

软件配置管理中的 模式与反模式

AntiPatterns and Patterns in Software Configuration Management

William J. Brown

(美) Hays W. "Skip" McCormick III 著

Scott W. Thomas

缪准扣 曾红卫 许庆国 译

机械工业出版社
China Machine Press



中信出版社
CITIC PUBLISHING HOUSE

TP31
138

软件配置管理中的 模式与反模式

AntiPatterns and Patterns in Software
Configuration Management

William J. Brown

(美) Hays W. "Skip" McCormick III 著

Scott W. Thomas

缪淮扣 曾红卫 许庆国 译



机械工业出版社
China Machine Press



中信出版社
CITIC PUBLISHING HOUSE

软件配置管理是软件开发过程中非常重要却又往往被忽视的重要过程。本书以生动有趣的语言,分析透彻的实例,结构清晰的写作方法,深入浅出地详细论述了软件配置管理中常见的主要模式和反模式。内容包括:模式和反模式的一般概念,软件配置管理模式和反模式,管理和过程模式与反模式,需求和测试模式与反模式,等等。

本书既可作为在校大学生、研究生学习软件工程技术的教材或参考书,亦可作为从事项目管理和软件开发的相关人员在实际工作中的参考资料。

William J. Brown, Hays W. "Skip" McCormick III, Scott W. Thomas: AntiPatterns and Patterns in Software Configuration Management (ISBN: 0-471-32929-0)

Authorized translation from the English language edition published by John Wiley & Sons, Inc.
Copyright © 1999 by William J. Brown, Hays W. "Skip" McCormick III, Scott W. Thomas.
All rights reserved.

本书中文简体字版由约翰-威利父子公司授权机械工业出版社独家出版。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。

本书版权登记号:图字:01-2002-1709

图书在版编目(CIP)数据

软件配置管理中的模式与反模式/(美)布朗(Brown, W.J.)等著;缪淮扣等译.-北京:机械工业出版社,2004.9

(软件工程技术丛书 项目管理系列)

书名原文:AntiPatterns and Patterns in Software Configuration Management
ISBN 7-111-14915-7

I. 软… II. ①布… ②缪… III. 软件-配置-管理 IV. TP31

中国版本图书馆CIP数据核字(2004)第070970号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑:王子恢

北京牛山世兴印刷厂印刷·新华书店北京发行所发行

2004年9月第1版第1次印刷

787mm×1092mm 1/16 · 15.5印张

印数:0 001-4 000册

定价:35.00元

凡购本书,如有倒页、脱页、缺页,由本社发行部调换
本社购书热线:(010) 68326294

译者序

当构建计算机软件时，变化是不可避免的。软件配置管理（SCM）是一组用于在计算机软件的整个生命期内管理变化的保护性活动。SCM是如此的重要，但遗憾的是，很多软件开发人员却对SCM的技术知之甚少。软件开发过程中的许多错误的观念和和行为都可以归结为没有正确地应用SCM的技术。

我们都知道经验的重要价值。当我们用某种方法解决了一个问题时，如果能把问题的细节和解决它的方法记录下来，那我们以后就可以复用这种经验。教训对于我们来说同样重要，当我们因采用了错误的方式导致某项任务执行失败时，如果能找到引起错误的根源，并把它记录下来，那以后就可以避免再犯同样的错误。这也就是我们常说的，“吃一堑，长一智”。

模式已引起了广泛的关注。Christopher Alexander和他的同事们最先将模式语言应用于城市建筑领域，他们的思想和其他人的贡献已经根植于面向对象软件界。模式是一种命名的指导性见解的结晶，它表达了在给定背景和多竞争因素下针对常见问题的解决方案的精华。模式也是一种用于收集专家设计师的智慧和经验的文献格式。简而言之，软件领域中的设计模式为开发人员提供了一种使用专家设计经验的有效途径。

反模式描述了不该做的事情，或者用在错误背景下的解决方法。反模式可以帮助我们更深入地理解原则。有时反模式发生在强调了某一个原则而忽视了其他原则的情况下。好的解决方法用在错误的情况下也可能发生反模式。

模式和反模式都是对不同学习方式的补充，而且反模式在补充模式方面尤其具有价值。

本书三位作者都是长期从事系统工程和软件系统开发的资深工程师，有丰富的项目开发经历，目前正领导着分布式面向对象框架的开发或正进行着安全合作的环境的开发。Willam J. Brown和Hays W. “Skip” McCormick III是第一本关于反模式的著作《*AntiPatterns: Refactoring Software, Architectures and Projects in Crisis*》(John Wiley & Sons, Inc., 1998)的合著者。而本书是一本介绍在系统生命周期的软件开发过程中遇到的一系列模式和反模式

RT918/03

IV

的力作，特别注重的准则是软件配置管理、需求分析和测试。它可能是国内第一本有关软件配置管理反模式和模式的正式译著。

本书针对 SCM 相关活动，生动有趣、击中要害地详细阐述了各种反模式和模式。书中提供了涉及到最常见的 SCM 错误和陷阱的 16 种反模式，其中有 8 种 SCM 反模式、4 种管理和过程反模式、4 种需求和测试反模式以及 3 种过程模式，解决了过程、人员和工具问题。

本书对高级主管、项目经理和软件开发人员都有帮助。它可以帮助从事软件开发的相关人员吸取成功的项目开发和管理的经验，避免各种错误和陷阱。

本书可作为计算机专业高年级本科生、研究生的软件工程教材或参考书，亦可作为软件开发人员和项目管理人员的参考书。

本书的翻译工作得到了国家自然科学基金项目（60373072）和国家“973”计划的项目（2002CB312001）的支持，并得到了上海大学，特别是计算机学院软件工程研究室的同事的支持和帮助，在此表示感谢。

译者

2004 年 5 月

前 言

我们写作本书旨在帮助你管理你的软件开发。经验表明，现今从事软件开发的大部分人中，对于真正的软件配置管理（Software Configuration Management, SCM）技术的理解只是有限的，或者对此几乎没有任何经验。我们认为，这不是因为项目经理不想知晓这些原理，而是因为软件配置管理看起来难于掌握，更难于实现，并且几乎不可能在实践中应用。我们中的大多数人觉得软件配置管理就好像是某类神秘精妙的充满了“霸王条款”的黑色科学，很可能会让我们陷入许多麻烦之中，并吞噬我们大脑中创造性的火花。

如果你有这方面的顾虑，即使是轻微的，那么我们希望本书会打消这种顾虑。

不幸的是，到目前为止，大家所阅读到的关于软件配置管理方面的大多材料都过于乏味，而且让人读起来昏昏欲睡。在本书中，我们力图使该主题闪耀出简洁性和易理解性的光芒，而不是再提供一个非常有效的治疗失眠的方法。我们的前一本关于反模式的著作的反馈意见表明，反模式不仅读起来有趣，而且是将许多软件人员所追求的智慧 and 知识系统化的一种行之有效的方法。熟悉那本书的大多数人说道：“很有趣，我有些喜欢它……”然后讲述了他们的个人经历如何恰好与其读过的许多反模式的部分相匹配。很多人甚至阅读了全书！

这正是我们向你推荐此书的希望之所在。我们希望你能阅读并使用它，而且我们认为你将发现它也是活生生的资源。此外，你将发现该书具有很强的娱乐性，因为你会发现关于软件配置管理的两个重要的事实：

1. SCM 不是可怕的、不可知的、恶梦般的，也不是魔鬼；而是可理解的、基于常识的。
2. 你能够成功地实现并维护一个 SCM 程序，并且由此为防止你的工程项目可能失效提供强大的防护措施。

我们建议，在你还不需要解决一个具体问题之前，为获得全面的理解，你最好立即通读全书（并且本书篇幅并不大）。实际上你需要一个特定的问题。不过，假如你的项目恰好有麻烦，并且立即需要一些关于模式或反模式的参考资料，我们建议你直接跳到“引言”并阅读

附录 A 以快速确定和引用行之有效的模式或反模式。在（合适地）解决了你的危机之后，你可以再完成本书的阅读；或者是将之放于一边，当你下一次遇到危机时再阅读。

模式的概念有些复杂并且学术性较强，因为它同面向对象软件开发相关。不过，在本书的写作过程中，我们提升了这些概念，强调了基于现实的、在实践中实用的项目管理问题。因此，本书倾向于实践型的读者，适用于正因为一个误用的模式（原理或关键过程领域）引起的危机而挣扎的项目领导、配置经理和测试经理等。本书将为之提供一个重构方案，而无须先行全面理解面向对象开发原理或者模式原理。

全书分为三个部分。第一部分：模式和反模式引论；第二部分：软件工程的反模式和模式；第三部分：结论和资源。

第一部分包括三章，第 0 章和第 1 章解释了本书介绍的内容的重要性，还包括一个所讨论的论题的导向指南。另外，给出了模式和反模式的概念的综述，使读者对模式的原理概念得以基本了解。（我们估计许多实践型的读者对面向对象程序设计的模式可能并不熟悉，而这样的理解并非绝对必要，建议了解而已。）第 2 章旨在作为在本书中包含的所有其他反模式和模式的样板。我们希望该章能为读者较早地了解其余部分打下基础。第一部分只有反模式，从第二部分开始，则突破了此限。

第二部分是本书的主体，包含由系统工程原理分解而成的三章。具体说来，第 3 章介绍软件配置管理模式和反模式，第 4 章介绍管理和过程的模式与反模式，第 5 章介绍需求分析和测试的模式与反模式。以这种方式划分模式和反模式的目的是为了有利于读者快速确定由原理定位的模式和反模式位于何处。本书可以以线性方式来阅读，但也可用来作为不同情形和环境读者的参考资料。反模式和模式的结构超越，能够使读者在引用该书时，迅速地辨别并发现自己感兴趣的主体。每章的开头还提供了模式和反模式的总结性列表，从而提供附加的快速引用。

第三部分给出了本书的结论和其他参考文献。所谓“结论”，实际上是反模式和模式的列表，为每一模式提供了梗概、确定了相应的原则。最后，我们为读者提供了我们认为对实践型读者来说非常宝贵的参考资料，包括专著、期刊论文和直接相关主题的 Web 网站。

致 谢

在我们的第一本书，由四位合著者编写的《*AntiPatterns: Refactoring Software, Architecture, and Projects in Crisis*》(John Wiley & Sons, Inc., 1998) 中，我们列举了大约 98 个人。我们试图表达许多人帮助我们达到了这样的境界，才得以写成《*AntiPatterns*》，并且我们想尽可能多地去认识他们。

从那时以来，列举人员的名单已大得无法想象了，因为我们在 PLoP、OOPSLA 和其他讨论会上结识了大批新朋友，除了通过诸如 WikiWeb 和 AntiPattern.com 的 Web 空间和许多模式研讨会名单进行交流的几乎几百个新的因特网协会会员之外，还有几个模式研究组，以及遍及全球的从事反模式研究的大学教师。感谢许许多多鼓励我们，在智力上参与我们的合作，以及热情激励我们的人，他们在这一全新的工作中发挥了至关重要的作用。

另外有几位关键人物对本书有直接的和重要的贡献，有必要一一提及。首先感谢 John Vlissides 充满友情和鼓励的话语。他在管理反模式概念的拓展方面处于领先地位。感谢 James Coplien 热情而主动地花费时间激励我们认识、接受并尽力负起作为模式社团中的作者对全体知识分子负有的伟大职责。还要感谢 Hillside Gang，尤其是 Joe Yoder、Brian Foote 和 Ralph Johnson，感谢这些朋友对 PLoP '98 上“让我们一起来”论坛的支持并加入了这个团体。

在提供大多数改进的内容方面，我们感谢 John Marsh 和 Walt Stoneburner 对手稿的出色评论和倾心鉴定，他们对于细节也给予了巨大的耐心和关注，很少有人能像 John 和 Walt 那样高效主动。

最后要感谢的是对本书的完成不可或缺的人物，她呵护、责备、引导、鼓励并促使我们去我们所能胜任的最好的工作，并极其细心、非常有效地指导着我们这个作者小组中的每一个人，她就是我们的编辑 Terri Hudson。她一直是并将继续是最好的幻想家、合作者、支持者和我们期望的编辑……她是一名很棒的朋友。谢谢你，Terri，希望我们再次合作。



作者小传

William J. Brown 是位于弗吉尼亚州的 McLean 的第五元素技术的产品工程师，他是《*AntiPatterns: Refactoring Software Architectures and Projects in Crisis*》(John Wiley & Sons, Inc., 1998) 一书的合著者之一，也是《分布式计算》的专栏作者。Bill 领导过分布式对象系统的开发，他有电信、卫生和军事实时系统方面的工作经历，但他最擅长的是金融业。他目前一直担任保险服务、财政及相关机构、金融产品开发的架构师和项目经理。在他还是金融业任务特别工作组的架构师时，Bill 就因 OMG 而声名远扬，是该领域的权威。他的前一种职业是变革行为中的社会活动者，这为其进行软件开发提供了独特的社会哲学视角。

Hays W. “Skip” McCormick 是位于弗吉尼亚州的 McLean 的 MITRE 公司的分布式构件、体系结构和框架部的资深工程师。他也是《*AntiPatterns: Refactoring Software Architectures and Projects in Crisis*》(John Wiley & Sons, Inc., 1998) 一书的合著者之一，他有在很多信息系统领域工作的广泛背景，包括计算机安全、人工智能和分布式计算等。他最近的工作集中在知识管理领域，追求的是一种艰深集成的、分布式合作系统的可行的具体化。Skip 具有马里兰州安纳波利斯市的美国海军军官学校的计算机科学学士学位，他居住在弗吉尼亚州的马纳萨斯市，是 Teddy 和 Janine 引以为自豪的父亲和 Kim 的幸福丈夫。

Scott W. Thomas 是位于弗吉尼亚州 McLean 的 MITRE 公司的信息技术部的资深工程师和项目领导。他具有在政府部门 20 年的工作经历，其工作涉及到系统工程生命周期的方方面面。他现在正进行着安全合作环境的开发工作。Scott 还是美国海军预备队的海军上尉、工程责任官，美国大西洋舰队 606 派遣队的海军水面军队的指挥官。Scott 从弗吉尼亚州的 Blacksburg 的弗吉尼亚技术学院获得了工业工程和运筹学的学士学位以及系统工程的硕士学位。他居住在弗吉尼亚州的 Vienna，他有两个孩子 Shara 和 Ryan，妻子是 Susan。

引言

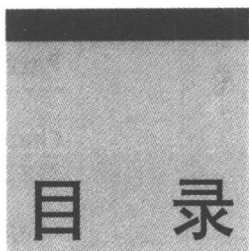
本书将帮助你辨别和克服反复发生的妨碍软件配置管理和相关的管理的常见路障以成功地交付软件。反模式确定了通用错误并提供了重构方案，而模式则确定良好的工作方法。本书将帮助你解决组织和工程方面的现有问题，并帮助你开发实用的工作方法，以阻止某些经常发生的问题。

许多人偶尔会忘记我们需要严格的控制，初看起来严格的控制似乎造成了额外的工作负担而又几乎看不到有什么回报。软件配置管理通常就是这样。实际上，SCM的缺乏或不正确的实施将造成更多的工作负担，甚至是巨大的更为频繁的失败以及投资的巨大浪费。简而言之，所有的软件都需要相应级别的软件配置管理，除非该软件打算直接进入垃圾箱。这就是说，SCM解决方案必须与其他软件开发过程很好地平衡以确保获得效率，而不是获得非生产性的官僚机制。

这一点对于需求的可跟踪性和测试同样正确，两者往往在交付的压力下被削减。另外，如果没有原则，那么结果也将是无原则的。策略就是实现一个最小的容易遵循的过程，并产生有意义的结果。本书涉及很常见的问题并提出实际的解决方案。

文化和管理从最高层开始。如果掌握公司方向的高级主管忽视细节，那么这种做法就会变成公司的准则。高级主管既不会试图去定义也不会去实现关键产品的开发管理过程，而是看重在状况报告中的结果，批准软件发布以及满足客户的需要。这样一来，软件开发公司的高级主管必须既要懂得实现过程的缺失和糟糕的实现过程所带来的问题，又要掌握避免和解决这些问题的方法。

如果你了解一个已经失败了的项目，就很有可能利用本书中的信息发现至少一条失败原因。并且，我们编写的反模式丛书的前一本书《*AntiPatterns: Refactoring Software Architectures and Projects in Crisis*》可用于找出其他原因。目的是为了预先发现可能存在的问题，实现解决现有问题的重构方案。在某些情况下，这可能太迟了，但却打开了获知该开发之所以失败的原因以及防止其再次发生的机会之门。因为媒体广为宣传失败的项目，业界似乎已对软件开发交付失败见怪不怪了。但是，人类总是追求进步的，现况是可以改善的。不要再犹豫了，抓紧时间。

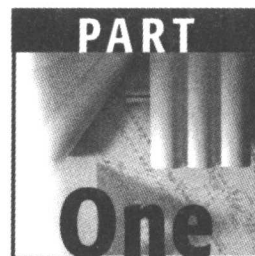


目 录

译者序
前言
致谢
作者小传
引言

第一部分 模式和反模式引论	1
第 0 章 模式本质简介	3
模式和反模式概论	4
过程和管理模式模板	5
反模式模板	7
无名质量	9
第 1 章 没有章法：系统工程透视	13
软件配置管理	16
需求开发	18
测试	19
知难而进	20
第 2 章 所有管理反模式之父	23
Nil Desperandum 反模式	24

第二部分 软件工程的反模式和模式	39
第 3 章 软件配置管理模式和反模式	41
“软件配置管理组织”模式	43
“银弹”反模式	48
“CM 接管”反模式	55
“开发人员 - 软件配置管理”反模式	63
“分散式配置管理”反模式	71
“不审计”反模式	83
“面向对象的配置管理”反模式	92
“软件配置管理专家”反模式	111
“事后计划”反模式	123
第 4 章 管理和过程模式与反模式	133
P ² 模式	135
Détente 反模式	144
“涅槃”反模式	151
“企业系统神话”反模式	161
第 5 章 需求和测试模式与反模式	173
“需求危险”反模式	175
“信息完整性管理”模式	184
“媒体上是什么”反模式	193
“想当然”反模式	201
“大期望”反模式	210
第三部分 结论和资源	221
附录 A 反模式和模式梗概	223
附录 B 其他参考资料	229



第一部分
模式和反模式引论





INTRODUCTION

第 0 章 模式本质简介

为了从本书中充分获益，我们建议首先阅读如下书籍，以便理解本书提出的模式和反模式的语义（在本章的“参考文献”部分有完整的信息出处）。

Timeless Way of Building

A Pattern Language: Towns, Buildings, Construction

Design Patterns: Elements of Reusable Object-Oriented Software

Pattern Languages of Program Design 1-3

如果你没有阅读过这些书，仍然可以从本书中获益，但会失去一些内涵。然而，为了帮助模式和反模式的初学者，我们在本章定义了概念和模式格式。因此，阅读反模式丛书中的前一本书《*AntiPatterns: Refactoring Software, Architecture, and Projects in Crisis*》是必要的。

目前，已经出版了关于各种模式的论著、期刊文章和书籍。Martin Fowler 已认识到定义模式这个术语的难度。尽管如此，出于本书的目的，对

模式和反模式有一个清晰的语义定义是非常有用的，这样在本书的后续章节中，就可以在一个陈述清晰的语境中对它们进行分门别类地描述。模式的根本原则是质量，能够影响软件开发的方方面面的质量，在有关模式的一般讨论中并没有讲到这一点。

模式和反模式概论

由于大多数可用的模式用于处理分析、体系结构和设计制品，因此它们是面向应用的。这些模式给出了一个本质上通用的模板，能够应用于多种项目的软件构造中。下面是试图定义模式的三种尝试。

模式是一种已经在一个实际语境中很有用的思想，该思想在其他语境中可能也很有用。[Fowler 1977]

设计模式记录经长久发展和演变的用于特定问题的解决方案。[Gamma 1995]

模式主要描述特定设计情形下重复出现的设计问题并提出一种解决方案。它记录现有的、经良好证实的设计经验。[Buschmann 1996]

模式的本质

“模式就是对想要解决的问题的一种解决方案。关于模式，读者的问题通常是确定应该何时使用和怎样使用模式。”

反模式定义为“针对不可避免要产生负面结果的问题给出一种普遍的现实解决方案”。换言之，在软件行业中，反模式一般是一种重复的实践行为，是由于某种方式的缺陷造成的，并且造成的非期望的症状比得到的积极结果要多。反模式可以分解成以下三个关键部分：

- 用于标识问题（如厂商锁定（Vendor Lock-In））的实例。在该形式下，反模式是一个普遍重复的问题。
- 重构解决方案。同样，它也是一个普遍的可重复使用的解决方案。
- 声明反模式的每个部分的可重复的结构；换言之，就是反模式模板。

所以，如果问题和方案以结构化的形式表示，并被重复地用来说明其他实例，那么就像 Bobby Woolf 在 PLoP '98（见图 0-1）指出的那样，反模式

只不过是模式的另一种形式。主要的不同在于反模式有如下几点：

- 主要以在模式中展示负面结果的方式对问题进行分类。
- 提供对问题直接重构的一种模式解决方案。
- 通常处理开发领域的问题而不是设计层次的模板，例如管理和体系结构、过程和组织结构。

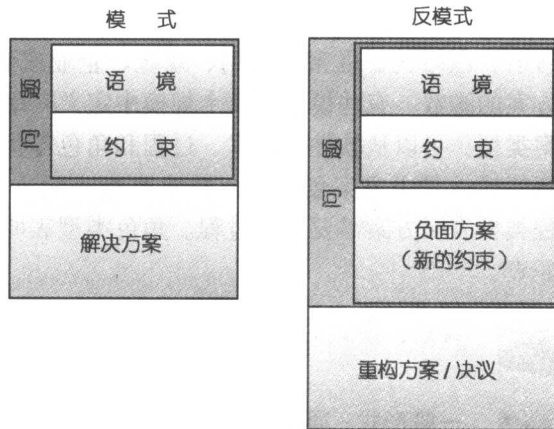


图 0-1 作为模式的反模式 (PloP '98)

对模式和反模式的观察角度是至关重要的，因为它表达解决方案的形式使得以这种角度看待问题的角色能成功地实现这种解决方案。本书从管理角度来考察软件配置管理 (SCM)。

表示这个主题的模板是反模式模板及一个派生的过程模板和管理模式模板。模式和反模式的顺序基于本章后面将要讲述的软件开发过程生命周期。

本书内的模式和反模式参考文献引用了在本书使用的专用名称，例如 CM 接管 (CM Takeover)。外部的模式和反模式的参考文献均按照原始作者和年份的形式来表示，例如 [Coplien 1995]，这些被引用的文献在章末有详细的说明。

过程和管理模式模板

过程和管理模式模板对过程和管理领域的解决方案的反模式模板的部分进行了抽象。它包含的元素来自如下几份资料，其中有 Gerard M. Mezaros 和 James Doble 提交给 PloP '96年会的“A Pattern Language for Pattern Writ-