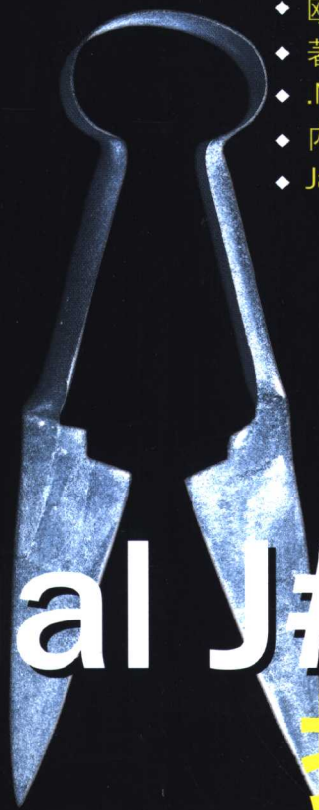


微软.NET程序员系列

- ◆ 欧美读者评价 ★★★★★
- ◆ 著名编程专家精心编著
- ◆ .NET核心开发组鼎力推荐
- ◆ 内容经典，示例典型
- ◆ Java和J++程序员通往.NET世界的金钥匙



Microsoft®

Visual J# .NET 技术内幕

**MICROSOFT Visual J# .NET
CORE REFERENCE**

许志庆 王卫国 译
Visual Studio .NET产品组 审校

John Sharp
[美] *Andy Longshaw* 著
Peter Roxburgh

Microsoft
.net



清华大学出版社

微软.NET 程序员系列

Visual J# .NET 技术内幕

(美)John Sharp, Andy Longshaw, Peter Roxburgh

许志庆 王卫国

Visual Studio .NET 产品组

著

译

审校

清华大学出版社

北 京

内 容 简 介

J#是一种采用 Java 语法的编程语言, 它使 Java 程序员能够使用 .NET 框架开发各种应用程序。

本书全面、深入地讲述了如何使用 J#和 Visual Studio .NET, 开发在 .NET 环境下运行的企业应用程序。本书分为五大部分。第I部分, “.NET 的体系结构”, 介绍 .NET 平台的结构、主要概念、公共语言运行库以及 .NET 中提供的 GUI 库等。第II部分, “管理和操作数据”, 重点介绍在 J#中如何处理数据。第III部分, “企业开发”, 介绍如何开发可以通过局域网运行的多线程应用程序。第IV部分, “与 Windows 进行集成”, 介绍如何开发能够利用 Windows 的特性与服务的应用程序。第V部分, “生成 Web 应用程序”, 介绍使用 J#开发能够连接到 Internet 的应用程序, 包括 ASP.NET 模型、Web 服务开发等。

本书适合具有 Java 开发经验、想利用 .NET 平台开发各种应用程序的程序员。

Visual J# .NET 技术内幕

Microsoft Visual J# .NET Core Reference

John Sharp, Andy Longshaw, Peter Roxburgh

Copyright © 2003 by Microsoft Corporation.

Original English Language Edition Copyright © 2003 by Microsoft Corporation.

Published by arrangement with the original publisher, Microsoft Press,

a division of Microsoft Corporation, Redmond, Washington, U.S.A.

本书中文版由 Microsoft Press 授权清华大学出版社出版。

北京市版权局著作权合同登记号 图字: 01-2003-0839

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

Microsoft Visual J# .NET 技术内幕/(美)夏普(Sharp,J.), (美)朗萧(Longshaw,A.), (美)罗克斯巴勒(Roxburgh,P.)

著; 许志庆, 王卫国译.—北京: 清华大学出版社, 2003

(微软.NET 程序员系列)

书名原文: Microsoft Visual J# .NET Core Reference

ISBN 7-302-07711-8

I.M… II.①夏…②朗…③罗…④许…⑤王… III.JAVA 语言—程序设计 IV.TP312 .

中国版本图书馆 CIP 数据核字(2003)第 108659 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

译 者: 许志庆 王卫国

文稿编辑: 张 莉 赵飞虎

封面设计: 陈刘源

印 刷 者: 北京牛山世兴印刷厂

装 订 者: 三河市李旗庄少明装订厂

发 行 者: 新华书店总店北京发行所

开 本: 185 × 230 印张: 44.25 插页: 5 字数: 949 千字

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

书 号: ISBN 7-302-07711-8/TP · 5646

印 数: 1 ~ 3000

定 价: 78.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770175-3103 或 (010)62795704。

《微软.NET 程序员系列》序

自 2000 年 6 月微软宣布自己的 .NET 战略以来,在不到两年的时间里,.NET 已经从战略变成现实。.NET 带来了全新的、快速而敏捷的企业计算能力,也给软件开发商和软件开发人员提供了支持未来计算的高效 Web 服务开发工具。作为微软.NET 战略的重要组成部分——Visual Studio .NET (中文版)已经于 2002 年 3 月 22 日正式在中国推出。

Visual Studio .NET 是一个功能强大、高效并且可扩展的编程环境。它充分展现了应用程序开发的潜能,并提供了生成应用程序所需的工具和技术。这些应用程序将给当今的企业、机构提供强大的支持,并推动下一代基于 XML Web 服务软件的发展。

有了 Visual Studio .NET,那些对全世界数百万的专业和业余程序员来说曾一度极端复杂、费时费力,甚至让人望而生畏的编程任务现在已不再神秘。更重要的是,Visual Studio .NET 使开发人员能运用既有的技能和知识来迎接新的编程挑战。

在 10 年前,Visual Basic 1.0 成为数以百万计的开发人员的革命性的应用程序开发语言。现在,Visual Studio .NET 为未来的 10 年做好了准备。

微软出版社为了配合 Visual Studio .NET 的推广以及.NET 技术的普及,邀请 Visual Studio .NET 项目开发组的核心开发人员和计算机图书专业作家精心编写了英文版《微软.NET 程序员系列》丛书;该丛书自面市以来,在美国图书销量排行榜上一直高居前列,颇受好评,成为程序开发人员和网络开发人员了解.NET 技术的权威工具书。尤其是《Microsoft .NET Framework 程序设计》一书,长期占据美国及欧洲此类书籍的排行榜冠军位置,程序开发人员不可不读此书。

清华大学出版社为了满足中国广大程序开发人员、网络开发人员学习最新技术的渴望,在微软出版社的配合下,从《微软.NET 程序员系列》这套丛书中精选了 50 余本翻译成中文,以满足国内广大读者的需要。这套丛书阵容庞大(且在不断扩充之中),几乎涵盖了.NET 技术及其应用的各个方面;也正因为如此,翻译和编辑加工的工作量也大得惊人。但为了保持国外优秀技术图书的魅力,同时使读者领会新技术的真谛,本丛书的翻译和编辑都是经过严格筛选的、具有很高的翻译水平或丰富编辑经验的技术人员;另外,我们还聘请微软公司 Visual Studio .NET 产品组的技术专家审读每一本书,确保在技术上准确无误。

相信这套丛书定会帮助程序开发人员、网络开发人员以及那些具有一定编程基础的中、高级读者,快速、全面地掌握.NET 技术,协助他们为技术生涯的下一个 10 年做好准备,为培养新一代软件人才,并推动中国软件产业的快速发展起到积极的作用!

这套丛书分为 3 个子系列:技术内幕系列、语言参考系列和程序员系列。目前,已出版和在编的共有 36 本,已从 2002 年 6 月份起陆续和读者见面。

● 技术内幕系列

目前共有 7 本：

◆ 《Visual C++ .NET 技术内幕(第 6 版)》

本书是 Visual C++ 和 MFC 开发的经典著作。它秉承了第 4 版和第 5 版的风格，已根据该编程语言的最新版本 Visual C++ .NET 进行了全面更新和补充，是 .NET 时代的 C++ 程序员必读的教材。此外，本版仍由第 4 版译者潘爱民先生翻译。

◆ 《Microsoft .NET Compact Framework 技术内幕》

◆ 《Visual Basic .NET 技术内幕》

◆ 《Visual C# .NET 技术内幕》

◆ 《ADO.NET 技术内幕》

◆ 《Microsoft .NET 程序设计技术内幕》

◆ 《Visual J# .NET 技术内幕》

● 语言参考系列

目前共有 3 本：

◆ 《Visual Basic .NET 语言参考手册》

◆ 《Visual C# .NET 语言参考手册》

◆ 《Visual C++ .NET 语言参考手册》

● 程序员系列

目前共有 27 本。详细书目，请参见本书彩插页。

其中，Microsoft .NET Framework 1.1 类库(1-4 卷)是 .NET System 命名空间大全，直接源自 Microsoft .NET Framework。这 4 卷书包含了 .NET Framework Class Library System 命名空间的全部详细信息。帮助开发人员充分利用 .NET Framework 来开发 Web 应用程序、客户端应用程序和 XML Web 服务。

随着技术的发展，我们将根据读者的需要，不断增加新的书目。

丛书版式特色

本丛书在风格上力求文字精炼。并采用小 5 号字编排，内容紧凑，版面清晰美观，易于阅读。此外，书中还安排了一些特色段落，提供正文之外的一些细节知识：



注意：提醒阅读和操作过程中应注意的事项，避免出现错误或问题。



提示：指点一些操作捷径或实用技巧，使您少走弯路，阅读和操作更为高效。



要点：总结关键知识点或操作细节，帮助您适时掌握要领。

 **注：**提示首次出现的编程元素，以及书中涉及到该元素的其他位置以供参考。

 **警告：**阅读和操作过程中，应特别注意的事项，如果使用不当，可能会导致严重后果。

尽管我们倾心相注，精心而为，总有疏忽纰漏，恳请广大读者不吝赐教与指正，我们定会全力改进，以期在后续工作中得以完善。

本丛书在创作过程中得到了微软(中国)公司的大力支持。本丛书能够顺利出版，更是倾注了无数幕后人员的汗水和心力。在此，对他们的辛勤劳动一并表示衷心感谢！

编者

2003 年 3 月

前言

Blaise Pascal、Ada Lovelace、Edna Cobol 以及现在的 John Sharp，我们有什么共同之处？正如您所见的，我的名字与这门语言非常类似。如果 Microsoft 是根据我的名字为 Visual J#命名就好了，但很遗憾，事实并非如此。这只是偶然发生的一个巧合而已。

那么，到底什么是 J#？Microsoft 为何开发出 J#？从某种程度上来说，说明 J#不是什么要更简单些。J#不是 Java。J#是一种编程语言，它使用了 Java 的语法，从而使 Java 开发人员可以利用 Microsoft .NET 框架构建应用程序。您会发现可以使用 J#来重新编译许多现有的 Java 类，从而生成可在 .NET 公共语言运行库中运行的可执行部分。J#还是一种开发新应用程序的强大语言。它为不想放弃使用熟悉语法这一便利的 Java 和 J++开发人员提供了一条通向 .NET 的大路。

本书描述了如何在 .NET 环境中使用 J#和 Visual Studio .NET 开发企业级应用程序。开发 .NET 的目的就在于帮助您建立分布式系统。它提供了丰富的类框架，从而为您提供了一种基础结构，您可以根据自己的需求对其进行扩展。本书还详细介绍了在使用 J#进行编程时，如何使用绝大多数 .NET 特性。因此，通过本书您会找到关于 .NET 和公共语言运行库的许多信息。

本书的目的是什么？这本书的目标不是教您如何使用 Java 编写程序，我们假设本书读者对 Java 语言的语法和句法均非常熟悉。书中的许多概念和思想都以一种 Java 开发人员能够理解的方式引入，但是我们会对 J#概念的截然不同之处详细阐述。

编写一本关于未发布的产品的书非常具有挑战性。因为一种未完成的产品可能会发生一些轻微的变动和更改(如果您拥有任何公开发布的 Beta 版，那么都应该对其进行升级，这是因为对于 Beta 版而言，正式版做出了一些重大的变化)。但是，我们对本书中出现的所有错误之处和不准确的地方都具有全责，非常乐于了解到这些错误并修订它们。

附：在给我发 E-mail 之前，我应该说明 COBOL 也不是根据 Edna Cobol 命名的。她是我的宠物猫的心理医生，我在这里提到她，已得到她的同意。

本书组织方式

第 I 部分，.NET 的体系结构，这部分提供了 .NET 自身的一些技术信息。第 1 章汇总了分布式系统所带来的挑战。第 2 章描述了 .NET 平台的整体目标和结构。它介绍了一些关键性概念，如 .NET 框架、JIT 编译以及 .NET 和其他重要的技术(如 COM+)之间的关系。第 3 章详细介绍了公共语言运行库，并描述了 MSIL 和托管执行环境。本章还介绍了公共语言库中面向组件的特性——属性、命名空间、

委托和事件并且为您介绍如何将 Java 的实现映射到 J#中, 第 4 章将 .NET 中的 GUI 库与 Java 开发者工具包中的 GUI 库进行了比较。

第 II 部分, 管理和操作数据, 集中讨论了在 J#中处理数据的内容。第 5 章介绍了 J#中可用的 XML 支持, 并且介绍了如何产生和使用有效的 XML 数据。第 6 章介绍了如何使用 .NET 内建的 XSLT 支持将 XML 从一种格式转换为另外一种。第 7 章介绍了 ADO.NET。这一章为您介绍如何使用 ADO.NET 以有效、可扩展的方式连接数据源、识别查询并执行更新和删除操作。

第 III 部分, 企业开发, 这一部分介绍了如何建立 LAN 的多线程应用程序。第 8 章介绍了 .NET 线程模型, 并为您介绍通过 J#使用多个线程的方法。第 9 章介绍了基本网络机制, 它们可用于与其他机器或平台(如 UNIX)进行通信, 并且覆盖了 .NET 中实现套接字的内容。第 10 章描述的是 .NET 用于在线程和进程之间封送对象的机制。本章介绍了序列化和非序列化的操作方式, 并且比较了 XML 与二进制序列化。第 11 章介绍的是 .NET 远程处理体系结构, 包括值和引用对象远程处理、信道、客户消息、远程事件处理、激活模型、安全性和加密。第 12 章为您介绍如何使用 MSMQ 来实现可靠、同步、使用 J#和 .NET 的消息处理。

第 IV 部分, 与 Windows 进行集成。本部分主要介绍如何生成使用具有 Windows 功能与服务的应用程序。其中第 13 章介绍如何将原有的 Java 组件、DLL 和 COM 组件集成进 J#应用程序。同时还介绍如何在支持早期与晚期绑定的 J#中创建 COM 可调用组件。第 14 章说明如何创建使用 Component Services(COM+)的 J#组件。同时也说明如何创建使用事务的组件以便维护数据的完整性。第 15 章介绍如何在 J#中编写 Windows 服务。同时介绍了服务应用程序的结构与生命周期, 并且还包括了安装与安全性主题。

第 V 部分, 生成 Web 应用程序。这一部分介绍如何使用 Internet 生成 J#应用程序以便获得全球性访问。其中第 16 章说明了 ASP.NET 模型如何工作并且揭示了 J#中 ASP.NET 应用程序的结构。第 17 章介绍如何在 J#中生成 Web 服务并且介绍如何通过 Web 访问服务与组件。同时还详细的介绍了 SOAP 并且介绍了使用 Web 服务作为 COM 与其他组件外表的技术。第 18 章介绍如何生成能够同步和异步调用 Web 服务的 Web 服务客户端。同时还介绍了动态 Web 服务搜索以及 UDDI 的使用。

系 统 需 求

本书中的所有示例均是使用 Windows XP Professional 和 Windows .NET Server(Beta 3)建立并测试的。除第 15 章中的示例之外, 所有的代码也可在 Windows 2000 下运行(推荐使用 SP 2)。当然, 您的计算机上还需要安装 Visual Studio .NET 和 Visual J# .NET。

要想运行这些示例, 您需要:

- Microsoft Internet Explorer 5.5 或以上版本

- Internet Information Services(IIS)
- 随 Microsoft Visual Studio .NET 安装的 .NET Framework SDK

支持信息

为了降低书的成本,减轻读者负担,对于因内容很少而不值得单独配盘的图书,我们将其范例代码或练习文件放在我们的网站上,供读者下载。敬请访问以下网址: <http://www.wenyuan.com.cn>, 查找本书的有关链接。

如果您对本书或配书文件有任何建议、意见或想法,请通过以下电子邮件与清华大学出版社第三事业部客户服务部取得联系:

service@wenyuan.com.cn

或致函:

北京 100084-157 信箱

读者服务部

邮编: 100084

亦可致电: 010-62792098-220。

请注意,上述地址并不提供软件产品的支持。

目 录

前言	XIII
----------	------

第 I 部分 .NET 的体系结构

第 1 章 N 层开发面临的挑战 3

1.1 客户端/服务器和 N 层体系结构	3
1.1.1 两层体系结构	3
1.1.2 N 层体系结构	5
1.1.3 Northwind Traders 示例	6
1.2 连接与数据格式	8
1.2.1 套接字	9
1.2.2 远程过程调用	10
1.2.3 对象 RPC	11
1.2.4 远程方法启用	12
1.2.5 Web	13
1.3 数据访问	15
1.3.1 开放式数据库连接	15
1.3.2 Java 数据库连接	16
1.3.3 ADO 和 ADO.NET	16
1.4 非功能性需求	17
1.4.1 安全性	17
1.4.2 可缩放性和可用性	18
1.4.3 完整性和事务	18
1.5 .NET 企业服务器	19
1.6 Java 和 .NET	19
1.7 本章小结	20

第 2 章 Microsoft .NET 21

2.1 编译器和语言支持	21
2.1.1 公共语言运行库	21
2.1.2 跨语言开发	29

2.1.3 内存管理	33
------------------	----

2.1.4 将非托管代码集成到 .NET 应用程序中	37
-------------------------------------	----

2.2 程序集和部署 38

2.2.1 DLL 的乐趣：电影剧本	38
2.2.2 .NET 程序集	39
2.2.3 私有程序集	41
2.2.4 全局程序集高速缓存	42
2.2.5 配置应用程序	45

2.3 .NET 框架类库 49

2.3.1 命名空间	50
------------------	----

2.4 .NET 远程处理和 Web 51

2.4.1 .NET 远程处理体系结构	51
2.4.2 远程对象激活	52
2.4.3 ASP.NET	53
2.4.4 Web 服务	54

2.5 .NET 安全性 54

2.5.1 应用程序域	55
2.5.2 基于角色的安全性	57
2.5.3 代码访问安全性	60

2.6 本章小结 65

第 3 章 Java 和公共语言运行库 66

3.1 组件和 Java 67

3.1.1 JavaBeans	68
3.1.2 .NET 中的组件	73

3.2 Java、J# 和 .NET 框架 80

3.2.1 类的打包和查找	81
3.2.2 Java 类的层次结构	82
3.2.3 其他问题	89
3.2.4 迁移到 J#	92

3.3 元数据和属性.....	94	5.2.1 使用 XMLReader 类 分析 XML 文档.....	153
3.3.1 .NET 中的反射.....	94	5.2.2 使用 XMLTextReader 实例 处理 XML.....	153
3.3.2 属性(Attribute).....	101	5.2.3 用于读取和导航的 其他选项.....	160
3.4 企业版 Java.....	102	5.2.4 类型和命名空间.....	161
3.4.1 J2EE 的元素.....	103	5.2.5 异常处理.....	163
3.4.2 .NET 与 J2EE 的比较.....	106	5.2.6 使用 XmlWriter 类编写 XML 文档.....	163
3.4.3 Java 宠物商店.....	109	5.2.7 写入时的转义和复制.....	166
3.5 本章小结.....	111	5.3 验证和实体解析.....	166
第 4 章 图形用户界面.....	112	5.3.1 验证 XML 文档.....	166
4.1 桌面 GUI.....	112	5.3.2 解析实体.....	171
4.1.1 重访 Java GUI 开发.....	113	5.4 在内存中处理 XML 文档.....	171
4.1.2 Windows Forms 库.....	115	5.4.1 内存中处理.....	172
4.2 移植现有 Java 应用程序.....	115	5.4.2 将 XML 加载到 XmlDocument 类.....	173
4.3 使用 .NET 类编写 GUI 应用 程序.....	124	5.4.3 从 DOM 文档获取信息.....	173
4.3.1 创建 Windows 窗体.....	124	5.4.4 将 DOM 片段视为流.....	178
4.3.2 向窗体添加控件.....	128	5.5 编写和操作内存中 XML 文档.....	178
4.3.3 处理事件.....	134	5.5.1 更改 DOM 树的内容.....	178
4.3.4 使用文件对话框.....	135	5.5.2 大量更改 XML 文档.....	179
4.3.5 使用系统剪贴板.....	139	5.5.3 写出 DOM 树.....	180
4.3.6 生成并运行应用程序.....	141	5.6 XML 与数据.....	181
4.4 本章小结.....	141	5.6.1 XML 与 ADO.NET 数据 之间的联系.....	181
第 II 部分			
管理和操作数据			
第 5 章 处理 XML.....	145	5.6.2 将 XML 视为关系数据.....	182
5.1 在 .NET 应用程序中使用 XML.....	145	5.6.3 将 XML 作为关系数据 操作.....	182
5.1.1 作为数据格式的 XML.....	146	5.6.4 将关系数据视为 XML.....	184
5.1.2 XML 的角色.....	148	5.7 本章小结.....	184
5.1.3 XML 支持所需应用程序.....	149	第 6 章 转换 XML.....	185
5.1.4 处理 XML 数据.....	149	6.1 在 .NET 应用程序中转换 XML.....	185
5.1.5 Visual J# 和 .NET 框架中对 XML 的支持.....	150		
5.2 XML 文档的单个传递处理.....	152		

6.1.1 转换需求.....	185	7.3.4 从数据源中检索单条记录.....	223
6.1.2 XSLT 处理模型.....	187	7.3.5 将 DataReader 对象用于只读数据检索.....	223
6.1.3 应用转换.....	190	7.4 使用 DataSet 对象进行数据访问.....	225
6.1.4 .NET 对 XML 转换的支持.....	191	7.4.1 从数据适配器填充 DataSet 表.....	225
6.2 对 XML 文档应用样式表.....	192	7.4.2 导航类型化的 DataSet.....	227
6.2.1 使用 XslTransform 进行的简单转换.....	192	7.4.3 导航非类型化的 DataSet.....	229
6.2.2 转换源和目标.....	194	7.4.4 从数据适配器操作和更新数据存储区.....	230
6.2.3 转换 DataSet.....	195	7.4.5 定义关系和约束.....	233
6.3 优化和部分转换.....	195	7.5 事务管理.....	236
6.3.1 使用 XPath 搜索和导航.....	196	7.6 使用 XML 和 ADO.NET.....	237
6.3.2 优化 XSLT 转换.....	197	7.6.1 以 XML 形式编写 DataSet.....	237
6.3.3 部分转换.....	198	7.6.2 以 XML 的形式写入 DataRelation 对象.....	240
6.4 参数化和外部功能.....	198	7.6.3 由 XML 推断 DataSet 结构.....	240
6.4.1 传入参数.....	199	7.7 本章小结.....	242
6.4.2 调用外部函数.....	202		
6.5 本章小结.....	207		
第 7 章 ADO.NET.....	208	第 III 部分 企业开发	
7.1 ADO.NET 的体系结构.....	208	第 8 章 .NET 下的多线程.....	245
7.1.1 ADO.NET 的组件.....	209	8.1 线程和.NET.....	245
7.1.2 DataSet 对象.....	210	8.1.1 应用程序域和线程.....	247
7.1.3 连接和数据适配器.....	211	8.1.2 线程的创建.....	248
7.2 连接到数据存储区.....	211	8.1.3 线程和安全.....	249
7.2.1 连接到数据源(OLE DB 和 SQL).....	212	8.1.4 给线程传送参数.....	249
7.2.2 连接和 Visual Studio .NET.....	213	8.1.5 线程状态.....	250
7.2.3 连接池.....	214	8.1.6 终止线程.....	252
7.2.4 使用连接事件.....	214	8.1.7 调度线程.....	254
7.3 对数据存储区执行命令.....	215		
7.3.1 创建命令.....	215		
7.3.2 在语句中使用参数.....	217		
7.3.3 调用存储过程.....	223		

8.1.8 线程和自由(不可控) 代码	256	第 10 章 对象的序列化	323
8.2 同步	257	10.1 序列化与反序列化数据	323
8.2.1 手工同步	260	10.1.1 格式化数据	324
8.2.2 自动同步	267	10.1.2 反序列化	326
8.2.3 静态和线程数据	270	10.1.3 版本化	327
8.3 线程内部通信	271	10.1.4 进行选择	331
8.3.1 线程通知	271	10.2 高级序列化	331
8.3.2 计时器	280	10.2.1 自定义序列化	331
8.4 线程池	281	10.2.2 处理对象图	334
8.4.1 Threadpool 类	282	10.3 XML 序列化	339
8.4.2 异步 IO	283	10.3.1 XML 格式化	340
8.5 本章小结	284	10.3.2 反序列化 XML 流	346
第 9 章 网络编程基础	285	10.4 本章小结	349
9.1 套接字基础	285	第 11 章 .NET 远程处理	351
9.1.1 面向连接的套接字	286	11.1 公共语言运行库远程处理 体系结构	351
9.1.2 数据传输的问题	299	11.1.1 远程对象	351
9.1.3 无连接的套接字	300	11.1.2 .NET 远程处理模型	354
9.1.4 阻塞与非阻塞套接字	305	11.1.3 ObjRef 对象与代理	359
9.2 套接字的异步使用	306	11.1.4 消息、信道与信道接收	360
9.2.1 Poll 和 Select 方法	306	11.2 使用 TCP 远程处理编程	361
9.2.2 网络流	308	11.2.1 服务器端激活对象的 远程处理	361
9.3 Web 网络编程	309	11.2.2 客户端激活对象的 远程处理	368
9.3.1 能够插入的协议	310	11.2.3 管理对象的生存期与 租约	370
9.3.2 使用 HTTP 进行数据 请求和接收	312	11.2.4 TCP 远程处理安全性	372
9.3.3 发布数据	314	11.2.5 远程方法参数	373
9.3.4 请求的异步处理	315	11.2.6 远程事件	373
9.3.5 使用 WebClient 对象	316	11.3 HTTP 远程处理	378
9.3.6 HTTP 连接管理和池	317	11.3.1 远程处理服务器宿主	378
9.4 Internet 的安全性	318	11.3.2 使用 IIS 作宿主	380
9.4.1 认证和授权	319	11.3.3 HTTP 远程处理安全性	382
9.4.2 加密	321		
9.5 本章小结	321		

11.4 自定义远程处理	383	13.2 在非托管 DLL 中调用方法	428
11.4.1 单向远程处理	383	13.2.1 使用 J/Direct	429
11.4.2 RemotingServices 类	383	13.2.2 平台调用服务	433
11.4.3 跟踪处理程序	385	13.2.3 其他的 P/Invoke 问题	447
11.4.4 自定义信道接收和信道	388	13.3 调用 COM 组件	450
11.5 本章小结	389	13.3.1 创建并使用 RCW	451
第 12 章 使用消息队列	390	13.3.2 接收 COM 事件	455
12.1 Message Queuing 3.0 的 体系结构	391	13.3.3 使用没有类型库的 COM 对象	457
12.1.1 队列、服务器和 Active Directory	391	13.4 将 .NET 组件集成进 COM 中	464
12.1.2 事务性消息队列	392	13.4.1 为 COM Interop 设计 .NET 组件	465
12.1.3 管理队列	392	13.4.2 创建 COM 可调用包装	469
12.1.4 系统队列	393	13.4.3 测试 CCW	472
12.1.5 消息传送	394	13.5 与其他技术的互用性	474
12.1.6 消息队列触发器	394	13.5.1 实际的解决方案: XML Web 服务	475
12.2 消息队列编程	395	13.6 本章小结	475
12.2.1 发送和接收消息	395	第 14 章 服务组件与 COM+	476
12.2.2 处理消息	404	14.1 使用现有的 COM+ 组件	477
12.2.3 管理队列	408	14.1.1 再论 FourthCoffee 组件	477
12.3 异步操作	412	14.1.2 配置 Fourth Coffee COM+ 应用程序	478
12.3.1 异步接收消息	412	14.1.3 使用 Fourth Coffee COM+ 应用程序	481
12.3.2 断开连接的队列	413	14.1.4 订阅松耦合事件	482
12.3.3 请求确认	416	14.2 生成服务组件	488
12.4 实际环境中的消息	417	14.2.1 服务组件基础	489
12.4.1 可靠性与事务	418	14.2.2 注册与使用服务组件	494
12.4.2 消息的鉴别与加密	421	14.3 服务组件的功能	497
12.4.3 通过 HTTP 传送消息	423	14.3.1 同步、活动与上下文	497
12.5 本章小结	424	14.3.2 静态方法	500
		14.3.3 服务组件的激活	500
		14.3.4 缓存共享的状态	506
第 IV 部分			
与 Windows 进行集成			
第 13 章 非托管组件的集成	427		
13.1 托管与非托管代码	428		

14.3.5 关于事务的更多说明	510	16.1.3 ASP.NET 的基本 编程模型	544
14.4 .NET 与 COM+ 的安全性	513	16.2 HTML 表单和 ASP 表单	548
14.4.1 代码访问安全性需求	513	16.3 ASP.NET Web 表单	552
14.4.2 .NET 的基于角色的 安全性模型	513	16.3.1 服务器端控件	555
14.4.3 COM+ 的基于角色的 安全性模型	514	16.3.2 页面后面的代码	556
14.4.4 从 .NET 中实现 COM+ 安全性	515	16.3.3 事件的处理	560
14.4.5 COM+ 的强制安全性	517	16.3.4 客户端验证	563
14.5 异步组件	518	16.4 从 ASP 页面中迁移	568
14.5.1 创建排队组件	518	16.4.1 语言与代码	568
14.5.2 支持松耦合事件	519	16.4.2 用户接口	569
14.6 本章小结	521	16.5 页面、控件与数据	571
第 15 章 实现 Windows 服务	522	16.5.1 Page 类	571
15.1 控制 Windows 服务	522	16.5.2 公共控件	574
15.1.1 显示服务信息	523	16.5.3 创建自己的控件	576
15.1.2 服务的启动和停止	526	16.5.4 数据绑定	580
15.2 编写 Windows 服务	528	16.6 生成 ASP.NET Web 应用程序	583
15.2.1 服务应用程序的结构	528	16.6.1 Web.config	583
15.2.2 了解安装程序类	532	16.6.2 Global.asax	584
15.2.3 创建新安装程序	533	16.6.3 部署 ASP.NET 应用 程序	586
15.2.4 增加服务说明	535	16.6.4 状态的管理	586
15.2.5 安装并测试服务	536	16.6.5 错误处理程序	593
15.2.6 卸载服务	538	16.6.6 安全性	598
15.3 本章小结	538	16.6.7 缓存	601
		16.7 本章小结	604
第 V 部分		第 17 章 生成 Web 服务	605
生成 Web 应用程序		17.1 Web 服务概述	605
第 16 章 ASP.NET 是更好的 ASP	541	17.1.1 Web 服务是什么	605
16.1 ASP.NET 简介	542	17.1.2 Web 服务技术	606
16.1.1 基于浏览器的 Web 应用程序	542	17.1.3 .NET 中的 Web 服务	608
16.1.2 ASP.NET 环境	543	17.2 创建 Web 服务	609
		17.2.1 简单的 Web 服务	609

17.2.2 使用 Visual Studio .NET 创建 Web 服务	613	服务来提供	653
17.3 Web 服务说明与数据类型	616	17.5 本章小结	654
17.3.1 提供 Web 服务接口	616	第 18 章 创建 Web 服务客户端	655
17.3.2 调用服务	627	18.1 Web 服务客户端	655
17.3.3 传递复杂的数据类型	628	18.1.1 Web 服务的客户端视图	656
17.3.4 传递 DataSet 对象	639	18.1.2 使用 Visual Studio .NET 创建 Web 服务客户端	657
17.3.5 传递 XML 文档	641	18.1.3 跨越简单的客户端方案	660
17.4 创建 XML Web 服务应用 程序	642	18.1.4 其他客户端类型	677
17.4.1 作为 ASP.NET 应用的 Web 服务	643	18.2 动态搜索 Web 服务	680
17.4.2 事务处理和 Web 服务	651	18.2.1 搜索服务器上的服务	681
17.4.3 将现存的应用程序作为 Web		18.2.2 通过 UDDI 搜索服务	686
		18.3 本章小结	689

第 I 部分

.NET 的体系结构

这部分提供了.NET 自身的一些技术信息。第 1 章汇总了分布式系统所带来的挑战。第 2 章描述了.NET 平台的整体目标和结构。它介绍了一些关键性概念,如.NET 框架、JIT 编译以及.NET 和其他重要的技术(如 COM+)之间的关系。第 3 章详细介绍了公共语言运行库,并描述了 MSIL 和托管执行环境。本章还介绍了公共语言库中面向组件的特性——属性、命名空间、委托和事件并且为您介绍如何将 Java 的实现映射到 J# 中,第 4 章将.NET 中的 GUI 库与 Java 开发者工具包中的 GUI 库进行了比较。