重构：改善既有代码的设计

作者：[美]马丁·福勒（Martin Fowler）

目录

重构列表

PEARSON

Refactoring Improving the Design of Existing Code

重构改善既有代码的设计

[美]Martin Fowler 著

熊节 译

人民邮电出版社

北京

重构：改善既有代码的设计/（美）福勒（Fowler,M.）著；熊节译.--2版.--北京：人民邮电出版社，2015.8

书名原文：Refactoring:Improving the Design of Existing Code

ISBN 978-7-115-36909-3

I.①重… II.①福…②熊… III.①机器代码程序—程序设计 IV.①TP311.11

中国版本图书馆CIP数据核字（2015）第137778号

内容提要

本书清晰揭示了重构的过程，解释了重构的原理和最佳实践方式，并给出了何时以及何地应该开始挖掘代码以求改善。书中给出了70多个可行的重构，每个重构都介绍了一种经过验证的代码变换手法的动机和技术。本书提出的重构准则将帮助你一次一小步地修改你的代码，从而减少了开发过程中的风险。

本书适合软件开发人员、项目管理人员等阅读，也可作为高等院校计算机及相关专业师生的参考读物。

◆著 ［美］Martin Fowler

译 熊节

责任编辑 杨海玲

责任印制 张佳莹 焦志炜

◆人民邮电出版社出版发行 北京市丰台区成寿寺路11号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 http://www.ptpress.com.cn

三河市中晟雅豪印务有限公司印刷

◆开本：800×1000 1/16

印张：28.25

字数：490千字 2015年8月第2版

印数：1-6000册 2015年8月河北第1次印刷

著作权合同登记号 图字：01-2009-5707号

定价：69.00元

读者服务热线：(010)81055410 印装质量热线：(010)81055316

重构的重新认识（再版序）

光阴荏苒，从当年译完这本《重构》，到如今重新整理译稿，不知不觉已经过去6年了。6年来，在各种大型系统中进行重构和指导别人重构，一直是我的一项工作。对于这本早已烂熟于心的书，也有了一些新的认识。

不得不遗憾地说，尽管“重构”已经成了常用词汇，但重构技术并没有像我当初乐观认为的那样“变得像空气与水一样普通”。一方面，一种甚嚣尘上的观点认为只要掌握重构的思想就足够了，没必要记住那些详细琐碎的重构手法；另一方面，倒是有很多人高擎“重构”大旗，刀劈斧砍进行着令人触目惊心的大胆修改——有些干脆就是在重做整个系统。

这些人常常忘了一个最基本的定义：重构是在不改变软件可观察行为的前提下改善其内部结构。当你面对一个最需要重构的遗留系统时，其规模之大、历史之久、代码质量之差，常会使得添加单元测试或者理解其逻辑都成为不可能的任务。此时你唯一能依靠的就是那些已经被证明是行为保持的重构手法：用绝对安全的手法从“焦油坑”中整理出可测试的接口，给它添加测试，以此作为继续重构的立足点。

六年来，在各种语言、各种行业、各种软件形态，包括规模达到上百万行代码的项目中进行重构的经验让我明白，“不改变软件行为”只是重构的最基本要求。要想真正让重构技术发挥威力，就必须做到“不需了解软件行为”——听起来很荒谬，但事实如此。如果一段代码能让你容易了解其行为，说明它还不是那么迫切需要被重构。那些最需要重构的代码，你只能看到其中的“坏味道”，接着选择对应的重构手法来消除这些“坏味道”，然后才有可能理解它的行为。而这整个过程之所以可行，全赖你在脑子里记录着一份“坏味道”与重构手法的对应表。

而且，尽管Java和.NET的自动化重构工具已经相当成熟，但另一些重要的面向对象语言（C++、Ruby、Python……）还远未享受到这样的便利。在重构这些语言编写的程序时，我们仍然必须遵循这些看似琐碎的做法指导（加上语言特有的细节调整），按部就班地进行——如果你还想以安全的方式重构的话。

所以，仅仅掌握思想是没用的。如果把重构比作一门功夫的话，它的威力全都来自日积月累的勤学苦练。记住所有的“坏味道”，记住它们对应的重构手法，记住常见的重构步骤，然后你才可能有信心面对各种复杂情况——学会所有的招式，才可能“无招胜有招”。我知道这听起来很难，但我也知道这并不像你想象的那么难。你所需要的只是耐心、毅力和不断重读这本书。

熊节

2009年10月21日