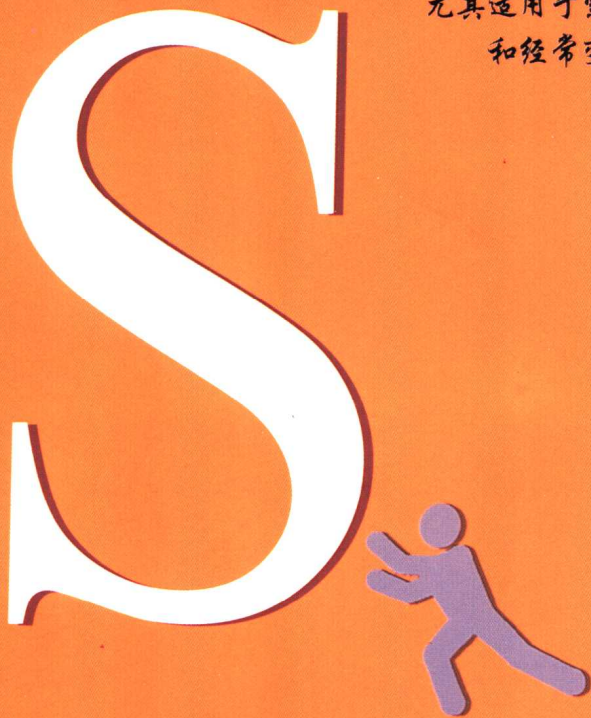




软件管理与软件工程译丛

荣获软件开发杂志 JOLT WINNER 大奖，

尤其适用于紧张、高度竞争
和经常变更的软件项目



Adaptive Software Development

自适应软件开发

[美] 詹姆斯·海斯
(James A. Highsmith III)

/著

钱岭等 /译



清华大学出版社

软件管理与软件工程译丛

Adaptive Software Development

自适应软件开发

[美]詹姆斯·海斯 / 著
(James A. Highsmith III)

钱岭 等 / 译



清华大学出版社

(京)新登字 158 号

Adaptive Software Development: a collaborative approach to managing complex systems, by James A. Highsmith III
EISBN: 0-932633-40-4
Copyright © 2000 by James A. Highsmith III. Original language published by Dorset House Publishing Co., Inc.
All rights reserved.

本书中文简体字版由 Dorset House Publishing 授权清华大学出版社在中国境内(香港、澳门特别行政区和台湾地区除外)独家出版、发行。
未经出版者书面许可,不得以任何方式复制或抄袭本书的任何部分。

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。
北京市版权局著作权合同登记号:图字:01-2002-0843

图书在版编目(CIP)数据

自适应软件开发:一种管理复杂系统的协作模式/(美)海斯著;钱岭等译.—北京:清华大学出版社,2002

(软件管理与软件工程译丛)

书名原文:Adaptive Software Development: a collaborative approach to managing complex systems

ISBN 7-302-06053-3

I. 自… II. ①海…②钱… III. 自适应控制—应用—软件开发—项目管理
IV. TP311.52

中国版本图书馆 CIP 数据核字(2002)第 086586 号

出版者:清华大学出版社(北京清华大学学研大厦,邮编 100084)
<http://www.tup.com.cn>

责任编辑:熊妍妍

封面设计:孙剑波

版式设计:肖米

印刷者:清华大学印刷厂

发行者:新华书店总店北京发行所

开本:880×1230 1/32 印张:16.625 插页:2 字数:290千字

版次:2003年1月第1版 2003年1月第1次印刷

书号:ISBN 7-302-06053-3/TP·3609

印数:0001~8000

定价:39.80元

编委会名单

主编

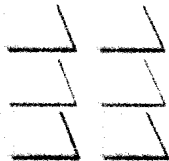
王文京 用友软件股份有限公司, 董事长

总策划

麻众志 闲 洁

编委会委员 (以下按姓氏笔画排序)

万起光 北京湘计立德信息技术有限公司
方春旭 UMLChina
邓俊辉 清华大学计算机系计算机软件研究所
乐林峰 UMLChina
任 伟 北京翰联世纪企业管理技术有限公司
朱于军 朗讯科技(中国)有限公司贝尔实验室
杨作兴 北京方舟科技有限公司
汪 颖 IBM(中国)有限公司上海分公司
陈江北 用友软件股份有限公司
周浩宇 安捷伦科技软件有限公司
孟迎霞 《程序员》杂志社
钱 岭 朗讯科技(中国)有限公司贝尔实验室
熊 节 《程序员》杂志社
潘加宇 UMLChina



策划人语

印象里,从谭浩强老师的《BASIC 语言》在全国畅销,最终销量超过 1 200 万册时,民族软件产业就承载了国人太多的期待。计算机图书出版者也与读者结下了不解之缘。人们将软件产业看作中华民族腾飞的一次历史性机遇。的确,软件产业是头脑产业,中华民族最不缺的就是有智慧的人才。几台微机,一本指南,似乎就可以开公司编程赚钱,而且可以出口赚美元。然而,20 多年转瞬即逝,我们听说过很多软件英雄和编程奇才的故事,却没有产生一个有国际影响的软件品牌或有国际竞争力的民族软件企业。就在我们身边,印度的软件产业却已迅速崛起,产值 10 年增长了 10 倍,数十家企业通过了 CMM5 级认证,印度已经成为世界第二大软件出口国。目前,中国经济发展水平可能要领先印度 10 年,但中国软件产业却落后印

度将近 10 年,原因何在?

几乎所有的人都认为,软件开发是年轻人的职业。程序员们一边挥着汗水,辛苦地熬夜写代码,一边又对自己 30 岁以后的职业发展方向充满惶恐。实际上,我国最缺的是有 10 年以上经验的软件工程师。金山软件公司总裁雷军曾说过:“在印度,包括在美国,我见到的项目经理都是三四十岁的人,他们‘越老越值钱’,有些人甚至拥有超过 20 年的行业经验。”为什么中国的程序员总是在不断学习新的开发工具、钻研程序代码,而不能逐步提升自己的视野、思维和经验?

作为出版者,我们一直关注着中国软件产业的发展,也时刻检讨着自己的责任。长期以来,在各大城市书城和计算机专业书店的书架上,很难找到几本软件管理和软件工程的图书,更多的书店里甚至根本没有软件工程图书的分类。同样在印度,软件管理的图书比比皆是,数量远远超过编程指南的图书。在全球最大的网上书店亚马逊网站上,软件管理和软件工程方面的图书也是门类清晰,为数众多。对这种强烈反差长期的漠视,也许正是我们出版者的责任?

于是,在 2001 年的秋季,我们在一个月的时间内,走访了软件行业中的各大专业媒体、网站、院校和企业,与 50 多位业界专家进行了交流和沟通。最终确定了引进出版一套国外软件管理和软件工程经典图书的设想。希望借



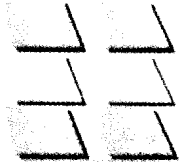
此能将国外成熟先进的软件管理思想和管理经验系统引进到国内,从而推动中国的软件产业的发展,对广大软件工程师的成长和素质提升有所帮助。我们相信,中国软件产业的繁荣和发展,不仅有赖于国家产业政策的支持;还有赖于从业者管理理念的突破和管理思想的普及;更有赖于软件产业中有人文精神的优秀人才的涌现和具有核心竞争力的优秀企业的诞生。

经过将近一年的努力,在业界许多专家的支持下,我们抱着宁缺毋滥的想法,将国外软件管理和软件工程的经典图书完整地筛选了一遍后,选定了这套丛书,其中的每一本均是相关领域中最经典或最具代表性的作品,有的书曾在海外屡获大奖;有的书已经出版 20 年,却仍然畅销不衰。此外,我们对翻译和编辑质量严格把关,虽然两度延期出版,但让我们欣慰的是,如今终于可以向读者和原著者们提交一份负责任的答卷。

在本系列丛书引进和出版的过程中,我们得到了《IT 经理世界》、《程序员》杂志;UMLChina 网站等机构和众多业界专家学者的真诚理解和大力支持,在此向他们一并致以衷心的感谢。读者如有任何意见和建议,请致 hjvp@sohu.com。

和君创业·闲洁工作室

2002 年 10 月



致中国读者

自从 1999 年 12 月《自适应软件开发》一书出版以后，敏捷软件开发/自适应软件开发领域中已经发生了显著的变化。这些最初被称为“轻量级”方法的倡议者们经过一系列的会议和讨论之后，17 个此类方法的作者在美国犹他州的雪鸟城整整开了三天会，并完成了《敏捷方法宣言》（参见 [www. agilealliance. org](http://www.agilealliance.org)）。与会者包括如下方法的代表：极限编程（Extreme Programming）、Scrum、自适应软件开发、水晶方法（Crystal Methods）、特性驱动的开发（Feature-driven development）以及动态系统开发方法（Dynamic Systems Development Method）。

在 2001 年和 2002 年，敏捷方法学已经进入主流社会。很多主流期刊，诸如“ComputerWorld”、“Software Development”、“IEEE Software”、“IEEE Computer”、

“Software Test and Quality Engineering”，甚至包括“Economist”上都出现了关于敏捷方法学的文章。世界范围内的软件研讨会上充满了关于敏捷方法/自适应方法的讨论。很多个人和公司都对在软件开发中使用这些方法表现出浓厚的兴趣，因为它们为多变环境下发布高质量软件提供了一种新视角。这些方案首先关注人、人的技能以及人和人的交流；将过程放到第二位。这些方案首先关注发行能工作的产品；然后才是作为副产品的一系列文档。这些方案首先关注与客户的协作和对变更的反应；然后才根据需要讨论合同事宜和计划。

在世界范围内广为接受自适应软件开发概念，真的令人振奋。在最近两年中，我和很多国家的公司进行合作，并在很多研讨会上发表演说，这些国家包括：英国、爱尔兰、意大利、挪威、波兰、印度、加拿大、新西兰、澳大利亚和美国。本书中文版的发行更进一步表明这种新型软件开发方法对那些创造价值的人——开发人员、测试人员、分析师以及项目经理的重视。

从本质上看，自适应开发和敏捷开发方法都关注三个领域：刚好够用的方法学、协作和混沌观。方法学、过程、文档和度量很快就会变成繁重的包袱和官僚主义。因此对它们的要求是刚好够用，但又不是不够用。作为《自适应软件开发》重要主题的协作，是当今分布式异地开发、技术和业务知识密集型开发中的一种关键技能。最后，在混



致中国读者

S

沌和有序间保持平衡的混沌观为我们提供了一种在扰动的、有时无法预测的世界中保持镇静的能力；这时需要仔细回顾传统的、按计划行事的项目管理策略。

詹姆斯·海斯

2002年8月

亚利桑那州的弗拉格斯塔夫市



To Chinese Readers

Since the publication of Adaptive Software Development in December 1999, there have been significant events in the Agile/Adaptive approaches to software development. After a series of meetings and discussions among proponents of what was initially called "lightweight" methodologies, a group of 17 authors of these approaches met for 3 days in Snowbird, Utah, USA and wrote the Agile Manifesto (www.agilealliance.org). Present at this meeting were representatives of Extreme Programming, Scrum, Adaptive Software Development, Crystal Methods, Feature-driven development, and Dynamic Systems Development Method.

In 2001 and 2002, Agile methodologies hit the popular press. Articles appeared in Computerworld, Software Development, IEEE Software, IEEE Computer, Software Test and Quality Engineering, and even the Economist. Extensive talks about Agile/Adaptive development have filled software conferences around the world. More and more individuals and

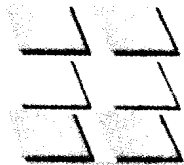
companies are interested in these approaches to software development because they offer a different vision of how to quickly deliver high quality software that is responsive to change. These approaches focus on people, their skills and their interactions first, and process a distant second. These approaches focus on delivering working software first, and streamlined documentation as a byproduct. These approaches focus on collaboration with customers and responding to change first, and contract negotiation and following plans only as needed.

Acceptance of the concepts of Adaptive Software Development around the world has been gratifying. In the past two years I have worked with companies and delivered conference talks in England, Ireland, Italy, Norway, Poland, India, Canada, New Zealand, Australia, and the U. S. Publication of this Chinese edition of Adaptive Software Development further indicates the wide interest in new styles of software development focused on those who actually deliver value—developers, testers, analysts, and project managers.

The essence of both Adaptive and Agile development focuses on three areas—barely sufficient methodology, collaboration, and a chaotic perspective. Methodology, process, documentation, and metrics can rapidly grow bur-

densome and bureaucratic. They should all be kept to a barely sufficient, but not insufficient level. Collaboration, a major theme of Adaptive Software Development, becomes a critical skill in today's world of distributed development and extensive technical and business knowledge. Finally, a chaotic perspective, balancing between chaos and order, provides us with the understanding that the world is a turbulent and somewhat unpredictable place in which our traditional conformance-to-plan approach to project management needs serious reconsideration.

Jim Highsmith, Flagstaff, Arizona August 2002



译者序

这是一本关于项目管理的书。本书作者 James A. Highsmith III 提出了一种适合在当今这种需求多变、技术迅猛发展、市场竞争激烈的分布式开发环境中的项目管理方法。与多数关于项目管理的书不同,本书在内容方面有几个鲜明的特点:

(1) **创新性**。本书介绍的方法虽然与一些传统方法有一定关系,但绝非以往任何一种方法的翻版。此外,作者还提出或改进了很多成功的项目管理技术,包括联合应用开发(JAD)技术和客户中心组技术(CFG)。

(2) **理论基础**。本书在介绍了作者几十年项目管理和项目咨询经验的同时,通过对复杂自适应系统研究领域中最新研究成果的借鉴,成功地把自适应控制理论和软件开发实践结合在一起。自适应理论将各自独立的成功实践

整合为一套完整的方法论。

(3) **可扩展性**。以往的开发方法通常只能适用于某种规模的项目开发。而 James 提出的这种方法不仅可用于管理小型 RAD 项目,而且通过引入新型管理技术,包括工作态管理、结构化协作和时间区间技术,使之同样适用于大型软件开发项目的管理。

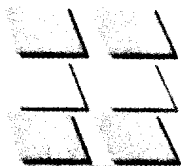
在以往的软件开发和管理实践中,我既使用过类似 XP 那样的轻量级的管理方法,也实施过类似 SW-CMM 这样的重量级方法。然而两者的缺点都是明显的。轻量级方法相对松散,在开发人员数量超过一定限度时,就难以控制;重量级方法的额外开销太大——过程定义、过程部署、过程维护以及过程审计都给正常开发带来了大量额外负担。看到本书提出的自适应软件开发方法,顿时耳目一新,感到它为解决我面临的问题提供了极好的思路。

对于面临中国入世、新技术涌现带来的大好契机的同时,又面对国外大型软件企业巨大压力的国内中小型软件企业来说,如何生存并实现从作坊到大企业的平稳转型就显得尤为重要。自适应软件开发方法从总体结构和具体技术两个层面都给这些企业提供了很多可借鉴的经验。对于企业中的广大开发人员,特别是各级项目管理人员来说,如果能够真正领会并掌握书中介绍的管理思想和各种管理、协作技术,就一定可以应付各种复杂环境,创造出满足用户要求的软件产品。

本书是由一个翻译小组共同完成的。其中第 1、2、3 章由唐忆、徐尧强等人翻译,第 7、8、9 章由朱于军翻译,杨敏强翻译了第 10、11 章的各一部分内容。我翻译了本书的其余部分并对全书做统一校对。此外,衷心感谢清华大学出版社及和君创业闻洁工作室给予的大力支持和协作。

译者 钱岭

2002 年 9 月,于北京



英文版前言

George Johnson 的书《记忆宫殿》(Palaces of Memory)编织了一个令人着迷的故事——关于所有生物现象中最复杂的事情之一：人类大脑。从 Johnson 的书中关于研究神经及其部件——轴突、神经键和枝状结构，并且试图对神经细胞如何互相激励建模并建立工作模式的神经生物学家，到开发神经网络和人工智能的计算机科学家，读者可以得出这样的结论：记忆的工作原理仍然被包裹在谜团之中。

软件开发或许和复杂商业行为一样接近纯脑力活动。正如我们迷惑于大脑如何工作那样，我们被软件产品如何从创造者的大脑中形成这个谜团所吸引。

软件看起来简单。几个操作符，几个操作数然后程序就被创造出来了。然而，组合数学保证了任何长度的程序