

# *Microsoft* .NET

## 最佳实践

陆永宁 李烨等 译

麦中凡 审校

北京航空航天大学出版社

## 内容简介

Microsoft .NET 是微软公司推出的新一代软件开发环境,旨在支持下一代 Web 服务的软件开发。本书是对 .NET 包括的各部分的原理及主要工具 Visual Studio .NET 的入门向导。读者对象是接受 .NET 相关培训的人员以及第一线的软件开发人员,特别是想转向 .NET 环境上开发软件的人员。

本书分为九个模块。基本每个模块都有相应的演示和实践。主要内容为:.NET 和 .NET 框架的原理及基本思想;C# 语言的基础语法结构和相关技术,为读者学习 C# 语言提供了一些入门指导;XML 语言的基本原理、XML 文档的生成方法和相关技术。.NET 框架中重要的组成部分 ASP.NET;Web 控件的原理及应用;如何使用工具 ADO.NET 访问数据;用 ASP.NET 生成一个 Web 应用;对 Web 服务原理及创建 Web 服务的讲解。

本书对于希望了解下一代网络计算技术的高校师生和高层软件工作者也是一本好教材。

## 图书在版编目(CIP)数据

Microsoft .NET 最佳实践/陆永宁等译. —北京:

北京航空航天大学出版社,2001. 10

ISBN 7-81077-110-8

I. M... II. 陆... III. 网站 程序设计

IV. TP393.092

中国版本图书馆 CIP 数据核字(2001)第 066610 号

本书版权归 Microsoft 公司所有。Microsoft Press 为 Microsoft 公司的下属出版社。本书简体字中文版由 Microsoft Press 授权北京航空航天大学出版社独家出版。未经出版者书面允许,不得用任何手段复制或抄袭本书部分内容或全部内容。

版权号 图字:01-2001-3890 号

## Microsoft .NET 最佳实践

陆永宁 李 焱等 译

麦中凡 审校

责任编辑 曾昭奇

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100083) 发行部电话:82317024 传真:82328026

<http://www.buaapress.com.cn>

E-mail:pressell@publica.bj.cninfo.net

北京宏文印刷厂印装 各地书店经销

开本:787×1092 1/16 印张:20 字数:512 千字

2001 年 9 月第 1 版 2001 年 9 月第 1 次印刷 印数:5000 册

ISBN 7-81077-110-8/TP·059 定价:36.00 元

# 译者前言

当今,计算机日新月异地发展同时对自身提出了很多挑战:对开发者而言,在开发上要选择语言和操作系统,交付上要选择合适的方式,稳定性问题包括操作系统不稳定、DLL 兼容等,维护上的用户参与和各种媒介;对用户而言,部署上需要时间、人员、各种兼容测试和许可管理,稳定性和安全性上需要测试防止无意破坏,维护上的部署需要安全问题;对应用和服务而言,当前的服务都是作为应用级的服务,而且 Internet 服务是通过门户交付、不可随身携带、浏览器和卖主及设备相互依赖,需要交互式访问。所有这些都是我们正在或者将要面临的问题。这中间有个业界的巨大转变:从分布式客户/服务器计算转向“软件就是服务”的网络计算。后者把资源共享、软件重用又提升到一个新阶段,租赁、下载在 Internet 上进行。在 ASP (应用服务提供者)提供的各种各样的服务基础上,业务系统的应用开发者几乎不用编制、维护(升级、纠错)软件就可以快速交付十分复杂的系统。而 ASP 用 C# 语言开发的,Web 服务也是大量地使用其他 ASP 提供的服务(当然也包括以 Java,甚至 C++ 提供的服务构件)。新一代的网络计算对我们提出了挑战,而微软公司提供的 Microsoft .NET 正是为完善新一代网络计算的软件平台环境。

Microsoft .NET 的前身是下一代 Windows 服务(NGWS),始于 1998 年。它主要基于包含实际语言和执行平台的 .NET Framework,以及提供了丰富内建功能的基本类库。Microsoft .NET 是把 Internet 转换成一个分布式计算平台的技术。Microsoft .NET 蕴含的基本理念就是将注意力从单独的 Web 站点和与 Internet 相连的各种设备,转移到由各种计算机、设备和服务协同工作的架构之上,从而提供涉及面更广、功能更全面的解决方案。Microsoft.NET 的设计理念即在于允许在合作解决方案架构内集成或协调 Internet 上的任意资源组。它为创建分布式 Web 应用提供了所有的工具和技术,在应用的各个层次上提供了一个语言中立的统一编程模型,同时对已有的技术进行了无缝的集成。

微软在开发推行 .NET 平台时是非常稳健的,直到本书出版之时,.NET 平台还没有正式发布 1.0 版,网上可见的是 Beta1 和 Beta2 两个版本,原因是网络计算还在纵深发展,.NET 还在不断试探中完善。截止至本书译出时,Microsoft Visual Studio .NET Beta 2 刚刚推出,所以本书还是基于 Visual Studio .NET Beta 1 的,请读者注意。

为了适应新一代专业软件开发人员培训计划,教育部计算机课程教学指导委员会与微软中国有限公司协商进行新一代专业软件开发人员培训,本书正是在这种背景下出版的。本书的使命是全面介绍 .NET。全书共分为九个模块,分别包括:.NET 和 .NET 框架的原理及基本思想;C# 语言的基础语法结构和相关技术;XML 语言的基本原理;XML 文档的生成方法和相关技术;.NET 框架中重要的组成部分 ASP.NET;Web 控件的原理及应用;使用工具 ADO .NET 访问数据;用 ASP.NET 生成一个 Web 应用;Web 服务原理及创建 Web 服务。为了使

读者能够尽快地掌握有关知识,本书安排了非常丰富的演示和亲自实践的练习,读者在学习完本书并进行相应实践后,相信能够对.NET 有一个初步的认识。

本书在成书过程中,第 1 个模块由麦中凡翻译,第 2 个模块由陆永宁翻译,李烨翻译了第 3、4、9 个模块,樊平和张晓昉共同翻译了第 5、6、7、8 个模块,还有戴彩霞、尹瑞、潘红莲在翻译校对编排中作了大量的工作。最后的审校由麦中凡和陆永宁完成。

关于批评、指正、帮助的联系地址是:[oops@buaa.edu.cn](mailto:oops@buaa.edu.cn)

译 者

2001.8

# 目 录

## 1 模块 1: .NET 框架概述

|                                  |    |
|----------------------------------|----|
| 1.1 概 述 .....                    | 2  |
| 1.2 回 望 .....                    | 2  |
| 1.2.1 对开发组织的挑战 .....             | 3  |
| 1.2.2 对客户合作的挑战 .....             | 4  |
| 1.2.3 应用与服务 .....                | 5  |
| 1.3 .NET 框架概述 .....              | 6  |
| 1.3.1 什么是.NET .....              | 6  |
| 1.3.2 .NET 和.NET 框架 .....        | 7  |
| 1.3.3 .NET 框架图表 .....            | 7  |
| 1.3.4 通用语言运行时 .....              | 8  |
| 1.3.5 通用语言运行时图表 .....            | 9  |
| 1.3.6 更多的.NET 框架 .....           | 10 |
| 1.3.7 统一了不同的编程模型 .....           | 11 |
| 1.3.8 服 务 .....                  | 11 |
| 1.3.9 Web 服务 .....               | 12 |
| 1.3.10 什么是 Web 服务 .....          | 13 |
| 1.3.11 .NET 如何使 Web 服务成为可能 ..... | 14 |
| 1.3.12 什么是商业案例 .....             | 16 |
| 1.3.13 Orchestration(1) .....    | 17 |
| 1.3.14 Orchestration (2) .....   | 18 |
| 1.3.15 .NET 企业服务器 .....          | 18 |
| 1.3.16 BizTalk Server 2000 ..... | 19 |
| 1.3.17 SQL Server 2000 .....     | 20 |
| 1.3.18 Exchange 2000 服务器 .....   | 21 |
| 1.3.19 集成、安全与加速 .....            | 22 |
| 1.3.20 应用、服务和商务 .....            | 23 |
| 1.3.21 .NET 与.NET 企业服务器 .....    | 24 |
| 1.3.22 迁移路径——从任何地方到.NET .....    | 24 |
| 1.3.23 .NET 的主要优点(1) .....       | 25 |
| 1.3.24 .NET 的主要优点(2) .....       | 26 |
| 1.3.25 .NET 的主要优点(3) .....       | 27 |
| 1.4 小 结 .....                    | 28 |

## 2 模块 2: C# 概述

|                                             |    |
|---------------------------------------------|----|
| 2.1 概 述.....                                | 30 |
| 2.2 C# 程序的结构.....                           | 31 |
| 2.2.1 Hello, World .....                    | 31 |
| 2.2.2 类.....                                | 32 |
| 2.2.3 Main 方法 .....                         | 33 |
| 2.2.4 using 指令和 System 名字空间 .....           | 34 |
| 2.2.5 示例:使用 Visual Studio 创建 C# 程序 .....    | 35 |
| 2.3 基本输入/输出操作 .....                         | 36 |
| 2.3.1 Console 类 .....                       | 36 |
| 2.3.2 Write 和 WriteLine 方法 .....            | 37 |
| 2.3.3 Read 和 ReadLine 方法 .....              | 39 |
| 2.4 推荐练习.....                               | 40 |
| 2.4.1 注释应用程序.....                           | 40 |
| 2.4.2 生成 XML 文档 .....                       | 41 |
| 2.4.3 示例:生成和查看 XML 文档 .....                 | 42 |
| 2.4.4 异常处理.....                             | 43 |
| 2.5 编译、运行和调试 .....                          | 45 |
| 2.5.1 调用编译器.....                            | 45 |
| 2.5.2 运行应用程序.....                           | 47 |
| 2.5.3 示例:编译和运行一个 C# 程序 .....                | 48 |
| 2.5.4 调 试.....                              | 48 |
| 2.5.5 多媒体:使用 Visual Studio 的调试器 .....       | 50 |
| 2.5.6 SDK 工具 .....                          | 50 |
| 2.5.7 示例:使用 Ildasm .....                    | 52 |
| 2.6 实验:创建一个简单的 C# 程序 .....                  | 53 |
| 2.6.1 练习 1:创建一个简单的 C# 程序 .....              | 53 |
| 2.6.2 练习 2:从 Command Line 编译和运行 C# 程序 ..... | 54 |
| 2.6.3 练习 3:使用调试器 .....                      | 55 |
| 2.6.4 练习 4:为 C# 程序添加异常处理 .....              | 55 |
| 2.7 复 习.....                                | 58 |

## 3 模块 3: XML 文档概述

|                       |    |
|-----------------------|----|
| 3.1 概 述.....          | 60 |
| 3.2 结构化数据.....        | 61 |
| 3.2.1 结构化与非结构化数据..... | 62 |
| 3.2.2 信息处理的问题.....    | 63 |
| 3.2.3 搜索数据.....       | 64 |

|                                |    |
|--------------------------------|----|
| 3.2.4 对 XML 的需要 .....          | 65 |
| 3.2.5 用 XML 标记描述数据 .....       | 66 |
| 3.3 什么是 XML .....              | 67 |
| 3.3.1 XML 和 W3C 的历史 .....      | 67 |
| 3.3.2 XML 和 Web .....          | 68 |
| 3.3.3 XML 与 HTML 的比较 .....     | 69 |
| 3.3.4 XML 的优势和劣势 .....         | 70 |
| 3.3.5 XML 的应用 .....            | 72 |
| 3.4 创建 XML 文档 .....            | 73 |
| 3.4.1 XML 语法 .....             | 74 |
| 3.4.2 XML 文档的内容 .....          | 75 |
| 3.4.3 XML 文档的结构 .....          | 77 |
| 3.4.4 创建格式良好的 XML 文档 .....     | 78 |
| 3.4.5 XML 工具 .....             | 80 |
| 3.4.6 示例:创建并浏览 XML 文档 .....    | 84 |
| 3.4.7 实践:创建一个 XML 文档 .....     | 84 |
| 3.4.8 映射一个数据规则说明 .....         | 85 |
| 3.5 文法和名字空间 .....              | 86 |
| 3.5.1 通用 XML 文法 .....          | 87 |
| 3.5.2 创建自己的 XML 文法 .....       | 88 |
| 3.5.3 使用名字空间 .....             | 89 |
| 3.6 示例: LitWare Books 应用 ..... | 91 |
| 3.7 复 习 .....                  | 92 |

## 4 模块 4: XML 技术概述

|                                    |     |
|------------------------------------|-----|
| 4.1 概 述 .....                      | 95  |
| 4.2 XML 技术: 纵览 .....               | 96  |
| 4.2.1 介绍 XML 术语 .....              | 96  |
| 4.2.2 动画: 一个 XML 文档的生命周期 .....     | 99  |
| 4.3 使用样式表浏览 XML .....              | 100 |
| 4.3.1 介绍层叠样式表(CSS) .....           | 100 |
| 4.3.2 和 XML 一起使用层叠样式表 .....        | 101 |
| 4.3.3 示例: 对 XML 文档应用层叠样式表 .....    | 102 |
| 4.3.4 XML 的 XSL 样式表 .....          | 103 |
| 4.3.5 应用 XSL 样式表 .....             | 104 |
| 4.3.6 示例: 对 XML 文档应用 XSL 样式表 ..... | 105 |
| 4.3.7 实践: 将样式表加到 XML 文档里 .....     | 105 |
| 4.4 将 XML 绑定到 HTML .....           | 108 |
| 4.4.1 将数据送到客户端 .....               | 108 |

|       |                              |     |
|-------|------------------------------|-----|
| 4.4.2 | 创建 XML 数据岛 .....             | 109 |
| 4.4.3 | 将 XML 数据绑定到 HTML 元素 .....    | 110 |
| 4.4.4 | 在 HTML 表中显示 XML 数据 .....     | 111 |
| 4.4.5 | 示例:将 XML 数据绑定到 HTML 元素 ..... | 112 |
| 4.5   | 用文档对象模型访问 XML .....          | 113 |
| 4.5.1 | 介绍文档对象模型 .....               | 113 |
| 4.5.2 | XML 文档树 .....                | 114 |
| 4.5.3 | DOM 对象 .....                 | 115 |
| 4.5.4 | 使用 DOM 操作 XML 数据岛 .....      | 116 |
| 4.5.5 | 示例:使用 DOM 访问 XML 数据 .....    | 117 |
| 4.6   | 验证 XML .....                 | 117 |
| 4.6.1 | 验证的需要 .....                  | 118 |
| 4.6.2 | 格式良好和验证文档 .....              | 119 |
| 4.6.3 | 验证技术:DTD 和 XML 模式 .....      | 120 |
| 4.6.4 | 介绍 DTD .....                 | 121 |
| 4.6.5 | 将 DTDs 加入到 XML 文档中 .....     | 122 |
| 4.6.6 | 实践:应用 DTD 并验证数据 .....        | 123 |
| 4.6.7 | DTD 的局限 .....                | 124 |
| 4.6.8 | 介绍 XML Schemas .....         | 125 |
| 4.6.9 | 应用 XML 模式 .....              | 126 |
| 4.7   | 复 习 .....                    | 127 |

## 5 模块 5: ASP.NET 使用

|       |                                                       |     |
|-------|-------------------------------------------------------|-----|
| 5.1   | 概 述 .....                                             | 130 |
| 5.2   | ASP.NET 简介 .....                                      | 131 |
| 5.2.1 | .NET 框架 .....                                         | 131 |
| 5.2.2 | ASP.NET 的特征 .....                                     | 133 |
| 5.2.3 | ASP.NET 的特征(续) .....                                  | 135 |
| 5.2.4 | 动画:ASP.NET 执行模型 .....                                 | 136 |
| 5.3   | 创建 Web Forms .....                                    | 137 |
| 5.3.1 | 什么是 Web Forms .....                                   | 138 |
| 5.3.2 | 什么是服务器控件 .....                                        | 139 |
| 5.3.3 | 服务器控件的类型 .....                                        | 140 |
| 5.3.4 | 服务器控件是如何工作的 .....                                     | 142 |
| 5.3.5 | 示例:为 ASP.NET 页面添加服务器控件 .....                          | 144 |
| 5.4   | 添加 ASP.NET 代码到页面 .....                                | 146 |
| 5.4.1 | 创建 ASP.NET <SCRIPT> 区 .....                           | 146 |
| 5.4.2 | Visual Basic 6.0 和 Visual Basic.NET Version 7.0 ..... | 147 |
| 5.4.3 | 创建事件过程 .....                                          | 149 |

|                                   |     |
|-----------------------------------|-----|
| 5.4.4 示例:给控件添加代码.....             | 150 |
| 5.5 处理页面事件 .....                  | 151 |
| 5.5.1 页面事件生命周期 .....              | 151 |
| 5.5.2 处理 Postback Forms .....     | 153 |
| 5.5.3 示例:使用回送 Form .....          | 154 |
| 5.6 讨论:ASP 和 ASP.NET .....        | 155 |
| 5.7 实验:使用 ASP.NET 来输出文本 .....     | 155 |
| 5.7.1 练习 1: 创建 ASP.NET 页面.....    | 156 |
| 5.7.2 练习 2: 创建页面事件过程 .....        | 159 |
| 5.8 复 习 .....                     | 160 |
| <b>6 模块 6: 使用 Web 控件</b>          |     |
| 6.1 概 述 .....                     | 162 |
| 6.2 什么是 Web 控件 .....              | 163 |
| 6.3 使用 Intrinsic 控件 .....         | 164 |
| 6.3.1 什么是 Intrinsic 控件 .....      | 165 |
| 6.3.2 Intrinsic 控件列表 .....        | 167 |
| 6.3.3 处理 Intrinsic 控件的事件 .....    | 168 |
| 6.3.4 示例:给页面添加 Intrinsic 控件 ..... | 170 |
| 6.3.5 链接两个控件 .....                | 171 |
| 6.3.6 示例:链接两个控件.....              | 172 |
| 6.4 使用输入验证控件 .....                | 173 |
| 6.4.1 输入验证控件 .....                | 174 |
| 6.4.2 在 Form 中添加输入验证控件 .....      | 175 |
| 6.4.3 验证整个页面 .....                | 178 |
| 6.4.4 示例:添加输入验证控件.....            | 179 |
| 6.4.5 使用输入验证控件的优点 .....           | 180 |
| 6.5 为应用程序选择控件 .....               | 181 |
| 6.6 实验: 使用 Web 控件 .....           | 182 |
| 6.6.1 练习 1: 在页面中添加 Web 控件 .....   | 183 |
| 6.6.2 练习 2: 给页面添加日历控件 .....       | 184 |
| 6.6.3 练习 3: 验证用户输入 .....          | 186 |
| 6.6.4 练习 4: 汇总验证错误 .....          | 188 |
| 6.7 复 习 .....                     | 190 |
| <b>7 模块 7: 使用 ADO.NET</b>         |     |
| 7.1 概 述 .....                     | 192 |
| 7.2 ADO.NET 概述 .....              | 193 |
| 7.2.1 ADO.NET 对象模型 .....          | 194 |

|                                             |     |
|---------------------------------------------|-----|
| 7.2.2 动画:使用 ADO.NET 访问数据 .....              | 196 |
| 7.2.3 记录集和数据集 .....                         | 196 |
| 7.2.4 使用名字空间 .....                          | 198 |
| 7.3 连接数据源 .....                             | 199 |
| 7.4 用 DataSets 访问数据 .....                   | 200 |
| 7.4.1 用 DataSets 读数据 .....                  | 201 |
| 7.4.2 在 DataSet 中存储多个表 .....                | 202 |
| 7.4.3 使用 DataViews .....                    | 203 |
| 7.4.4 在 DataGrid 控件中显示数据 .....              | 204 |
| 7.4.5 示例:在 DataGrid 中显示数据 .....             | 206 |
| 7.4.6 使用模板 .....                            | 207 |
| 7.4.7 使用 Repeater 控件 .....                  | 208 |
| 7.4.8 示例:在控件 Repeater 中显示数据 .....           | 211 |
| 7.5 使用存储过程 .....                            | 212 |
| 7.5.1 调用存储过程 .....                          | 213 |
| 7.5.2 传递参数 .....                            | 214 |
| 7.5.3 调用活动存储过程 .....                        | 216 |
| 7.5.4 示例:调用储备过程 .....                       | 217 |
| 7.6 实验:使用 ADO.NET 访问数据库 .....               | 219 |
| 7.6.1 练习 1:使用 ADO.NET 访问 SQL Database ..... | 220 |
| 7.6.2 练习 2:使用 ASP.NET 控件显示数据 .....          | 222 |
| 7.6.3 练习 3(可选):创建菜单页面 .....                 | 223 |
| 7.7 用 DataReaders 访问数据 .....                | 225 |
| 7.7.1 创建 DataReader .....                   | 226 |
| 7.7.2 从 DataReader 读数据 .....                | 227 |
| 7.7.3 示例:使用 DataReaders 访问数据 .....          | 228 |
| 7.7.4 使用 DataSets 还是 DataReaders .....      | 229 |
| 7.8 绑定到 XML 数据 .....                        | 230 |
| 7.8.1 XML 概述 .....                          | 231 |
| 7.8.2 把 XML 数据读进 DataSet .....              | 232 |
| 7.8.3 示例:把 XML 数据读入到 DataSet .....          | 234 |
| 7.9 复 习 .....                               | 235 |

## 8 模块 8: 创建 ASP.NET Web 应用程序

|                               |     |
|-------------------------------|-----|
| 8.1 概 述 .....                 | 237 |
| 8.2 Web 应用程序的要求 .....         | 238 |
| 8.3 ASP.NET 有什么创新 .....       | 239 |
| 8.3.1 Cookie-less 会话 ID ..... | 239 |
| 8.3.2 配置文件(config.web) .....  | 240 |

|                          |                                             |     |
|--------------------------|---------------------------------------------|-----|
| 8.3.3                    | 设立 Cookie-less 会话 .....                     | 242 |
| 8.3.4                    | 全局应用程序文件 (global.asax) .....                | 243 |
| 8.3.5                    | 示例: 在 Global.asax 中使用事件过程 .....             | 244 |
| 8.3.6                    | 页面缓存 .....                                  | 245 |
| 8.4                      | 在页面间共享信息 .....                              | 246 |
| 8.4.1                    | 使用 ASP.NET 缓存 .....                         | 246 |
| 8.4.2                    | 使用 Config.web 变量 .....                      | 248 |
| 8.4.3                    | 示例: 使用变量 Config.web .....                   | 249 |
| 8.4.4                    | 使用会话和应用程序变量 .....                           | 250 |
| 8.4.5                    | 示例: 使用会话变量 .....                            | 252 |
| 8.4.6                    | 把会话和应用程序变量存储到数据库中 .....                     | 253 |
| 8.4.7                    | 讨论: 共享信息的方式 .....                           | 254 |
| 8.5                      | 使 ASP.NET 应用程序安全 .....                      | 254 |
| 8.5.1                    | 什么是鉴别 .....                                 | 255 |
| 8.5.2                    | 基于 Cookie 的鉴别的体系结构 .....                    | 256 |
| 8.5.3                    | 在 Config.web 里设置安全性 .....                   | 257 |
| 8.5.4                    | 创建登录 Form .....                             | 260 |
| 8.5.5                    | 示例: 在 Config.web 中设置安全 .....                | 263 |
| 8.6                      | 实验: 创建 ASP.NET Web 应用程序 .....               | 264 |
| 8.6.1                    | 练习 1: 存储数据 .....                            | 265 |
| 8.6.2                    | 练习 2: 使用 Cookie 鉴别 .....                    | 266 |
| 8.6.3                    | 练习 3: 在 ASP.NET 页面中获取 Cookie 数据 .....       | 268 |
| 8.6.4                    | 练习 4: 在 Register.aspx 中使用 Cookie 进行鉴别 ..... | 269 |
| 8.7                      | 复 习 .....                                   | 271 |
| <b>9 模块 9: 使用 Web 服务</b> |                                             |     |
| 9.1                      | 概 述 .....                                   | 274 |
| 9.2                      | 什么是 Web 服务 .....                            | 275 |
| 9.2.1                    | Web 服务的特征 .....                             | 275 |
| 9.2.2                    | Web 服务执行模型 .....                            | 276 |
| 9.2.3                    | 查找已有的 Web 服务 .....                          | 278 |
| 9.2.4                    | 已有 Web 服务的实例 .....                          | 279 |
| 9.2.5                    | 示例: 从浏览器调用 Web 服务 .....                     | 280 |
| 9.3                      | 从浏览器调用 Web 服务 .....                         | 280 |
| 9.3.1                    | 从浏览器调用 Web 服务(续) .....                      | 282 |
| 9.4                      | 使用代理调用 Web 服务 .....                         | 282 |
| 9.4.1                    | 将一个 .asmx 文件编译为代理 .....                     | 283 |
| 9.4.2                    | 将代理编译为 DLL .....                            | 285 |
| 9.4.3                    | 调用 Web 服务的方法 .....                          | 286 |

---

|                                               |     |
|-----------------------------------------------|-----|
| 9.4.4 示例:使用代理调用 Web 服务 .....                  | 287 |
| 9.5 使用 Visual Basic 创建一个简单的 Web 服务 .....      | 288 |
| 9.5.1 编写一个 Web 服务 .....                       | 288 |
| 9.5.2 编写一个 Web 服务(续) .....                    | 289 |
| 9.5.3 示例:创建一个 Web 服务 .....                    | 290 |
| 9.6 使用 Visual Studio.NET 创建并调用 Web 服务 .....   | 290 |
| 9.6.1 使用 Visual Studio.NET 创建 Web 服务 .....    | 291 |
| 9.6.2 示例:使用 Visual Studio.NET 创建 Web 服务 ..... | 293 |
| 9.6.3 使用 Visual Studio.NET 调用 Web 服务 .....    | 294 |
| 9.6.4 示例:使用 Visual Studio.NET 调用 Web 服务 ..... | 296 |
| 9.7 实验:使用 Web 服务 .....                        | 297 |
| 9.7.1 练习 1:使用 Web 服务 .....                    | 298 |
| 9.7.2 练习 2:创建一个 Web 服务 .....                  | 300 |
| 9.7.3 练习 3(如果时间允许):其他 Web 服务 .....            | 301 |
| 9.8 复 习 .....                                 | 305 |

**msdn** training

**1**

## 模块 1: .NET 框架概述

概 述

回 望

.NET 和 .NET 框架概述

小 结

**Microsoft**



## 1.1 概 述



- 对开发组织的挑战
- 对客户合作的挑战
- 应用和服务

### 1.2.1 对开发组织的挑战

- 发展
  - 语言和操作系统选择
- 交付
  - CD-ROM、产品、包装、运输
  - 电子交付需要人工参与
- 稳定性
  - DLL 兼容性问题
  - 操作系统——导致不稳定的污点
  - “流行”的问题解决方案的兼容性测试
- 维护
  - 用户参与
  - 包装、产品和交付的媒介

#### 开 发

开发者和(或)开发组织曾经对解决方案的研究提出疑问:“应该用什么作目标平台呢?”答案可能是以下几种: Microsoft Windows 95, Windows 98, Windows Millennium Edition, Windows NT, 或 Windows 2000; Apple Macintosh; UNIX (任何形式); 或是 Web 浏览器。选定了目标平台后, 研发小组在此基础上选择一种语言或工具——有时他们甚至会选择同一种目标平台和开发平台。通常的情况是, 采取哪种平台并不是出于做出最佳解决方案的考虑, 而是出于在目标平台上可用资源的考虑。你可能有亲身感受, 一旦选择了某种平台并进行开发后, 最后更改的可能性就很小了。

#### 交 付

传统的软件解决方案的交付方式主要是 CD 或者软盘。尽管目前也可以选择电子交付, 但有很多缺陷。

CD 和软盘交付需要把物理媒介发送给客户。这有什么不好呢? 首先是费用, 生产者和消费者都要在以下几方面花钱: 买媒介(CD 或软盘), 做成可交付的形式, 进行包装, 邮寄或派人递送等等。而电子交付最大的优点就是不需要包装, 不需要刻光盘, 不需要给光盘作包装设计和印刷, 也不需要邮寄或人员派送, 连宣传手册也可有可无。不过, 订购和下载当今基于 Internet 系统的软件的过程还需要用户的参与, 对坏码也要警惕。

#### 稳定性

如果你是电脑用户, 一定经历过所谓的“DLL 地狱”的 DLL 兼容问题。当制售商的软件

需要使用某种通常已经装载在操作系统(OS)的功能即动态链接库(DLL)时,常出现功能与要求不符的情况。而当前系统和开发模型要求制售商创建一个定制的版本和软件一起发布。在安装过程中,制售商的版本替换了原有的 DLL。

问题在于:原来安装的运行软件不能和新安装的版本一起工作。如果操作系统受到破坏,这种程序错误就成了非常恒定的事件,而且发生得极其频繁。不幸的是,还没有一个使用现有系统,且开发方和用户共同认可的解决这个问题的方案。DLL 兼容性也存在于 OS 以外的其他领域。

## 维 护

维护是软件的致命弱点,因为我们经常需要对现有的版本进行更新。在现有的开发、生产和交付方式下,维护一个软件方案的更新常常要交给客户几乎整个的版本。而且维护同样会碰到前面所述的所有的困难。

### 1.2.2 对合作客户的挑战

- 发展
  - 安装通常需要时间和人员
  - 对现存应用程序和OS进行兼容性测试
  - 许可管理
- 稳定性和安全性
  - 需要测试
  - 可能无意中导致对安全性的破坏
  - 可能动摇系统或其他解决方案
- 维护
  - 访问所有关于发展和安全的问题

## 部 署

如今用户或具体的说是网络管理组,必须手工安装所有需要的软件。这就需要对新软件测试以保证安全性和稳定性,需要人员进行安装,需要授权管理以保证依从性。

### 稳定性和安全性

在公司网络上部署软件还面临安全性和稳定性的风险。

- 该软件来自何处?
- 它是否稳定?
- 安装该软件会不会有意无意地破坏安全性?
- 安装是否使该系统或其他应用软件不稳定?

只有不断进行测试,测试,再测试,才能找到上面问题的答案。这无疑需要巨大的资源

耗费。

### 维 护

维护是公司最为困扰的问题。信息技术小组花费大量时间和资源来测试和部署软件,仅仅是重复进行冗长而昂贵的过程,而此时另一个新版本或服务升级版已经出现了!

## 1.2.3 应用与服务

- 作为服务解决方案的应用
  - 今天的解决方案是作为应用的服务
  - 未来不是应用而是服务
    - 桌面服务 当前的应用 + 任选的 Web 服务
    - Web 服务 交付给任意装置的应用
- 今天的 Internet “服务”
  - 通过协议通过 Internet 交付
    - www hotmail com
    - www lycos com
  - 不是“便携式”不能随身携带
  - 存在一些中间层解决方案
    - www passport com
  - 浏览器、卖主和设备相互依赖
  - 需要交互式访问

### 应用就是服务

也许你已经注意到:今日的软件服务主要以应用程序的形式出现,而且通常在 Windows 桌面或服务器上运行。.NET 和 .NET 框架引入了另外一种可能性:作为服务的软件解决方案。

你将可以开发、安装、使用基于桌面的解决方案。这些解决方案是耦合了基于 Web 服务的应用解决方案。基于 Web 的服务只是按要求通过 Web 交付给应用程序的部分。另一种服务形式是 100% 的纯 Web 服务:应用软件可以在任何时候、任何地方,在任何设备上交付!

### 今日的互联网服务

今日的互联网服务大多是披着狼皮的羊,貌似汹汹。它们大多只提供服务的门户,在其他地方根本无法使用。这意味着在每个站点都必须重复键入联系信息和个人信息。随着“中间层”解决方案的迅猛发展,如 Microsoft Passport,这种局面已经开始改变。不过还没有达到合理的预期目标。

现有网络软件的另一个大缺陷是对设备的依赖性。多数网站假定你是通过个人电脑上网,有一个最低屏幕分辨率的限制,如 800 × 600 像素。有的网址甚至限制了浏览器的版本,如 Internet Explorer 5。很少有网站考虑了用户的个人数字助理(PDA)。