

微软.NET程序员系列

- ◆ 欧美读者评价:★★★★☆
- ◆ 如何用Windows窗体开发GUI元素
- ◆ 如何用ADO.NET访问和存储数据
- ◆ 如何开发可以跨无线网络连接和断开连接的应用程序
- ◆ 如何与SQL Server集成数据

Microsoft®

.NET Compact Framework 技术内幕

Microsoft .NET Compact Framework

Andy Wigley
Stephen Wheelwright 著

罗 兵 徐雅骥 江 楠 译
Visual Studio .NET产品组 审校

Microsoft
.net



清华大学出版社

微软.NET 程序员系列

Microsoft .NET Compact Framework 技术内幕

Andy Wigley, Stephen Wheelwright

著

罗 兵 徐雅骥 江 楠

译

Visual Studio .NET 产品组

审校

清华大学出版社

北 京

内 容 简 介

.NET Compact Framework 为手持设备(如 Pocket PC 和智能电话)提供了.NET Framework 的强大功能。

本书是《微软.NET 程序员系列》丛书之一。全书分为 5 个部分,分别介绍了.NET Compact Framework 的体系结构、如何用.NET Compact Framework 开发应用程序、如何访问数据,以及一些可用来开发高级移动应用程序的技术,可帮助读者轻松地完成从桌面设备开发者向移动设备开发者的转变。

本书适合有一定程序开发经验的中、高级读者阅读。

Microsoft .NET Compact Framework 技术内幕

Microsoft .NET Compact Framework(ISBN 0-7356-1725-2)

Andy Wigley, Stephen Wheelwright

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:01-2003-3534

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 .NET Compact Framework 技术内幕/维葛利,惠尔瑞特著;罗兵,徐雅骥,江楠译。

—北京:清华大学出版社,2003

(微软.NET 程序员系列)

书名原文:Microsoft .NET Compact Framework

ISBN 7-302-07228-0

I.M… II.①维…②惠…③罗…④徐…⑤江… III.①移动通讯—通讯设备—程序设计

②微型计算机—程序设计 IV.①TN929.5②TP368.33

中国版本图书馆 CIP 数据核字(2003)第 080748 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:罗 兵 徐雅骥 江 楠

文稿编辑:马云艳

封面设计:陈刘源

印 刷 者:世界知识印刷厂

发 行 者:新华书店总店北京发行所

开 本:787×960 1/16 印张:45.75 插页:5 字数:1090 千字

版 次:2003 年 9 月第 1 版 2003 年 9 月第 1 次印刷

书 号:ISBN 7-302-07228-0/TP·5257

印 数:1~3500

定 价:76.00 元

《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，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

越来越多的企业开始使用手持设备的主要因素有两个，其一是硬件的费用越来越低。在笔者起草本书前言的这个星期内，Dell 正在以一种意在压倒对手的超低价格发布她的第一款 Pocket PC，HP 已经宣布了新的低端和高端 iPaq，而 Microsoft 和 Samsung 宣布将有一种全新的 Pocket PC 概念设计，要“大大地降低成本并减少移动设备 OEM(原始设备制造商)的开发时间”。在移动设备提供商之间的这种竞争使价格越来越低，也使得各企业开始认识到为自己企业内的大量职工购买移动设备是物有所值的。

这些智能设备也越来越能干了。处理器的速度越来越快，低端市场中的设备现在也有了 32 MB 的 RAM，要知道，就在不久以前，它们的 RAM 还只有 16 MB。较高端的设备仍然内置了蓝牙技术和 802.11 无线联网功能，而电话与 PDA 的“混血”产品，如 Pocket PC 电话版，能够在移动电话运营商提供的网络中使用通用分组无线业务(GPRS)或 CDMA2000 包交换数据通信技术。

第二个关键因素是软件的体系结构得到了改善。.NET Compact Framework 就是托管执行环境的一个例子。运行库引擎位于设备上，负责管理.NET 应用程序的执行。这种功能性会带来很多益处，例如内存管理、跨语言互操作性和改善的错误处理机制。.NET Compact Framework 还实现了一个统一的、面向对象的、分层的和可扩展的类库，这些类库将常用的和不常用的任务分别封装成类，使用任意语言编写的任意的应用程序都可以调用这些类。

最重要的好处是.NET Compact 设备的编程模型与那些用.NET 为桌面 PC 和服务器生成应用程序的开发者所使用的编程模型是完全一样的。.NET Compact Framework 实现了.NET Framework 完整版的一个可兼容的子集，所以开发者使用同样的开发工具，利用同样的开发技巧，完成相似的任务。在.NET 以前，为 Windows CE 开发应用程序必须使用的工具集与为桌面环境开发时使用的工具集虽然有些相似，但是它们之间的不同却足以使一些公司拒绝在培训开发员工方面进行必要的投资。现在，桌面环境开发者可以很轻松地将他们的.NET 开发技术转移到为手持设备开发应用程序，因此就减少了培训开发人员的费用。

有些“正统”的软件开发者抱怨说.NET 使 Windows 的开发过程“毫无生趣”。其实不是这样的。.NET 确实使简单的事情更为简单了，但是它也使一些较为复杂的操作(如访问 Web 服务和生成组件)更易完成。您完全可以用节省出来的时间去开发一些更具有创新意义的软件解决方案。一些最具有创新意义的解决方案将是一种分布式的系统，具有运行在手持设备上的丰富的客户应用程序，在这些手持设备上处理的企业数据可以通过 Web 服务或 SQL Server CE 的数据集成功能来访问。随着成本的降低、性能的提高，以及软件开发过程的不断改善，手持设备即将成为企业应用程序的核心组件。

系统要求

要使用本书的示例文件，需要安装如下的软件：

- Microsoft Visual Studio .NET 2003
- Microsoft Windows 2000、Microsoft Windows XP 或 Microsoft .NET Server

开发 PC 的最低硬件配置应该是 Pentium II 级别的处理器，450 MHz 的 RAM、最小 128 MB (Windows 2000)或 256 MB (Windows XP 或 Windows .NET Server)的 RAM。要安装 Visual Studio .NET 2003，需要大约 5 GB 的硬盘空间。

Visual Studio .NET 2003 中提供了 Windows CE .NET 和 Pocket PC 2003 设备的仿真器，所以并不一定需要一个真实的手持设备。但是，如果想在真实的设备上测试应用程序，您需要下面的硬件和软件：

- Pocket PC 2000 或 2002 设备，或者 Windows CE .NET 4.1 或更高版本的设备
- ActiveSync 3.6 或更高版本(可从 <http://www.microsoft.com/mobile/downloads> 下载)
- Serial/USB 同步电缆或红外线，用以进行与桌面 PC 的同步。在该设备与您的开发 PC 建立了伙伴关系之后，如果您的设备支持，也可以使用以太网(或无线以太网)。

示例代码

本书中的大部分示例代码都是用 C#编写的。许多读者可能倾向于使用 Visual Basic，所以我们也用 Visual Basic .NET 实现了大部分示例，并将 C# 和 Visual Basic .NET 版本的示例文件都放在 Web 站点上供读者下载。希望 Visual Basic .NET 开发者在阅读本书时可以下载这些文件并参考这些文件。幸运的是，在 .NET 中，语言之间的区别再也不像以前那么明显了。C#和 Visual Basic .NET 应用程序在结构上是相同的，只是语法略有变化，所以介绍如何在 C#中使用某一编程技巧的内容对于使用 Visual Basic .NET 版本示例文件的读者来说，也是可以理解的。

可以在 <http://www.microsoft.com/mspress/books/5960.asp> 下载本书的示例文件。单击该页右侧的 More Information 框中的 Companion Content 链接可以打开 Companion Content 页。Companion Content 页中提供了将示例代码安装在您自己的 PC 上的操作指导。通过 Support: Book And CD 链接可以查看本书的勘误和更新。

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。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计算机应用编辑二室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	xvii
----------	------

第 I 部分 .NET Compact Framework 体系结构

第 1 章 .NET Compact Framework

介绍 3

1.1 智能移动计算设备简史 3

1.1.1 便携式计算的起始 4

1.1.2 PSION 发明了个人管理器 5

1.1.3 PDA 市场的发展 5

1.1.4 Microsoft 与手持设备 6

1.2 Microsoft .NET 7

1.2.1 .NET 出现之前的 COM 解决方案 7

1.2.2 .NET 是更好的 COM 10

1.2.3 这就是.NET 了吗 12

1.2.4 .NET 与移动性 12

1.3 .NET Compact Framework 目标平台 15

1.3.1 运行.NET Compact Framework 的平台 15

1.3.2 不支持.NET Compact Framework 的平台 15

1.4 开发工具 16

1.4.1 Visual Studio .NET 2003 16

1.4.2 eMbedded Visual Tools 第 3 版 17

1.4.3 eMbedded C++第 4 版 18

1.4.4 在 Visual Studio .NET 中 开始智能设备项目 19

1.4.5 管理项目 22

1.4.6 创建一个 Hello World 应用程序 24

1.4.7 使用仿真器 26

1.5 移动设备开发者所面临的 独特挑战 29

1.5.1 获得数据 29

1.5.2 资源受限的设备 30

1.5.3 安全模式 31

1.6 本章小结 32

第 2 章 .NET 框架基础 33

2.1 Compact Framework 执行环境 33

2.1.1 公共语言运行库 34

2.1.2 程序集 39

2.2 使用类对象 40

2.2.1 构造函数和终结器 40

2.2.2 用 IDisposable 释放资源 42

2.2.3 委托与事件 44

2.3 .NET Compact Framework 类库 48

2.3.1 类库结构 49

2.3.2 与 .NET Framework 完整版 的主要不同 55

2.4 本章小结 63

第 II 部分 使用 .NET Compact Framework 开发应用程序

第 3 章 使用 Windows 窗体进行

GUI 开发 67

3.1 System.Windows.Forms 命名空间 67

3.2 创建 Windows 窗体	68	4.8.1 为应用程序关联 一个图标	146
3.2.1 窗体设计器	68	4.8.2 在 Start 菜单中添加一个 快捷方式	146
3.2.2 Windows 窗体目标平台 的差别	68	4.9 一般性设计原则	148
3.3 .NET Framework 与 .NET Compact Framework 窗体行为的比较	69	4.10 本章小结	149
3.3.1 支持和不支持的控件	70	第 5 章 测试和调试	150
3.3.2 图形支持	72	5.1 发现并纠正编程错误	150
3.4 探讨 Windows 窗体控件	72	5.2 使用调试器	150
3.4.1 控件的创建	72	5.2.1 调试一个示例应用程序	151
3.4.2 控件的成员	73	5.2.2 理解默认的异常处理	153
3.4.3 .NET Compact Framework 控件	75	5.2.3 在调试器中处理异常	153
3.5 本章小结	129	5.3 使用调试器特性	154
第 4 章 构建 Windows 窗体 应用程序	130	5.3.1 进入中断模式	154
4.1 使用窗体	130	5.3.2 创建一个断点	155
4.1.1 设置 Form 类的属性	130	5.3.3 监视变量和表达式	155
4.1.2 遵循窗体显示原则	132	5.3.4 逐步执行一个应用程序	157
4.1.3 创建自定义的对话框	132	5.3.5 从中断模式继续一个 程序	157
4.1.4 使用内置的对话框	134	5.3.6 终止一个程序	157
4.2 使用 InputPanel 控件	136	5.3.7 删除断点	157
4.3 使用 MainMenu 控件	137	5.3.8 设置更为复杂的断点	157
4.4 使用 ToolBar 控件	139	5.3.9 开始执行(不调试)	160
4.5 在 Pocket PC 上编程实现 窗体的激活和停用	142	5.4 不支持的调试器特性	160
4.5.1 编程实现激活和停用	143	5.5 跟踪从应用程序得到的输出	160
4.5.2 编程实现窗体关闭	143	5.6 在应用程序内部使用断言	161
4.5.3 通过 Pocket PC Shell 关闭应用程序	144	5.7 在调试器中处理异常	161
4.6 处理轻击并保持事件	144	5.8 控制发布代码的大小	163
4.7 处理 Pocket PC 硬件键	145	5.9 在异常中生成更长的错误消息	164
4.8 为应用程序提供启动方式	146	5.10 测试和调试的策略	164
		5.11 从 Visual Studio 中配置仿真器	165
		5.12 手动地启动仿真器	167
		5.13 使用仿真器外观	168

5.13.1 使用映像文件	169	6.4 使用 Trickle-Feed 部署	193
5.13.2 仿真硬件按钮	171	6.5 将程序集安装到全局 程序集缓存中	194
5.13.3 用外观文件启动仿真器	172	6.5.1 理解强名称程序集	195
5.14 本章小结	173	6.5.2 使用并行的版本控制	197
第 6 章 完成并分发应用程序	174	6.6 本章小结	199
6.1 选择项目设置	174	第 III 部分 常用的编程任务	
6.1.1 使用文件的 Build Action 属性	174	第 7 章 使用集合类	203
6.1.2 将一个图标关联到 应用程序	175	7.1 .NET Compact Framework 集合类介绍	203
6.1.3 设置部署目录	175	7.2 使用简单的数组	205
6.1.4 应用程序私有程序 集的优点	176	7.2.1 初始化引用类型的数组	206
6.2 对应用程序进行打包	177	7.2.2 使用多维数组	207
6.2.1 创建针对处理器的 CAB 文件	177	7.3 使用 .NET Compact Framework 集合	209
6.2.2 使用 Visual Studio .NET 生成 CAB 文件	177	7.3.1 创建一个 ArrayList 集合	209
6.2.3 使用 CAB 向导生成 CAB 文件	178	7.3.2 使用 Hashtable 集合 存储键值对	210
6.2.4 使用资源文件生成器	179	7.3.3 使用 Stack 类	210
6.2.5 理解配置文件	179	7.3.4 使用 ListDictionary 集合 存储元素	211
6.3 分发应用程序	186	7.3.5 使用 HybridDictionary 集合	211
6.3.1 安装 .NET Compact Framework	187	7.3.6 使用 NameValueCollection 存储字符串键值对	211
6.3.2 从一个 Web 站点 进行部署	188	7.4 在集合中对对象进行排序	211
6.3.3 从一个文件共享 进行部署	188	7.4.1 使用 IComparable 进行排序	212
6.3.4 从另外的设备进行部署	189	7.4.2 用 IComparer 实现多个 排序顺序	213
6.3.5 使用内存存储卡 进行部署	189	7.5 创建您自己的集合	219
6.3.6 通过 ActiveSync 进行部署	190	7.6 本章小结	222

第 8 章 使用日期和字符串 223

- 8.1 理解 String 类 223
 - 8.1.1 使用字符串 223
 - 8.1.2 与 .NET Framework 完整版中的 String 类的区别 226
- 8.2 使用 StringBuilder 类 226
 - 8.2.1 使用 StringBuilder 的方法和属性 227
 - 8.2.2 与 .NET Framework 完整版中的 StringBuilder 类的区别 227
- 8.3 使用日期 228
 - 8.3.1 构造日期 228
 - 8.3.2 使用 DateTime 属性 228
 - 8.3.3 使用 DateTime 的算术方法 229
 - 8.3.4 使用 DateTime 的实用方法 229
 - 8.3.5 使用 DateTime 运算符 230
 - 8.3.6 与 .NET Framework 完整版的不同之处 230
- 8.4 使用正则表达式 230
 - 8.4.1 理解正则表达式 231
 - 8.4.2 使用简单的正则表达式匹配和 Match 对象 231
 - 8.4.3 用 MatchCollection 对象处理多个匹配 232
 - 8.4.4 使用分组和捕获 232
 - 8.4.5 替换文本模式 233
 - 8.4.6 使用 Split 方法 234
- 8.5 设置输出字符串的格式 234
 - 8.5.1 理解概念和术语 234
 - 8.5.2 使用格式说明符 235

- 8.5.3 使用格式规范和 String.Format 方法 241

- 8.5.4 理解格式提供程序 241

- 8.6 分析 243

- 8.6.1 解决分析中的问题 243

- 8.6.2 分析日期 244

- 8.7 本章小结 244

第 9 章 使用 XML 245

- 9.1 在 .NET Compact Framework 中使用 XML 246

- 9.2 与 .NET Framework 完整版的不同点 247

- 9.3 使用 XmlReader 类读取 XML 247

- 9.3.1 理解 XmlTextReader 类 247

- 9.3.2 构造 XmlTextReader 对象 247

- 9.3.3 用 XmlTextReader 类读取 XML 文档 248

- 9.3.4 不处理不必要的节点 250

- 9.3.5 读取属性节点 250

- 9.3.6 用 XmlTextReader 处理命名空间 252

- 9.3.7 处理验证、DTD 和实体 253

- 9.3.8 处理注释节点 253

- 9.3.9 处理 CDATA 节点 254

- 9.4 用 XmlWriter 类写入 XML 254

- 9.4.1 用 XmlTextWriter 类写入 XML 文档 254

- 9.4.2 构造 XmlTextWriter 255

- 9.4.3 写入 XML 数据 255

- 9.4.4 用 XmlTextWriter 处理命名空间 256

- 9.5 用 XmlDocument 类处理 XML 259

9.5.1 理解 XmlDocument 类	259	10.2.2 用 StreamReader 对象 读取文件流	295
9.5.2 与 .NET Framework 完整版 的不同点	260	10.2.3 在 StreamReader 和 StreamWriter 对象 中使用编码	296
9.5.3 用 XmlDocument 类 读取 XML	260	10.2.4 使用编码方式时指定 字节顺序	297
9.5.4 用 GetElementsByTagName 查找 XML 文档	266	10.2.5 使用 StreamWriter 写入 文件流	297
9.5.5 修改 XML 文档	267	10.3 用 StringReader 和 StringWriter 处理字符串	298
9.5.6 保存 XML 文档	271	10.4 使用 File、Path 和 Directory 类	298
9.6 本章小结	271	10.4.1 用 File 类创建文件	299
第 10 章 输入和输出	273	10.4.2 使用 Directory 类	300
10.1 理解 Stream 类	273	10.4.3 与 .NET Framework 完整版的不同	301
10.1.1 创建 FileStream 对象	274	10.4.4 使用 Path 类	301
10.1.2 为流指定文件访问模式	274	10.5 本章小结	302
10.1.3 为流指定文件共享模式	275	第 11 章 联网	303
10.1.4 使用 FileStream 对象 读取和写入字节	276	11.1 使用 System.Net 命名空间	303
10.1.5 使用 BinaryReader 和 BinaryWriter 类	276	11.2 使用 WebRequest 和 WebResponse 类	304
10.1.6 在流中查找	279	11.2.1 理解 WebRequest 和 WebResponse 的基本要点	304
10.1.7 锁定文件	279	11.2.2 使用 HTTP 获取资源	305
10.1.8 使用 NetworkStream 对象	279	11.2.3 处理错误	307
10.1.9 使用 NetworkStream 时处理网络错误	281	11.2.4 使用 HTTP 标头	308
10.1.10 对 NetworkStream 对象 执行异步读/写操作	281	11.2.5 网络传输编码	310
10.1.11 使用 MemoryStream 类	283	11.2.6 使用 HTTP GET 和 HTTP POST 向服务器 发送数据	311
10.1.12 产品订购应用 程序示例	286	11.2.7 cookie 的替代方法	313
10.2 使用 StreamReader 和 StreamWriter 类	295		
10.2.1 创建 StreamReader 和 StreamWriter 对象	295		

11.2.8	使用 SSL 确保网络传输的安全.....	315	12.3.1	在 .NET Compact Framework 应用程序中提供登录凭据.....	356
11.2.9	使用 NetworkCredential 类验证请求.....	316	12.3.2	为移动用户授予适当的权限.....	357
11.2.10	异步 Web 请求	317	12.4	确保 XML Web 服务的安全.....	357
11.2.11	通过代理服务器进行网络请求	323	12.5	确保 SQL Server CE 连接的安全	362
11.3	判断设备是否连接	323	12.6	确保数据在设备上的安全	363
11.4	使用套接字.....	324	12.6.1	在手持设备上使用加密.....	363
11.4.1	了解使用仿真器和 ActiveSync 的 IP 网络	324	12.6.2	确保 SQL Server CE 数据库的安全	363
11.4.2	执行 DNS 查找.....	326	12.7	编程设计 .NET Compact Framework 的安全性	365
11.4.3	使用 System.Net.Sockets.TcpListener	326	12.7.1	理解 .NET Compact Framework 的安全性策略.....	366
11.4.4	使用 System.Net.Sockets.TcpClient	328	12.7.2	使用安全编码惯例	366
11.4.5	使用 System.Net.Sockets.UdpClient.....	333	12.7.3	在代码中存储用户名和密码	367
11.4.6	使用 Socket 类.....	334	12.7.4	避免恶意下载.....	368
11.4.7	红外线通信	336	12.8	本章小结.....	369
11.5	串行通信	342			
11.6	本章小结	348			
第 12 章 .NET Compact Framework 应用程序安全性.....			第 IV 部分 连接数据		
12.1	手持设备安全性的基础.....	350	第 13 章 访问数据.....		
12.1.1	用户验证	350	13.1	缓存、复制更改和解决冲突	373
12.1.2	使用杀毒软件.....	351	13.1.1	缓存	374
12.1.3	使用防火墙	351	13.1.2	复制更改	374
12.2	确保无线数据传输的安全性.....	351	13.1.3	解决冲突	375
12.2.1	使用虚拟专有网络	352	13.2	避免复制.....	375
12.2.2	使用 SSL	353	13.3	在网络大部分时间断开的解决方案中同步数据	376
12.2.3	防止欺骗	354	13.3.1	使用 ActiveSync 文件复制进行同步	376
12.3	验证和授权移动客户	354			

13.3.2 使用 SQL Server CE RDA 进行同步	378	14.3.3 对 DataRow 对象进行 读取、更新或删除操作	391
13.3.3 使用 SQL Server 2000 合并复制进行同步	379	14.3.4 查找 DataRow 对象	391
13.4 在大部分时间都连接的解决 方案中同步数据	380	14.3.5 查看 DataSet 对象的 XML 表示	392
13.4.1 直接访问 SQL Server 2000	380	14.3.6 使用类型化的 DataSet 对象	393
13.4.2 实现移动 Web 应用 程序	381	14.3.7 使用 DataSet Designer 创建 DataSet 对象	393
13.4.3 使用 Web 服务	381	14.3.8 加载存储的架构	394
13.5 使用其他网络解决方案	382	14.4 使用多表 DataSet 对象	395
13.5.1 使用套接字	382	14.4.1 在一个多表 DataSet 对象 中定义关系	395
13.5.2 使用红外线	383	14.4.2 在 DataRelation 对象中 设置约束	398
13.6 本章小结	384	14.5 在本地保存 DataSet 对象	399
第 14 章 ADO.NET 数据对象	385	14.5.1 写入数据	400
14.1 ADO.NET 的优点	385	14.5.2 读取数据	400
14.1.1 ADO.NET 是无连接的	385	14.5.3 综合使用所有的技术	401
14.1.2 ADO.NET 以 XML 为中心	386	14.6 使用更新、数据视图和 数据绑定	404
14.1.3 ADO.NET 允许 数据绑定	386	14.6.1 跟踪 DataSet 更新	404
14.2 引入 ADO.NET 对象	386	14.6.2 使用 DataView 对象	405
14.2.1 理解 DataSet 对象	387	14.6.3 数据绑定	407
14.2.2 表示数据表	388	14.7 本章小结	411
14.2.3 用 DataRelation 对象 定义关系	388	第 15 章 与 SQL Server 集成数据	412
14.2.4 理解 DataView 对象	388	15.1 分析数据访问体系结构	412
14.3 从零开始构建一个 DataSet 对象	389	15.1.1 使用 SQL Server 2000 托管提供程序	412
14.3.1 添加 DataSet、DataTable 和 DataColumn 对象	390	15.1.2 与桌面版 .NET Framework 的差异	413
14.3.2 添加 DataRow 对象	390	15.1.3 System.Data.SqlClient 命名空间	414

15.2 检索数据	416	16.2.3 XML Web 服务描述	442
15.2.1 创建 SqlConnection 对象	417	16.2.4 XML Web 服务 连网格式	443
15.2.2 使用 SqlCommand 对象	419	16.3 XML Web 服务的运行概况	444
15.2.3 使用 SqlDataReader 对象	420	16.3.1 XML Web 服务客户端	444
15.2.4 使用 SqlException 对象	421	16.3.2 运行时的 Web 服务	445
15.3 绑定数据	421	16.4 Compact Framework 的限制	445
15.4 更新数据	425	16.5 一个简单的 XML Web 服务	446
15.4.1 更新被绑定数据	426	16.6 这个 XML Web 服务的 客户程序	449
15.4.2 使用 SqlCommand- Builder	428	16.6.1 Web 引用	449
15.4.3 处理更新过程	429	16.6.2 生成代理	451
15.5 使用 SQL Server 存储过程	431	16.6.3 代理类允许动态 Web 服务 URL	451
15.5.1 在 .NET Compact Framework 中使用事务	432	16.6.4 使用 XML Web 服务	451
15.5.2 使用获取数据的 存储过程	432	16.7 返回数据集的 Web 服务	452
15.5.3 使用带参数的 存储过程	435	16.7.1 TimesheetService Web 服务	453
15.5.4 执行不返回数据的 存储过程	436	16.7.2 使用类型化数据集的 XML Web 服务客户程序	455
15.6 SQLXML	436	16.7.3 修改 Web 服务使用 标准数据集	455
15.6.1 在应用程序中使用 SQLXML	437	16.7.4 使用标准数据集的 Web 服务客户程序	456
15.6.2 其他阅读材料	438	16.7.5 异步操作	457
15.7 本章小结	438	16.8 返回数据集进行更新	458
第 16 章 使用 XML Web 服务	439	16.9 设计 XML Web 服务时需要 考虑的问题	459
16.1 XML Web 服务的体系结构	439	16.10 本章小结	460
16.2 XML Web 服务的基础结构	440	第 17 章 使用 SQL Server CE	461
16.2.1 XML Web 服务目录	441	17.1 在智能设备上使用 SQL Server	461
16.2.2 XML Web 服务发现	441	17.2 编制 SQL Server CE 应用程序	464
		17.2.1 SqlServerCe 数据编程 对象概述	466