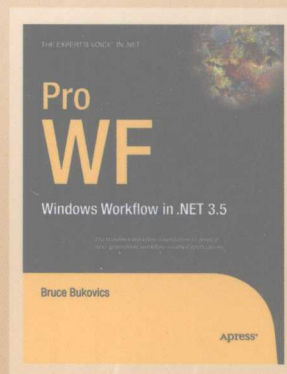


Pro WF Windows Workflow in .NET 3.5

WF 高级程序设计

[美] Bruce Bukovics 著
柴晓伟 译

- 一场高级程序设计的丰盛大宴
- 深入WF精髓，满足你无限的求知欲
- 分析透彻的丰富示例，手把手教你设计绝活



TURING

图灵程序设计丛书

微软技术系列

Pro WF Windows Workflow in .NET 3.5

WF高级程序设计

人民邮电出版社
北 京

图书在版编目(CIP)数据

WF 高级程序设计 / (美) 布科威茨 (Bukovics, B.) 著;
柴晓伟译. —北京: 人民邮电出版社, 2009.10
(图灵程序设计丛书)
书名原文: Pro WF: Windows Workflow in .NET 3.5
ISBN 978-7-115-21241-2

I. W… II. ①布…②柴… III. 窗口软件, Windows —
程序设计 IV. TP316.7

中国版本图书馆CIP数据核字(2009)第146500号

内 容 提 要

本书全面介绍了 WF 高级程序设计。WF 不仅是实现用户界面的新方法, 也是在应用程序和服务之间通信的新方法, 更是一种体现了开发应用程序的全新方式。全书围绕如何在应用程序中整合 WF 展开讨论, 主要涵盖 WF 基础知识、承载 workflow 运行时、流控制、本地服务、workflow 持久性, 状态机 workflow、异常和错误处理、动态 workflow 更新, 以及在 Web 服务和 ASP.NET 开发中使用 workflow 等内容。另外, 本书示例丰富, 最大程度地阐明了使用 WF 技术所需要掌握的方方面面。

本书适合希望在自己的应用程序中使用 WF 的所有 .NET 开发人员。

图灵程序设计丛书

WF高级程序设计

-
- ◆ 著 [美] Bruce Bukovics
 - 译 柴晓伟
 - 责任编辑 傅志红
 - 执行编辑 边晓娜
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
 - 邮编 100061 电子函件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 北京顺义振华印刷厂印刷
 - ◆ 开本: 800×1000 1/16
 - 印张: 44
 - 字数: 1236千字 2009年10月第1版
 - 印数: 1—3 000册 2009年10月北京第1次印刷
 - 著作权合同登记号 图字: 01-2009-2902号
 - ISBN 978-7-115-21241-2
-

定价: 99.00 元

读者服务热线: (010)51095186 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Original English language edition, entitled *Pro WF: Windows Workflow in .NET 3.5* by Bruce Bukovics, published by Apress L. P., 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705 USA.

Copyright © 2008 by Bruce Bukovics. Simplified Chinese-language edition copyright © 2009 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由Apress L.P.授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

前言

我开始应用微软新的WinFX技术时，该技术还处于beta版和CTP（Community Technology Preview，社区技术预览）版阶段。这些基础结构开始时称作WinFX（包含Windows Presentation、Windows Communication和Windows Workflow），在2006年11月它们作为.NET Framework 3.0的一部分正式发布。实际上，我在日常工作中开始学习并使用这3个基础结构也是从那时开始的。下面就来谈谈我的不平凡的学习过程。

WPF（Windows Presentation Foundation）和WCF（Windows Communication Foundation）所具有的灵活性及其功能给我留下了深刻的印象，同时，WF（Windows Workflow Foundation）也有点莫名地吸引我。WF并不仅仅是实现用户界面的新方法，也不仅仅是在应用程序和服务之间通信的新方法，而是一种体现了开发应用程序的全新方式。它是可声明、可视化且无限灵活的，其倡导的模型可以清晰地将工作内容和工作时间分隔开来。这种分隔允许你改变工作流模型（工作时间），而不会影响到工作内容。可将业务逻辑实现为一组独立并且可测试的组件，它们能像构建块一样装配到工作流中。

工作流并不是新概念。但是当微软公司花费了数年开发工作流基础结构并免费提供给我们时，这便值得我们去关注了。相比于其他工作流框架，WF实际上是Windows应用程序的标准工作流框架。

工作流（特别是WF）让我很振奋，所以我编写了本书的第1版。我也很振奋于它为和我一样的应用程序开发人员提供的各种机会。直到今天我仍然十分振奋，特别是在.NET Framework 3.5发布之后。在这个版本中，微软公司为WF提供了更多的增强功能——尤其是对工作流服务的支持（工作流服务其实就是以工作流的方式实现的WCF服务）。

我衷心希望，你读过本书之后也能够使用WF来构建令人振奋的新一代实用的工作流应用程序。

本书读者

本书面向所有想要学习如何在自己的应用程序中使用WF的.NET开发人员。它不是.NET或C#语言的入门读物。想要更深刻地理解本书的示例，你需要良好的.NET 1.1知识（.NET 2.0更好）。本书所有的示例都采用C#编写，所以你需要通晓C#。

Visual Studio 2008已经内置了对WF的支持。我使用它来生成本书中的示例和图示，也推荐你采用它作为开发平台。在本书的示例代码中，我会使用一些省时的C# 3.0新特性，比如自动实现属性等。所以，如果你熟悉C# 3.0的新功能，将会对你理解这些示例非常有帮助。

本书内容

本书共分为19章，每一章都基于前一章的内容。这样组织材料是为了让你不需要为理解某些概念而跳着阅读。不过，由于每一章都基于前一章的知识点，我会假设你已经按顺序读完并理解了该章之

前的所有内容。

下面是每一章的简短摘要。

第 1 章：WF 快速入门

本章简单介绍了WF。在本章中，我将引领你开发第一个工作流（“Hello Workflow”），并向你介绍WF的一些基本概念，例如，如何向工作流传递参数以及如何在工作流中制定决策等。

第 2 章：基础结构概述

本章的目标是对WF进行全面而高层次的概述。本章不会教你如何使用具体的WF功能，但会让你了解WF的设计时和运行时功能。可以将本章看作是一张涵盖了本书剩余章节所有内容的路线图。

第 3 章：活动

活动是WF的构建块，用于构造完整的工作流。本章总结了WF内置的标准活动，而且还比较了两种将业务逻辑引入工作流的方法：CodeActivity和构建自定义活动。

第 4 章：承载工作流运行时

WF并不是独立的应用程序，而是允许你构建自己的工作流应用程序的框架。本章演示了在应用程序中如何承载工作流运行时，还介绍了一套自定义工作流管理器类，它们帮助你承载工作流运行时，而且在之后的章节中会大量使用这些辅助类。

第 5 章：流控制

WF包含了丰富的标准活动，支持的特性包括简单的分支决策、While循环，以及并行执行和复制。这些流控制活动控制着工作流的执行顺序，这也是本章的主要内容。大多数活动都支持Boolean类型的条件，该条件既可以通过代码来指定，也可以设置为声明性规则条件。本章将对这两种类型的条件进行比较。

第 6 章：本地服务

WF的一些核心功能是采用可插拔的服务机制来实现的。这种机制允许你根据自己的需要去实现这些服务。WF还提供了本地服务，你可以基于任何目的来实现这些服务。在工作流实例和宿主应用程序之间通信就是本地服务的一个常见用途。本章的重点是实现和使用自定义的本地服务。

第 7 章：事件驱动活动

本章的主要内容是事件驱动活动，事件驱动活动允许工作流等待外部事件的到达。第6章演示了如何实现本地服务和从工作流实例中调用这些服务。本章则演示如何从这些本地服务中触发事件并由工作流来处理这些事件，另外还将讨论自定义事件驱动活动的开发以及如何使用工作流队列进行通信。

第 8 章：工作流持久性

工作流持久性允许自动保存正在运行中的工作流实例的状态，并在将来重新加载它们。持久性对长时间运行的工作流来说尤为重要。在使用持久性服务时，你可以从内存中卸载空闲和等待外部事件

的工作流。

第 9 章：状态机工作流

WF支持两种主要的工作流类型：顺序和状态机。到现在为止，你在本书中接触到的所有工作流都是顺序工作流，它们经常用来与系统交互。如果在设计时就知道准确的任务执行顺序，那么采用顺序工作流最合适。本章主要关注状态机工作流，经设计它可以轻松地响应外部事件。在解决有人类参与交互的工作流问题时，状态机工作流尤其有用，因为在这种情况下，准确的任务执行顺序往往不能在设计时决定。

第 10 章：事务和补偿

本章的目标是演示两种用来控制由工作流执行的工作的完整性和一致性的方法。事务允许将多个活动排列到一个工作逻辑单元中执行。当启用事务时，所有的工作只能被全部提交或者全部回滚，而不存在部分更新的问题。另一方面，补偿则是撤销已经完成的工作的过程。如果当某个活动成功完成之后，工作流才决定刚才的工作必须撤销的话，就需要进行补偿。

第 11 章：工作流规则

WF包含通用的规则引擎，它带来声明业务逻辑的另一种方法。作为简单语句或相关的数据断言来使用规则是最好的思路（相对于使用程序指令）。单独的规则组成了规则集，并且可以通过WF内置的规则引擎进行计算。每个规则都允许定义一些动作，并根据规则的结果（true或者false）执行相应的动作。

第 12 章：异常和错误处理

异常处理对任何应用程序来说都至关重要，WF提供了一种机制来允许声明性地处理异常。本章的目的是说明各种在工作流模型中处理异常的方法。本章的内容还包括取消处理程序；当执行中的活动被取消时，可以使用取消处理程序来执行一组活动。

第 13 章：高级自定义活动

本章提供了关于自定义活动开发的高级信息，探索了通用的组合活动以及长时间运行的自定义活动的开发。

第 14 章：动态工作流更新

大多数情况下，你会静态地定义工作流，然后在运行时创建它的实例。WF还提供了动态更新执行中的工作流的功能，从而改变工作流的内部结构。本章演示了如何从宿主应用程序和执行中的工作流内部动态更新工作流。

第 15 章：工作流跟踪

WF提供了跟踪框架来跟踪工作流的执行过程。跟踪框架支持可插拔的跟踪服务，可以根据自己的需求来实现这种服务。跟踪框架基于跟踪配置，而跟踪配置又允许自定义每一种工作流被跟踪的数量和数据类型。本章的重点是使用标准的跟踪服务和开发自定义跟踪服务。

第 16 章：Web 服务和 ASP.NET

WF允许在工作流内部声明性地访问Web服务。也可以将工作流作为Web服务公开给任何Web服务客户端。本章会通过ASP.NET Web Form应用程序中使用WF来演示这些主题。

第 17 章：工作流服务和 WCF 集成

本章的重点是.NET Framework 3.5中新增的WCF与WF的集成。这种集成的主要示例就是能够承载以工作流的方式实现的WCF服务。这就是工作流服务。从.NET 3.5开始，WF还允许在工作流中直接调用WCF服务。

第 18 章：工作流序列化和标记

本章的目的是演示如何使用工作流标记和序列化。每个工作流定义都可以用几种方式来声明和表示，其中就包括标记。标记用简单的XML格式声明了工作流，该格式不需要编译就可以直接被工作流运行时引擎解析并执行。使用标记的好处是可以非常容易地脱离Visual Studio来修改工作流定义，因为它不需要编译。

第 19 章：承载工作流设计器

第18章介绍了工作流序列化和标记，本章则演示如何创建自己的工作流设计器。WF包含了一些类来在应用程序中承载工作流设计器。本章演示了一个承载工作流设计器的应用程序，它允许定义和修改仅以标记表示的工作流。

本书要求

WF最初是.NET 3.0的一部分，开发环境则是通过插件的形式添加到Visual Studio 2005中。现在Visual Studio 2008已经内置了对WF的支持，而且.NET 3.5还包含了一些新的WF功能，例如支持工作流服务。

在运行本书提供的示例之前，需要安装一些软件。至少需要：

- Visual Studio 2008 专业版、标准版或Team System版（Express版不支持WF开发）；
- .NET 3.5运行时（与Visual Studio 2008同时安装）。

注意上述软件支持的操作系统。通常来说，可以使用Windows XP Services Pack 2及之后的操作系统。

Visual Studio 2008和.NET 3.5的组合版允许使用WF的所有最新功能和本书的全部示例，而且其安装也很简单。

当然，也可以使用Visual Studio 2005和.NET 3.0来进行WF开发。但如果这样做的话，会被限制只能使用WF最初版本的功能。请查阅第1章以了解使用Visual Studio 2005需要安装的软件明细。

Microsoft .NET Framework开发中心（<http://msdn2.microsoft.com/en-us/netframework/default.aspx>）是一个不错的起点，在这个站点可以找到需要的任何文件。

本书源代码

我发现学习并掌握一门新技术的最佳途径就是动手练习示例。基于这个原因，本书包含了许多示例源代码。面对没有在书中印出所有源代码的技术图书，我不止一次地感到懊恼。尽管代码可以通过

下载的方式得到，但阅读的时候就会需要一台电脑在手边。这要是在海滩上就无能为力了。因此，我认为有必要将实际构建和执行示例所需的代码全部展现出来。

当你准备运行这些示例代码时，不必自己输入，可以访问Apress网站（<http://www.apress.com>），然后在Source Code/Download栏目中下载示例代码；本书的所有源代码都可以在这里下载^①。我将所有可下载的源代码归类到每一章单独的文件夹中，并且每一章都有一个Visual Studio解决方案。唯一的例外是一个共享项目，本书的许多章节都引用了它。我建议你最好也像本书的示例一样按章节分离你的代码。

与作者联系

如果你对本书或者WF有任何问题，都可以发送电子邮件到workflow@bukovics.com来告诉我。为了避免邮件被垃圾邮件过滤器屏蔽，请确保邮件主题中包含ProWF字样。

^① 读者也可以访问图灵公司网站<http://www.turingbook.com>，免费注册后，即可在本书页面上下载源代码。

致 谢

许多人值得我去感谢。最重要的是，我必须感谢我的家人，Teresa和Brennen。编写本书又一次占用了我应当给予他们的许多时间。我很抱歉不得不打破我“不再出书”的承诺。我非常爱你们，也要感谢你们在我又编写一本书时再次给予我的耐心和支持。

还要感谢使本书的第1版获得成功的读者。我真的惊讶于有那么多读者花时间给我发电子邮件。许多人发邮件来向我提问题，还有一些只是想告诉我他们有多么喜欢这本书。正是因为你们的支持才坚定我编写本书的信心。

这是我和Apress合作出版的第3本书，并且又一次让我觉得本书的编写过程很愉快。我由衷地感谢整个Apress团队，他们的专业精神让我钦佩。

这次Matthew Moodie是我的编辑。Matt检查了每一章，并提出了能使本书更优秀的建议。他还发现了一些差点被忽略的技术错误。Matt，感谢你为了让本书尽可能优秀所付出的努力。

Beth Christmas作为项目编辑，负责协调所有出版前的工作及进度控制。一旦发现有可能提前出版本书，她就及时调整了工作计划，力争早日出书。Beth，感谢你成就了一个没有出问题的项目。

还要感谢Kim Wimpsett、Ellie Fountain、Susan Glinert和Brian Myers。Kim是本书的文字编辑。我由衷地感谢她在提高我的写作水平的同时还允许我保留自己的风格。Ellie是生产编辑，Susan负责将我的原始文字排版到你现在看到的纸面上，这是多么伟大的工作啊！Brian是本书的技术审稿人。如果你能找到错误的话，那都完全是他的失误——开个玩笑。Brian，感谢你帮忙确保本书尽可能准确。

3年3本书，我不确定这是否可行，但我想我发现了一种规律。

目 录

第 1 章 WF 快速入门	1	2.2.2 运行时引擎	29
1.1 为什么需要工作流	1	2.2.3 运行时服务	30
1.1.1 工作流各不相同	1	2.2.4 设计时工具	30
1.1.2 为什么选择WF	2	2.2.5 .NET 2.0运行时	30
1.2 开发环境	3	2.3 运行时环境	30
1.3 Hello Workflow	3	2.3.1 应用程序宿主进程	31
1.3.1 创建工作流项目	4	2.3.2 运行时引擎	32
1.3.2 工作流设计器	6	2.3.3 已注册的核心服务	33
1.3.3 使用工作流活动	6	2.3.4 已注册的本地服务	36
1.3.4 输入代码	7	2.3.5 工作流实例	37
1.3.5 承载工作流运行时	8	2.4 设计时环境	39
1.3.6 运行应用程序	10	2.4.1 工作流创作模式	39
1.4 传递参数	10	2.4.2 项目模板	40
1.4.1 声明属性	11	2.4.3 工作流设计器	41
1.4.2 在运行时中传递值	11	2.4.4 活动设计器	45
1.5 作出决策	13	2.4.5 规则条件编辑器	46
1.5.1 创建工作流库	13	2.4.6 Ruleset编辑器	48
1.5.2 添加工作流属性	13	2.4.7 工作流调试器	51
1.5.3 添加IfElse活动	14	2.4.8 命令行工作流编译器	51
1.5.4 添加计算逻辑	17	2.4.9 命令行通信活动生成器	52
1.5.5 创建计算器客户端	19	2.5 小结	52
1.5.6 测试和调试计算器	23	第 3 章 活动	53
1.6 小结	25	3.1 理解活动	53
第 2 章 基础结构概述	26	3.1.1 活动的两个访问者	53
2.1 工作流类型	26	3.1.2 类层次结构	54
2.1.1 顺序工作流	26	3.2 探讨标准活动	55
2.1.2 状态机工作流	26	3.2.1 自定义工作流逻辑	55
2.1.3 选择工作流类型	27	3.2.2 流控制	55
2.2 基础结构交付项	28	3.2.3 状态管理	57
2.2.1 类库和基础结构	28	3.2.4 事件处理	58

3.2.5 本地服务通信	59	4.3.2 实现 workflow 管理器类	119
3.2.6 规则	59	4.3.3 承载 workflow 管理器	127
3.2.7 Web 服务	59	4.3.4 通过 App.config 配置运行时	133
3.2.8 工作流服务	60	4.4 控制工作流实例	136
3.2.9 事务、补偿和同步	61	4.5 同步的工作流执行	139
3.2.10 异常和错误处理	62	4.6 小结	142
3.3 标准活动摘要	62	第 5 章 流控制	143
3.4 添加工作流逻辑	64	5.1 条件类型	143
3.5 使用 CodeActivity	64	5.2 使用 IfElseActivity	144
3.5.1 创建项目	65	5.2.1 使用代码条件的 IfElseActivity	144
3.5.2 定义工作流参数	65	5.2.2 使用规则条件的 IfElseActivity	150
3.5.3 验证账户	69	5.3 使用 WhileActivity	153
3.5.4 验证产品	70	5.3.1 实现工作流	153
3.5.5 录入订单	70	5.3.2 测试工作流	156
3.5.6 运行工作流	74	5.4 使用 ParallelActivity	158
3.5.7 总结此方法	77	5.4.1 实现工作流	159
3.6 开发自定义活动	77	5.4.2 测试工作流	162
3.6.1 为什么使用自定义活动	77	5.4.3 添加 DelayActivity	164
3.6.2 两种访问者的设计	78	5.5 使用 ReplicatorActivity	165
3.6.3 创建项目	78	5.5.1 实现顺序工作流	166
3.6.4 实现账户验证活动	78	5.5.2 测试顺序工作流	169
3.6.5 实现产品验证活动	82	5.5.3 实现并行工作流	170
3.6.6 实现订单录入活动	84	5.5.4 测试并行工作流	174
3.6.7 定义工作流参数	87	5.5.5 使用 ReplicatorActivity 事件	176
3.6.8 定义工作流	88	5.5.6 中断执行	176
3.6.9 运行工作流	92	5.6 使用 ConditionedActivityGroup	176
3.6.10 总结此方法	95	5.6.1 实现工作流	177
3.7 增强设计体验	96	5.6.2 测试工作流	180
3.7.1 验证活动	96	5.7 使用 InvokeWorkflowActivity	182
3.7.2 自定义工具箱行为	100	5.8 使用 TerminateActivity	183
3.7.3 自定义设计器	104	5.9 使用 SuspendActivity	183
3.8 小结	108	5.10 小结	183
第 4 章 承载工作流运行时	109	第 6 章 本地服务	184
4.1 承载概述	109	6.1 了解本地服务	184
4.2 简单的工作流承载	110	6.1.1 实现本地服务	185
4.2.1 实现测试工作流	110	6.1.2 使用本地服务	185
4.2.2 实现简单宿主应用程序	112	6.2 实现并使用本地服务	186
4.3 改进的工作流承载	117		
4.3.1 实现工作流实例包装	117		

6.2.1 实现账户类	186	7.6.1 了解 workflow 队列	237
6.2.2 声明服务约定	186	7.6.2 事件驱动活动的需求	238
6.2.3 实现本地服务类	187	7.6.3 实现 ProcessGuessActivity	239
6.2.4 实现 workflow	188	7.6.4 实现 SendHintActivity	246
6.2.5 测试 workflow	191	7.6.5 修改 workflow	248
6.3 从 App.config 加载	193	7.6.6 修改宿主应用程序	249
6.4 使用自定义活动	195	7.6.7 测试 workflow	252
6.4.1 实现自定义活动	195	7.7 小结	253
6.4.2 修改 workflow	198		
6.4.3 测试 workflow	199	第 8 章 workflow 持久性	254
6.5 使用 CallExternalMethodActivity	199	8.1 了解持久性	254
6.5.1 实现 workflow	199	8.1.1 持久化 workflow 的原因	254
6.5.2 测试 workflow	201	8.1.2 持久性概述	255
6.6 小结	201	8.2 使用 SqlWorkflowPersistenceService	256
第 7 章 事件驱动活动	202	8.2.1 为持久性准备数据库	257
7.1 使用事件驱动活动	202	8.2.2 实现本地服务	257
7.2 使用 HandleExternalEventActivity	203	8.2.3 实现 workflow	259
7.2.1 创建项目	204	8.2.4 实现宿主应用程序	261
7.2.2 实现事件参数类	204	8.2.5 测试应用程序	271
7.2.3 定义服务接口	205	8.3 实现自定义持久性服务	273
7.2.4 实现本地服务	206	8.3.1 了解抽象方法	274
7.2.5 实现 workflow	209	8.3.2 实现服务	276
7.2.6 实现宿主应用程序	213	8.3.3 测试自定义服务	282
7.2.7 测试 workflow	217	8.4 小结	284
7.3 生成通信活动	218	第 9 章 状态机 workflow	285
7.3.1 生成活动	218	9.1 了解状态机	285
7.3.2 修改 workflow	219	9.1.1 为什么需要状态机 workflow	285
7.4 手动控制关联	220	9.1.2 状态机概述	286
7.4.1 实现事件参数类	221	9.2 实现状态机 workflow	290
7.4.2 定义服务接口	221	9.2.1 设计 CarStateMachine	290
7.4.3 实现本地服务	222	9.2.2 定义本地服务接口	291
7.4.4 实现 workflow	223	9.2.3 实现本地服务	292
7.4.5 测试 workflow	225	9.2.4 实现 workflow	295
7.5 使用 EventHandlingScopeActivity	227	9.2.5 实现宿主应用程序	302
7.5.1 定义服务接口	228	9.2.6 测试应用程序	308
7.5.2 实现本地服务	228	9.3 消除重复的事件处理程序	309
7.5.3 实现 workflow	230	9.3.1 重构 CarWorkflow	309
7.5.4 测试 workflow	234	9.3.2 测试修改后的 workflow	310
7.6 开发自定义事件驱动活动	236	9.4 标识现有事件	311

9.4.1 询问 workflow 队列	311	11.3 使用 PolicyActivity 来定义规则	369
9.4.2 修改 CarStateMachine	311	11.3.1 实现 SalesItem 类	369
9.4.3 测试应用程序	313	11.3.2 声明规则	370
9.5 访问 workflow 运行时信息	314	11.3.3 测试 workflow	375
9.5.1 修改 CarStateMachine	314	11.4 跟踪规则	379
9.5.2 测试应用程序	316	11.5 调整规则顺序	381
9.6 小结	316	11.5.1 设置规则优先级	381
第 10 章 事务和补偿	317	11.5.2 测试 workflow	381
10.1 事务	317	11.6 在规则中使用方法	383
10.1.1 事务的方式	317	11.6.1 添加操作方法	383
10.1.2 WF 对事务的支持	318	11.6.2 在规则集中使用方法	384
10.2 使用 TransactionScopeActivity	319	11.6.3 标识间接关系	385
10.2.1 实现 AccountAdjustment- Activity	322	11.7 在代码中执行规则集	386
10.2.2 实现 AccountTransfer- Workflow	326	11.7.1 实现 SellItemSerialized- Workflow	386
10.2.3 测试 workflow	329	11.7.2 测试 workflow	389
10.3 了解补偿	334	11.8 在代码中构造规则集	389
10.4 使用可补偿活动	335	11.8.1 实现 SellItemInCode- Workflow	389
10.4.1 实现 InventoryUpdate- Activity	335	11.8.2 测试 workflow	392
10.4.2 实现 OrderDetailActivity	339	11.9 小结	393
10.4.3 实现 OrderEntryWorkflow	344	第 12 章 异常和错误处理	394
10.4.4 测试 workflow	350	12.1 理解 workflow 异常处理	394
10.5 参与工作批次	353	12.2 探讨默认行为	396
10.6 使用 IPendingWork 接口	353	12.2.1 实现 ExceptionWorkflow	396
10.6.1 实现本地服务	354	12.2.2 测试 workflow	398
10.6.2 实现 BatchedWorkWorkflow	356	12.3 使用 FaultHandlerActivity	400
10.6.3 测试 workflow	357	12.3.1 处理 ArithmeticException	400
10.7 小结	359	12.3.2 处理 DivideByZeroException	403
第 11 章 workflow 规则	361	12.4 抑制异常	405
11.1 了解 workflow 规则	361	12.5 重新抛出异常	407
11.1.1 规则的组成	362	12.6 补偿和异常	408
11.1.2 使用规则的原因	362	12.6.1 实现 CompensateWorkflow	408
11.2 在 WF 中使用规则	363	12.6.2 测试 workflow	410
11.2.1 定义规则	363	12.7 使用 CancellationHandlerActivity	412
11.2.2 定义规则集	366	12.7.1 实现 CancelHandler- Workflow	412
11.2.3 使用特性来标识依赖项	368	12.7.2 测试 workflow	414
		12.8 小结	414

第 13 章 高级自定义活动	415	14.5.1 实现DynamicCondition- Workflow	463
13.1 开发专用的复合活动	415	14.5.2 实现宿主应用程序	464
13.1.1 开发自定义活动	416	14.5.3 测试工作流	468
13.1.2 开发工作流	417	14.6 替换规则条件	468
13.1.3 测试活动	418	14.6.1 修改RuleDefinitions	468
13.2 开发通用的复合活动	420	14.6.2 修改宿主应用程序	470
13.2.1 了解活动执行计划	420	14.6.3 测试修改后的应用程序	472
13.2.2 复合活动的执行模式	421	14.7 小结	472
13.2.3 实现顺序活动	422	第 15 章 工作流跟踪	473
13.2.4 实现测试工作流	426	15.1 了解工作流跟踪	473
13.2.5 测试顺序活动	428	15.1.1 跟踪服务	473
13.3 开发优先级顺序活动	429	15.1.2 跟踪事件类型	474
13.3.1 实现优先级顺序活动	429	15.1.3 自定义跟踪配置	475
13.3.2 实现扩展提供程序	432	15.1.4 使用跟踪数据	475
13.3.3 实现自定义设计器	434	15.1.5 受益于工作流跟踪	476
13.3.4 实现测试工作流	435	15.2 使用SqlTrackingService	476
13.3.5 测试优先级顺序活动	436	15.2.1 准备跟踪SQL数据库	477
13.4 开发长时间运行的活动	436	15.2.2 开发测试工作流	477
13.4.1 实现本地服务	437	15.2.3 开发宿主应用程序	479
13.4.2 实现自定义活动	440	15.2.4 执行宿主应用程序	481
13.4.3 实现测试工作流	443	15.2.5 获取跟踪数据	482
13.4.4 测试长时间运行的活动流	443	15.3 创建用户跟踪点	489
13.5 小结	445	15.3.1 增强TrackingExample- Workflow	490
第 14 章 动态工作流更新	446	15.3.2 测试修改后的工作流	490
14.1 了解动态更新	446	15.4 跟踪规则计算	491
14.1.1 使用动态更新的原因	447	15.4.1 实现TrackingRules- Workflow	491
14.1.2 应用动态更新	447	15.4.2 测试工作流	492
14.1.3 防止动态更新	448	15.5 使用自定义跟踪配置提取数据	493
14.2 从宿主应用程序中应用更新	449	15.5.1 使用跟踪配置	494
14.2.1 实现DynamicWorkflow	449	15.5.2 实现TrackingProfile- Helper	495
14.2.2 实现自定义活动	450	15.5.3 创建跟踪配置	499
14.2.3 实现宿主应用程序	452	15.5.4 测试跟踪配置	501
14.2.4 测试工作流	456	15.6 维护SQL跟踪数据库	502
14.3 限制动态更新	456	15.6.1 分区	502
14.4 从工作流中应用更新	458	15.6.2 设置分区间隔时间	502
14.4.1 实现SelfUpdatingWorkflow	458		
14.4.2 实现宿主应用程序	461		
14.4.3 测试工作流	462		
14.5 更新规则条件	463		

15.6.3 自动或手动分区	502	17.1.2 配置终结点和绑定	542
15.6.4 访问分区数据	503	17.1.3 承载和配置	543
15.6.5 分离或删除分区	503	17.2 了解 workflow 服务	543
15.7 开发跟踪服务	503	17.2.1 使用 ReceiveActivity	543
15.7.1 实现跟踪通道	504	17.2.2 使用 SendActivity	544
15.7.2 实现跟踪服务	506	17.2.3 承载和配置	545
15.7.3 测试自定义跟踪服务	510	17.2.4 实例化和上下文管理	547
15.8 小结	512	17.3 开发无状态的工作流服务	548
第 16 章 Web 服务和 ASP.NET	513	17.3.1 定义服务约定	548
16.1 将 workflow 发布为 Web 服务	513	17.3.2 实现工作流	549
16.1.1 了解 Web 服务活动	513	17.3.3 配置 workflow 服务	550
16.1.2 发布和配置	514	17.3.4 测试 workflow 服务	551
16.2 开发 Web 服务 workflow	515	17.4 IIS 承载和客户端开发	552
16.2.1 定义 Web 服务接口	515	17.4.1 创建 IIS 虚拟目录	553
16.2.2 定义 MathServiceWorkflow	516	17.4.2 创建 .svc 文件	553
16.2.3 发布工作流	518	17.4.3 配置 Web.config 文件	553
16.2.4 测试工作流	519	17.4.4 部署服务	554
16.3 返回 Web 服务错误	521	17.4.5 开发客户端应用程序	555
16.3.1 修改 MathServiceWorkflow	522	17.5 自承载的工作流服务	557
16.3.2 测试修改后的 Web 服务	523	17.5.1 开发服务宿主	558
16.4 开发状态化的 Web 服务	523	17.5.2 配置服务宿主	559
16.4.1 定义 Web 服务接口	524	17.5.3 更改客户端应用程序	560
16.4.2 实现 MathServiceStatefulWorkflow	525	17.5.4 测试服务宿主	561
16.4.3 发布新工作流	528	17.6 定义 workflow 优先的服务	561
16.4.4 测试 Web 服务	528	17.7 开发状态化的 workflow 服务	562
16.5 在 workflow 中调用 Web 服务	528	17.7.1 定义服务约定	563
16.5.1 实现 InvokeWebServiceWorkflow	529	17.7.2 实现工作流	564
16.5.2 测试工作流	532	17.7.3 在 IIS 中承载 workflow 服务	566
16.6 在 ASP.NET 中使用工作流	533	17.7.4 开发客户端应用程序	567
16.6.1 实现 DivideNumberWorkflow	534	17.7.5 测试 workflow 服务	569
16.6.2 实现 UseWorkflowWebsite	534	17.8 使用 SendActivity	570
16.6.3 测试工作流	538	17.8.1 实现工作流	570
16.7 小结	539	17.8.2 开发客户端应用程序	573
第 17 章 workflow 服务和 WCF 集成	540	17.8.3 测试工作流	575
17.1 了解 WCF	540	17.9 上下文和对话	575
17.1.1 定义服务约定	541	17.9.1 定义服务约定	577
17.1.2 配置终结点和绑定	542	17.9.2 实现库存工作流	579
17.1.3 承载和配置	543	17.9.3 实现库存宿主	583
17.2 了解 workflow 服务	543	17.9.4 实现客户端工作流	586
17.2.1 使用 ReceiveActivity	543	17.9.5 实现客户端宿主	592
17.2.2 使用 SendActivity	544		
17.2.3 承载和配置	545		
17.2.4 实例化和上下文管理	547		
17.3 开发无状态的工作流服务	548		
17.3.1 定义服务约定	548		
17.3.2 实现工作流	549		
17.3.3 配置 workflow 服务	550		
17.3.4 测试 workflow 服务	551		
17.4 IIS 承载和客户端开发	552		
17.4.1 创建 IIS 虚拟目录	553		
17.4.2 创建 .svc 文件	553		
17.4.3 配置 Web.config 文件	553		
17.4.4 部署服务	554		
17.4.5 开发客户端应用程序	555		
17.5 自承载的工作流服务	557		
17.5.1 开发服务宿主	558		
17.5.2 配置服务宿主	559		
17.5.3 更改客户端应用程序	560		
17.5.4 测试服务宿主	561		
17.6 定义 workflow 优先的服务	561		
17.7 开发状态化的 workflow 服务	562		
17.7.1 定义服务约定	563		
17.7.2 实现工作流	564		
17.7.3 在 IIS 中承载 workflow 服务	566		
17.7.4 开发客户端应用程序	567		
17.7.5 测试 workflow 服务	569		
17.8 使用 SendActivity	570		
17.8.1 实现工作流	570		
17.8.2 开发客户端应用程序	573		
17.8.3 测试工作流	575		
17.9 上下文和对话	575		
17.9.1 定义服务约定	577		
17.9.2 实现库存工作流	579		
17.9.3 实现库存宿主	583		
17.9.4 实现客户端工作流	586		
17.9.5 实现客户端宿主	592		

17.9.6 测试应用程序.....	595	18.8 编译包含规则的工作流.....	625
17.10 小结.....	596	18.9 在命令行中编译.....	628
第 18 章 工作流序列化和标记.....	598	18.10 反序列化标记.....	628
18.1 了解工作流创作模式.....	598	18.11 小结.....	632
18.1.1 仅限代码创作模式.....	598	第 19 章 承载工作流设计器.....	633
18.1.2 代码分离创作模式.....	599	19.1 工作流设计器.....	633
18.1.3 无代码创作模式.....	601	19.1.1 设计器命名空间.....	634
18.2 开发仅限代码工作流.....	602	19.1.2 设计器类.....	634
18.2.1 实现工作流.....	603	19.1.3 设计器服务.....	635
18.2.2 测试工作流.....	604	19.2 构建设计器应用程序.....	635
18.2.3 查看生成的代码.....	605	19.2.1 创建设计器项目.....	636
18.3 开发代码分离工作流.....	607	19.2.2 实现WorkflowLoader.....	637
18.3.1 实现工作流.....	607	19.2.3 实现WorkflowMenuService.....	645
18.3.2 测试工作流.....	608	19.2.4 实现WorkflowEvent- BindingService.....	649
18.3.3 查看标记.....	608	19.2.5 实现EventProperty- Descriptor.....	653
18.4 开发无代码工作流.....	609	19.2.6 实现WorkflowProperty- ValueService.....	657
18.4.1 实现工作流基类.....	609	19.2.7 实现WorkflowToolbox- Service.....	659
18.4.2 实现自定义活动.....	610	19.2.8 实现WorkflowDesigner.....	668
18.4.3 定义工作流标记.....	611	19.2.9 实现MainForm.....	674
18.4.4 增强WorkflowRuntime- Manager.....	612	19.2.10 实现AssemblyReference- Form.....	679
18.4.5 测试工作流.....	614	19.2.11 实现NewWorkflowForm.....	681
18.5 在无代码工作流中使用规则.....	615	19.3 使用设计器.....	683
18.5.1 定义规则条件.....	616	19.4 小结.....	687
18.5.2 修改工作流标记.....	617		
18.5.3 测试工作流.....	618		
18.6 序列化到标记.....	618		
18.7 编译工作流.....	622		