

# 《C#高级编程(第9版)——C# 5.》

## 图书基本信息

书名：《C#高级编程(第9版)——C# 5.0 &.NET 4.5.1》

13位ISBN编号：9787302380023

出版时间：2014-10-1

作者：(美)内格尔 (Nagel,C.) 等著

页数：1552

译者：李铭 译

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu111.com](http://www.tushu111.com)

## 内容概要

本书由.NET专家的梦幻组合编写，包含开发人员使用C#所需的所有内容。C#是编写.NET应用程序的一种语言，本书适合于希望提高编程技巧的、有经验的C#程序员，也适用于刚开始使用C#的专业开发人员。本书探讨了Visual Studio 2013和.NET Framework 4.5.1、新的测试驱动开发和并发编程功能。所有示例的源代码都可以下载，读者可以立即开始编写Windows桌面应用程序、Windows Store应用程序和ASP.NET Web应用程序。

## 作者简介

Christian Nagel是Microsoft RD、Microsoft MVP、thinktecture的合作伙伴、CN革新技术的奠基人，他还是一位软件架构师和开发人员，为开发Microsoft .NET解决方案提供培训和咨询服务。他具备超过25年的软件开发经验。Christian从PDP 11和VAX/VMS系统开始其计算机生涯，熟悉各种语言和平台。他具备Microsoft技术的深厚功底，编写了大量图书，并获得了Microsoft认证培训师和专业开发人员证书。Jay Glynn开发软件的时间超过20年，使用PICK Basic为PICK操作系统编写应用程序。到目前为止，他使用过Delphi、VBA、Visual Basic、C、Java和C#编写软件。他目前是VGT的高级软件工程师，编写基于Web的应用程序。

Morgan Skinner是一位自由顾问，他在开始自己的顾问生涯之前，在Microsoft工作了将近10年。

## 书籍目录

### 第 部分 C# 语言

#### 第1章 .NET体系结构

2

##### 1.1 C#与.NET的关系

2

##### 1.2 公共语言运行库

3

###### 1.2.1 平台无关性

3

###### 1.2.2 提高性能

3

###### 1.2.3 语言的互操作性

4

##### 1.3 中间语言

6

###### 1.3.1 面向对象和接口的支持

6

###### 1.3.2 不同的值类型和引用类型

7

###### 1.3.3 强数据类型化

7

###### 1.3.4 通过异常处理错误

12

###### 1.3.5 特性的使用

12

##### 1.4 程序集

12

###### 1.4.1 私有程序集

13

###### 1.4.2 共享程序集

13

###### 1.4.3 反射

14

###### 1.4.4 并行编程

14

###### 1.4.5 异步编程

14

##### 1.5 .NET Framework类

15

##### 1.6 名称空间

15

##### 1.7 用C#创建.NET应用程序

16

###### 1.7.1 创建ASP.NET应用程序

16

###### 1.7.2 使用WPF

17

1.7.3 Windows Store应用程序	18
1.7.4 Windows服务	18
1.7.5 WCF	18
1.7.6 Windows WF	19
1.8 C#在.NET企业体系结构中的作用	19
1.9 小结	20
第2章 核心C#	22
2.1 C#基础	23
2.2 第一个C#程序	23
2.2.1 代码	23
2.2.2 编译并运行程序	23
2.2.3 详细介绍	24
2.3 变量	26
2.3.1 变量的初始化	26
2.3.2 类型推断	27
2.3.3 变量的作用域	28
2.3.4 常量	30
2.4 预定义数据类型	31
2.4.1 值类型和引用类型	31
2.4.2 CTS类型	32
2.4.3 预定义的值类型	32
2.4.4 预定义的引用类型	35
2.5 流控制	37
2.5.1 条件语句	37
2.5.2 循环	

40	
2.5.3	跳转语句
43	
2.6	枚举
44	
2.7	名称空间
46	
2.7.1	using语句
47	
2.7.2	名称空间的别名
48	
2.8	Main()方法
48	
2.8.1	多个Main()方法
49	
2.8.2	给Main()方法传递参数
50	
2.9	有关编译C#文件的更多内容
50	
2.10	控制台I/O
52	
2.11	使用注释
54	
2.11.1	源文件中的内部注释
54	
2.11.2	XML文档
54	
2.12	C#预处理器指令
56	
2.12.1	#define和#undef
57	
2.12.2	#if、#elif、#else和#endif
57	
2.12.3	#warning和# error
58	
2.12.4	#region和#endregion
58	
2.12.5	#line
59	
2.12.6	#pragma
59	
2.13	C#编程规则
59	
2.13.1	关于标识符的规则
59	
2.13.2	用法约定
60	
2.14	小结
66	

## 第3章 对象和类型

67

### 3.1 创建及使用类

67

### 3.2 类和结构

68

### 3.3 类

69

#### 3.3.1 数据成员

69

#### 3.3.2 函数成员

69

#### 3.3.3 只读字段

81

### 3.4 匿名类型

82

### 3.5 结构

82

#### 3.5.1 结构是值类型

84

#### 3.5.2 结构和继承

84

#### 3.5.3 结构的构造函数

85

### 3.6 弱引用

85

### 3.7 部分类

86

### 3.8 静态类

87

### 3.9 Object类

88

#### 3.9.1 System.Object()方法

88

#### 3.9.2 ToString()方法

89

### 3.10 扩展方法

90

### 3.11 小结

91

## 第4章 继承

92

### 4.1 继承

92

### 4.2 继承的类型

92

#### 4.2.1 实现继承和接口继承

92

#### 4.2.2 多重继承

93	
4.2.3	结构和类
93	
4.3	实现继承
93	
4.3.1	虚方法
94	
4.3.2	隐藏方法
95	
4.3.3	调用函数的基类版本
96	
4.3.4	抽象类和抽象函数
97	
4.3.5	密封类和密封方法
97	
4.3.6	派生类的构造函数
98	
4.4	修饰符
102	
4.4.1	可见性修饰符
103	
4.4.2	其他修饰符
103	
4.5	接口
104	
4.5.1	定义和实现接口
105	
4.5.2	派生的接口
108	
4.6	小结
109	
第5章	泛型
110	
5.1	泛型概述
110	
5.1.1	性能
111	
5.1.2	类型安全
112	
5.1.3	二进制代码的重用
112	
5.1.4	代码的扩展
113	
5.1.5	命名约定
113	
5.2	创建泛型类
113	
5.3	泛型类的功能
117	



5.3.1 默认值	118
5.3.2 约束	118
5.3.3 继承	120
5.3.4 静态成员	121
5.4 泛型接口	122
5.4.1 协变和抗变	122
5.4.2 泛型接口的协变	123
5.4.3 泛型接口的抗变	125
5.5 泛型结构	125
5.6 泛型方法	128
5.6.1 泛型方法示例	128
5.6.2 带约束的泛型方法	129
5.6.3 带委托的泛型方法	130
5.6.4 泛型方法规范	131
5.7 小结	132
第6章 数组	133
6.1 同一类型和不同类型的多个对象	133
6.2 简单数组	134
6.2.1 数组的声明	134
6.2.2 数组的初始化	134
6.2.3 访问数组元素	135
6.2.4 使用引用类型	136
6.3 多维数组	137
6.4 锯齿数组	138
6.5 Array类	

139	
6.5.1 创建数组	139
6.5.2 复制数组	140
6.5.3 排序	141
6.6 数组作为参数	144
6.6.1 数组协变	144
6.6.2 ArraySegment<T>	144
6.7 枚举	145
6.7.1 IEnumerator接口	146
6.7.2 foreach语句	146
6.7.3 yield语句	147
6.8 元组	152
6.9 结构比较	152
6.10 小结	155
第7章 运算符和类型强制转换	156
7.1 运算符和类型转换	156
7.2 运算符	156
7.2.1 运算符的简化操作	158
7.2.2 运算符的优先级	162
7.3 类型的安全性	163
7.3.1 类型转换	163
7.3.2 装箱和拆箱	167
7.4 比较对象的相等性	168
7.4.1 比较引用类型的相等性	168
7.4.2 比较值类型的相等性	169

7.5 运算符重载	169
7.5.1 运算符的工作方式	170
7.5.2 运算符重载的示例：Vector结构	171
7.6 用户定义的类型强制转换	178
7.6.1 实现用户定义的类型强制转换	179
7.6.2 多重类型强制转换	185
7.7 小结	188
第8章 委托、lambda表达式和事件	189
8.1 引用方法	189
8.2 委托	190
8.2.1 声明委托	190
8.2.2 使用委托	191
8.2.3 简单的委托示例	194
8.2.4 Action<T>和Func<T>委托	196
8.2.5 BubbleSorter示例	197
8.2.6 多播委托	199
8.2.7 匿名方法	203
8.3 lambda表达式	204
8.3.1 参数	204
8.3.2 多行代码	205
8.3.3 闭包	205
8.3.4 使用foreach语句的闭包	206
8.4 事件	207
8.4.1 事件发布程序	207
8.4.2 事件侦听器	

209	
8.4.3 弱事件	
210	
8.5 小结	
214	
第9章 字符串和正则表达式	
215	
9.1 System.String类	
216	
9.1.1 创建字符串	
217	
9.1.2 StringBuilder成员	
220	
9.1.3 格式字符串	
221	
9.2 正则表达式	
227	
9.2.1 正则表达式概述	
227	
9.2.2 RegularExpressionsPlayaround示例	
228	
9.2.3 显示结果	
230	
9.2.4 匹配、组合和捕获	
232	
9.3 小结	
233	
第10章 集合	
234	
10.1 概述	
235	
10.2 集合接口和类型	
235	
10.3 列表	
236	
10.3.1 创建列表	
238	
10.3.2 只读集合	
247	
10.4 队列	
247	
10.5 栈	
251	
10.6 链表	
252	
10.7 有序列表	
258	
10.8 字典	
259	

10.8.1 键的类型	260
10.8.2 字典示例	261
10.8.3 Lookup类	265
10.8.4 有序字典	265
10.9 集	266
10.10 可观察的集合	268
10.11 位数组	269
10.11.1 BitArray类	270
10.11.2 BitVector32结构	272
10.12 不变的集合	274
10.13 并发集合	276
10.13.1 创建管道	277
10.13.2 使用BlockingCollection	279
10.13.3 使用ConcurrentDictionary	281
10.13.4 完成管道	282
10.14 性能	284
10.15 小结	285
第11章 LINQ	286
11.1 LINQ概述	286
11.1.1 列表和实体	287
11.1.2 LINQ查询	290
11.1.3 扩展方法	291
11.1.4 推迟查询的执行	292
11.2 标准的查询操作符	294
11.2.1 筛选	

296	
11.2.2	用索引筛选
296	
11.2.3	类型筛选
297	
11.2.4	复合的from子句
297	
11.2.5	排序
298	
11.2.6	分组
299	
11.2.7	对嵌套的对象分组
300	
11.2.8	内连接
301	
11.2.9	左外连接
303	
11.2.10	组连接
303	
11.2.11	集合操作
306	
11.2.12	合并
308	
11.2.13	分区
309	
11.2.14	聚合操作符
310	
11.2.15	转换操作符
311	
11.2.16	生成操作符
312	
11.3	并行LINQ
313	
11.3.1	并行查询
313	
11.3.2	分区器
314	
11.3.3	取消
314	
11.4	表达式树
315	
11.5	LINQ提供程序
318	
11.6	小结
319	
第12章	动态语言扩展
320	
12.1	DLR
320	

12.2 dynamic类型	321
12.3 包含DLR ScriptRuntime	325
12.4 DynamicObject和ExpandoObject	328
12.4.1 DynamicObject	328
12.4.2 ExpandoObject	330
12.5 小结	331
第13章 异步编程	332
13.1 异步编程的重要性	332
13.2 异步模式	333
13.2.1 同步调用	340
13.2.2 异步模式	341
13.2.3 基于事件的异步模式	342
13.2.4 基于任务的异步模式	343
13.3 异步编程的基础	345
13.3.1 创建任务	345
13.3.2 调用异步方法	346
13.3.3 延续任务	346
13.3.4 同步上下文	347
13.3.5 使用多个异步方法	347
13.3.6 转换异步模式	348
13.4 错误处理	349
13.4.1 异步方法的异常处理	350
13.4.2 多个异步方法的异常处理	350
13.4.3 使用AggregateException信息	351
13.5 取消	

352	
13.5.1	开始取消任务
352	
13.5.2	使用框架特性取消任务
352	
13.5.3	取消自定义任务
353	
13.6	小结
353	
第14章	内存管理和指针
354	
14.1	内存管理
354	
14.2	后台内存管理
354	
14.2.1	值数据类型
355	
14.2.2	引用数据类型
356	
14.2.3	垃圾回收
358	
14.3	释放非托管的资源
360	
14.3.1	析构函数
360	
14.3.2	IDisposable接口
361	
14.3.3	实现IDisposable接口和析构函数
362	
14.4	不安全的代码
364	
14.4.1	用指针直接访问内存
364	
14.4.2	指针示例：PointerPlayground
373	
14.4.3	使用指针优化性能
377	
14.5	小结
380	
第15章	反射
381	
15.1	在运行期间处理和检查代码
381	
15.2	自定义特性
382	
15.2.1	编写自定义特性
382	
15.2.2	自定义特性示例：WhatsNewAttributes
386	



15.3 反射	388
15.3.1 System.Type类	388
15.3.2 TypeView示例	391
15.3.3 Assembly类	393
15.3.4 完成WhatsNewAttributes示例	394
15.4 小结	397
第16章 错误和异常	398
16.1 简介	398
16.2 异常类	399
16.3 捕获异常	400
16.3.1 实现多个catch块	402
16.3.2 在其他代码中捕获异常	406
16.3.3 System.Exception属性	406
16.3.4 没有处理异常时所发生的情况	406
16.3.5 嵌套的try块	407
16.4 用户定义的异常类	409
16.4.1 捕获用户定义的异常	410
16.4.2 抛出用户定义的异常	411
16.4.3 定义用户定义的异常类	414
16.5 调用者信息	416
16.6 小结	417
第 部分 Visual Studio	
第17章 Visual Studio 2013	419
17.1 使用Visual Studio 2013	419
17.1.1 项目文件的改进	421

17.1.2 Visual Studio的版本	422
17.1.3 Visual Studio设置	423
17.2 创建项目	423
17.2.1 面向多个版本的.NET Framework	424
17.2.2 选择项目类型	426
17.3 浏览并编写项目	429
17.3.1 Solution Explorer	429
17.3.2 使用代码编辑器	435
17.3.3 学习和理解其他窗口	437
17.3.4 排列窗口	441
17.4 构建项目	441
17.4.1 构建、编译和生成	441
17.4.2 调试版本和发布版本	442
17.4.3 选择配置	443
17.4.4 编辑配置	444
17.5 调试代码	445
17.5.1 设置断点	445
17.5.2 使用数据提示和调试器可视化工具	446
17.5.3 监视和修改变量	447
17.5.4 异常	448
17.5.5 多线程	449
17.5.6 IntelliTrace	449
17.6 重构工具	450
17.7 体系结构工具	451
17.7.1 依赖项关系图	

452	
17.7.2	层关系图
453	
17.8	分析应用程序
454	
17.8.1	代码地图
454	
17.8.2	序列图
454	
17.8.3	探查器
455	
17.8.4	Concurrency Visualizer
457	
17.8.5	Code Analysis
458	
17.8.6	Code Metrics
459	
17.9	单元测试
459	
17.9.1	创建单元测试
459	
17.9.2	运行单元测试
460	
17.9.3	预期异常
462	
17.9.4	测试全部代码路径
462	
17.9.5	外部依赖
463	
17.9.6	Fakes Framework
466	
17.10	Windows Store应用程序、WCF、WF等
467	
17.10.1	使用Visual Studio 生成WCF应用程序
467	
17.10.2	使用Visual Studio 生成WF应用程序
468	
17.10.3	使用Visual Studio 2013生成Windows Store应用程序
469	
17.11	小结
470	
	第18章 部署
471	
18.1	部署是应用程序生命周期的一部分
471	
18.2	部署的规划
472	
18.2.1	部署选项
472	

18.2.2 部署要求	472
18.2.3 部署.NET运行库	473
18.3 传统的部署选项	473
18.3.1 xcopy部署	474
18.3.2 xcopy和Web应用程序	475
18.3.3 Windows Installer	475
18.4 ClickOnce	475
18.4.1 ClickOnce操作	476
18.4.2 发布ClickOnce应用程序	476
18.4.3 ClickOnce设置	477
18.4.4 ClickOnce文件的应用程序缓存	479
18.4.5 应用程序的安装	479
18.4.6 ClickOnce部署API	480
18.5 Web部署	481
18.5.1 Web应用程序	481
18.5.2 配置文件	482
18.5.3 创建Web Deploy包	482
18.6 Windows Store应用程序	483
18.6.1 创建应用程序包	484
18.6.2 Windows App Certification Kit	485
18.6.3 旁加载	486
18.6.4 Windows部署API	486
18.7 小结	488
第 部分 基础	
第19章 程序集	

490	
19.1	程序集的含义
490	
19.1.1	程序集的功能
491	
19.1.2	程序集的结构
492	
19.1.3	程序集清单
492	
19.1.4	名称空间、程序集和组件
493	
19.1.5	私有程序集和共享程序集
493	
19.1.6	附属程序集
493	
19.1.7	查看程序集
493	
19.2	构建程序集
494	
19.2.1	创建模块和程序集
494	
19.2.2	程序集的特性
495	
19.2.3	创建和动态加载程序集
497	
19.3	应用程序域
500	
19.4	共享程序集
504	
19.4.1	强名
504	
19.4.2	使用强名获得完整性
505	
19.4.3	全局程序集缓存
506	
19.4.4	创建共享程序集
506	
19.4.5	创建强名
507	
19.4.6	安装共享程序集
508	
19.4.7	使用共享程序集
508	
19.4.8	程序集的延迟签名
509	
19.4.9	引用
510	
19.4.10	本机映像生成器
511	

19.5 配置.NET应用程序	512
19.5.1 配置类别	512
19.5.2 绑定程序集	513
19.6 版本问题	514
19.6.1 版本号	515
19.6.2 通过编程方式获取版本	515
19.6.3 绑定到程序集版本	516
19.6.4 发行者策略文件	517
19.6.5 运行库的版本	518
19.7 在不同的技术之间共享程序集	519
19.7.1 共享源代码	519
19.7.2 可移植类库	520
19.8 小结	521
第20章 诊断	522
20.1 诊断概述	522
20.2 代码协定	523
20.2.1 前提条件	524
20.2.2 后置条件	525
20.2.3 不变量	526
20.2.4 纯粹性	527
20.2.5 接口的协定	527
20.2.6 简写	528
20.2.7 协定和遗留代码	529
20.3 跟踪	529
20.3.1 跟踪源	

530	
20.3.2	跟踪开关
532	
20.3.3	跟踪侦听器
532	
20.3.4	筛选器
534	
20.3.5	相关性
535	
20.3.6	使用ETW进行跟踪
539	
20.3.7	使用EventSource
539	
20.3.8	使用EventSource进行高级跟踪
541	
20.4	事件日志
543	
20.4.1	事件日志体系结构
543	
20.4.2	事件日志类
544	
20.4.3	创建事件源
546	
20.4.4	写入事件日志
546	
20.4.5	资源文件
547	
20.5	性能监视
551	
20.5.1	性能监视类
551	
20.5.2	性能计数器生成器
551	
20.5.3	添加PerformanceCounter组件
554	
20.5.4	perfmon.exe
556	
20.6	小结
557	
第21章	任务、线程和同步
558	
21.1	概述
558	
21.2	Parallel类
560	
21.2.1	用Parallel.For()方法循环
560	
21.2.2	使用Parallel.ForEach()方法循环
563	

## 21.2.3 通过Parallel.Invoke()方法调用多个方法

564

## 21.3 任务

564

### 21.3.1 启动任务

565

### 21.3.2 Future——任务的结果

567

### 21.3.3 连续的任务

568

### 21.3.4 任务层次结构

569

## 21.4 取消架构

570

### 21.4.1 Parallel.For()方法的取消

570

### 21.4.2 任务的取消

571

## 21.5 线程池

572

## 21.6 Thread类

574

### 21.6.1 给线程传递数据

575

### 21.6.2 后台线程

576

### 21.6.3 线程的优先级

577

### 21.6.4 控制线程

578

## 21.7 线程问题

578

### 21.7.1 争用条件

578

### 21.7.2 死锁

581

## 21.8 同步

583

### 21.8.1 lock语句和线程安全

583

### 21.8.2 Interlocked类

588

### 21.8.3 Monitor类

589

### 21.8.4 SpinLock结构

590

### 21.8.5 WaitHandle基类

591

### 21.8.6 Mutex类



591
21.8.7 Semaphore类
593
21.8.8 Events类
595
21.8.9 Barrier类
598
21.8.10 ReaderWriterLockSlim类
600
21.9 Timer类
602
21.10 数据流
604
21.10.1 使用动作块
604
21.10.2 源和目标数据块
605
21.10.3 连接块
606
21.11 小结
608
第22章 安全性
609
22.1 概述
609
22.2 身份验证和授权
610
22.2.1 标识和Principal
610
22.2.2 角色
611
22.2.3 声明基于角色的安全性
612
22.2.4 声称
613
22.2.5 客户端应用程序服务
614
22.3 加密
619
22.3.1 签名
621
22.3.2 交换密钥和安全传输
622
22.4 资源的访问控制
625
22.5 代码访问安全性
628
22.5.1 第2级安全透明性
628

22.5.2 权限	629
22.6 使用证书发布代码	634
22.7 小结	635
第23章 互操作	636
23.1 .NET和COM技术	636
23.1.1 元数据	637
23.1.2 释放内存	638
23.1.3 接口	638
23.1.4 方法绑定	639
23.1.5 数据类型	640
23.1.6 注册	640
23.1.7 线程	640
23.1.8 错误处理	641
23.1.9 事件	642
23.1.10 封送	642
23.2 在.NET客户端中使用COM组件	643
23.2.1 创建COM组件	643
23.2.2 创建运行库可调用包装	649
23.2.3 使用RCW	650
23.2.4 通过动态语言扩展使用COM服务	651
23.2.5 线程问题	652
23.2.6 添加连接点	652
23.3 在COM客户端中使用.NET组件	654
23.3.1 COM可调用包装	655
23.3.2 创建.NET组件	

655	
23.3.3	创建类型库
656	
23.3.4	COM互操作特性
658	
23.3.5	COM注册
660	
23.3.6	创建COM客户端应用程序
661	
23.3.7	添加连接点
662	
23.3.8	使用sink对象创建客户端
663	
23.4	平台调用
665	
23.5	小结
669	
第24章	文件和注册表操作
670	
24.1	文件和注册表
670	
24.2	管理文件系统
671	
24.2.1	表示文件和文件夹的.NET类
672	
24.2.2	Path类
674	
24.2.3	FileProperties示例
674	
24.3	移动、复制和删除文件
679	
24.3.1	FilePropertiesAndMovement示例
679	
24.3.2	FilePropertiesAndMovement示例的代码
680	
24.4	读写文件
682	
24.4.1	读取文件
683	
24.4.2	写入文件
684	
24.4.3	流
685	
24.4.4	缓存的流
687	
24.4.5	使用FileStream类读写二进制文件
687	
24.4.6	读写文本文件
692	

24.5 映射内存的文件	698
24.6 读取驱动器信息	699
24.7 文件的安全性	701
24.7.1 从文件中读取ACL	701
24.7.2 从目录中读取ACL	702
24.7.3 添加和删除文件中的ACL项	704
24.8 读写注册表	705
24.8.1 注册表	706
24.8.2 .NET注册表类	708
24.9 读写独立存储器	710
24.10 小结	714
第25章 事务处理	715
25.1 简介	715
25.2 概述	716
25.2.1 事务处理阶段	716
25.2.2 ACID属性	717
25.3 数据库和实体类	717
25.4 传统的事务	719
25.4.1 ADO.NET事务	719
25.4.2 System.EnterpriseServices	721
25.5 System.Transactions	722
25.5.1 可提交的事务	723
25.5.2 事务处理的升级	725
25.5.3 依赖事务	727
25.5.4 环境事务	

729	
25.6	隔离级别
736	
25.7	自定义资源管理器
737	
25.8	文件系统事务
743	
25.9	小结
747	
第26章	网络
748	
26.1	网络
748	
26.2	HttpClient类
749	
26.2.1	异步调用Web服务
749	
26.2.2	标题
750	
26.2.3	HttpContent
752	
26.2.4	HttpMessageHandler
752	
26.3	把输出结果显示为HTML页面
753	
26.3.1	从应用程序中进行简单的Web浏览
754	
26.3.2	启动Internet Explorer实例
755	
26.3.3	给应用程序提供更多IE类型的功能
756	
26.3.4	使用WebBrowser控件打印
761	
26.3.5	显示所请求页面的代码
761	
26.3.6	WebRequest类和WebResponse类的层次结构
763	
26.4	实用工具类
763	
26.4.1	URI
763	
26.4.2	IP地址和DNS名称
764	
26.5	较低层的协议
766	
26.5.1	使用SmtpClient
767	
26.5.2	使用TCP类
769	

26.5.3 TcpSend和TcpReceive示例

769

26.5.4 TCP和UDP

771

26.5.5 UDP类

771

26.5.6 Socket类

772

26.5.7 WebSocket

776

26.6 小结

779

第27章 Windows服务

780

27.1 Windows服务

780

27.2 Windows服务的体系结构

781

27.2.1 服务程序

782

27.2.2 服务控制程序

783

27.2.3 服务配置程序

783

27.2.4 Windows服务的类

783

27.3 创建Windows服务程序

784

27.3.1 创建服务的核心功能

784

27.3.2 QuoteClient示例

787

27.3.3 Windows服务程序

791

27.3.4 线程化和服务

795

27.3.5 服务的安装

795

27.3.6 安装程序

796

27.4 Windows服务的监控和控制

800

27.4.1 MMC管理单元

800

27.4.2 net.exe实用程序

801

27.4.3 sc.exe实用程序

801

27.4.4 Visual Studio Server Explorer

801	
27.4.5	编写自定义ServiceController类
802	
27.5	故障排除和事件日志
809	
27.6	小结
810	
第28章	本地化
811	
28.1	全球市场
811	
28.2	System.Globalization名称空间
812	
28.2.1	Unicode问题
812	
28.2.2	区域性和区域
813	
28.2.3	使用区域性
817	
28.2.4	排序
823	
28.3	资源
824	
28.3.1	创建资源文件
824	
28.3.2	资源文件生成器
824	
28.3.3	ResourceWriter
825	
28.3.4	使用资源文件
826	
28.3.5	System.Resources名称空间
830	
28.4	使用Visual Studio的Windows Forms本地化
830	
28.4.1	通过编程方式修改区域性
835	
28.4.2	使用自定义资源消息
836	
28.4.3	资源的自动回退
837	
28.4.4	外包翻译
837	
28.5	ASP.NET Web Forms的本地化
838	
28.6	用WPF本地化
839	
28.6.1	用于WPF的.NET资源
840	

28.6.2 XAML资源字典	841
28.7 自定义资源读取器	845
28.7.1 创建DatabaseResourceReader类	845
28.7.2 创建DatabaseResourceSet类	847
28.7.3 创建DatabaseResourceManager类	847
28.7.4 DatabaseResourceReader的客户端应用程序	848
28.8 创建自定义区域性	848
28.9 用Windows Store应用程序进行本地化	850
28.9.1 使用资源	851
28.9.2 使用多语言应用程序工具集进行本地化	851
28.10 小结	852
第29章 核心XAML	853
29.1 XAML的作用	853
29.2 XAML概述	854
29.2.1 元素如何映射到.NET对象上	854
29.2.2 使用自定义.NET类	856
29.2.3 把属性用作特性	857
29.2.4 把属性用作元素	857
29.2.5 基本的.NET类型	858
29.2.6 使用集合和XAML	858
29.2.7 用XAML代码调用构造函数	859
29.3 依赖属性	859
29.3.1 创建依赖属性	860
29.3.2 强制值回调	861
29.3.3 值变更回调和事件	



862	
29.3.4	事件的冒泡和隧道
863	
29.4	附加属性
866	
29.5	标记扩展
868	
29.6	创建自定义标记扩展
869	
29.7	XAML定义的标记扩展
870	
29.8	读写XAML
871	
29.9	小结
872	
第30章	Managed Extensibility Framework
873	
30.1	概述
873	
30.2	MEF的体系结构
874	
30.2.1	使用属性的MEF
875	
30.2.2	基于约定的部件注册
880	
30.3	定义协定
882	
30.4	导出部件
883	
30.4.1	创建部件
883	
30.4.2	导出属性和方法
888	
30.4.3	导出元数据
890	
30.4.4	使用元数据进行惰性加载
892	
30.5	导入部件
893	
30.5.1	导入连接
895	
30.5.2	部件的惰性加载
897	
30.5.3	用惰性实例化的部件读取元数据
898	
30.6	容器和出口提供程序
900	
30.7	类别
902	

30.8 小结	904
第31章 Windows运行库	905
31.1 概述	905
31.1.1 .NET与Windows运行库的比较	906
31.1.2 名称空间	906
31.1.3 元数据	908
31.1.4 语言投射	909
31.1.5 Windows运行库中的类型	911
31.2 Windows运行库组件	912
31.2.1 集合	912
31.2.2 流	913
31.2.3 委托与事件	914
31.2.4 异步操作	914
31.3 Windows Store应用程序	915
31.4 应用程序的生命周期	917
31.4.1 应用程序的执行状态	918
31.4.2 Suspension Manager	919
31.4.3 导航状态	921
31.4.4 测试暂停	922
31.4.5 页面状态	922
31.5 应用程序的设置	924
31.6 小结	927
第 部分	
数 据	
第32章 核心ADO.NET	929
32.1 ADO.NET概述	

929	
32.1.1	名称空间
930	
32.1.2	共享类
931	
32.1.3	数据库专用类
931	
32.2	使用数据库连接
932	
32.2.1	管理连接字符串
933	
32.2.2	高效地使用连接
934	
32.2.3	事务
936	
32.3	命令
938	
32.3.1	执行命令
938	
32.3.2	调用存储过程
941	
32.4	快速数据访问：数据读取器
944	
32.5	异步数据访问：使用Task和await
946	
32.6	管理数据和关系：DataSet类
948	
32.6.1	数据表
949	
32.6.2	数据列
950	
32.6.3	数据关系
955	
32.6.4	数据约束
956	
32.7	XML架构：用XSD生成代码
958	
32.8	填充DataSet类
965	
32.8.1	用数据适配器填充DataSet
965	
32.8.2	从XML中填充DataSet类
966	
32.9	持久化DataSet类的修改
966	
32.9.1	通过数据适配器进行更新
966	
32.9.2	写入XML输出结果
969	

32.10 使用ADO.NET	970
32.10.1 分层开发	970
32.10.2 生成SQL Server的键	972
32.10.3 命名约定	974
32.11 小结	975
第33章 ADO.NET Entity Framework	976
33.1 用Entity Framework编程	976
33.2 Entity Framework映射	978
33.2.1 逻辑层	978
33.2.2 概念层	980
33.2.3 映射层	982
33.2.4 连接字符串	983
33.3 实体	983
33.4 对象上下文	984
33.5 关系	986
33.5.1 一个层次结构一个表	986
33.5.2 一种类型一个表	988
33.5.3 懒惰加载、延迟加载和预先加载	989
33.6 查询数据	991
33.6.1 Entity SQL	991
33.6.2 使用DbSqlQuery	992
33.6.3 LINQ to Entities	993
33.7 把数据写入数据库	994
33.7.1 对象跟踪	994
33.7.2 改变信息	

995	
33.7.3	附加和分离实体
996	
33.7.4	使用最后一个更改操作 写入实体的更改
997	
33.7.5	使用第一个更改操作写入实体的更改
997	
33.7.6	写入实体的更改并处理冲突
998	
33.8	使用Code First编程模型
999	
33.8.1	定义实体类型
999	
33.8.2	创建数据上下文
1000	
33.8.3	创建数据库，存储实体
1000	
33.8.4	数据库
1001	
33.8.5	查询数据
1001	
33.8.6	定制数据库的生成
1002	
33.8.7	数据库的自动填充
1003	
33.8.8	连接的弹性
1004	
33.8.9	架构的迁移
1005	
33.9	小结
1007	
第34章	处理XML
1008	
34.1	XML
1008	
34.2	.NET支持的XML标准
1009	
34.3	System.Xml名称空间
1009	
34.4	使用System.Xml类
1010	
34.5	读写流格式的XML
1011	
34.5.1	使用XmlReader类
1011	
34.5.2	使用XmlReader类进行验证
1015	
34.5.3	使用XmlWriter类

1017
34.6 在.NET中使用DOM
1018
34.7 使用XPathNavigator类
1023
34.7.1 System.Xml.XPath名称空间
1023
34.7.2 System.Xml.Xsl名称空间
1028
34.7.3 调试XSLT
1032
34.8 XML和ADO.NET
1034
34.8.1 将ADO.NET数据转换为XML文档
1034
34.8.2 把XML文档转换为 ADO.NET数据
1040
34.9 在XML中序列化对象
1041
34.10 LINQ to XML和.NET
1051
34.11 使用不同的XML对象
1051
34.11.1 XDocument对象
1051
34.11.2 XElement对象
1052
34.11.3 XNamespace对象
1053
34.11.4 XComment对象
1055
34.11.5 XAttribute对象
1056
34.12 使用LINQ查询XML文档
1057
34.12.1 查询静态的XML文档
1057
34.12.2 查询动态的XML文档
1058
34.13 XML文档的更多查询技术
1060
34.13.1 读取XML文档
1060
34.13.2 写入XML文档
1061
34.14 小结
1063
第 部分 显 示

第35章 核心WPF	1065
35.1 理解WPF	1066
35.1.1 名称空间	1066
35.1.2 类层次结构	1067
35.2 形状	1069
35.3 几何图形	1070
35.4 变换	1072
35.5 画笔	1073
35.5.1 SolidColorBrush	1073
35.5.2 LinearGradientBrush	1074
35.5.3 RadialGradientBrush	1074
35.5.4 DrawingBrush	1075
35.5.5 ImageBrush	1076
35.5.6 VisualBrush	1076
35.6 控件	1077
35.6.1 简单控件	1077
35.6.2 内容控件	1078
35.6.3 带标题的内容控件	1079
35.6.4 项控件	1081
35.6.5 带标题的项控件	1081
35.6.6 修饰	1081
35.7 布局	1082
35.7.1 StackPanel	1082
35.7.2 WrapPanel	1083
35.7.3 Canvas	

1084	
35.7.4	DockPanel
1084	
35.7.5	Grid
1085	
35.8	样式和资源
1086	
35.8.1	样式
1087	
35.8.2	资源
1088	
35.8.3	系统资源
1090	
35.8.4	从代码中访问资源
1090	
35.8.5	动态资源
1090	
35.8.6	资源字典
1091	
35.9	触发器
1092	
35.9.1	属性触发器
1093	
35.9.2	多触发器
1094	
35.9.3	数据触发器
1095	
35.10	模板
1096	
35.10.1	控件模板
1097	
35.10.2	数据模板
1099	
35.10.3	样式化列表框
1101	
35.10.4	ItemTemplate
1102	
35.10.5	列表框元素的控件模板
1103	
35.11	动画
1105	
35.11.1	时间轴
1105	
35.11.2	非线性动画
1108	
35.11.3	事件触发器
1109	
35.11.4	关键帧动画
1111	



35.12 可见状态管理器	1112
35.12.1 可见的状态	1113
35.12.2 变换	1115
35.13 3-D	1116
35.13.1 模型	1116
35.13.2 照相机	1118
35.13.3 光线	1118
35.13.4 旋转	1118
35.14 小结	1119
第36章 用WPF编写业务应用程序	1120
36.1 概述	1120
36.2 菜单和功能区控件	1121
36.2.1 菜单控件	1121
36.2.2 功能区控件	1122
36.3 Commanding	1124
36.3.1 定义命令	1125
36.3.2 定义命令源	1126
36.3.3 命令绑定	1126
36.4 数据绑定	1127
36.4.1 BooksDemo应用程序内容	1128
36.4.2 用XAML绑定	1129
36.4.3 简单对象的绑定	1131
36.4.4 更改通知	1134
36.4.5 对象数据提供程序	1136
36.4.6 列表绑定	

1138	
36.4.7 主从绑定	
1141	
36.4.8 多绑定	
1142	
36.4.9 优先绑定	
1144	
36.4.10 值的转换	
1145	
36.4.11 动态添加列表项	
1147	
36.4.12 动态添加选项卡中的项	
1148	
36.4.13 数据模板选择器	
1149	
36.4.14 绑定到XML上	
1151	
36.4.15 绑定的验证和错误处理	
1153	
36.5 TreeView	
1161	
36.6 DataGrid	
1165	
36.6.1 自定义列	
1167	
36.6.2 行的细节	
1168	
36.6.3 用DataGrid进行分组	
1168	
36.6.4 实时成型	
1171	
36.7 小结	
1177	
第37章 用WPF创建文档	
1178	
37.1 简介	
1178	
37.2 文本元素	
1179	
37.2.1 字体	
1179	
37.2.2 TextEffect	
1180	
37.2.3 内联	
1181	
37.2.4 块	
1183	
37.2.5 列表	
1185	

37.2.6 表	1185
37.2.7 块的锚定	1186
37.3 流文档	1188
37.4 固定文档	1192
37.5 XPS文档	1196
37.6 打印	1197
37.6.1 用PrintDialog打印	1198
37.6.2 打印可见元素	1198
37.7 小结	1200
第38章 Windows Store应用程序：用户界面	1201
38.1 概述	1201
38.2 Microsoft的现代设计	1202
38.2.1 内容，不是边框	1202
38.2.2 快速流畅	1203
38.2.3 可读性	1204
38.3 示例应用程序的核心功能	1204
38.3.1 文件和目录	1204
38.3.2 应用程序页面	1205
38.4 应用程序工具栏	1210
38.5 启动与导航	1213
38.6 布局的变化应用程序数据	1215
38.7 存储	1220
38.7.1 定义数据协定	1220
38.7.2 写入移动数据	1222
38.7.3 读取数据	

1223	
38.7.4 写入图像	
1224	
38.7.5 读取图像	
1226	
38.8 选择器	
1227	
38.9 活动的磁贴	
1228	
38.10 小结	
1230	
第39章 Windows Store应用程序：协定和设备	
1231	
39.1 概述	
1231	
39.2 搜索	
1232	
39.3 共享协定	
1234	
39.3.1 共享源	
1234	
39.3.2 共享目标	
1237	
39.4 相机	
1239	
39.5 定位	
1240	
39.6 感应器	
1243	
39.6.1 光线	
1244	
39.6.2 罗盘	
1244	
39.6.3 加速计	
1245	
39.6.4 倾斜计	
1246	
39.6.5 陀螺仪	
1246	
39.6.6 方向	
1246	
39.6.7 Rolling Marble示例	
1247	
39.7 小结	
1249	
第40章 核心ASP.NET	
1250	
40.1 用于Web应用程序的.NET Framework	
1250	

40.1.1 ASP.NET Web Forms	1251
40.1.2 ASP.NET Web Pages	1251
40.1.3 ASP.NET MVC	1252
40.2 Web技术	1252
40.2.1 HTML	1253
40.2.2 CSS	1253
40.2.3 JavaScript和jQuery	1253
40.3 托管和配置	1254
40.4 处理程序和模块	1256
40.4.1 创建自定义处理程序	1257
40.4.2 ASP.NET处理程序	1258
40.4.3 创建自定义模块	1259
40.4.4 通用模块	1260
40.5 全局的应用程序类	1261
40.6 请求和响应	1262
40.6.1 使用HttpRequest对象	1262
40.6.2 使用HttpResponse对象	1264
40.7 状态管理	1264
40.7.1 视图状态	1265
40.7.2 cookie	1266
40.7.3 会话	1267
40.7.4 应用程序状态	1270
40.7.5 缓存	1270
40.7.6 配置文件	1271
40.8 ASP.NET身份系统	

1275	
40.8.1 基础知识	
1276	
40.8.2 存储和检索用户信息	
1277	
40.8.3 安全启动	
1278	
40.8.4 使用注册和身份验证	
1278	
40.9 小结	
1279	
第41章 ASP.NET Web Forms	
1281	
41.1 概述	
1281	
41.2 ASPX页面模型	
1282	
41.2.1 添加控件	
1283	
41.2.2 使用事件	
1284	
41.2.3 使用回送	
1284	
41.2.4 使用自动回送	
1285	
41.2.5 回送到其他页面	
1285	
41.2.6 定义强类型化的跨页面回送	
1286	
41.2.7 使用页面事件	
1287	
41.2.8 ASPX代码	
1288	
41.2.9 服务器端控件	
1290	
41.3 母版页	
1291	
41.3.1 创建母版页	
1291	
41.3.2 使用母版页	
1293	
41.3.3 在内容页中定义母版页内容	
1294	
41.4 导航	
1296	
41.4.1 站点地图	
1296	
41.4.2 Menu控件	
1296	

41.4.3 菜单路径	1297
41.5 验证用户输入	1297
41.5.1 使用验证控件	1298
41.5.2 使用验证摘要	1299
41.5.3 验证组	1300
41.6 访问数据	1300
41.6.1 使用Object Framework	1301
41.6.2 创建库	1302
41.6.3 使用Object Data Source	1303
41.6.4 编辑	1306
41.6.5 定制列	1307
41.6.6 在网格中使用模板	1308
41.7 安全性	1309
41.7.1 建立 ASP.NET身份	1309
41.7.2 用户注册	1310
41.7.3 用户的身份验证	1311
41.7.4 用户授权	1312
41.8 Ajax	1313
41.8.1 ASP.NET AJAX的概念	1314
41.8.2 ASP.NET AJAX网站示例	1316
41.8.3 支持ASP.NET AJAX的网站配置	1319
41.8.4 添加ASP.NET AJAX功能	1320
41.9 小结	1326
第42章 ASP.NET MVC	1328
42.1 ASP.NET MVC概述	

1328	
42.2 定义路由	
1330	
42.2.1 添加路由	
1330	
42.2.2 路由约束	
1331	
42.3 创建控制器	
1331	
42.3.1 动作方法	
1332	
42.3.2 参数	
1332	
42.3.3 返回数据	
1333	
42.4 创建视图	
1335	
42.4.1 向视图传递数据	
1336	
42.4.2 Razor语法	
1337	
42.4.3 强类型视图	
1338	
42.4.4 布局	
1339	
42.4.5 部分视图	
1342	
42.5 从客户端提交数据	
1346	
42.5.1 模型绑定器	
1347	
42.5.2 注释和验证	
1348	
42.6 HTML Helper	
1350	
42.6.1 简单的Helper	
1350	
42.6.2 使用模型数据	
1351	
42.6.3 定义HTML特性	
1352	
42.6.4 创建列表	
1352	
42.6.5 强类型化的Helper	
1353	
42.6.6 编辑器扩展	
1354	
42.6.7 创建自定义Helper	
1355	



42.6.8 模板	1355
42.7 创建数据驱动的应用程序	1356
42.7.1 定义模型	1357
42.7.2 创建控制器和视图	1358
42.8 动作过滤器	1364
42.9 身份验证和授权	1366
42.9.1 登录模型	1366
42.9.2 登录控制器	1366
42.9.3 登录视图	1368
42.10 小结	1370
第 部分	
通 信	
第43章 WCF	1372
43.1 WCF概述	1372
43.1.1 SOAP	1374
43.1.2 WSDL	1374
43.1.3 REST	1375
43.1.4 JSON	1375
43.2 创建简单的服务和客户端	1375
43.2.1 定义服务和数据协定	1376
43.2.2 数据访问	1378
43.2.3 服务的实现	1379
43.2.4 WCF服务宿主和WCF测试客户端	1380
43.2.5 自定义服务宿主	1382
43.2.6 WCF客户端	1384
43.2.7 诊断	

1386	
43.2.8	与客户端共享协定程序集
1388	
43.3	协定
1389	
43.3.1	数据协定
1390	
43.3.2	版本问题
1390	
43.3.3	服务协定
1391	
43.3.4	消息协定
1392	
43.3.5	错误协定
1392	
43.4	服务的行为
1394	
43.5	绑定
1397	
43.5.1	标准的绑定
1397	
43.5.2	标准绑定的特性
1399	
43.5.3	Web套接字
1400	
43.6	宿主
1403	
43.6.1	自定义宿主
1403	
43.6.2	WAS宿主
1404	
43.6.3	预配置的宿主类
1405	
43.7	客户端
1406	
43.7.1	使用元数据
1406	
43.7.2	共享类型
1407	
43.8	双工通信
1408	
43.8.1	双工通信的协定
1408	
43.8.2	双工通信的服务
1408	
43.8.3	双工通信的客户端应用程序
1409	
43.9	路由
1410	

43.9.1 示例应用程序	1411
43.9.2 路由接口	1412
43.9.3 WCF路由服务	1412
43.9.4 为失败使用路由器	1413
43.9.5 改变协定的桥梁	1415
43.9.6 过滤器的类型	1415
43.10 小结	1415
第44章 ASP.NET Web API	1416
44.1 概述	1416
44.2 创建服务	1417
44.2.1 定义模型	1418
44.2.2 创建控制器	1418
44.2.3 错误处理	1420
44.3 创建.NET客户程序	1420
44.3.1 发送GET请求	1421
44.3.2 发送POST请求	1422
44.3.3 发送PUT请求	1423
44.3.4 发送DELETE请求	1423
44.4 Web API路由和操作	1424
44.4.1 给操作添加HTTP方法	1424
44.4.2 基于特性的路由	1425
44.5 使用OData	1427
44.5.1 创建数据模型	1427
44.5.2 创建服务	1428
44.5.3 OData查询	

1430	
44.5.4 WCF Data Services客户程序	
1431	
44.6 保护Web API	
1436	
44.6.1 创建账户	
1437	
44.6.2 创建验证令牌	
1439	
44.6.3 发送验证过的调用	
1440	
44.6.4 获取用户信息	
1440	
44.7 自驻留	
1441	
44.8 小结	
1443	
第45章 Windows Workflow Foundation	
1444	
45.1 工作流概述	
1444	
45.2 Hello World示例	
1445	
45.3 活动	
1446	
45.3.1 If活动	
1447	
45.3.2 InvokeMethod活动	
1448	
45.3.3 Parallel活动	
1448	
45.3.4 Delay活动	
1449	
45.3.5 Pick活动	
1449	
45.4 自定义活动	
1450	
45.4.1 活动的验证	
1451	
45.4.2 设计器	
1452	
45.4.3 自定义复合活动	
1454	
45.5 工作流	
1456	
45.5.1 实参和变量	
1457	
45.5.2 WorkflowApplication	
1458	

45.5.3 存放WCF工作流	1461
45.5.4 工作流的版本	1465
45.5.5 驻留设计器	1466
45.6 小结	1471
第46章 对等网络	1472
46.1 P2P网络概述	1472
46.1.1 客户端-服务器体系结构	1473
46.1.2 P2P体系结构	1473
46.1.3 P2P体系结构的挑战	1474
46.1.4 P2P术语	1475
46.1.5 P2P解决方案	1475
46.2 PNRP	1475
46.3 构建P2P应用程序	1478
46.4 小结	1484
第47章 消息队列	1485
47.1 概述	1486
47.1.1 使用消息队列的场合	1486
47.1.2 消息队列功能	1487
47.2 Message Queuing产品	1488
47.3 消息队列体系结构	1489
47.3.1 消息	1489
47.3.2 消息队列	1489
47.4 Message Queuing管理工具	1490
47.4.1 创建消息队列	1490
47.4.2 消息队列属性	

1491	
47.5	消息队列的编程实现
1492	
47.5.1	创建消息队列
1492	
47.5.2	查找队列
1493	
47.5.3	打开已知队列
1493	
47.5.4	发送消息
1495	
47.5.5	接收消息
1497	
47.6	课程订单应用程序
1499	
47.6.1	课程订单类库
1499	
47.6.2	课程订单消息发送程序
1502	
47.6.3	发送优先级和可恢复的消息
1504	
47.6.4	课程订单消息接收应用程序
1505	
47.7	接收结果
1511	
47.7.1	确认队列
1511	
47.7.2	响应队列
1512	
47.8	事务队列
1512	
47.9	消息队列和WCF
1514	
47.9.1	带数据协定的实体类
1514	
47.9.2	WCF服务协定
1515	
47.9.3	WCF消息接收应用程序
1516	
47.9.4	WCF消息发送应用程序
1519	
47.10	消息队列的安装
1520	
47.11	小结
1520	

## 精彩短评

- 1、太厚了1500页 杀人防身必备
- 2、这货应该分成六册出版，一个部分一册，太厚了。
- 3、一般般的工具书，广而不深，排版捉急，妄称经典。
- 4、翻译水平已经难以用很差来形容了，糟蹋了经典原著 翻译人员简直在骗钱，很多章节只比有道词典的翻译好一点点，完全不建议购买中文版，建议直接看英文原著。
- 5、连续畅销16年，经典巨著
- 6、c#,WPF,WCF,ado.net,asp.net,asp.net mvc,windows store各种技术大杂烩，了解基础可以，但深入较难，另翻译太敷衍了，Annotation竟然翻译为注释，太外行了
- 7、啃字典
- 8、非常好，非常全面，桌案必备佳作！
- 9、快扫了一遍，确实高级

### 精彩书评

1、刚开始几章还可以，但是看到中间以后，感觉这本书的翻译人员纯粹是在骗钱，有些地方的翻译，只是比用有道字典的翻译稍微强一些。而且感觉很多地方的翻译，是由根本不懂技术的人来翻译的，完全的词不达意。本来这本书的原著英文版是非常经典的，但是被这帮混饭吃的翻译给弄成垃圾了。这本中文书，买的必要不是很大，建议有能力的读者直接读英文版。没有能力的，也别读中文版了，因为词不达意的翻译，会让你赶到完全是在看天书，原因并不是难度，而是翻译出来的，根本就不是人话。



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:[www.tushu111.com](http://www.tushu111.com)