第一图书网, tushu007.com

<<.NET探秘>>

图书基本信息

书名：<<.NET探秘>>

13位ISBN编号：9787115201768

10位ISBN编号：7115201765

出版时间：2009-9

出版时间：人民邮电出版社

作者：利丁

页数：421

译者：包建强

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：http://www.tushu007.com

前言

不要惊讶，坐在你面前娓娓而谈的，并不是一个技术Geek，而是流连于淮海路迤逦风光的性情中人。
编程只是我诸多业余爱好中的一项。
当然，有时会稍微热情些，比如说，从狄更斯的《双城记》中能联想到设计模式的精华。
一直在酝酿一篇有些不同的序言，即开头没有诸多名人的溢美之词，结尾也没有感谢父母家人的肺腑之言，我只想谈一谈阅读本书时所要注意的问题，以及本书的不足之处。
作者Serge Lidin是位慈眉善目的老伯，我们曾坐在微软总部的长椅上攀谈良久。
他曾经是微软IL小组的成员，所写的内容都是第一手资料，所以写起本书来当仁不让，章节划分相当合理。
但是，正如作者所说，他不善笔墨，所以本书读起来非常枯燥。
当然，CLR底层本身就是非常枯燥的，要把它写得精彩生动，很难做到。
除非借助C#来讲解IL和元数据，才会吸引人，如某位印度作家编著的C#to IL。
建议大家先去阅读后面这本书，它只是基于．NET 1．1的，没有涉及泛型，内容没有前者丰富，但是，它要简单得多。
言归正传。
本书是学习和研究IL和元数据的首选图书，它可以作为一本技术参考手册，睡觉前翻翻也是不错的。
本书并没有介绍任何IL性能调优的方法，而这正是开发人员所关心的，这不能不说是一个遗憾。
IL和元数据，也就是PE结构体，是静态的内容。
对于运行期间的动态执行过程，完全不是这么回事。
本书没有涉及后面这部分内容，毕竟，这超出了本书的范围。
有兴趣的读者可以去研究Rotor，可以参考SharedSource CLI Essentials一书。
作者非常认真负责。
我们书信往来半年之久，修正了原书中的一些错误。
同时，统一了.NET和汇编语言中的中文术语，其中，参考了侯捷先生和李建忠先生译作中的诸多神来之笔。
此外，还在字里行间添加了大量的译注，有助于读者理解。

第一图书网, tushu007.com

内容概要

本书是IL汇编语言的权威之作，深入地讲解了IL的全部内容，是.NET底层开发人员必备的参考书，对于从概念设计到实现和维护等软件开发所有阶段都很有价值。
本书内容基于.NET 2.0版本，同时，考虑到.NET 3.0/3.5在CLR/IL上没有作任何改动，且即将推出的.NET 4.0也只是在CLR上稍作修改而并没有涉及IL语言，所以，本书对于目前各个版本的.NET Framework而言都是适用的。
本书适合所有.NET开发人员、讲师、研究人员阅读。

作者简介

作者：(加拿大)利丁(Serge Lidin) 译者：包建强Serge Lidin，曾是．NET CLR小组核心开发人员之一，是MSlL语言的设计者以及该语言规范的作者之一，负责设计并开发了IL汇编器、IL反汇编器等核心组件。
目前任职于微软Phoenix小组。
开发新一代编译技术底层框架。
包建强，复旦人。
微软2008年MVP。
现就职于HP某航空项目，从事Sllverlight开发和Prism框架设计工作。
技术博客：http：／／jax．cnblogs．corn／。

书籍目录

第一部分　快速入门　第1章　简单示例　1.1　CLR基础　1.2　简单示例：The Code　1.2.1　程序头　1.2.2　类声明　1.2.3　字段声明　1.2.4　方法声明　1.2.5　全局项　1.2.6　映射字段　1.2.7　数据声明　1.2.8　作为占位符的值类型　1.2.9　调用非托管代码　1.3　类的预先声明　1.4　小结　第2章　代码增强　2.1　精简代码　2.2　保护代码　2.3　小结　第3章　使代码更简单　3.1　别名　3.2　编译控制伪指令　3.3　关联当前类及其引用项　3.4　小结　第二部分　底层结构　第4章　托管可执行文件的结构　4.1　PE/COFF头　4.1.1　MS-DOS头/Stub和PE签名　4.1.2　COFF头　4.1.3　PE头　4.1.4　节头　4.2　CLR头　4.2.1　头结构　4.2.2　Flags字段　4.2.3　EntryPointToken字段　4.2.4　VTableFixups字段　4.2.5　StrongNameSignature字段　4.2.6　重定位节　4.2.7　文本节　4.2.8　数据节　4.2.9　数据常量　4.2.10　V表　4.2.11　非托管导出表　4.2.12　线程局部存储(TLS)　4.2.13　资源　4.2.14　非托管资源　4.2.15　托管资源　4.3　小结　4.3.1　第一阶段：初始化　4.3.2　第二阶段：源代码解析　4.3.3　第三阶段：映像生成　4.3.4　第四阶段：完成　第5章　元数据表的组织　5.1　什么是元数据　5.2　堆和表　5.2.1　堆　5.2.2　通用元数据头　5.2.3　元数据表流　5.3　RID和标记　5.3.1　RID　5.3.2　标记　5.3.3　编码标记　5.4　元数据有效性检查　5.5　小结　第三部分　基本组件　第6章　模块和程序集　6.1　什么是程序集　6.2　私有程序集和共享程序集　6.3　作为逻辑执行单元的应用程序域　6.4　清单　6.5　Assembly元数据表和声明　6.6　AssemblyRef元数据表和声明　6.7　加载程序搜索程序集　6.8　Module元数据表和声明　6.9　ModuleRef元数据表和声明　6.10　File元数据表和声明　6.11　托管资源元数据表和声明　6.12　ExportedType元数据表和声明　6.13　ILAsm中清单声明的次序　6.14　单模块程序集和多模块程序集　6.15　元数据有效性规则概要　6.15.1　Assembly表有效性规则　6.15.2　AssemblyRef表有效性规则　6.15.3　Module表有效性规则　6.15.4　ModuleRef表有效性规则　6.15.5　File表有效性规则　6.15.6　ManifestResource表有效性规则　6.15.7　ExportedType表有效性规则　第7章　命名空间和类　7.1　类的元数据　7.1.1　TypeDef元数据表　7.1.2　TypeRef元数据表　7.1.3　InterfaceImpt元数据表　7.1.4　NestedClass元数据表　7.1.5　ClassLayout元数据表　7.2　命名空间和类的完整名称　7.2.1　ILAsm命名约定　7.2.2　命名空间　7.2.3　类的完整名称　7.3　类的特性　7.3.1　标志　7.3.2　类的可见性和友元程序集　7.3.3　类的引用　7.3.4　父类型　7.3.5　接口实现　7.3.6　类的布局信息　7.4　接口　7.5　值类型　7.5.1　值的装箱和拆箱　7.5.2　值类型的实例成员　7.5.3　值类型的派生　7.6　枚举　7.7　委托　7.8　嵌套类型　7.9　类的扩充　7.10　元数据有效性规则概要　7.10.1　TypeDef表有效性规则　7.10.2　特定于枚举的有效性规则　7.10.3　TypeRef表有效性规则　7.10.4　InterfaceImpt表有效性规则　7.10.5　NestedClass表有效性规则　7.10.6　ClassLayout表有效性规则　第8章　基本类型和签名　8.1　CLR中的基本类型　8.1.1　基本数据类型　8.1.2　数据指针类型　8.1.3　函数指针类型　8.1.4　向量和数组　8.1.5　修饰符　8.1.6　本地类型　8.1.7　可变类型　8.2　签名中类的表示　8.3　签名　8.3.1　调用约定　8.3.2　字段签名　8.3.3　方法和属性签名　8.3.4　MemberRef签名　8.3.5　间接调用签名　8.3.6　局部变量签名　8.3.7　类型说明　8.4　签名有效性规则概要　第9章　字段和数据常量　第10章　方法　第11章　泛型类型　第12章　泛型方法　第四部分　深入执行引擎　第13章　IL指令　第14章　托管异常处理　第五部分　特殊组件　第15章　事件和属性　第16章　自定义特性　第17章　安全特性　第18章　托管代码和非托管代码的互操作　第19章　多语言工程　第六部分　附录　附录A　ILAsm语法　附录B　元数据表　附录C　IL指令集　附录D　IL汇编器和IL反汇编器的命令行选项　附

第一图书网, tushu007.com

录E　离线验证工具　部分术语翻译说明

章节摘录

插图：第1章简单示例本章概要介绍几Asm——MsIL汇编语言。
（Ms几表示微软中间语言，本章稍后将会讨论。
）本章先讨论一个相对简单的ILAsm程序，随后提出了一些修改意见，展示了如何用ILAsm表示.NET程序设计中的概念和元素。
本章并没有讲如何用ILAsm编写程序，但是会帮助用户理解IL汇编器（ILASM）和IL反汇编器（ILDASM）做了什么，以及如何通过这些理解并借助这些工具来分析.NET程序的内部结构。
你还将了解到一些发生在CLR中的神秘而有趣的事件——相当吸引人，希望这能促使你读完本书的其他部分。
注解简单起见，在本书中将IL汇编语言简写ILAsm。
不要将其与ILAsM相混淆，后者是IL汇编器（也就是ILAsm汇编器）在.NET文档中的简写形式。
1.1CLR基础.NETcLR不仅是.NET诸多概念中的一个，更是.NET的核心。
（注意，有时候我会将CLR称为运行库。
）我们将重点关注.NETd?真正发生行为的部分——CLR，而不是.NET平台的完全描述。

媒体关注与评论

想真正理解．NET框架的内部机理吗？
阅读本书吧。
”　　——Gunnar Peipman。
微软MVP“凭借本书和微软的规范、文档，我们编写IL代码生成器是如此美妙过瘾！
”　　——Robert Salita，资深编译器工程师

编辑推荐

《.NET探秘MSIL权威指南》由微软资深工程师、MSIL语言设计者之一撰写，详尽而深入地讨论了．NET CLR的内部结构和运作。
首次阐述了MSIL语言的方方面面，是极为宝贵的．NET技术资料。
通过研读《.NET探秘MSIL权威指南》，你将大大加深自己对．NET框架的理解，从而开发出更加紧凑、更高性能的．NET代码。
．NET框架已经是微软技术的核心平台，但是，对于许多开发人员而言，它仍然是黑盒子。
遇到各种实际的性能、调试等问题时很容易让人陷入不知所措的境地。
深入洞悉．NET技术内幕，MSIL唯一权威指南，Amazon五星巨著，学习和参考皆宜。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:http://www.tushu007.com