什么不用易用性很好的 Visual Basic .NET 进行开发呢?

本书是面向有经验的程序员的,可以帮助开发人员快速了解如何使用 VB .NET,并继续深入理解 .NET 的框架。本书使用的例子是贯穿全书的综合实例,步骤详细,分析透彻,对于开发项目的专业工作很实用。

通过本书的学习,读者将能够轻松学会 Visual Basic .NET 开发项目的过程和技巧。Visual Basic 是 .NET 框架提供的最流行、最易学的语言之一,通用语言规范和通用类型系统使得不同 .NET 编程语言间无缝地协同工作,使得使用 Visual Basic .NET 开发项目成为一种流行。本书着眼于有 Visual Basic 早期版本编程知识的读者,假定读者是有经验的应用程序开发人员,有使用 Visual Basic 任意一种早期版本开发应用程序的知识。本书将完成读者从用 VB 学习编程到用 VB .NET 进行开发的过渡。

本书的翻译人员均为从事实际软件开发和应用的专业人员,主要由潘英、陈君翻译,陈河南审校。参与本书翻译工作的还有:赵晓坦、贺军、贺民、龚亚平、王学农、樊鹏、陈安华、谢高联、戴文军、李志云、王朝阳、杨敏、王宝成、毕子让、王春桥、胡新、李丽、董英材、王雷、蒋方帅等,在此一并表示感谢。

前 言

多年来,微软的 Visual Basic 一直被用做 Windows 应用软件基本的快速开发工具。它的易用性推进了 Windows 编程的发展,而其一系列成功的 Visual Basic 版本使它不断具有更加强大的功能。VB6 可以用来实现复杂的多层应用,这需要用到 COM 和 COM + 组件以及传统的 Windows 桌面程序。但是,不可能依靠 Visual Basic 来做每一件事,还需要使用例如 C++ 这样较低级的编程语言来完成某些工作,尤其在很多项目中,需要使用多种语言的混合编程,但这会导致在多种环境下连接的复杂性。

微软的 .NET 给软件开发带来了一个巨大的变革。强大的通用语言运行时 (Common Language Runtime) 和丰富的 .NET 框架类库为所有 .NET 编程语言提供了一个兼容的软件平台。实际上 Visual Basic 目前拥有同其他编程语言相同的功能,包括刚推出的 Visual C# 以及曾经是主流的 Visual C++, 而且同时 Visual Basic 还保留着最基本的易用性特点和强大的 RAD 功能,这些优势都使它更加流行。通用语言规范 (Common Language Specification) 和通用类型系统 (Common Type System) 使得不同 .NET 编程语言间的互操作几乎是无缝的,而其中每一种语言的功能都足够强大,对于大多数项目,无须使用混合语言编程,除非是原有的代码或你的编程习惯等因素迫使你这样做。

现在, Visual Basic 本身发生了本质的变化,未经修改的 VB6 代码不能在 .NET 环境中运行,而且新的编程语言 Visual Basic .NET (或简称为 VB.NET) 是一种真正的面向对象的语言,具有诸如接口、继承、多态等特性。所以说,从 VB6 转到 VB.NET 确实需要有一段学习过程,而且学习编程语言只是挑战的一部分,主要的挑战是学习 .NET 框架及其所有功能,包括 Windows Forms, ADO.NET, ASP.NET Web 窗体和 Web 服务。

本书是为有经验的程序员编写的,是为了帮助其快速了解如何使用 VB.NET 并继续深入理解 .NET 的框架。本书对于专业人员很实用,很多章节后面都配有例子和实际的案例分析,目的是使读者能够利用 Visual Basic .NET 和 .NET 框架开发良好的应用软件。

本书是 Object Innovations 和 Prentice Hall PTR 联合出版的 .NET 丛书之一,丛书的其他书对 VB.NET 给出了更加基本的介绍,讨论了如何移植到 VB.NET 的问题,并涉及到其他重要的 .NET 编程语言和框架中的内容。本书从内容和结构来说非常接近另外两本:“Application Development Using C# and .NET” 和 “.NET Architecture and Programming Using Visual C++”。本书和这两本书主要的区别在于,除了是使用 VB.NET 作为编程语言外,还大量地涉及图形用户界面 (GUI) 编程,包括有关 Windows 窗体的三章内容和关于 GDI+ 的一章内容。

本书结构

本书分成六部分,以便于展示所需要学习的内容。第一部分包括开始的两章,给出了每个人都需要了解的综述性内容,并回答了一个基本的问题:什么是 Microsoft .NET? 另外还给出了 .NET 框架的编程模型。

第二部分包括第 3 章到第 6 章,主要介绍了 VB.NET 编程语言。即使你了解原来的 Visual

Basic, 仍然需要阅读这一部分, 注意其中在数据类型方面的变化(第 3 章)和新的面向对象的功能(第 5 章)。第 6 章是关于重要的 VB.NET 和 .NET 框架之间交互的问题。顶点旅行社(Acme Travel Agency)案例从第 5 章开始引入, 并贯穿本书的始终。

第三部分包括第 7 章和第 8 章, 介绍了 Windows 窗体的基础知识。Windows 窗体是一套 .NET 框架中的类, 这些类用于描述图形用户界面, 熟悉以前 Visual Basic 版本的程序员应该注意到, 它们显著地改变了编程模型, 同时也带来了以前版本中所没有的灵活性。

第四部分包括第 9 章和第 10 章, 介绍了 .NET 框架中重要的基础性内容。第 9 章讨论了程序集和部署, 它们使开发 Windows 应用程序的简易性和健壮性更加先进, 并终结了臭名昭著的“DLL 地狱”。第 10 章引入了重要的 .NET 框架类, 并讨论了元数据、序列化、线程、属性、异步编程、远程化以及内存管理等内容。

第五部分包括第 11 章和第 12 章, 介绍了其他用 VB.NET 进行 UI 编程的高级内容。第 11 章介绍 GDI+, 它提供了一种图形编程模型, 这种模型比原来传统 Windows 的 GDI 模型功能更加强大并且更加简单, GDI+ 也完全可以通过 Visual Basic 来访问。第 12 章引入了几个重要的内容, 比如虚拟继承、MDI 以及在 .NET 应用程序中 ActiveX 控件的应用等。

第六部分包括第 13 章到第 17 章, 介绍了 .NET 框架中几个重要的部分, 这些内容主要用于创建各种不同的应用程序。第 13 章是关于 ADO.NET 的, 它提供了一些类用于访问关系数据和 XML 数据。第 14 章介绍了 ASP.NET 的基础概念, 包括用于开发网站的 Web 表单的使用。第 15 章介绍了 SOAP 和 Web 服务, 这些给异类系统提供了一个易用和健壮的交互机制。第 16 章详细讨论了安全的问题, 包括代码访问安全(Code Access Security)和声明安全。第 17 章介绍了 .NET 与以前的 COM 和 Win32 应用程序的交互性问题。

附录介绍了 Visual Studio.NET 以及 .NET 提供的用于调试和跟踪的类。

示例程序

真正掌握一个主要框架的惟一途径就是读写大量具有合理长度的程序。本书提供了很多小程序分别用来说明 .NET 的各种相关的功能, 从而使得这些相关功能更加容易理解。

一个主要的案例分析(即顶点旅行社的案例)从第 5 章开始, 并在对各章的学习中不断地得到完善。它解释了在 VB.NET 和 .NET 框架中使用的很多功能, 这些功能在实际中有着广泛应用。

示例程序可以在本书网站提供的的一个自解压文件中得到, 解压缩时, 将建立一个默认名为 c:\OI\NetVb 的目录。示例程序从第 2 章开始, 依次为子目录 Chap02, Chap03, 等等, 目录的名字与所在章同名。

作为 .NET 系列丛书的一部分, 本书中的示例程序与其他书中的示例程序有一定的联系, 这一系列中每本书的示例程序在 \OI 目录下都有各自的子目录, 所以当安装时, 书中的所有 .NET 示例程序都将被安装到一个共同的目录中。

这些程序只用于教学目的而不能应用于任何商业软件产品, 它们没有得到任何形式的授权。

网址

本套丛书的网址为 www.objectinnovations.com/dotnet.htm。

在该网址上提供了一个用于下载本书示例程序的链接,有关.NET技术的进一步信息可以在以下网址得到:www.mantasoft.com/dotnet.htm。本书的示例程序在这个网址也可以找到。

本书的这个网址中还有一些关于.NET技术的其他学习资源,并将不断更新。

目 录

第一部分 .NET 体系结构综述

第 1 章 什么是 Microsoft .NET	2
1.1 微软和 Web	2
1.2 桌面上的 Windows	4
1.3 全新的编程平台	5
1.4 XML 的角色	7
1.5 VB.NET 和 .NET 框架	8
1.6 小结	8
第 2 章 .NET 基础	10
2.1 Windows 开发中的问题	10
2.2 未来的应用程序	10
2.3 .NET 初步	10
2.4 小结	18

第二部分 VB.NET 语言

第 3 章 VB.NET 基础 (I)	20
3.1 Visual Studio.NET 和控制台应用程序	20
3.2 VB.NET 中的 Hello World	24
3.3 在 VB.NET 中进行计算	28
3.4 类	37
3.5 VB.NET 类型系统	48
3.6 小结	52
第 4 章 VB.NET 基础 (II)	54
4.1 字符串	54
4.2 数组和参数化的属性	62
4.3 关于方法的更多内容	68
4.4 可选择的参数	72
4.5 VB.NET 应用函数	73
4.6 命令行参数	79
4.7 小结	80
第 5 章 VB.NET 中的继承和异常	81
5.1 面向对象的概念	81
5.2 顶点旅行社案例:设计	84
5.3 VB.NET 中的继承	90

5.4	访问控制	92
5.5	顶点旅行社案例:实现	93
5.6	关于继承的更多内容	99
5.7	异常	108
5.8	小结	117
第 6 章	.NET 框架中的 VB.NET	118
6.1	System.Object	118
6.2	集合	122
6.3	接口	125
6.4	解决接口的多义性	132
6.5	顶点旅行社案例:第二步	134
6.6	.NET 中的通用接口	137
6.7	委托	148
6.8	事件	155
6.9	属性	162
6.10	小结	165

第三部分 Windows 窗体基础

第 7 章	Windows 窗体	168
7.1	Windows 窗体的层次结构	168
7.2	使用 .NET SDK 建立 Windows 应用程序	168
7.3	Windows 窗体事件处理	172
7.4	Visual Studio .NET 和窗体	177
7.5	VS.NET Windows 应用程序的“黑匣子”	183
7.6	对话框	185
7.7	控件	194
7.8	复选框的使用	196
7.9	使用列表控件(列表框和组合框)	198
7.10	小结	203
第 8 章	控件的使用	204
8.1	菜单	204
8.2	工具栏	210
8.3	状态栏	214
8.4	日历控件	216
8.5	范围控件	219
8.6	列表控件	225
8.7	DataGrid 控件	235
8.8	TreeView 控件	238
8.9	标准对话框控件	243

8.10 小结	245
---------------	-----

第四部分 .NET 框架基础

第 9 章 程序集和部署	248
9.1 组件	248
9.2 程序集	253
9.3 私有程序集部署	265
9.4 共享程序集部署	266
9.5 程序集配置	268
9.6 多模块程序集	272
9.7 安装和部署工程	274
9.8 小结	277
第 10 章 .NET 框架类	278
10.1 元数据和反射	278
10.2 .NET 的输入和输出	283
10.3 序列化	288
10.4 .NET 应用程序模型	292
10.5 上下文	303
10.6 应用程序隔离	305
10.7 异步编程	309
10.8 远程化	315
10.9 自定义属性	320
10.10 无用单元收集和终止	323
10.11 小结	330

第五部分 使用 VB.NET 的高级用户接口编程技术

第 11 章 GDI+ 简介	332
11.1 与设备无关的图形和抽象	332
11.2 GDI+ 体系结构	333
11.3 GDI+ 结构	337
11.4 笔和笔刷	342
11.5 一个绘图程序	348
11.6 裁剪矩形	352
11.7 位图和元文件	356
11.8 GDI+ 内存管理	361
11.9 文本和字体	362
11.10 滚动	364
11.11 小结	365

第 12 章 高级 Windows 窗体	366
12.1 无模式对话框	366
12.2 MDI 应用程序	368
12.3 形象继承	375
12.4 ActiveX 控件	385
12.5 小结	389

第六部分 .NET 应用程序

第 13 章 用 ADO.NET 编程	392
13.1 Visual Studio .NET 数据库试验台	393
13.2 ADO.NET 体系结构	397
13.3 连接式数据访问	399
13.4 参数和存储过程	405
13.5 SqlDataAdapter 和 DataSet 类	407
13.6 DataSet 基础	409
13.7 数据库事务处理和更新	412
13.8 乐观和悲观锁定以及数据集	413
13.9 使用 DataSets	414
13.10 顶点旅行社个案研究	427
13.11 XML 数据访问	428
13.12 AirlineBrokers 数据库	429
13.13 带有关系的概要	435
13.14 类型化 DataSet	436
13.15 小结	439
第 14 章 ASP.NET 和 Web 表单	440
14.1 什么是 ASP.NET	440
14.2 Web 表单体系结构	444
14.3 请求/响应编程	453
14.4 使用 Visual Studio .NET 开发 Web 应用程序	459
14.5 顶点旅行社个案研究	468
14.6 ASP.NET 应用程序	473
14.7 ASP.NET 应用程序中的状态	475
14.8 ASP.NET 配置	479
14.9 服务器控件	482
14.10 ASP.NET 中的数据库存取	488
14.11 小结	491
第 15 章 Web 服务	493
15.1 协议	493
15.2 Web 服务体系结构	496

15.3	SOAP 差别	506
15.4	WebService 类和 Visual Studio	510
15.5	Hotel Broker Web 服务(案例研究)	518
15.6	小结	521
第 16 章	安全措施	522
16.1	基于用户的安全	522
16.2	代码访问安全	523
16.3	Internet 安全	524
16.4	.NET 中基于角色的安全	525
16.5	基于表单的鉴别	542
16.6	代码访问权限	544
16.7	代码身份	554
16.8	安全策略	556
16.9	小结	565
第 17 章	互操作性	566
17.1	从托管代码调用 COM 组件	566
17.2	从 COM 客户内调用托管组件	576
17.3	平台调用服务(PInvoke)	580
17.4	小结	582
附录 A	Visual Studio .NET	583
A.1	Visual Studio .NET 概述	583
A.2	创建控制台应用程序	586
A.3	项目配置	590
A.4	调试	591
A.5	小结	597
附录 B	.NET 中的跟踪和调试	599
B.1	TraceDemo 示例	599
B.2	启动调试和跟踪输出	600
B.3	使用 Debug 和 Trace 类	600
B.4	使用开关来启动诊断	601
B.5	启用或者禁用开关	601
B.6	TraceListener	602
B.7	Listeners 集合	603
B.8	小结	603

第一部分

.NET 体系结构综述

第一部分包括开始的两章,给出了一个所有人都需要了解的综述。它主要解释了一个基本的问题:什么是 Microsoft .NET? 这一部分勾画出了 .NET 框架的编程模型。这些信息在学习 Visual Basic .NET 时有助于理解 .NET 的环境以及本书其他部分介绍的 .NET 框架内容。

第 1 章 什么是 Microsoft .NET

Visual Basic .NET(简称 VB .NET)代表了自从最早的 1.0 版本以来 Visual Basic 编程语言最大的转变。VB .NET 最大的特点就在于它能够良好地集成进 .NET 框架,它是 Internet 时代微软应用程序的版本。.NET 提供了基于 Internet 标准的增强的互操作性,提高了经典 Windows 应用程序的健壮性,为开发人员提供了一个全新的编程平台和有力的开发工具,同 XML 一起发挥着重要的作用。

微软的 .NET 是一个构建在操作系统之上的开发平台。在正式发布 .NET 之前的三年中,微软为此投入了很多力量,而其他先进的技术(如 XML, Java 和 COM)对 .NET 也有一定的影响。

微软的 .NET 提供如下内容:

- 通用语言运行时(Common Language Runtime):这是一个健壮的运行时平台。
- 多语言开发:没有一种语言会比其他语言更好。
- .NET 框架:这是一个可扩展的编程模型,它提供了非常丰富并且代码可以被其他 .NET 语言复用的类库。
- 支持构建在 Internet 标准之上的网络结构,使得应用程序之间可以实现高层次的通信。
- 支持 Web 服务的新行业标准:Web 服务代表了一种新的应用程序发送机制,它拓展了对 Internet 的基于组件开发的概念。
- ASP .NET:这允许使用标准化编程习惯来开发网络应用程序。
- 一个部署模型:用来实现版本控制并可以避免所谓的“DLL 地狱”。
- 一个安全模型:程序员可以很容易地用在自己的程序中。
- 一个互操作性机制:这允许 .NET 程序可以访问原有的代码,包括访问 COM 组件。
- 强大的开发工具。

在本章,我们将了解到 .NET 的总体情况,然后来看一看 VB .NET 是如何融入其中的。

1.1 微软和 Web

万维网(World Wide Web)已经给微软带来了巨大的挑战,而事实上,Web 与微软的传统领域 PC 共同存在得很融洽:通过一个 PC 应用程序,即浏览器,每个人都可以获取到整个世界的信息。

Web 依赖于 HTML,HTTP 和 XML 之类的标准,这些标准对于在具有各种计算机系统和设备的形形色色用户间进行通信是十分关键的。

Windows PC 尽管很复杂,但是非常标准化。而 Web 虽然基于各种标准的协议,但面临着许多复杂的问题,如涉及多种语言、数据库、开发环境以及在这些协议之上运行的各种设备。这种技术上巨大的复杂性更加剧了对能够使用新技术来构造所需系统的人才的持续需求,而 .NET 提供了整套的结构,程序员只需要在他们的应用程序中添加数值,无须重新建立通用编

程问题的解决方案。

1.1.1 Internet 时代的应用程序

从本质上说,Web 是一个巨大的信息仓库,浏览器可以对带有信息的页面提出请求,Web 服务器会将这些信息以静态 HTML 页面的形式发送。即使当交互式的 Web 应用程序被引入以后,HTML 仍然因为包含有详细的用于浏览的版本信息而得到继续应用。

XML 提供了一种标准的方式来传输数据,而与数据的格式无关,因此,XML 可以帮助公司解决文档和信息流的标准化,比如建立标准的订单或发货单等,由此电子商务可以在各公司间实现自动化(即 B2B)。但是 XML 只描述数据,并不对数据执行任何动作。所以,我们还需要有 Web 服务。

1.1.2 Web 服务

.NET 最重要的特性之一就是支持 Web 服务。基于工业标准 SOAP 协议,Web 服务允许通过 Internet 来传递应用程序的功能。从一个 .NET 程序员的角度看,Web 服务和其他任何通过 .NET 语言的类来实现的服务没有什么区别,这是因为调用函数时编程模型是一样的,无论是在一个应用程序中、在同一台机器上的不同组件中、还是不同机器上的一个 Web 服务中都是这样。

这种固有的简易性使得各个公司可以非常容易地创建自己的宿主应用程序。如果愿意的话,完全可以购买一个完整的应用程序,不必亲自开发、部署和维护。或者也可以使用在开发应用程序时还不存在的第三方 Web 服务。

1.1.3 ASP.NET

.NET 包括一个完全重新开发的流行 Active Server Pages 技术的版本,即 ASP.NET。虽然 ASP 依赖于在具有一定页面格式化命令的编程语言中已解释的脚本代码,但是 ASP.NET 的代码可使用任何 .NET 语言来编写,如 C#, VB.NET, JScript 及具有托管扩展的 C++。因为 ASP.NET 代码经过了编译,因此可以将接口代码放在一个单独的“代码隐藏”文件中,以便和商业逻辑分离开。虽然 C#, VB.NET, JScript 可以作为网页中嵌入的脚本,但托管的 C++ 必须置于代码隐藏文件中。

ASP.NET 提供了 Web 表单,这极大地简化了创建 Web 用户界面的工作。

Visual Studio.NET 的拖放功能使布局表单非常容易,可以给诸如单击按钮之类的表单事件添加代码。

ASP.NET 可以自动探测浏览器的功能。对于高端浏览器来说,代码可以在客户机上运行,而对于低端的浏览器,是由服务器来完成处理并生成标准的 HTML。ASP.NET 完成的所有这些工作对于开发者来说都是透明的。

Web 服务与经过编译的成熟的编程语言(例如 C#、VB.NET 及托管 C++)结合在一起,就可以使得 Web 编程遵循面向对象的编程模型,而这对于 ASP 脚本语言和 COM 组件来说是不可能的。

1.1.4 开放的标准和互操作性

现代化的计算环境一般包括规模庞大的硬件和软件系统。计算机的范围涉及大型机和高档服务器、工作站和 PC 机以及个人数字助理(PDA)和蜂窝电话等小型移动设备。形形色色的操作系统包括有传统的大型机系统、各种版本的 UNIX, Linux 和 Windows, 实时系统以及一些特殊用途的系统, 比如用于移动设备的 PalmOS。许多不同的编程语言、数据库、应用程序开发工具及各种中间设备都得到了应用。

在现代的环境中, 很少有应用程序作为孤岛存在, 即使是部署在一台单机上的应用软件, 也需要通过 Internet 注册和升级。应用程序间互操作性的关键是使用标准, 因为应用程序主要运行在网络上, 所以主要的标准是所应用的通信协议。

1.1.5 通信协议

TCP/IP 套接字被高度地标准化并得到了广泛的应用, 但是, 程序员必须掌握太多的细节才能高效率地写出可靠的分布式应用程序。更高的标准是远程过程调用(RPC), 但是 RPC 也非常复杂, 而且版本众多。现在流行的是一些更高层次的协议, 比如 CORBA, RMI 和 DCOM, 但是在每个终端还具有复杂性, 并且需要符合特殊要求的环境。这些协议又都有一些缺点, 如通过防火墙时的困难性等。

有一种通信协议已经得到了普遍应用, 这就是 HTTP。因此, Microsoft, IBM 以及其他供应商引入了一种新的协议 SOAP (Simple Object Access Protocol, 简单对象访问协议)。SOAP 使用基于文本的 XML 根据对象请求和当时的数据来进行编码。而 SOAP 最大的优点就是它的简单性, 从而在多种设备上实现起来同样简单。而且 SOAP 可以运行在任何协议之上, 特别是它可以运行在像 HTTP 这样的标准 Internet 协议之上, 这使它可以没有任何连接问题而穿过防火墙。

1.2 桌面上的 Windows

微软开始于桌面。现在的 Windows 环境已经得到了普遍应用, 由于有了无数的应用程序, 大多数计算机用户可以在家中使用 Windows。虽然微软在改进 Windows 的过程中取得了很大的成果, 但是还有很多问题需要解决。

1.2.1 Windows 的问题

维护一台装有 Windows 的 PC 机会带来很多麻烦事, 因为应用程序太复杂了, 包括有众多的文件、注册表项、快捷方式等。不同的应用程序可能会共用一些 DLL 文件, 而安装一个新的应用程序可能会覆盖一个其他应用程序需要的 DLL 文件, 这也许会破坏一个以前安装的应用程序(即所谓的 DLL 地狱), 删除一个应用程序也有着复杂的操作过程, 而且常常并不能比较理想地删除掉。

一台 PC 机可能会变得越来越不稳定, 经常需要重新格式化硬盘和进行划痕的恢复。但是使用 PC 机有很大的经济优势, 因为标准的应用程序并不昂贵, 而且又功能强大, 所需要的硬件也很便宜, 不过维修的开销使这种经济上的优势打了折扣。

刚开始开发 Windows 的时候,个人计算机还没有连接在一个网络上,那时候安全问题也就不算什么问题。但当安全机制被构建在 Windows NT 和 Windows 2000 中后,编程模型就很难使用了。

1.2.2 玻璃工厂和稀少客户

原来的由中央计算机的“玻璃工厂”模式来控制所有应用程序只是一种要求,也曾有过向“稀少客户”靠拢的意愿。但是曾经一度被炒作得沸沸扬扬的“网络 PC”从来也没有真正流行过。毕竟标准的 PC 机应用程序还有很高的价值,用户喜欢使用他们自己的 PC 机,他们的数据存储在本地上计算机上不但安全而且方便。一旦没有了宽带连接,一个基于服务器的应用程序(比如一个字处理程序)运行得不会很好,同时安全对于稀少客户来说也是一个很严重的问题。因此,个人计算机毫无疑问会继续存在下去。

1.2.3 可靠的 Windows

关于 .NET 和 Internet 有着各种评价,但是有一点需要澄清的是 .NET 改变了编程模型,使得程序员能够开发出更可靠的 Windows 应用程序。应用程序不再需要在脆弱的 Windows 注册表中储存过多的配置信息,自我描述的 .NET 应用程序在其可执行文件中包含了元数据,不同版本的应用程序或组件可以并排部署,应用程序也可以通过全局集合缓存(Global Assembly Cache)共用组件,而且在部署模型中加入了版本控制。同时,一个简单易懂的安全模型已经成为 .NET 的组成部分,Windows Forms 技术更是建立 Windows GUI 应用程序的全新范例。

1.3 全新的编程平台

下面看一下刚刚讨论过的 .NET 作为一个全新的编程平台有什么特点:

- 代码验证可以防止未经授权的行为发生。
- 比 Win32 API 和 COM 更易于编程。
- 开发平台的全部和部分可以应用在各种不同类别的计算机上(在这点上和 Java 一样)。
- 所有的编程语言使用同一个类库。
- 各种编程语言具有互操作性。

另外 .NET 开发平台还有一些重要特性:

- .NET 框架
- 通用语言运行时
- 多种语言开发
- 开发工具

1.3.1 .NET 框架

现代的编程很大程度上依靠程序库中提供的可复用源代码。面向对象的语言方便了类库的建立,类库不但使用起来很灵活,在一定程度上还有较好的抽象性,而且可以通过增加新类和从原有的类中产生新类的方式扩展来“继承”现有的功能。