<<C#与.NET 3.5高级程序设计>>

图书基本信息

书名：<<C#与.NET 3.5高级程序设计>>

13位ISBN编号：9787115196910

10位ISBN编号：7115196915

出版时间：2009-3

出版时间：人民邮电出版社

作者：特罗尔森

页数：1107

前言

~应用程序开发的目标始终如一，就是在最短时间内制作出最好的软件。
当今最流行的快速开发平台是什么，毫无疑问，是微软公司的.NET Framework。
经过近8年的发展，微软的.NET已经变得非常庞大和成熟，而且.NET发展的步伐越来越快。
2002年年初，微软发布了Visual Studio.NET 2002开发工具，也带来了.NET框架的第一个版本1.0。
2003年春天，微软发布了Visual Studio.NET 2003以及.NET Framework 1.1。
尽管只是1.1，但是.NET Framework 1.1比1.0多了几项重大更新，例如对移动设备的支持（精简版.NET）和对ODBC／Oracle数据库的支持。
2005年年底，.NET Framework 2.0随着Visual Studio 2005以及SQL Server2005一起发布。
无论是Visual Studio 2005、C#2.0还是ASP.NET 2.0，在易用性、安全性等方面都比前辈优秀很多。
C{fj}2.0中的泛型、迭代器、可空类型、匿名方法、分部类等特性确实给了我们很多方便。
2006年年底，微软随Vista操作系统推出了.NET Framework 3.0。
.NET Framework 3.0在2.0的基础上增加了几个重要组件：WCF、WPF、WF和WCS（本书厉面会对WCF、WPF~~HWF进行比较详细的介绍）。
WCF：Windows Communication Foundation，最完整的集成化通信管理框架。
WCF编程模型把Web服务、.NET远程技术、分布式事务和消息队列统一到单个面向服务的编程模型中，从而实现了真正意义上的分布式计算。
WPF：Windows Presentation Foundation，最华丽的界面技术框架。
WPF提供了一种一致的方案来构建编程模型，并且支持使用更为丰富的控件和设计技术来开发Windows程序。
WF：Windows Workflow Foundation，最灵活和最强大的工作流平台。
WF是微软用来定义、执行和管理工作流的编程模型、引擎和工具的总称。
WCS：Windows CardSpace，最安全的个人数字标识解决方案。
2007年年底，微软发布了Visual Studio 2008以及.NET Framework 3.5。
在.NET Framework 3.0的基础上，3.5又新增了一些程序集，并引入了LrNQ、ASP.NET AJAX等功能（本书基于C≠}3.0，而且也会对LINQ技术进行简单介绍）。
c#3.0：提供了隐式类型变量、自动属性、扩展方法、对象初始化语法、Lambda表达式、匿名方法、分部方法等新特性，其实几乎所有的新特性都是因LINQ而生的。
LINQ：语言集成查询，官方组件包括LINQ to SQL、LINQ to XML、LINQ to DataSet等。
当然，还有很多第三方公司或个人推出的LINQ to XXX。
可以通过这些组件使用统一的查询语言对各种基于磁盘、网络甚至服务的数据源进行查询。
ASP.NET AJAX：其实ASP.NET AJAX最早并不是在.NET Framework 3.5中出现的，只不过.NET Framework 3.5集成了ASP.NET AJAX。
关于.NET Framework 3.0中四大技术的扩展，我们也可以认为.NET Framework 3.5提供了WCF／WPF／WF／WCS的2.0版本。
~

<<C#与.NET 3.5高级程序设计>>

内容概要

《C#与.NET 3.5高级程序设计》(第4版)是C#领域久负盛名的经典著作，深入全面地叙述了C#编程语言和.NET平台核心，并以大量示例剖析相关概念。
书中介绍了C#的各种语言构造、.NET 2.0的类、核心API、公共中间语言(CIL)、动态程序集和ASP.NET扩展等内容；同时也介绍了.NET 3.0和.NET 3.5中新的编程API，包括WPF、WCF和WF的功能；另外，还介绍了最新的C# 3.0编程语言、LINQ编程技术、COM与.NET的互操作性以及平台无关的.NET开发。

作者简介

作者：(美国)特罗尔森 (Andrew troelsen) 译者：朱晔 肖逵 张大磊AndFew Troelsen，世界级c#专家。
微软；Visual C#MVP，他是著名的微软技术咨询企业Intertech的合伙人和副总裁。
该公司的客户包括微软、霍尼韦尔、美国宇航局等。
他曾为MSDN网站和MacTech网站撰写了有关各种操作系统平台上．NET技术的文章，并经常在业界主要技术会议上发表演讲和开设技术讲座。
除本书外，他还撰写了COM and．NET Interoperability和Visual Basic.NET andthe．NET Platform：An Advanced Guide等十多部．NET技术方面的著作。
译者简介：朱晔，微软ASP.NET MVP，熟悉.NET、c#、ASP.NET、SQL Server等技术。
擅长基于.NET的分布式电子商务网站、互联网网站和网游平台的架构设计。
近期出版了畅销图书《ASP.NET第一步——基于c#和ASP.NET 2.0》和译著《SQL server 2005范例代码查询辞典》等。
个人博客：http：//www.cnblogs.com/lovecherry。
张大磊，超过10年的软件研发实践，在网络安全、地理信息、医疗、生产制造和商业智能等多个领域积累了较为丰富的经验。
近年来主要关注微软平台技术，曾任微软公司开发技术经理。
业余时间与大中华区各地的微软技术架构师和最有价值专家进行广泛的技术交流，并多次受邀在TechED、MSDN等技术活动中发表主题演讲。
也是多个开源项目的贡献者，多个在线社区的特邀技术专家与微软金牌讲师的评委。
可以通过http：\\kmsdpe.enblogs.com与他联系。
肖逵资深软件开发工程师，高级技术讲师，毕业后一直效力于公司核心开发部门，主持研发数个大型企业级软件产品，对.NET Framework及相关技术有深入研究，曾就职于HoneyWell、HP等公司。
王少葵，2004～2008年四届微软Visual Developer.Visual C#方面MVP，通过MCP、MCDBA、MCSA、MCAD、MCSD等多项微软认证，有10多年IT行业从业经验，现为ABB（中国）有限公司金属部高级工程师。
主要为生产制造企业提供自动控制整体解决方案，与微软、宝信、用友等软件公司有良好的合作。
曾主持和参与宝钢、太钢、攀钢、酒钢、涟钢等多个企业的自动化控制、MES、ERP等若干大型系统设计与研发，涉及办公自动化、物流、数据存储、计算机通信、人机接口、模型计算等诸多领域。
活跃在CSDN社区，致力于研究.NET新技术。
范睿，（网名：fancyf）有5年.NET框架编程经验，超过6年使用C／C++的程序开发经验，对.NET框架有比较深入的研究。
经常活跃在CSDN社区，热心解答网友们的问题，致力于宣传与推广微软.NET技术的应用。
曾多次受邀进行有关.NET技术的讲座和培训。

<<C#与.NET 3.5高级程序设计>>

书籍目录

章节摘录

当初步了解.NET的语言无关性后，开发者会提出许多问题。
其中最普遍的问题可能就是：“ 如果所有的.NET语言都会编译成‘ 托管代码 ’，为什么我们还需要多种编译器呢？
”这个问题有多个答案。
首先，程序员在选择编程语言时有各自不同的喜好（包括我自己）。
一些人喜欢充满分号和圆括号而且关键字相当少的语言；另一些人喜欢更具有“ 可读性 ”语法标记的语言（如EIVB）；还有一些人在开始转向.NET平台时还希望可以使用他们已掌握的技能（通过COBOl.NET）。
现在，平心而论，如果微软推出一门派生自BASIC语言系列的“ 官方 ”.NET语言，你认为所有的程序员会喜欢这样的选择吗？
或者，如果这个唯一的“ 官方 ”.NET语言是基于Fortran语法的，那么可以想象所有人都会对.NET置之不理。
因为.NET运行库并不在意一段托管代码是由哪种语言生成的，所以.NET程序员可以继续使用他们熟悉的语法，且与组员、部门甚至其他公司共享编译的程序集（不管他们用的是哪种.NET语言）。
将各种.NET语言集成为一个统一软件方案的另一个好处，就是能够取长补短。
所有的编程语言都有各自的优点和缺点。
例如，一些编程语言对高级的数学处理有相当完美的内在支持能力。
另一些则精于支持财务计算、逻辑计算和与大型机交互等。
当你学习到某种编程语言的优点并将其融合于.NET平台时，大家就都能受益。
当然，实际上我们大部分时间还是在用自己习惯的.NET语言来编写程序。
但是，一旦学会了一种.NET语言的语法，就很容易掌握其他的了。
这是非常有益的，对软件技术顾问而言尤其如此。
如果你熟悉C#，在为只使用Visual Basic.NET的客户做咨询时，你仍然能够使用.NET Framework的功能，并且可以毫不费力地掌握代码的整体结构。
够棒的吧。
1.5.NET程序集概览不管选择了哪种.NET语言编程，需要明白的是，尽管.NET二进制文件COM服务器和非托管Win32二进制文件（*.dll或*.exe）具有相同的文件扩展名，但它们的内部却是完全不同的。
例如，*.dll的.NET二进制文件不会导出与COM运行库进行通信的方法（因为.NET不是COM）。
此外，.NET二进制文件不使用COM类型库文件描述而且不用在系统注册表中注册。
也许更重要的是，.NET二进制文件不包含特定于平台的指令，它包含的是平台无关的IL（中间语言）和类型元数据。
图1.3清楚显示了这个流程。
插图：

<<C#与.NET 3.5高级程序设计>>

编辑推荐

《C#与.NET 3.5高级程序设计》(第4版)由微软C# MVP Andrew Troelsen编写，历经多次修订，适合各层次.NET开发人员阅读。
Amazon超级畅销书，全面涵盖.c#3.O，用IL深入揭示各语言特性，深度剖析.NET 3.5平台。
让你知其然，更知其所以然，国内多位微软MVP联手翻译。
C#语言作为．NET平台上的第一语言．已经成为目前功能最强大的通用语言之一。
《C#与.NET 3.5高级程序设计(第4版)》是被誉为“ C样圣经 ”的经典巨著，因语言生动流畅、剖析深入、涵盖全面而广受推崇。
畅销不衰。
曾经获得Referenceware编程图书大奖。
并入选Jolt大奖提名。
书中探讨了C#语言和．NET平台的各种特性。
包括重载运算符、指针、泛型等高级功能和CIt、远程处理、Windows Forms、ASP．NET、ADO．NET等技术。
不少概念都通过lL代码透视其背后的本质，使你知其然。
更知其所以然。
新版更透彻阐述了C#3，0新功能（包括自动属性、扩展方法、匿名类型等）和．NET 3．5的最新特性（包括LINQ、WPF、WCF和WF等相关技术）。
附录中包括了COM与．NET的互操作和Mono发等主题。
与同类图书不同。
全书由世界级C#专家Andrew Troelsen以一人之力完成．因此写作思路和布局谋篇都独具匠心。
中文版由国内多位微软MVP联手译出。
强大的译者阵容有力地保证了这部权威原著原汁原味的重现。

版权说明