

AspectJ Cookbook



AspectJ Cookbook™ 中文版

Russ Miles 著

程利剑 译

O'REILLY®



清华大学出版社

AspectJ Cookbook™

中文版

Russ Miles 著

程利剑 译

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Paris • Sebastopol • Taipei • Tokyo

O'Reilly Media, Inc. 授权清华大学出版社出版

清华大学出版社

图书在版编目 (CIP) 数据

AspectJ Cookbook™ 中文版 / (美) 米尔斯 (Miles, R.) 著; 程利剑译. —北京: 清华大学出版社, 2006. 11

书名原文: AspectJ Cookbook

ISBN 7-302-14011-1

I. A… II. ①米… ②程… III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 121805 号

北京市版权局著作权合同登记

图字: 01-2006-5903 号

Copyright ©2004 by O'Reilly Media, Inc.

Authorized Simplified Chinese translation edition, by O'Reilly Media, Inc., is published by Tsinghua University Press, 2006. Authorized translation of the original English edition, 2004 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

本书之英文原版由 O'Reilly Media, Inc. 于 2004 年出版。

本中文简体翻译版由 O'Reilly Media, Inc. 授权清华大学出版社于 2006 年出版。此翻译版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc. 的许可。

版权所有, 未经书面许可, 本书的任何部分和全部不得以任何形式复制。

版权所有, 翻印必究。举报电话: 010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

本书防伪标签采用特殊防伪技术, 用户可通过在图案表面涂抹清水, 图案消失, 水干后图案复现; 或将表面膜揭下, 放在白纸上用彩笔涂抹, 图案在白纸上再现的方法识别真伪。

书 名 / AspectJ Cookbook™ 中文版

书 号 / ISBN 7-302-14011-1/TP · 8415

责任编辑 / 常晓波

封面设计 / Emma Colby, 张健

出版发行 / 清华大学出版社 (www.tup.tsinghua.edu.cn)

地 址 / 北京清华大学学研大厦 (邮政编码 100084)

经 销 / 各地新华书店

印 刷 / 清华大学印刷厂

装 订 / 三河市李旗庄少明装订厂

开 本 / 178 毫米 × 233 毫米 16 开本 20.75 印张 451 千字

版 次 / 2006 年 11 月第一版 2006 年 11 月第 1 次印刷

印 数 / 1 ~ 3000 册

定 价 / 40.00 元 (册)

作者简介

Russ Miles 是一位具有 10 年编程经验的开发人员，其涉足的语言包括：从 Atari 800XL 上的 Basic 到 Mac OS X 上的 Mono。Russ 是一位虔诚的 Mac 信徒，近年来，他专注于多平台环境下复杂的分布式系统开发，包括 Windows、Linux、Unix 和 Mac OS X 上多种风格的开发。Russ 目前是 General Dynamic UK Limited 的一位高级技术专家，还是许多项目的顾问，包括从内部工具开发到国际标准出版物。只要有可能，Russ 就会回到编程领域中来，并为多个开源项目做出了贡献。他还创立了他自己的一些开源项目。

在 Russ 意外地参加英国计算机协会的高级编程专家组的一次错误的会议时，他“发现”了面向方面编程。Russ 无法记起他参加会议的实际目的，但是他很幸运，他参加了一次名为“Aspect-Oriented Programming Using AspectJ”的讲座，演讲人是 IBM 的 Adrian Colyer。Russ 对面向方面的潜力和 AspectJ 的成熟印象深刻，他关于软件体系结构的观点也发生了永久性的变化。他立即利用这种新方法，改变了他及其团队开发软件的方式。很快两年过去了，Russ 仍在开发和倡导面向方面的方法。本书是 Russ 幸运地与 AspectJ 倡导者第一次碰面的结晶。

Russ 持有格林威治大学计算机科学的优等学士学位，目前是牛津大学 Kellogg 学院的成员，他在这里完成了他的计算机科学硕士学位。他还为 ONJava.com 和 java.net 写了许多文章，并且是 O'Reilly 的技术顾问。

封面介绍

本书封面上的动物是一只长尾猴。长尾猴是一种小毛猴（其大小不会超过硕大的老鼠），居住在南美洲的热带和亚热带森林里。它们是树栖动物，并且异常敏捷。它们强有力的后腿、灵巧得像爪子一样的手以及特别长的尾巴，使得它们可以轻而易举地从一根树枝摆动到另一根树枝。

长尾猴不太挑食。它们偏爱树液（树越粗越好），但是它们也享用蜥蜴、青蛙、蜗牛、昆虫、水果和花蜜。

O'Reilly Media, Inc. 介绍

为了满足读者对网络和软件技术知识的迫切需求，世界著名计算机图书出版机构 O'Reilly Media, Inc. 授权清华大学出版社，翻译出版一批该公司久负盛名的英文经典技术专著。

O'Reilly Media, Inc. 是世界上在 Unix、X、Internet 和其他开放系统图书领域具有领导地位的出版公司，同时也是联机出版的先锋。

从最畅销的《The Whole Internet User's Guide & Catalog》（被纽约公共图书馆评为二十世纪最重要的 50 本书之一）到 GNN（最早的 Internet 门户和商业网站），再到 WebSite（第一个桌面 PC 的 Web 服务器软件），O'Reilly Media, Inc. 一直处于 Internet 发展的最前沿。

许多书店的反馈表明，O'Reilly Media, Inc. 是最稳定的计算机图书出版商——每一本书都一版再版。与大多数计算机图书出版商相比，O'Reilly Media, Inc. 具有深厚的计算机专业背景，这使得 O'Reilly Media, Inc. 形成了一个非常不同于其他出版商的出版方针。O'Reilly Media, Inc. 所有的编辑人员以前都是程序员，或者是顶尖级的技术专家。O'Reilly Media, Inc. 还有许多固定的作者群体——他们本身是相关领域的技术专家、咨询专家，而现在编写著作，O'Reilly Media, Inc. 依靠他们及时地推出图书。因为 O'Reilly Media, Inc. 紧密地与计算机业界联系着，所以 O'Reilly Media, Inc. 知道市场上真正需要什么图书。

目录

前言	1
第 1 章 面向方面概述	7
面向方面简史	7
AspectJ	7
面向方面的定义	8
在哪里找到更多信息	10
第 2 章 初识 AspectJ	12
2.0 简介	12
2.1 安装 AspectJ	13
2.2 开发一个简单的方面	15
2.3 编译一个方面和多个 Java 文件	18
2.4 织入方面到 jar 中	19
2.5 在加载时织入方面	20
2.6 生成 Javadoc 文档	23
2.7 使用 Eclipse 编译一个 AspectJ 项目	26

2.8 选择织入 Eclipse 内的构建中的方面	30
2.9 使用 Ant 构建一个 AspectJ 项目	34
第 3 章 部署 AspectJ 应用程序	36
3.0 简介	36
3.1 部署命令行 AspectJ 应用程序	36
3.2 将 AspectJ 应用程序部署为全包含的可执行 JAR 文件	39
3.3 部署使用 AspectJ 的 Java Servlet	41
3.4 部署使用 AspectJ 的 JSP	46
3.5 部署使用 AspectJ 的 Axis Web 服务	51
第 4 章 捕获方法上的连接点	55
4.0 简介	55
4.1 捕获方法调用	55
4.2 捕获方法调用上传递的参数值	59
4.3 捕获方法调用的目标	61
4.4 当执行一个方法时捕获它	62
4.5 在执行方法时捕获 this 引用的值	64
第 5 章 捕获异常处理上的连接点	66
5.0 简介	66
5.1 捕获何时捕捉异常	66
5.2 捕获抛出的异常	68
5.3 捕获处理异常的对象	70
第 6 章 捕获通知上的连接点	72
6.0 简介	72
6.1 捕获何时执行通知	72
6.2 排除作为通知执行结果的连接点	74
6.3 在发出通知时展示原始连接点	77

第 7 章 捕获类和对象构造上的连接点	79
7.0 简介	79
7.1 捕获对构造函数的调用	79
7.2 在执行构造函数时捕获它	81
7.3 捕获何时初始化对象	83
7.4 捕获何时将要初始化一个对象	86
7.5 捕获何时初始化类	88
 第 8 章 捕获属性上的连接点	 90
8.0 简介	90
8.1 捕获何时访问对象的属性	90
8.2 捕获访问的字段值	93
8.3 捕获何时修改对象的字段	94
8.4 在修改字段时捕获它的值	96
 第 9 章 捕获程序作用域内的连接点	 98
9.0 简介	98
9.1 捕获特定类中的所有连接点	98
9.2 捕获特定包中的所有连接点	100
9.3 捕获特定方法内的所有连接点	101
 第 10 章 捕获基于控制流程的连接点	 104
10.0 简介	104
10.1 捕获通过初始连接点开始的程序控制流程内的所有连接点	104
10.2 捕获程序控制流程内的所有连接点，不包括初始连接点	107
 第 11 章 捕获基于对象类型的连接点	 110
11.0 简介	110
11.1 捕获何时 this 引用是一个特定的类型	110

11.2 捕获何时连接点的目标对象是特定的类型	113
11.3 捕获何时连接点的参数是某个数字、类型和次序	116
第 12 章 捕获基于布尔或复合表达式的连接点	120
12.0 简介	120
12.1 捕获何时连接点上的运行时条件评估为 true	120
12.2 使用逻辑 AND (&&) 结合切入点	122
12.3 使用逻辑 OR () 结合切入点	125
12.4 捕获所有未通过切入点声明指定的连接点	128
12.5 声明匿名切入点	129
12.6 重用切入点	131
第 13 章 定义通知	134
13.0 简介	134
13.1 访问类成员	134
13.2 访问连接点环境	136
13.3 在连接点之前执行通知	139
13.4 在连接点周围执行通知	140
13.5 在连接点之后无条件执行通知	143
13.6 仅在从连接点正常返回之后才执行通知	144
13.7 仅当连接点中引发了一个异常之后才执行通知	147
13.8 控制通知优先级	148
13.9 通知方面	150
第 14 章 定义方面实例化	152
14.0 简介	152
14.1 定义单件方面	152
14.2 在每个实例上定义一个方面	154
14.3 在每个控制流程上定义一个方面	158

第 15 章 定义方面的关系	161
15.0 简介	161
15.1 继承切入点定义	161
15.2 实现抽象切入点	163
15.3 把类继承进方面中	165
15.4 在类内部声明方面	167
第 16 章 增强类和编译器	170
16.0 简介	170
16.1 扩展现有的类	170
16.2 声明类之间的继承关系	171
16.3 使用方面实现接口	172
16.4 声明默认的接口实现	174
16.5 减轻异常的影响	175
16.6 扩展编译	177
第 17 章 实现创建型面向对象设计模式	179
17.0 简介	179
17.1 实现单件模式	180
17.2 实现原型模式	183
17.3 实现抽象工厂模式	185
17.4 实现工厂方法模式	187
17.5 实现生成器模式	189
第 18 章 实现结构型面向对象设计模式	191
18.0 简介	191
18.1 实现复合模式	191
18.2 实现享元模式	195
18.3 实现适配器模式	198

18.4 实现桥接模式	200
18.5 实现装饰模式	202
18.6 实现代理模式	205
第 19 章 实现行为型面向对象设计模式.....	210
19.0 简介	210
19.1 实现观察者模式	210
19.2 实现命令模式	214
19.3 实现迭代器模式	219
19.4 实现调停者模式	221
19.5 实现责任链模式	224
19.6 实现备忘录模式	229
19.7 实现策略模式	232
19.8 实现访问者模式	236
19.9 实现模板方法模式	240
19.10 实现状态模式	241
19.11 实现解释器模式	243
第 20 章 应用类和组件级方面.....	245
20.0 简介	245
20.1 验证传递给方法的参数	245
20.2 重写在构造函数调用上实例化的类	249
20.3 添加持久性到类中	252
20.4 应用模拟组件支持单元测试	256
第 21 章 应用应用程序级方面.....	261
21.0 简介	261
21.1 应用面向方面的跟踪	262
21.2 应用面向方面的日志记录	265

21.3 应用延迟加载	269
21.4 管理应用程序属性	274
第 22 章 应用企业级方面	280
22.0 简介	280
22.1 应用开发指导和规则	280
22.2 应用事务	283
22.3 应用资源池	286
22.4 使用 RMI 透明地远程访问类	290
22.5 应用安全策略	292
第 23 章 应用面向方面设计模式	294
23.0 简介	294
23.1 应用 Cuckoo's Egg 设计模式	295
23.2 应用 Director 设计模式	297
23.3 应用 Border Control 设计模式	300
23.4 应用策略设计模式	303
附录 AspectJ 运行时 API	307

前言

本书重点关注的是用AspectJ完成任务。面向方面的软件开发 (Aspect-Oriented Software Development, AOSD) 给纳入了面向对象方法的某些旧思想贴上了新的商标。这个相当复杂的定义实际上概括了一种灵巧的方法，用于解决一些传统的面向对象的问题。

虽然本书并没有深入研究面向方面背后的理论细节，但是提供了一个简短的概述，从而给你提供了一个有用的基础，用以支持构成本书余下部分的编码秘诀。这个编码秘诀将指导你如何为面向方面的开发建立系统，构建你的第一个小程序，并最终把面向方面应用于复杂的现实问题。

本书的目标是成为你案头的那些有用书籍之一，并定期被要求“给我展示如何执行那个任务”。你应该能够在需要时直接跳到所需的秘诀上，以与 O'Reilly Cookbook 系列的主旨即“取精去粗”方法保持一致。在记住这一点后，本书中介绍的主题包括：

- 简述面向方面和 AspectJ
- 在广泛的构建和目标环境中建立、认真处理和运行 AspectJ 程序
- 实际地检查针对 Java 语言的每个 AspectJ 扩展
- 将 AspectJ 应用于可从面向方面的方法中获益的现实软件开发问题

本书读者对象

虽然可以说本书足以满足有兴趣学习 AspectJ 和面向方面的任何人的需要，但是本书对于那些在面向对象设计方面经验相当丰富的人最有用，特别是当利用 Java 实现时则更是如此。可以用新的方法来解决一些传统的面向对象方法的问题，对新方法持开放态度的

任何人都应该会在本书中找到一些内容，这些内容鼓励他们在自己的应用程序中尝试 AspectJ。

关于本书

本书涵盖了以下主题：

- 概述使用 AspectJ 的面向方面的各种元素、切入点 (pointcut)、连接点 (join point) 和通知 (advice)
- 建立你自己的环境，并开始 AspectJ 开发
- 在多种不同的环境中运行基于 AspectJ 的软件
- 把 AspectJ 应用程序部署到变化的目标环境中
- 如何使用 AspectJ 支持的多种形式的切入点
- 如何使用 AspectJ 中可用的多种形式的通知
- 控制如何创建和销毁方面
- 使用 AspectJ 更改软件和支持工具的静态结构
- 使用 AspectJ 增强现有的面向对象的设计模式实现
- 将方面应用于组件、系统和企业领域内的现实问题

很少见到从头至尾一页不漏地阅读 cookbook 图书，但是关于每一节重点的概述是有用的，它们便于理解每一章的地位及其意义。这本 cookbook 被组织成 4 大部分，如图 P-1 所示。

第 1~3 章提供了一组秘诀，用于让你初识支持使用 AspectJ 开发的工具和环境。这几章介绍了如何使用命令行工具、Eclipse，甚至 ANT 来开发面向方面的应用程序，并通过展示如何部署到各种各样的目标环境（如 Java Servlet、Java Server Page 和 Web 服务）来结束这几章的介绍。

第 4~16 章提供了 AspectJ 语言的深度语法分析。这些章旨在提供一个有用、详尽的参考，以理解如何使用 AspectJ 语言添加到 Java 中的不同构造。

第 17~23 章充分展示了 AspectJ 超过传统的面向对象软件实现的优点。在这几章中，使用方面来增强和改进各种软件解决方案的设计，包括从设计模式实现到企业应用程序。第 23 章通过介绍一些面向方面的设计模式来结束全书的内容，这些设计模式是作为工具（如 AspectJ）逐渐发现的，从而把面向方面引入软件开发的主流。

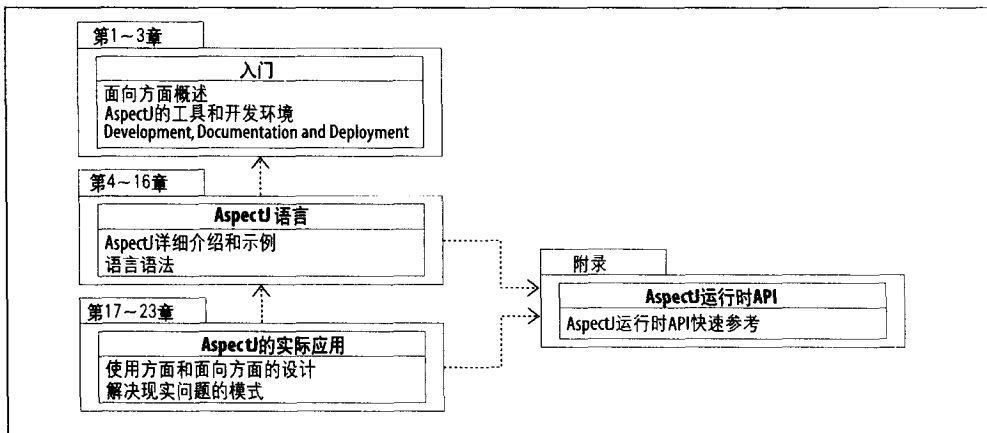


图 P-1：本书各章打算说明什么

既然你知道了本书介绍了哪些内容，现在就应该解释本书没有解释什么。如前所述，本书不是对 AspectJ 的理论研究。本书中的代码随处可见，以确保每个领域都有实际的关注。有许多优秀的在线文章，并且它们会逐渐出现在研究 OA 的更关注理论的印刷品中，因此，本书不会尝试论及这些方面。

本书也没有声明是对 AspectJ 的简要介绍。AspectJ 处在一种非常稳定的状态中，从而具有早期的实现，但是甚至现在对该语言和技术的扩展仍在考虑中，因此本书中的某些领域可能比其他一些领域更成熟。为了尝试确保本书具有尽可能最长的寿命，在编写本书时，只包括了最稳定的领域。

本书所做的假定

下面假定你具有对面向方面开发的相关支持工具、语言以及概念的了解和使用它们的经验。

- 假定你具有 Java 语言的相当高级的知识
- 假定你的系统上具有现有的 Java Source Development Kit (Java 源代码开发工具包, SDK)，它至少应该是 JSDK 1.2，最好是 1.4+
- 假定你对 Java Server Page、Java Servlet 和 Web 服务（确切地讲，是 Apache Axis）编程有一定程度的了解，并有一些使用它们的经验
- 假定你具有一些 UML 知识（它们是有用的，但是不必理解图）

排版约定

本书使用了以下排版约定：

斜体字 (*Italic*)

表示新出现的名词、URL、电子邮件地址、文件名、文件扩展名、路径、目录以及 Unix 实用程序。

等宽字体 (`Constant Width`)

表示命令、选项、分支、变量、类型、类、命名空间、方法、模块、属性、参数、值、对象、时间、时间处理程序和 XML 标记。

等宽斜体 (*Constant Width Italic*)

显示应被用户提供的值替换的文本。

等宽黑体 (**Constant Width Bold**)

在代码示例中强调重要文本内容。

使用代码示例

本书用于帮助你完成工作。一般而言，你可能在程序和文档中使用本书中的代码。你不必联系我们以获得许可，除非你正在复制代码的重要部分。例如，编写一个使用了本书中多块代码的程序不需要许可。销售或分发 O'Reilly 书籍中示例的 CD-ROM 确实需要许可。通过引用本书和引用示例代码来解答问题不需要许可。把本书中大量的示例代码纳入到你的产品文档中确实需要许可。

我们感谢但不要求注明出处。在注明出处时，通常要包括标题、作者、出版社和 ISBN。例如：“*AspectJ Cookbook* by Russ Miles, Copyright 2005 O'Reilly Media, Inc., 0-596-00654-3”。

如果你感到对代码示例的使用不在公平使用或者上面给出的许可之内，可通过 permissions@oreilly.com 自由地与我们联系。

我们很想听到你的建议

我们在 O'Reilly 尽最大努力确保本书内的秘诀是对作者最佳能力的测试和验证。不过，AspectJ 是一种快速发展的技术，其支持工具集或语言本身的某些特性可能发生了变化，或者我们自己犯了错误。如果是这样，请通过以下地址把关于本书的评论和问题发送给出版社：

美国：

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472

中国：

100080 北京市海淀区知春路 49 号希格玛公寓 B 座 809 室
奥莱理软件（北京）有限公司

本书的网页上列出了勘误表、示例和任何额外的信息。可登录以下网址查询：

<http://www.oreilly.com/catalog/aspectjckbk>

<http://www.oreilly.com.cn/book.php?bn=7-302-14011-1>

如果想要发表关于本书的评论和技术问题，请发邮件至：

bookquestions@oreilly.com

info@mail.oreilly.com.cn

关于图书、会议、资源中心和 O'Reilly 网络的更多信息，请查看我们的站点：

<http://www.oreilly.com>

<http://www.oreilly.com.cn>

致谢

无论如何，永远都不能把一本书的成果归于如此之少的几个人。本书是一个人编写的，但是得到了太多人的支持，我可以记住在这里提到的只是其中的一部分人，因此，如果漏掉了某个人，我预先表达我的歉意；我没有忘记你们。这只是由于我设法（但是未能）避免奥斯卡式的演讲。

我在这里有机会向那些使我集中注意力并支持我编写我的第一本书的人表示感谢。首先（并且是最重要的）必然是感谢我的家庭。感谢妈妈和爸爸，你们用无尽的笑容、温暖和理解见证了我通过最艰难的时刻。感谢 Bobs、Rich、Ad，甚至还要感谢 Aimee，你们总是让我保持做人的正道。你们全都帮助我完成这个项目，并让我保持脚踏实地，因此，我无论怎么感谢你们都是不够的，但是我将很开心地尝试这样做。

感谢我的编辑 Brett McLaughlin 和 O'Reilly 的所有员工，你们见证了我通过提案和编写的完整过程，并且感谢你们如此努力地工作，以使得本书出版。衷心感谢 Rob Romano，