



软件工程技术丛书

软件过程系列

软件过程改进

Software Process Improvement

Sami Zahran 著

陈新 罗劲枫 等译



机械工业出版社
China Machine Press



中信出版社
CITIC PUBLISHING HOUSE

软件工程技术丛书

软件过程系列

软件过程改进

Software Process Improvement

Sami Zahran 著

陈新 罗劲枫 等译



机械工业出版社
China Machine Press



中信出版社
CITIC PUBLISHING HOUSE

本书系统地讲述了软件过程改进的思维与方法, 内容包括: 过程思维、软件过程改进框架、启动软件过程改进、当前的软件过程改进模型与标准、软件过程改进的商业收益等五个部分的内容, 提供了有效进行软件过程改进的实际方法, 给出了创建过程支持架构的指南。此外, 作者还以通用软件过程改进框架为基础, 对当前各种主流改进模型的特点与优劣进行了分析与比较, 从而有助于企业在过程改进中对不同的模型加以取舍。

本书内容丰富, 结构合理, 适于计算机及相关专业的本科生和研究生以及软件技术人员使用。

Software Process Improvement: Practical Guidelines for Business Success

Copyright © 1998 by Pearson Education. All Rights Reserved.

Translation copyright © 2002 by China Machine Press & CITIC Publishing House.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as ADDISON WESLEY LONGMAN, a Pearson Education Company.

本书中文版由Prentice Education, Inc.授权机械工业出版社和中信出版社出版发行。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

本书版权登记号: 图字: 01-2002-0857

图书在版编目(CIP)数据

软件过程改进/(美)扎赫兰(Zahran, S.)著; 陈新等译. - 北京: 机械工业出版社, 2002.8

(软件工程技术丛书)

书名原文: Software Process Improvement Practical Guidelines for Business Success

ISBN 7-111-10845-0

I. 软… II. ①扎… ②陈… III. 软件开发 IV. TP311.52

中国版本图书馆CIP数据核字(2002)第063181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: 华章

北京忠信诚印刷厂印刷·新华书店北京发行所发行

2002年8月第1版第1次印刷

787mm×1092mm 1/16·23.5印张

印数: 0 001-5 000册

定价: 49.00元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

谨以此书献给软件过程的前辈——Watts Humphrey。

软件工程研究所提出了“将工程化的方法引入软件开发与维护”的理念，希望本书能为这一目标的实现尽一份力量。

译者序

从20世纪70年代算起，国内的软件开发已经走过了30多年的岁月。但放眼国内，除了极少数几个可称得上“软件企业”的公司之外，我们的绝大部分公司仍然是“手工作坊”式的软件公司，而在我们的近邻印度，其软件出口额每年已高达几十亿美元之多。产生这种差距的原因有很多，例如政策方面的，市场方面的，人员素质方面的，但关键的一点还在于处于国内软件开发一线的大部分软件行业的从业人员缺乏对软件开发科学的正确认识，缺乏正规有效的软件开发过程。

可喜的是，近一两年，国内越来越多的企业认识到了软件过程的重要性，也有越来越多的有识之士投入到对软件过程以及软件过程改进的研究领域之中。但由于整个行业水平的低下，许多企业在从事这方面的工作时仍感到缺少必要的指导，致使无从下手。Sami Zahran的这本书正是为此而写的。书中讲述了什么是软件过程，为什么要使用软件过程，为什么要进行软件过程改进以及如何进行软件过程改进，他为我们指出了一条通向持续软件过程改进的光明大道。但同时应注意的是：软件过程改进是一件“知易行难”的工作，虽然书中给出了一个软件过程改进的良好框架，但在具体实施过程中仍会遇到许多困难。

希望本书的翻译能为国内软件行业的发展尽一份微薄之力。参加本书翻译工作的还有胡旦旦、荣勃、耿岳、王江波、潘龙、张科峰、汪洋和付钢等。

由于本书中大量使用专门术语，加之译者水平有限，在翻译过程中难免有错误，欢迎指正。读者还可就软件过程改进方面的内容与我们交流。我们的E-mail地址为：spi_book@hotmail.com。

汉弗莱序

我荣幸地向大家介绍Sami Zahran撰写的这本书。他以一种有趣的方式讲述了一个重要的问题。对于我来说，阅读这本书是一种享受，我相信你也会如此。除了讲述过程改进之外，作者还从实践者的角度提供了实用的指导。他清晰明了地解释了软件过程改进的目的和方法，还对其中最主要的方法以及相应的重要特性进行了比较。最重要的是，他讨论了作为使用者来说在进行过程改进时将会面临的问题。

当你阅读本书并思考Sami所说的话时，我建议你在脑海中保留一些特别的主题。它们可以提供有用的观点，有助于以最有效的方式进行过程改进。

在商业的革新中，有一种称之为解冻的东西。从本能上来说，人们会抵制变革，而解冻可以打破他们的抵制。告诉工程师以及他们的经理，现在什么地方是错误的，让他们对现状感到更加不满，使他们相信还有更好的、更有效的工作方式。当工程师和经理们在一大堆问题中找到了所有的关键问题时，他们就会意识到他们真的需要做点什么来改变现状了。

完成一个合适的软件过程评估将走过软件过程的所有细节。打破坚冰，开始解冻。接着，你就可以开始考虑如何改进了。除非能够做到这一点，否则所有关于改进的讨论只会局限于讨论本身，而不会再有其他成效。这就是评估过程产生的原因。通过工程师与经理们的参与，询问他们什么地方做的不对、什么地方可以改进，这样的评估才真正能够成为组织内不断学习的方式。当你知道了人们关于改进有什么想法之后，才能让他们参加并支持改变过程。

另一个容易引起误解的问题是关于能力成熟度模型（CMM）目标的。首先，在开发CMM模型时，我们没有想拒绝任何好的想法。软件界中充满了富有创造力的人们，一旦参与到改进活动中来，他们就会发现还存在着许多进行改进的方式。而且软件工程本身也是一个相当新的并且正在快速变化的领域。今天看起来正确的东西，可能仅仅过了几年就过时了，毫无用处了。一个过分详细的CMM模型，很可能会妨碍而不是鼓励改变的发生。

这就是为什么说CMM的真正目的是目标，其余的都是关于如何满足这个目标的例子。因此我建议你应该关注于目标本身，并且要使用所有可能的方式去实现这个目标。

另一个问题则是关于成熟度等级的。等级有助于我们将注意力集中在最高优先级的问题上面。虽然这是一种最好的交流方式，但是有时也会导致错误的行为。人们容易忽略过程改进最基本的目的，而却把CMM中下一个等级作为自己的目标。

当人们把注意力集中于CMM的等级上时，他们就会认为评估是度量等级的一种方式。由于将CMM作为一种度量工具，人们接着就开始关注于它的准确性了。一些人甚至试图寻求一种复杂的、可以精确判断成熟度等级的度量方式。这真是荒谬。软件过程本身太过于复杂了，以至于无法通过一个或者几个数字来加以度量。我们最需要的是一个框架，通过这个框架，可以使得工程师以及经理们来检查并讨论他们的过程，最后根据对组织情况的详细了解，

可以确定出关键问题之所在，从而决定对什么进行改进和如何改进。

因为关于等级的度量还是具有许多价值的，所以我们仍应将其保留下来。我们建议你将注意力集中在少数的、可以立刻带来成效的改进上。CMM可以指导你去发现这些机会。除此之外，不要担忧成熟度等级。如果你持续地进行过程改进，那么CMM等级自然也会相应地提高。否则，如果只关注于CMM等级的话，那么就不可能进行什么改进了。

为了支持上述观点，建议你应做如下一些事情：

- 1.使用成熟度等级进行评估、评价并设置有关的优先级。
- 2.将目标集中在进行特定的改进方面。
- 3.将改进变成每位经理工作的一部分，对他或她们的目标实施情况加以追踪。
- 4.制定计划，以便尽早实现高优先级关键过程域（KPA）的改进。
- 5.当成功实施了有关计划后，为下一个最重要KPA的改进制定计划并加以实施。

如果完成了上面的工作，当进行下一次评估时，你会发现有明显的改进，而相应过程的等级也自然会随之提高。

Watts S.Humphrey

美国，佛罗里达州，萨拉索塔

波 克 序

熟悉计算机的人们通常都会感受到“软件危机”的痛苦。在创建那些与计算机相关的软件系统方面，虽然我们的能力与五十年前相比已有了巨大的提高，但人们对软件的要求增长得更快，而软件行业也仍然处于从手工作坊式方式到采用工程化方式进行开发的不断进化过程中。总的来说，我们最终得到的是一种“慢性的软件危机”：软件的推出（几乎）总是晚于预期的时间，成本总是高于预期的预算，功能总是少于预期的内容。但是，我们仍然有希望扭转当前这种软件危机的现状。

如果我们打算克服软件危机的话，方法之一就是采用Sami Zahran这本书中所讲述的内容——软件过程改进。在为Watts Humphrey的《管理软件过程》(Managing the Software Process)一书所写的前言中，Peter Freeman写道，“‘软件危机’已经消亡了！”。而Humphrey的这本书正是这种转变的最好标志。

经过了八年的时间，现在软件组织在规定的时间内能够开发出高质量软件产品的能力在不断地提高，而这也正说明了Freeman当年提出的观点的正确性——至少在系统地采用了软件过程改进方法的软件组织中是正确的。

但遗憾的是，只有少部分的软件组织采用了系统的改进方法。这其中的原因很多，但关键原因或许还在于软件工程知易行难的特点。

由于“改变我们做事的方式”将会引发行为的全面改变，而这也导致了大部分问题的产生。真正的软件过程改进需要管理层、特别是高层管理者的参与。与此同时，还需要处于不同层次的人员参与到对有效可用过程的定义以及实施中来，而这就意味着他们需要从所谓的“实际工作”中抽出身来。如果我们仅仅将软件改进看做是解决所有问题的“法宝”而不是对未来的一项投资，那么它就会成为另一种转瞬即逝的时尚潮流，它的价值也就无法体现出来。

改进的过程意味着我们将不得不面对一些令人不快的现实。软件危机的部分痛苦来源于人类的本性。对于“软件成本为什么如此之高”这一问题，Jerry Weinberg认为“这要看与谁相比”，而Tom Demarco认为这些提法过于负面。人们之所以抱怨是因为他们知道只有当他们抱怨时，我们才会更努力地工作。在一次调查中，大多数被调查的专业软件经理报告说：虽然他们的估计是令人沮丧的，但这并不是针对整个评估过程。虽然大部分专业软件人士同意Demarco的观点，但仍有许多软件经理和客户对“如何更有效地管理软件项目”这一问题十分感兴趣。

如果将进度压力以及加班加点作为开发中的激励手段，那么这样的客户和经理最终将不得不在质量方面加以取舍。那些对软件项目管理真正感兴趣的客户与经理们（有时也必须勇敢面对现实中那些令人不快的情况）可以采用一系列的方法对软件开发与维护过程加以系统地改进。

对这些方法的成功应用，将会带来希望的最终结果，而这也导致软件危机的最终终结。

最著名的软件过程改进方法包括：国际标准化组织的ISO 9001质量管理标准、软件工程研究所的软件能力成熟度模型（CMM）、用于软件过程评估的ISO 15504标准（通常称做SPICE）。在这些以及其他一些方法中，都将全面质量管理的原则运用到软件过程之中，而这在Zahran博士的书中都有所描述。希望读者通过本书中的有关论述，能够找到适合自己的改进方法。

作为Software CMM的产品经理，我对各种软件过程改进方法都带有偏见。在下一版的软件CMM中，我们融合了大量的来自于本书所提及的标准与模型中关于过程实现与管理的各种思想。虽然我相信Software CMM是进行软件过程改进的最好基础，而我们也积极努力地工作以便保持这种地位；但是应当说任何一种系统的改进方法都能够帮助组织获得成功。无论选择何种方法，过程改进都将成为在当今这个高度竞争的世界中生存下去的必要条件。

无论怎样强调软件产品质量的重要性都不过分。最近，英国法院对美国统一商业代码（US Uniform Commercial Code）的有关判决以及修改建议预示着来自不满客户的潜在诉讼。与其他商品一样，软件发售时应能按预定的要求工作并且没有重大的缺陷，这一点应该能够成为大多数软件开发人员的目标。

不用说是为了不断成长壮大，就是为了生存，现代的组织也必须不断地对各个商业方面加以改进。与软件相关的产品与服务的改进是十分关键的，同时也是很困难的。在巨大的压力之下，我们所面临的挑战是如何使软件组织能够实施优秀的软件工程以及管理实践。对于生存与发展来说，一个规范化的、系统的软件过程以及质量改进方法都是必须的（就像那些模型与标准所支持的那样）。

然而，要想取得成功，仅有过程改进是不够的，其他方面也是必不可少的，例如：

- 创建用户所要购买的合适产品。
- 雇用、选择并且维持具有竞争力的成员。
- 突破组织的界线（例如，系统工程师与软件工程师之间的界线）。

有效的软件过程改进应该与组织中的其他活动相配合（可能它们都属于全面质量管理的范畴），以便解决那些与过程改进相关的所有商业问题。

无论采用何种方法，竞争优势都是建立在过程改进之上的，而不是仅仅获得一个好分数（无论这个分数代表的是成熟度的第几级，还是表示获得认证）。Zahran博士的这本书可以帮助读者更好地理解软件过程改进以及与之相关的其他内容。

Mark C.Paulk

软件工程研究所

宾夕法尼亚州匹兹堡

前 言

软件开发是具有挑战性的工作

能够在规定的时间和预算内开发出可靠的软件对于许多组织来说都是一件非常困难的事情。在目前的商业活动以及人们的日常生活中，软件扮演着越来越重要的角色，因此由于软件产品延期、超过预算或者根本无法工作所引发的问题也变得日益严重。由于不可靠软件所导致的死亡或者在很大范围内引起的生活不便屡屡成为新闻媒体的头条。据估计，在最近几年中，由于软件缺陷导致的死亡人数已近4000人。在现代化的飞机中，如果软件停止运行200毫秒，就将造成不可挽回的损失。1996年6月，欧洲空间中心带着数颗卫星的火箭在发射数秒后发生爆炸，这次事故正是由于软件故障所导致。几年前，不可靠的软件成为英国国内的大新闻，从紧急情况处理系统的故障，到社会保障支付系统的失误，更不用说伦敦股票交易所这种大型项目中的故障了。提高软件质量是确保获得可靠产品与服务 and 获得客户满意度所不可或缺的。美国总审计局（US Government Accounting Office, GAO）的报告显示了以下一些情况：耗资数十亿美元的系统无法如预期的那样工作，成本以百万计的数字上升，进度的延误不是以月计而是以年计（Paulk等,1994）。

仅有CASE工具是不够的

让新闻界激动不已的失败的软件项目与20世纪80年代中后期CASE工具在媒体上所夸下的海口形成了强烈的对比。工业界已经意识到仅有CASE工具是不够的。软件业已经形成一种共识，即“有工具的笨蛋依然是笨蛋”。通常情况下，商业解决方案包含三方面的内容：人、过程与技术。已有的工业方面的经验显示，当实施商业解决方案或者引入改革时，通常最不可能产生的问题就是技术问题，而过程和人员是导致成功或失败的关键因素。人员是过程不可分割的一部分，因为人是整个过程活动、过程监控、过程管理的关键。

仅有合格的人员是不够的

软件行业使用CASE工具的经验已经证明，导致软件项目失败的主要原因几乎与技术、工具没有任何关系，更多的是由于缺乏过程规范。软件开发需要团队的努力，如果缺少过程规范，团队就可能遵循不同的过程。更普遍的情况则是团队根本就没有任何经过定义的过程。这就如同“在同一支球队中，一些人在踢足球，一些人在打棒球，而其他的人却在打橄榄球。在这样的环境下，即便拥有最好的球员，也只能组成一支最差的球队”（Humphrey,1995）。与此相反，一个遵循统一过程的团队可以很好地协调成员间的工作，为实现共同的目标，将团

队成员工作的注意力都集中在一起并且对进展情况可以进行更加精确的跟踪。

以过程为中心可以提供更多的成功机会

作为一种规范,软件开发已经有40多年的历史了,但是我们仍然无法将软件行业转变为工程规范化的行业。而近年来,对软件开发过程的关注则是向这个方向迈出的正确一步。只有通过创建规范的软件开发过程,我们才可能管理并控制软件产品的质量。各类组织已经意识到,他们的根本问题就在于软件开发过程的不成熟。所有的证据表明,投资于软件过程改进将会为软件行业带来更多的希望,而这就如同在制造行业和其他行业中所发生的那样。此外,提倡以工具为基础的软件开发活动的动机与基于过程改进的软件开发活动的动机是不同的。过程改进是软件开发组织的责任,其中并没有包含工具制造商的利益。通过使用工具来自动规范一个混乱的过程将导致更大的(自动化的)混乱。这样的例子比比皆是。在软件行业之外的最引人注目的一个例子就是:将复杂的武器工厂以及大规模杀伤性武器引入原始国家的两个部落之间时必定引起一场混乱的战争。这样的“技术性工具”不可能导致稳定,只可能增加人们的痛苦。经验显示,在不成熟或者没有规范的环境中,引入新的技术与工具可能会导致混乱的情况更加严重。没有规范的过程来加以控制与管理的话(例如质量保证、配置管理以及项目管理),这样的项目是不可能从工具中受益的。将工具用到这样的项目中只可能增加混乱,加速更多有缺陷的产品的生产,使客户的不满意度成倍提高。这样的项目最终将以各个方面的失望而告终。这里的失望有各种表现形式,例如浪费的人力、时间、金钱、资源以及可能带来的无法避免的灾难。

软件过程改进运动:软件行业的第二次浪潮

为了满足在软件需求和复杂性方面日益增加的要求以及因此导致的开发团队规模的不断扩大,结构化开发方法于20世纪70年代逐渐发展起来了。这是软件行业的第一次浪潮。它是为了满足那些利用共享系统去建立复杂的交互式商业应用以及对这些系统进行维护而发展的。结构化方法所关注的是方法本身,即需求定义的正规化以及设计、创建那些可以在最终系统中加以追踪的需求的方法。已有的自动化的工具可以在某种程度上有助于进行部分此类转换。尽管这是将软件开发从“手工作坊”向大规模生产进行转换的开端,但这还远远不够。而那些导致软件项目成功或者失败的真实原因(例如项目管理以及需求管理)还未成为关注的焦点。软件过程改进是对不断提高的软件项目失败率的一种反应。对过程的关注开始于美国国防部(DoD)最初对软件工程研究所(SEI)的赞助,SEI为美国国防部提出了评估软件转包商能力的方法。Watts Humphrey刚开始加入SEI时并没有明确的职位,在开始的数月中,他的头衔是SEI过程计划的主管。从那时开始,SEI所产生的关于过程的讯息就越来越多地影响到了全世界的软件行业。

人们可以很容易地将软件过程追溯到20世纪30年代,盛行于20世纪70、80以及90年代的质量控制运动。质量控制概念的领袖(例如Edwards Deming 和 Philip Crosby)在全世界的制造业获得了很高声望。Watts Humphrey在软件开发中引入了同样的质量控制概念。过程成熟度运动为软件行业的第三次浪潮铺平了道路,而这第三次浪潮就是“软件行业化”。在第三次

浪潮中，软件开发将变得如同装配与制造过程一样。发动第三次浪潮的技术因素包括面向对象技术以及重用组件库，这使得利用标准重用组件以装配软件成为可能。第三次浪潮启动的关键因素是规范化的软件工程过程，在这个过程中，质量、进度与功能都是可以预测的。

本书的目的

本书提供了关于有效进行软件过程改进的实际方法，给出了创建过程支持架构的指南，以便读者能够以过程的观点来看待软件开发。书中概述了建立规范的以及持续改进的软件过程环境的实用方法。总的说来，本书给出了建立有效而持续的软件过程改进环境的框架。

本书的有关论点强调了与人有关的因素，理解过程并遵循过程和过程的定义是同样重要的，书中同样强调了过程规范对团队能力和实现商业目标的影响。

书中所提供的概念源于笔者从事软件行业三十多年的实际工作经验。在这些年中，笔者花费了大部分的时间去从事软件工程各个不同方面的工作与研究，其中包括软件开发以及软件项目管理中所经历的各种酸甜苦辣。这些项目覆盖了各种商业以及工业应用，包括从石油、银行、政府部门到国防、制造以及航天业。同时也包括操作系统、数据库管理系统、数据字典系统、事务处理系统以及大量的商业应用的开发。因为经历过多次的成功与失败，所以笔者于1992年第一次欣然到Carnegie Mellon大学的软件工程研究所去接受关于过程的培训。从那时开始，我对软件过程的热情就再也没有中断过。

读者对象

本书有广泛的读者，包括具有软件过程评估与改进知识的人们以及那些仅仅知道这个主题对他们的工作有意义的人们。换句话说，本书包括一些新的想法与方法，可能会使那些已经具有相关知识的人感兴趣，同时它又简单易读，因此同样适合于那些对此问题感兴趣的人。本书主要是针对下列这些特殊人群：

软件工程师和专业人员

参与软件过程活动的人们应该会对本书感兴趣，这些活动包括管理、协调、开发与控制。涉及的人员包括：对软件感兴趣的商业经理、项目经理、团队领导、软件工程师、软件支持人员，例如品牌管理、质量保证以及过程改进组。同时，本书也可以作为从事软件过程改进工程研究的研究生读物。

过程改进组

本书第一部分采用了通常的概念对过程思维进行讨论。它应该使所有关注过程改进活动的人受益。这些活动包括业务流程再造、业务流程再设计以及业务流程改进。

从事过程研究的科学家

书中提供了过程改进环境的整体方法。书中讨论的概念与方法可以使读者保持进一步研

究过程建模和质量概念的热忱。

本书的结构

本书分为五个部分并附有词汇表和参考文献。

第一部分：过程思维

这部分是本书其余部分的知识基础。它定义并解释了什么是过程思维并将这个概念与过程规范联系在一起加以讨论。此部分描述了有效过程环境的特征，同时还将这些概念与软件过程环境相联系。第一部分包括4章：

第1章：过程思维

第2章：过程规范

第3章：有效的过程环境

第4章：过程成熟度：软件工业的第二次浪潮

第二部分：软件过程改进框架

这部分描述了软件过程环境框架，介绍了框架的组成、过程评估方法以及过程改进计划。这个框架将会带来一个持续的过程改进环境。第二部分包括5章：

第5章：一个软件过程改进的框架

第6章：软件过程架构

第7章：过程改进规划图

第8章：软件过程评估的基本原理

第9章：软件过程改进的行动计划

第三部分：启动软件过程改进

这部分描述了如何在组织中计划并启动软件过程改进活动的策略与计划，讨论了将评估结果转换为改进计划的步骤，强调了度量软件过程改进收益的必要性。此外，本部分还讨论了软件过程改进的关键成功因素，总结了实施软件过程并使其制度化、度量商业收益时建议采用的一些方法。第三部分包括3章：

第10章：启动软件过程改进

第11章：软件过程改进的实现和制度化

第12章：度量软件过程改进带来的收益

第四部分：当前的软件过程改进模型与标准

第四部分描述了一些进行软件过程改进以及评估的著名模型与标准，其中包括软件 engineering 研究所的过程成熟度模型（CMM）和基于CMM的评估、国际标准化组织的ISO/IEC 15504及其评估指导草案、BOOTSTRAP评估方法，此外还讨论了其他一些早期的工作，包括ISO 9001、MIL-STD-498、Trillium和V-模型。第四部分总共包括6章：

第13章：能力成熟度模型（CMM）

第14章：基于CMM的过程评估

第15章：ISO/IEC 15504软件过程评估的技术报告草案

第16章：ISO/IEC 15504评估操作指导草案

第17章：BOOTSTRAP软件过程评估

第18章：其他早期工作（ISO、MIL-STD-498、Trillium及V-模型）

第五部分：软件过程改进的商业收益

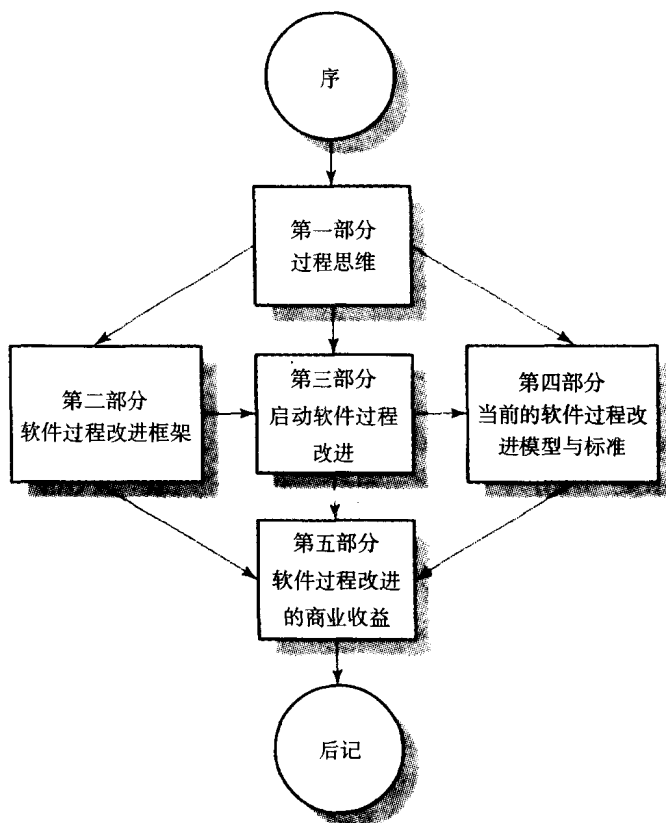
第五部分描述软件过程改进的商业收益，总结了关于追踪并度量收益方面的一些欧洲与美国的案例。最后，讨论了未来软件过程改进的一些设想。第五部分包括2章：

第19章：软件过程改进能够带来商业收益的证明

第20章：后记：软件过程改进的未来

如何使用本书

书中的每个部分都有一个主题，而且每个部分对于其主题来说都是自包含的。下图说明了书中的结构和可能的阅读顺序。



本书特点

本书将引导你从软件项目的“救火”行为过渡到规范化过程、缺陷避免以及持续改进的文化。本书有以下一些特点：

- 提供了用于支持软件过程改进环境的模型以及体系结构的整体框架，它们可以帮助组织建立一个有效的过程改进环境。
- 提出了关于建立软件过程改进架构以及过程规范的相应框架。
- 描述了著名的软件过程改进的早期工作与标准，包括CMM、ISO/IEC 15504、MIL-STD-498、Trillium、V-模型、ISO 12207以及BOOTSTRAP。
- 在软件过程评估的一般模型以及基于组织可获得的评估标准与方法从而设计自己的过程评估方法这两方面，本书提出了相应的一般模型。
- 提供了成功建立并管理软件过程环境的实际步骤。
- 每章都有说明与模型，这些说明与模型有助于强调所讨论的主要概念。此外，每章后面都有小结，用于帮助读者理解本章的主要概念。
- 本书包括五个部分。每个部分分别讨论了建议的软件过程环境框架中的一个主要组成部分。因此，你只需支付一本书的价钱，却可获得五本书的内容。

不做承诺

本书所总结的方法与标准仍处于不断的发展之中，因此笔者不能保证书中所提供的信息能够反映这些标准与方法的最新状态。这些内容基于本书写作时可以获得的信息，而且仅代表笔者本人的观点。希望有关的解释与总结能为读者提供关于这些方法和标准的一个合理的认识。如果对最新的信息感兴趣的话，建议读者直接与那些组织联系。

致谢

本书的写作历经了五年多的时间，书中的许多观点来自于与软件过程领域的几位世界著名大师的讨论结果。尽管不能一一列出每一个参与了讨论的人，但我仍非常感激下面这些对我的思想产生深远影响的同事们。

来自软件工程研究所的同事

感谢Watts Humphrey，我通过他的工作论文认识他已近十年了，1996年在西班牙的巴塞罗那我和他本人见面了。他对于过程领域的出现、领导和信仰使他成为软件过程活动中无可争议的“先知”。我同样感谢Tim Kasse、Jeff Purdue和Jim Hart，我在1992年去软件工程研究所接受关于软件过程的学习时遇见了他们。他们已经成为过程教育的忠诚的“传教士”，他们使我“皈依”于过程信仰。此外，还要感谢软件工程研究所的Mark Paulk，早在我们在伦敦和巴塞罗那见面的几年以前，我就已经通过他的CMM论文认识了他。Mark关于CMM和软件过程的深刻理解已经影响了我的关于软件行业过程改进的关键地位之观念。还要感谢软件工程研究所的

David Kitson, 我在英国剑桥见到他以前就已经看过他的CMM培训录像。上面提到的所有人都是软件过程运动的先驱, 他们已经成功地影响了全世界。最后, 感谢Charles Weber和Suzanne Garcia, 他们帮助我查阅CMM的资料, 还提供了关于CMM最新进展的有用的评论与指导。

来自欧洲委员会的同事

David Talbot、Brian Holmes和他们在比利时布鲁塞尔的欧洲委员会的同事推动了欧洲委员会的软件最佳实践计划。这个计划在整个欧洲资助并鼓励了相当多的软件过程改进项目。他们的欧洲软件与系统自主权(ESSI)及过程改进实验(PIE)在欧洲软件项目中蓬勃发展起来, 为软件行业带来了实际收益, 增加了软件过程意识。

来自英国的同事

感谢Harry Barker, 他是英国国防研究局的成员, 他的团队领导了英国国防部的软件过程改进计划, 感谢Colin Tully, 他是推动整个欧洲范围内软件过程改进信息传播的领导者; 感谢Jennifer Stapleton、Mike George、Jane Searles、Bev Wilson和Alec Dorling, 我们一起在英国的各个地方进行了多次讨论, 他们给出了一些有益的评论; 感谢国际标准组织的Roy Farmer和David Booth; 感谢Robin Whittey, 他提供了ami的资料以及与我就软件过程度量的问题进行了充分讨论; 感谢A.Furnham教授就组织变革策略进行的有益论述。感谢Peter Goodhew、Agnes Donaldson和Richard Thomas, 他们提供了IBM的软件开发实践基准的资料。最后特别感谢Essam Isa、Norman Ward和Sam Waters教授以及Peter King教授, 这些人都曾对我在计算机领域从事的事业产生过影响。

来自欧洲其他地方的同事

感谢希腊的Ch.Avratoglou博士, 他与我分享了运用三个ESPRIT软件过程改进项目的经验与总结; 感谢瑞士的Rolan Simon博士、Rolf Kaul和Hans Rene Klaey博士, 感谢他们的论述, 感谢他们与我分享了实现软件过程改进的经验; 感谢丹麦的Anne-Mett Jonassen、Jorn Johanssen和Carsten Joergensen, 他们支持并推动了丹麦的软件过程改进; 感谢葡萄牙的Jose Oliveira和Daniel Ferreira, 他们与我一起分享了软件过程改进的经验; 感谢Hans Stienen博士和Hartmut Gierszol, 感谢他们提供了BOOTSTRAP的资料与评论; 感谢BOOTSTRAP研究所的Pasi Kuvaja。

来自Addison Wesley Longman的同仁

这里真诚地感谢Addison Wesley Longman的执行编辑Sally Mortimore博士, 感谢她的耐心、鼓励、指导与评论; 感谢她的助手Fiona Kinnear, 感谢她高效的工作与积极配合。感谢参与本书出版工作的所有成员, 感谢他们辛苦的工作和奉献。

我的家庭

感谢我的妻子Nadia、我的女儿Reem和我的儿子Wesam, 感谢他们的鼓励、耐心与支持。

软件过程改进的拥护者

最后，我相信在全世界有数以千计的参与到软件过程改进网（SPIN, Software Process Improvement Networks）的致力于软件过程改进的软件工程师与经理们。感谢全世界各种类型组织内的软件过程改进的拥护者。

Sami Zahran

英国，白金汉郡

Email:sami@cloud9.win-uk.net