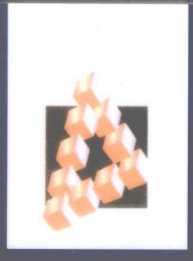
TURING 图灵程序设计丛书



WILEY

Pattern-Oriented Software Architecture Volume 4

A Pattern Language for Distributed Computing

面向模式的软件架构

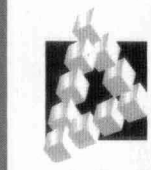
分布式计算的模式语言

卷4

[德] Frank Buschmann
[英] Kevlin Henney 著
[美] Douglas C.Schmidt
肖鹏 陈立 译

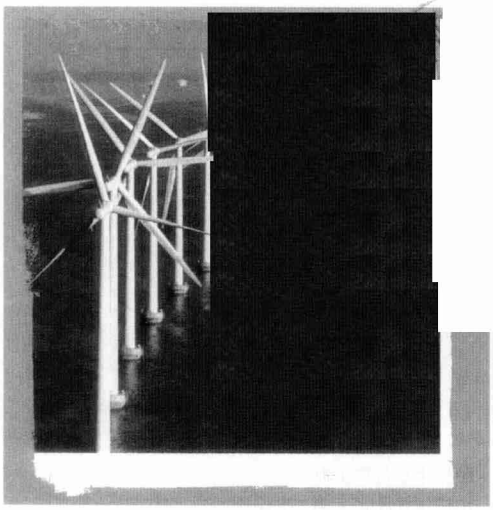
人民邮电出版社
POSTS & TELECOM PRESS

TURING 图灵程序设计丛书



Pattern-Oriented Software Architecture Volume 4
A Pattern Language for Distributed Computing

面向模式的软件架构
分布式计算的模式语言



卷4

[德] Frank Buschmann
[爱尔兰] Kevlin Henney
[美] Douglas C.Schmidt 著
肖鹏 陈立 译

人民邮电出版社
北 京

图书在版编目（CIP）数据

面向模式的软件架构 卷4：分布式计算的模式语言/（德）布施曼（Buschmann,F.），（英）亨尼（Henney,K.），（美）施密特（Schmidt,D.C.）著；肖鹏，陈立译. -- 北京 ：人民邮电出版社，2010.6
（图灵程序设计丛书）
书名原文：Pattern-Oriented Software Architecture
ISBN 978-7-115-22773-7

Ⅰ. ①面… Ⅱ. ①布… ②亨… ③施… ④肖… ⑤陈… Ⅲ. ①软件设计②分布式计算机系统 Ⅳ. ①TP311.5②TP338.8

中国版本图书馆CIP数据核字(2010)第071266号

内 容 提 要

本书关注分布式计算系统软件的设计和实现。书中首先介绍理解本书内容所需的核心的模式概念，分布式计算的好处和挑战；然后描述如何使用分布式计算模式语言，设计真实世界中仓库管理流程控制系统；最后重点讲述分布式计算模式语言，该语言陈述了创建分布式系统相关的技术主题。

本书适用于软件架构师和开发人员。

图灵程序设计丛书

面向模式的软件架构 卷4：分布式计算的模式语言

◆ 著 [德] Frank Buschmann [英] Kevlin Henney
[美] Douglas C. Schmidt

译 肖 鹏 陈 立

责任编辑 傅志红

执行编辑 贾利莹

◆ 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

中国铁道出版社印刷厂印刷

◆ 开本：800×1000 1/16

印张：23

字数：538千字 2010年6月第1版

印数：1－3 000册 2010年6月北京第1次印刷

著作权合同登记号 图字：01-2008-0488号

ISBN 978-7-115-22773-7

定价：69.00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

关于作者

Frank Buschmann

Frank Buschmann 是德国慕尼黑西门子技术公司的高级总工程师。他的研究领域包括对象技术、软件架构、产品线、模型驱动软件开发和模式。他在该领域著作甚多，其中最引人注目的便是POSA系列的前两卷[POSA1][POSA2]和最近的两卷：本书和[POSA5]。Frank在1992年至1996年曾是ANSI C++标准化委员会X3J16的成员，于1996年发起了首届EuroPLoP会议，与人合作汇编了数本模式方面的书籍[PLoPD3][SFHBS06]，现任Wiley软件设计模式丛书的主编。在西门子的开发工作中，Frank领导过几个大型工业软件项目的架构设计和实现，包括业务信息、工业自动化和电信系统。

工作之余，Frank的生活丰富多彩。他喜欢陪着妻子Martina和女儿Anna享受家庭的快乐；他喜欢骑着马儿Eddi外出遛弯，或者独自跑到慕尼黑啤酒花园打发时间；他喜欢在观看最爱的多特蒙德足球队的比赛时尽洒激情，也偶尔在慕尼黑歌剧院里的演出中尽情陶醉；临睡前他还喜欢来杯珍藏的苏格兰纯麦威士忌。

Kevlin Henney

Kevlin Henney住在英国布里斯托尔，是一名独立顾问。他主要从事自己感兴趣的领域的教授、辅导和实践，这些领域包括编程语言和技术、软件架构、模式和敏捷开发。他的客户既有全球化的公司，也有刚起步的小公司，所涉及的领域包括系统软件、电信、嵌入式系统、中间件开发、业务信息和金融。

Kevlin经常在各种软件会议上应邀演讲，同时也参与了多个会议的组织工作，包括EuroPLoP。他通过英国标准学会（BSI）和ISO参与到C++的标准中来，同时也参与了其他语言的标准化工作。Kevlin也因其写作受人关注，经常发表会议论文，主持各种出版物上定期的（和不定期的）专栏，比如*C++ Report*、*C/C++ Users Journal*、*Java Report*、*JavaSpektrum*、*Application Development Advisor*、*The Register*、*EXE*和*Overload*。

Kevlin的业余时间则是与妻子Carolyn还有两个孩子Stefan和Yannick度过的。他跟孩子摆摆积木，给孩子们修修玩具，看书，有时也啜几口啤酒或者来一杯葡萄酒。

Douglas C. Schmidt

Doug Schmidt是美国田纳西州纳什维尔市范德比尔特大学计算机科学教授，计算机科学与工程计划副主席。他的研究领域包括模式和模式语言、优化原理，还包括对于支持服务质量（QoS）的组件中间件相关技术的实证分析（empirical analysis）和支持分布式实时嵌入式系统的模型驱动的工程工具。

Doug是国际公认的软件开发方面的专家，尤其是模式、面向对象框架、实时中间件、建模工具和开源软件开发等方面。他已经在各种顶级技术杂志和会议上发表了300多篇论文；他与人合著了模式方面的著作[POSA2]和有关C++网络编程的[SH02][SH03]；同时他还与人合编过数本畅销的书，比如模式方面有[PLoPD1]，框架方面有[FJS99a][FJS99b]。除了学术研究，Doug还领导了ACE、TAO、CIAO和CoSMIC的开发，这些广泛应用的开源的中间件框架和模型驱动的工程工具包含了一整套丰富的可重用的组件，这些组件的实现大量使用了本书中介绍的模式。

在他闲暇之隙，Doug总是跟妻子Lori和儿子Bronson待在一起，他们一起练练举重、弹弹吉它、讨论世界历史和政治，或者开着他们的雪佛兰Corvette跑车四处兜风。

前　言

模式运动已经进行了十多年，从追捧到棒杀再到慢慢接受，模式已经经历了这个常见的轮回。Frank、Doug和Kevlin一直参与其中，受到过赞美，也遭遇过嘲讽，重要的是他们从中收集了大量好的想法，并将其描绘出来。POSA系列图书被认为是模式相关文献中最为坚实的基础性著作之一，它的每一卷都在我的书架上占有一席之地。

POSA的前几卷属于传统的模式书籍，描绘了某些特定领域中使用的模式，其中大部分以前均未有书面记录。本书则不同。分布式计算是一个相当宽泛的主题，一卷图书哪怕只是容纳已知的模式也是远远不够的。实际上这些模式分布在很多书里面，包括POSA系列和一些别的书。本书的目的是要把它们聚在一起。所以，这里列出的模式可能比你平时看到的要多，当然其描述也要简洁得多。有些模式可能并不是主要关于分布式的，但是多少都会和分布式系统有些关系。因此，本书是以分布式系统为背景来介绍这些模式的用法，并加以总结。

本书并不是仅仅介绍每个独立的模式的——同时也介绍它们之间的关系。在任何一个系统中都会同时使用多个模式，然而，就拿我的体会来说，讲述其中的关系要比介绍单独的模式难得多。本书没有回避这个问题，书中给出了很多关于在分布式场合下联合使用多种模式的建议。

分布式往往是一个棘手的难题。事实上，经常有人引用我的所谓分布式对象设计第一定律的“名言”：“不要使用分布式对象。”我这样说是有原因的——分布式使得软件设计更困难，所以我一直建议尽可能地避免采用分布式设计。然而无论我如何强烈地质疑分布式设计的范围，分布式毕竟是很多软件系统重要的组成部分。既然分布式这么难，分布式设计便更值得我们认真地研究一番——因此，本书也应该是每个程序员必备的图书了。

Martin Fowler[①]

① Martin Fowler，软件开发方面著名作家和演讲者，ThoughtWorks公司首席科学家，《重构：改善既有代码的设计》作者。——译者注

关于本书

分布式计算正在将整个世界连接起来，创造一个公平的竞争环境[Fri06]。今天，无处不在的Web和电子商务更为分布式计算提供了更大的动力：人们需要连接并访问大量分散在世界各地的信息和服务。即时消息和聊天室在Internet上的流行则带来了另一方面的需求：与家人、朋友、同事和客户保持连接。其他推动分布式计算的动力包括提高性能、可伸缩性和容错能力，同时通过共享昂贵的硬件和外围设备来降低成本。

鉴于分布式计算在我们的职业和个人生活中的重要性，软件文献中的众多模式都把目光投向了这个主题[POSA1] [POSA2] [POSA3] [Lea99] [VSW02] [VKZ04] [HoWo03] [PLoPD1] [PLoP2] [PLoPD3] [PLoPD4] [PLoPD5]。遗憾的是，这些模式大多都是孤立地描述，引用的几个其他模式也大多出自同一本出版物。尽管每个独立的模式也有其用处，但是孤立的描述毕竟无法提供完整的视角来着重展示，在分布式计算系统中相关的模式是如何相互取长补短的。于是，构建复杂的分布式系统成了只有少数天才和大师们才能掌握的暗黑艺术[①]。

为了展示更为完整的视角，本书描绘了一个单独的模式语言，它把多个与分布式计算有关的模式联系起来。在该语言中，每个模式或者直接与分布式计算有关，或者在分布式计算环境中扮演重要的支持角色。模式语言为分布式计算关键领域的最佳实践提供了指南，同时也提供了一个交流的工具。

目标读者

我们关注的是分布式计算系统软件的设计和实现。因此，本书的主要读者是专业的软件架构师或者是高阶的学生，他们参与开发分布式计算系统，设计新的应用或者改善和重构已有的应用。模式语言展示了一系列丰富的模式，目标是帮助架构师为分布式系统创建可持续的设计，并且帮助他们全面而专业地陈述其需求。

本书的第二类读者则是在工作中使用组件和通信中间件的开发人员。模式语言为开发人员提供了分布式系统设计的当前实践状态，以便他们可以更有效地使用中间件。可以从模式语言中获益的第三类人是项目和产品经理。这门语言使得产品经理对于他们所领导开发的系统的基本功能可以有更深入的理解，并且为他们和软件架构师以及开发人员沟通提供有用的词汇。

不过，我们并不打算让终端用户或者客户直接使用模式语言。正确地使用真实世界的隐喻可

① “暗黑艺术”，20世纪产生的一种艺术形态，其典型形式包括恐怖、抽象、神秘等。——译者注

能会使这些读者接受模式语言，但这需要换一种表达方式。此外，本书也不是分布式计算的全面指导手册。虽然我们讨论了这个主题的众多方面，并且包含了一个广泛的术语集，但是读者需要事先对分布式计算的核心概念和机制有所了解，比如死锁、事务处理、同步、调度和合意（consensus）。更多的和分布式计算相关的主题，比如网络协议和操作系统的设计，可以在参考文献中找到。

结构和内容

本书内容分为3个部分：概念、模式故事和模式语言。

第一部分“概念”介绍本书的背景：理解本书所需的核心的模式概念，分布式计算的好处和挑战的概述，支持分布式的技术的总结以及对模式语言的介绍。

第二部分“模式故事”描述如何使用分布式计算模式语言设计真实世界中仓库管理的流程控制系统。这个故事关注于该软件系统的3个领域：基线架构、通信中间件和仓库拓扑表现。

第三部分“模式语言”构成了本书的主体部分。它包括一门分布式计算模式语言，该语言陈述了与分布式系统结构相关的技术主题：

- 给出初始的软件基线架构
- 理解通信中间件
- 事件分离和分发
- 接口划分
- 组件划分
- 应用控制
- 并发
- 同步
- 对象间的交互
- 适配与扩展
- 模态行为
- 资源管理
- 数据库访问

每一章介绍了该章的主题，总结了主要的挑战，然后展示了帮助应对这些挑战的一系列模式。我们的分布式计算模式语言总共包含了114种模式，并且连接到其他文献中介绍的150多种模式。这可以称得上是目前为止较为巨大的，甚至是最大的已经文档化的软件模式语言了。

虽然分布式计算是该语言的关注点，但是其中很多内容具有更广泛的适用性。例如，大多数的应用程序必须具有某种形式的适应性和可扩展性，而且每个软件系统都需要良好设计的接口和组件。对于相应的技术领域，模式语言可以作为现代软件开发中最佳实践的总体指南，而不是仅限于分布式计算领域。

本书的结尾包含了对分布式计算模式语言的简单回顾、常用的术语词汇表、一个庞大的该领域参考文献列表。

毫无疑问，我们不可能覆盖到分布式系统所有的属性和模式，更何况随着时间的推移，在模式语言的实际应用和扩展实践中还会有更多的模式涌现出来。如果你有任何关于改进本书的风格和内容的评价、建设性的批评或建议，请通过电子邮件发送给siemens-patterns@cs.uiuc.edu。在模式主页http://hillside.net/patterns有一份注册指南。该链接还提供了一个关于模式的方方面面的重要信息源，比如已经出版的和即将出版的图书、模式会议、模式论文等。

致谢

我们很高兴可以在此感谢帮助我们完成此书的人们，他们有的是通过与我们分享他们的知识，有的则审阅了早期的各个章节的草稿。

审阅奖应该发给Michael Kircher，他认真审阅了我们的材料，包括其正确性、完整性、一致性和总体质量。Michael的反馈极大地提高了本书的质量。

另外，我们在3次EuroPLoP模式会议上展示了该语言的几个部分，同时也给分布式和模式方面的几个专家做了展示。Ademar Aguimar、Steve Berczuk、Alan O'Callaghan、Ekatarina Chtcherbina、Jens Coldewey、Richard Gabriel、Ian Graham、Prashant Jain、Nora Koch、Doug Lea、Klaus Marquardt、Andrey Nechypurenko、Kristian Sørensen、James Siddle、Michael Stal、Steve Vinoski、Markus Völter、Oliver Vogel和Uwe Zdun为我们提供了大量的反馈，该语言的很多大小修订均是由此而来。

非常感谢Mai Skou Nielsen，她在丹麦奥尔胡斯市的JAOO 2006会议上为Kevlin和Frank照了一张照片。Anton Brøgger帮助查明了我们在第12章所使用的照片的情况。Publicis Kommunikation-sagentur GmbH和Lutz Buschmann允许我们在本书中使用他们的一些照片。

特别感谢Lothar Borrmann和Reinhold Achatz给予的行政上的支持和德国慕尼黑西门子技术股份公司软件工程实验室的支持。

尤其感谢我们的编辑Sally Tickner，我们以前的编辑Gaynor Redvers-Mutton，以及John Wiley & Sons的其他人员，没有他们便不会有本书的出版。尽管我们的日常工作已经非常繁忙，Gaynor还是说服我们撰写了本卷POSA。接下来，Sally在我们撰写本书的数年里表现出极大的耐心。特别感谢我们的文字编辑，WordMongers公司的Steve Rickaby，他令本书增色不少。本书是由Steve负责的POSA系列的第4卷，我们希望后面的几卷还能跟他合作。

最后，还要感谢我们的家人，感谢他们在本书撰写期间给予的耐心和支持！

读者指南

你能做到，只是不要想一步登天。

——Oprah Winfrey[①]

本书的编写结构允许你用任何自己喜欢的方式阅读。最简单的方式就是从头读到尾。如果你自己知道想要了解哪一块，也可以自由地选择自己的阅读路径。这里，我们给出了一些线索，希望能帮助你找到自己关注的内容，并确定阅读的顺序。

关于模式和模式语言

本书为读者呈现了一种分布式计算模式语言，它包含了一系列相关的模式。这些模式定义了应该按照什么样的流程，系统地解决开发分布式系统软件过程中出现的问题。我们编写本书的目的是帮助你在日常的软件开发活动中应用这些模式来创建可工作、可支撑（sustainable）的分布系统软件架构。我们假定你对于模式和模式语言的概念已经有一定的了解，所以本书并不是专门全面介绍这两个概念的教程。

如果本书是你第一次接触到模式，我们建议你首先阅读《面向模式的软件体系结构（第一卷）：模式系统》[POSA1]和《设计模式：可复用面向对象软件的基础》[GoF95]中关于模式的介绍。这两本书介绍了软件架构和设计中有关模式的基本概念和术语。如果你已经熟悉了模式的概念，而不了解模式语言的概念，我们建议你阅读第1章，以及James O. Coplien写的*Software Patterns*的白皮书[Cope96]，从中你可以了解到模式语言的概念，这些内容能够保证你从本书的分布式计算模式语言中有所收获。以上两部分也简要地介绍了模式概念方面的高级内容，其思想要比[POSA1]和[GoF95]中的深一些。

关于分布式计算

本书假定你对分布式计算的关键概念和机制有一定的了解。第2章简要地描述了分布式计算的优点和挑战，并总结了支持分布式的技术，但并没有详细地讨论分布式计算和分布式系统。这一章的目的只是从总体上介绍一下本书的主题：要获得分布式计算的好处，你应该踏踏实实地根据我们的模式语言的指导去应对相关的挑战。

① 美国著名脱口秀主持人。——译者注

如果你需要更多关于分布式计算的背景知识，我们建议你阅读Andrew S. Tanenbaum和Maarten van Steen的*Distributed Systems: Principles and Paradigms* [TaSte02]和Ken Birman的*Reliable Distributed Systems* [Bir05]。

关于分布式计算模式语言

在开始阅读模式语言中的全部或者某几个模式之前，我们建议你先读第3章模式语言。这一章从总体上介绍了我们的模式语言，其中主要包括如下内容。

- 意图、范围和读者。
- 模式语言的通用结构，模式语言中与分布式计算相关的主题和要解决的挑战，以及该模式语言包含的具体模式。
- 介绍和阐述模式语言中的模式时，模式所使用的形式和标记法。

这一章也是模式语言的一个通用地图，有了这张地图，当你读到某一个或一组特定的模式时，就知道自己位于什么地方了。这个地图的目的就是帮助你在阅读某个模式的细节时，避免出现只见树木不见森林的尴尬。

模式语言实践

本书的第二部分，给出了一个具体的例子，阐明在实践中如何应用分布式计算模式语言为仓库管理流程控制系统创建架构。通过创建一个真实世界的系统，展示了分布式计算模式语言在构建高质量软件系统中可以为架构师和开发人员提供怎样的信息。如果你擅长通过例子学习，我们建议你在深入阅读模式语言之前先把这个例子看一遍，虽然先学习模式语言再回来看这个例子并不影响例子所表达的信息。这个例子演示了我们的模式语言如何帮助你创建和理解如下功能。

- 分布式系统的基线架构。基线架构可以高效地划分系统的功能和基础设施职责，并保证系统满足服务质量要求。
- 通信中间件。中间件使得分布式系统的组件之间能够高效地、健壮地、可移植地交互。
- 分布式系统中具体组件的详细设计。这些设计应该能够支持分配给自己的职责，满足各自的需求。

虽然这个例子本身就是完备的，但要达到融会贯通的程度还是需要阅读相应的章节，直到找到所描述问题的最基础的解决方案陈述。这时，如果你不熟悉某个模式，我们建议你阅读第三部分中相关模式的概要。如果你熟悉这个模式，可以继续阅读这个例子，看看这个模式是如何在仓库管理系统中应用的，备选的模式还有哪些，为什么没有选择备选的模式。

模式语言细节

本书的第三部分模式语言详细地介绍了分布式计算模式语言，包括与构建分布式系统相关的关键技术主题。我们建议你采用下面的方式阅读这一部分内容。

- 从头至尾。在这一部分有关语言的技术主题以及与之相关的模式，大体上是按照我们构

建分布式系统时的相关性和应用的顺序编写的。

- 按主题阅读。如果你对某个技术主题特别感兴趣，比如组件划分，你可以只阅读相应的章节。每一章的开头列举并讨论了与相应的技术主题有关的挑战，介绍了帮助我们应对这些挑战的模式，对比了它们的相同点和不同点。之后对每个模式的扼要介绍，也是每一章的重点。
- 按模式阅读。最后，如果你只是对某个模式感兴趣，你可以根据第3章中列出的模式，找到该模式在语言中的位置，直接阅读相关的内容。

模式概要并未介绍模式的实现细节，比如如何使用某个编程语言或者在某个特定的中间件平台上实现这个模式。我们在每个模式的概要中展示并讨论了该模式所解决的问题、模式背后的驱动力、它所包含的关键解决方案，以及应用该模式会带来的结果。同时，我们也说明了要实现这个模式可能用到的语言中的其他模式，以及该模式可以应用于哪些其他模式的实现。整本书中，当我们提及语言中某个模式的时候，我们会在其后用括号注明其所在页数。如果你对某个模式的实现细节感兴趣，我们建议你查阅我们引用的原始出处。

目　录

Part 1 第一部分

概　念

语言之城，人人皆有一砖一石之贡献。

——拉尔夫·瓦尔多·爱默生[1]

本书第一部分为分布式计算模式语言提供了背景。我们给出了模式和模式语言的概念、分布式计算的优点和挑战，并简单介绍了模式语言本身的概况。

本书关注的是模式和分布式计算模式语言。为了更好地理解这些模式和语言，并能在构建分布式系统产品的过程中成功应用它们，则了解模式和分布式计算相关的概念以及已有的分布式技术，不仅有益，而且必要。要在项目开发中有效地使用模式语言，你需要理解其范围、结构、内容和表现。

本书的第一部分总体描述了这些概念，同时也简单介绍了分布式计算模式语言。

第1章模式与模式语言介绍了模式和模式语言概念的各个方面，有助于我们理解分布式计算模式语言。我们给出了模式的基本概念，讨论了这些概念的核心属性，展示了这些模式是如何连接在一起组成模式语言的，以及模式所组成的网络如何相互配合、系统地解决软件开发中相互交织的各种问题。

第2章分布式系统简要地介绍了构建分布式系统的主要优点和挑战，指出了每一代分布式技术分别用于应对哪些挑战和如何应对这些挑战，以及还有哪些问题尚未解决，必须由分布式系统中的应用架构来解决。

第3章模式语言介绍了分布式计算模式语言。我们说明了语言的意图和范围，定义了它的适用性。我们还简单地介绍了所覆盖的13个问题领域，列出了114种模式，展示了语言的具体结构和范围。在本部分，我们还给出了语言的具体展现形式，比如模式的格式和用到的标记法，以及在产品开发项目中如何支持应用。

这3章为全书设定了背景，我们在第二部分模式故事中给出了一个关于仓库管理流程控制系统的模式案例，在第三部分模式语言中详细介绍模式语言。

---

① 爱默生（1803—1882），美国散文作家、思想家、诗人。——译者注