

实用软件度量

Practical Software Measurement Objective Information for Decision Makers

John McGarry,
(美) David Card, Cheryl Jones, 著
Beth Layman, Elizabeth Clark,
Joseph Dean, Fred Hall

吴超英 廖彬山 译



机械工业出版社
China Machine Press

实用软件度量

Practical Software Measurement
Objective Information for Decision Makers

John McGarry,
(美) David Card, Cheryl Jones, 著
Beth Layman, Elizabeth Clark,
Joseph Dean, Fred Hall

吴超英 廖彬山 译



机械工业出版社
China Machine Press

实用软件度量 (PSM) 是设计和实现基于项目的软件度量程序的过程, 它提供有关进度、资源分配和技术性能等方面的主要信息, 使项目决策者可以制定出正确的决策。

本书由PSM开发启动的领导者所撰写, 是对PSM的官方的、权威性的指导。它描述了开发、运行以及持续改进组织的度量程序的原理和实践, 使用真实世界的示例来说明实际的解决方案和特定的度量技术。本书适合软件管理人员和开发人员使用, 也可作为高等院校计算机专业及软件学院的参考教材。

Simplified Chinese edition copyright © 2003 by PEARSON EDUCATION ASIA LIMITED and China Machine Press.

Original English language title: *Practical Software Measurement: Objective Information for Decision Makers* (ISBN 0-201-71516-3) by John McGarry, David Card, Cheryl Jones, Beth Layman, Elizabeth Clark, Joseph Dean, Fred Hall, Copyright © 2002.

All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison-Wesley.

本书封面贴有Pearson Education (培生教育出版集团) 激光防伪标签, 无标签者不得销售。

版权所有, 侵权必究。

本书版权登记号: 图字: 01-2002-1837

图书在版编目 (CIP) 数据

实用软件度量 / (美) 麦加里 (McGarry, J.) 著; 吴超英等译. - 北京: 机械工业出版社, 2003.9

(软件工程技术丛书: 软件度量系列)

书名原文: *Practical Software Measurement: Objective Information for Decision Makers*
ISBN 7-111-12809-5

I. 实… II. ①麦… ②吴… III. 软件-测试 IV. TP311.5

中国版本图书馆CIP数据核字 (2003) 第068434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: 姚 蕾

北京昌平奔腾印刷厂印刷 · 新华书店北京发行所发行

2003年9月第1版第1次印刷

787mm × 1092mm 1/16 · 15.75印张

印数: 0 001- 5 000 册

定价: 33.00 元

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

译者序

本书详细、准确地论述了软件度量的意义、主要内容和实施方法，深入地考察了软件度量程序的基础，给出了大量可应用于软件项目的度量构造的示例，提供了两个不同类型项目的综合性的案例研究，并明确地指出：客观的、有意义的和量化的度量是成功地开发当今复杂软件的关键。

我们认为，本书是一本有关软件度量既实用又权威的著作，讲述了所有与计算机密集型系统有关的开发人员、管理人员、使用人员、技术支持人员以及后勤保障人员应该密切关心和逐步掌握的课题，同时阅读本书也是各个组织按照CMM/CMMI或其它模型深入进行过程改进的重要前提。通过阅读这本著作，你将会更深切地体会到软件工程确实是软件学科、人文学科和统计学科的交叉学科。

在本书翻译的过程中，我们曾部分地参阅了北京航空航天大学软件工程研究所李云辉、任发科和武硕的译稿，在某种意义上，可以说他们也是本书的译者。特别要提出的是，在全书的翻译过程中，周伯生始终对我们进行了帮助和指导。此外，北京赛柏科技有限责任公司周柳芳进行了部分录入工作。在此我们谨向他们表示诚挚的谢意。

吴超英 廖彬山

2003年6月于北京航空航天大学

序

软件工程的未来世界将与过去或现在有很大不同。我们的系统将更加复杂。它们大部分将是系统的系统，有不同的目标，但最终都强迫它们相互协调地协作并一起进化。

关于系统的系统的最安全的假设就是每个事物最终都是相互联系的。即使这样，事物有时仍不按给定的方式运作。移动系统在进入一个隧道或电池用完时将失去联系。由移动引起的混乱增加了这些复杂事情，各自独立的代理商就在网络空间做起生意来了。^①

不幸的是，当软件系统的复杂性增加时，可控能力就降低了。软件系统的内容越来越多的由商用成品（commercial off-the-shelf, COTS）驱动的。销售商可能会倾听你的需要，但他们的COTS特征及其最终的发展将对整个市场要求作出响应。除此之外，在日益竞争以及上市时间压力的驱使下，不得不在越来越短的时间内对软件的需求、构架、过程和资源作出决策。更进一步来说，信息技术快速变化的步伐意味着你的项目承诺必须适应技术、市场和公司结构的中途变更。

当然，如果你能明白如何最佳地使你的组织适应新的操作模式，伴随着所有这些挑战的将是获得巨大成功的机会。

在这方面已经取得成功的大多数组织都使用了一个完整的软件度量程序。那就是，他们已经而且正在收集和分析数据，用以表明他们在何处，遇到了什么问题，应采取哪些更好的策略来改进他们的状况。有些特别好的成功事迹被记载下来，如Bob Grady（1992）记载了惠普公司的经验，而Walker Royce（1998）记载了TRW的CCPDS-R的经验。

不过，盲目地投入一个软件度量程序，就会带来较高的失败风险。如下是一些常见的错误。

- 度量过度 收集数据太多，导致浪费了工作量和降低了士气
- 度量功能失常 将度量局限于个性化评价，导致“他们想听什么，就告诉他们什么”的数据
- 度量不匹配 挑选错误的、模棱两可的或不一致的度量，导致没有结论的数据分析
- 过程不匹配 挑选增强一些错误过程的度量（例如，最小化平均问题报告关闭时间，将导致人们先处理容易的问题）

幸运的是，你不需要盲目地使你的组织投入到一个度量程序中去。因为本书为开发、操作和持续地改进一个组织的度量程序提供了原理和实践。以下是一些关于实用软件度量（Practical Software Measurement, PSM）方法的重要论点：

- PSM基于几十年来数十个组织在学习如何最佳地实现一个成功的软件度量程序中所获得的经验，它的原理和实践已经成功地用在各种软件密集型项目中。
- PSM不是一个“万能的”方法，它也指导你裁剪度量框架和实践，以适合你的组织和项目

① 这句话的意思是不靠移动系统的管理了，靠网络各自做生意。——译者注

的情况。

- PSM不仅是一个抽象的指南集，它还通过案例研究来说明哪个指南能应用到嵌入式系统和商业系统领域内的特定项目中。案例研究不仅提供了“如何做”的信息，而且提供了“为什么这样做”的动机，同时为实践人员和管理人员提供复杂软件项目状态的可视化的评价，并为标识和控制问题状况提供了诊断信息的评价。
- PSM与新近启动的研究项目兼容并有助于它们的实现。这些项目包括能力成熟度模型集成（CMMISM）、ISO/IEC 15939（软件度量过程），以及正在由USC/UMD经验中心（Center for Empirically Based Software Engineering, CeBASE）对之进行试验的软件密集型的、系统级经验工厂。
- PSM是通过一组工具、培训课程、讲座和研讨会进行完善的，从中PSM经验得到了共享和讨论。

由于具备了所有这些支持优势，PSM为你提供了实现软件度量程序的重大效益的方法，同时帮助你理解和避免因“盲目投入”而带来的风险。你会发现本书对你未来的软件度量启动是很有价值的起点，也是持续地指导你绘制在复杂的机会之海中前进的航线图的来源。

Barry Boehm

南加州大学软件工程中心主任

2001年5月

前言

按事实进行管理已经在软件工程界和信息技术界成为日益流行的概念。组织正将关注点集中于度量和使用客观信息来制定决策。定量的性能信息是基于事实进行管理不可缺少的。本书在集成了度量信息模型（Measurement Information Model, MIM）和度量过程模型（Measurement Process Model, MPM）概念的基础上，为软件项目管理者描述了按事实进行管理的方法。尽管这些概念也适用于非软件活动，但本书介绍的示例和术语主要针对软件。

决策制定者的信息需要驱动软件度量和相关分析技术的选择。这是广为接受的软件度量方法的前提，包括目标/问题/度量元（Basili and Weiss, 1984）和待研究问题/分类/度量（McGarry et al., 1997）。信息需要由管理者的工作所产生，这些工作影响项目、过程和朝着已定义目标启动的成果。信息需要通常从两个来源派生：（1）管理者寻求达到的目标，（2）妨碍实现这些目标的障碍。障碍也属于待研究问题，包括风险、问题以及缺乏在目标相关领域中的信息。除非有一个管理者或其他具有信息需要的决策制定者，否则度量不起作用。软件项目管理者面临的问题是无数的，通常包括估计和分配项目资源、跟踪进展以及交付满足客户规格说明和期望的产品。

度量信息模型定义了管理者的信息需要与要收集的客观数据（常称为**度量**）之间的关系。它还为基本度量思想和概念建立了一致的术语，这对于决策制定者来交流度量信息是至关重要的。在实用软件度量中的信息模型定义了三个层次的度量：（1）基本度量，（2）派生的度量和（3）指示器。有趣的是，这三个层次的度量大致是与许多已有的度量方法所倡导的三层结构相对应的。已经在软件界中使用的例子包括目标/问题/度量元（Basili and Weiss, 1984）、因素/准则/度量元（Walters and McCall, 1977）和待研究问题/分类/度量（McGarry et al., 1997）方法。Kitchenham等人开发了一个定义度量的通用数据结构的类似方法，将它们的结构定义为一个实体关系图（1995）。

一个有效的度量过程必须首先选择合适的度量，同时提供对已收集数据的有效分析。**度量过程模型**描述了一般可应用于所有环境的相关的度量活动集，而不考虑任何特殊情况下的特定的信息需要。该过程由四个迭代的度量活动所组成，这四个度量活动包括建立、计划、执行和评价。这个过程类似于常见的计划、实施、检查和行动（Plan-Do-Check-Act）循环（Deming, 1986）。

基于事实的客观信息需要的认识，为一个项目或一个组织确立了一个度量过程。在度量策划期间，决策制定者和度量用户的特有的信息需要，驱动着选择和定义合适的度量。将一个项目特有的信息模型实例化，其度量方法的结果标识了要实施的基本度量、派生度量和指示器以及要应用的分析技术，以便处理项目已确定优先级的信息需要。

当实现或执行度量计划时，要收集和分析所要求的度量数据。把执行度量活动所产生的信息产品提供给决策制定者。从这些度量用户处反馈回来的信息，有助于评价度量和度量过程的有效性，以便能够在一个持续的基础上得到改进。

本书介绍的基本概念是从大量的度量经验和以前的研究发展而来的。它们在*Practical Software Measurement* (McGarry et al., 1997) 的已发布的序列版本中引入, 并且在ISO/IEC标准15939“软件度量过程(2001)”中进行了形式化。来自ISO/IEC 15939的度量过程模型和度量术语也已被软件工程研究所能能力成熟度模型集成(CMMI)项目(CMMI开发组, 2000)所采用, 作为一个新的度量和分析过程域的基础。本书基于ISO/IEC 15939和CMMI的已证实的度量概念, 解释了如何用实用的、可理解的方法来实现一个切实可行的度量过程。

简而言之, 要实现一个软件密集型项目的一个客观的按事实度量的过程, 包括通过开发一个项目特有的信息模型来定义项目决策制定者的信息需要, 并区分优先级, 然后裁剪和执行一个项目特有的度量过程活动集。实现这个过程的PSM方法, 综合了以前的经验和对许多应用领域的众多原始资料的研究结果。

如何使用本书

本书为实现一个软件度量过程提供了渐进的指导。它全面描述了度量信息模型和度量过程模型, 同时为在一个实际项目环境中应用这些模型提供了基于经验的指导。还不曾有哪本书能够收集了所有有关软件度量的相应信息和实用示例。同样地, 在实用软件度量Web站点www.psmc.com上, 包含了附加的信息、指导、示例和工具来补充本书的内容。

为了提高易读性, 作者大部分采用在正文中引用参考资料的形式对有关专题给出进一步阅读的建议。附加的参考资料在本书的“参考文献”部分中列出。

在本书中讨论了以下主题:

第1章 度量: 关键概念和实践。第1章综述性地介绍了软件度量, 解释了度量如何支持今天的面向信息的商业模型以及度量如何成为公司的资源。它描述了项目级和组织级度量之间的关系, 介绍了PSM的两个主要概念: 度量信息模型和度量过程模型。

第2章 度量信息模型。第2章深入讨论了度量信息模型及其度量构件。它把度量信息模型与度量计划和实现活动关联起来。

第3章 计划度量。第3章是详细讨论每个度量过程活动的五章中的头一章。第3章集中讨论计划度量活动, 并描述了需要什么信息来定义信息驱动的、项目特有的度量计划。

第4章 执行度量。第4章研究执行度量活动, 并讨论了如何收集和分析度量数据。本章介绍了几个与度量分析有关的概念。包括分析类型以及如何按因果关系将信息需要与相关的问题关联起来。

第5章 分析技术。第5章对估计、可行性分析和性能分析三种主要的度量分析类型进行了深入探讨。

第6章 评价度量。第6章描述了评价度量活动。它集中于已应用项目度量的评估、评价和改进以及已实现的项目度量过程。

第7章 建立和维持承诺。第7章解释了建立和维持承诺这一最后的度量活动, 说明了与实现一个切实可行的度量过程有关的组织级需求。第7章还说明了度量的“经验教训”。

第8章 成功的度量。第8章评审了本书中提出的一些主要概念, 并标识关键的度量成功因素。

附录A 度量构造示例。附录A提供了详细的已应用于软件密集型项目的典型的度量构造示例。

VIII

附录B 信息系统案例研究。附录B提供了一个综合的案例研究，说明了一个典型信息系统的度量过程的实现。

附录C Synergy集成复印机案例研究。附录C是第二个案例研究，描述了如何将度量应用于一个较大型的软件密集型升级项目。

目 录

译者序

序

前言

第1章 度量：关键概念和实践.....1

- 1.1 度量的动机2
- 1.2 度量作为组织的鉴别器3
- 1.3 基础：项目度量4
- 1.4 使度量有效的因素5
- 1.5 度量信息模型7
- 1.6 度量过程模型8

第2章 度量信息模型.....11

- 2.1 信息需要13
- 2.2 度量构造14
 - 2.2.1 属性（可度量的）.....17
 - 2.2.2 基本度量17
 - 2.2.3 派生度量19
 - 2.2.4 指示器19
 - 2.2.5 其它术语20
- 2.3 度量构造示例21
 - 2.3.1 生产率示例22
 - 2.3.2 质量示例22
 - 2.3.3 编码进展示例23

第3章 计划度量25

- 3.1 标识信息需要并设定优先级26
 - 3.1.1 信息需要的标识27
 - 3.1.2 将信息需要映射到信息分类28
 - 3.1.3 设定信息需要的优先级31
- 3.2 选择和指定度量32
 - 3.2.1 刻画项目语境33
 - 3.2.2 定义可度量概念34
 - 3.2.3 选择可应用的度量36

- 3.2.4 指定度量构造37

3.3 将度量方法集成到项目过程中40

- 3.3.1 标识度量时机40
- 3.3.2 制定度量规程43
- 3.3.3 文档化项目度量计划46

第4章 执行度量49

- 4.1 收集和处理数据50
- 4.2 分析数据55
 - 4.2.1 指示器生成56
 - 4.2.2 指示器表示60
 - 4.2.3 集成化分析模型62
- 4.3 提出建议68

第5章 分析技术71

- 5.1 估计71
 - 5.1.1 使用集成化分析模型来估计72
 - 5.1.2 估计量73
 - 5.1.3 估计过程概述74
 - 5.1.4 软件估计方法概述75
 - 5.1.5 选择估计方法78
 - 5.1.6 四种方法的相似点和不同点79
 - 5.1.7 在整个项目生命周期中使用不同的方法80
 - 5.1.8 映射与校准80
 - 5.1.9 计算估计82
 - 5.1.10 规模估计82
 - 5.1.11 工作量估计83
 - 5.1.12 进度估计84
 - 5.1.13 度量估计84
 - 5.1.14 评价估计85
- 5.2 可行性分析86
 - 5.2.1 可行性指示器88

5.2.2 可行性分析过程概述	90	6.2.9 评价建立和维持承诺活动	114
5.2.3 评价项目计划	91	6.2.10 度量过程成熟度	115
5.2.4 评估计划风险的影响	93	6.3 更新经验库	117
5.2.5 标识缓解的替代方案	93	6.4 标识和实现改进	118
5.3 性能分析	93	第7章 建立和维持承诺	119
5.3.1 为性能分析使用集成化分析模型	94	7.1 获得组织的承诺	119
5.3.2 性能指示器	95	7.2 定义度量职责	121
5.3.3 性能分析过程	98	7.3 提供资源	122
5.3.4 计划与实际比较	98	7.3.1 度量培训	123
5.3.5 评估影响	101	7.3.2 度量工具	123
5.3.6 预测结果	102	7.4 评审度量程序	124
5.3.7 评价替代方案	102	7.5 获得经验	125
第6章 评价度量	105	7.5.1 开始要小	125
6.1 评价度量	106	7.5.2 提供足够的培训	125
6.1.1 质量产品的使用	107	7.5.3 展示承诺	126
6.1.2 质量结果的可信度	107	7.5.4 最小化成本	126
6.1.3 质量对目的的适应性	107	7.5.5 采用面向行动的方法	127
6.1.4 度量结果的可理解性	108	7.5.6 交流	128
6.1.5 指示器模型假设的满足程度	108	第8章 成功的度量	129
6.1.6 度量的准确性	108	8.1 集成化规范	129
6.1.7 度量的可靠性	109	8.2 支持文化	130
6.2 评价度量过程	110	8.3 成功的指示器	131
6.2.1 性能	110	附录A 度量构造示例	133
6.2.2 及时性	111	附录B 信息系统案例研究	167
6.2.3 度量过程的效率	111	附录C Synergy集成复印机案例研究	199
6.2.4 缺陷抑制	112	术语表	215
6.2.5 客户满意度	112	参考文献	221
6.2.6 度量过程的依从性	112	索引	225
6.2.7 审计计划度量活动	113		
6.2.8 审计执行度量活动	114		

第 1 章

度量：关键概念和实践

所有成功的软件组织都将度量作为自己每日必做的管理和技术活动的一部分。度量为他们作出明智的决策提供了客观的信息，这些决策确实对他们的商业运作和工程性能产生了积极影响。在成功的软件组织中，度量得来的信息被视为一项重要的资源，并且使决策制定者可从各级管理处获取这些有用信息。

软件度量已经发展成为一门至关重要的软件工程学科。过去，许多软件组织都将度量视为额外的、非增值的任务，或者仅仅是“可有可无的工作”。现在，度量已经成为一项基本的软件工程实践，最明显的就是，它已经包括在软件工程研究所（SEI）的能力成熟度模型集成（CMMI）的2级成熟度需求及相关的商业软件过程标准中。

软件组织中实际实现和使用度量的方法决定其在商业运作和工程性能上的价值。当用度量来支持组织的商业和技术目标并与现存的定义软件项目的技术和管理活动集成在一起时，度量是最高效的。当度量提供的是可能影响项目既定目标的风险和问题的相关客观信息时，度量将发挥最佳效果。换句话说，当度量成为项目管理中重要的不可分割的一部分时，度量将发挥最佳效果。

1[⊖]

高性能的组织通过设计自己的技术和管理过程来使用客观的度量数据。度量数据和相关的分析结果既支持短期的也支持长期的决策制定。一个成熟的软件开发组织通常使用度量来帮助计划和评价一个已建议的软件项目，对照计划的目标客观地跟踪实际的性能，指导软件过程改进决策和投资，以及针对市场驱动的需求来帮助评估整个商业和技术性能。一个高性能的组织在软件项目的整个生命周期中，从初始阶段直至退役，都会用到度量。度量是

⊖ 边栏页码为英文原书页码，与索引中的页码相呼应。

作为一门主动性的学科实现的，度量得来的信息被视为是一项战略资源。

1.1 度量的动机

为什么要对软件进行度量？首先，软件已经成为公司的投资和商业战略中的一项主要因素，即使在“非软件密集型”组织中也是如此。在不断加剧的竞争环境中，度量已经成为组织能否跟上快速变化的信息技术发展的关键构件。从大公司在开发和维护关键信息资产的投资中不难看出，对软件密集型项目进行客观评估和管理的要求正在不断增长。

度量在项目级上最为重要。软件度量可以帮助项目经理更好地工作。它帮助定义和执行更加实际的计划，正确地分配宝贵的资源以使那些计划能够到位，并按计划准确地监控进展和性能。软件度量为作出关键项目决策和采取适当行动提供了所需要的信息。度量帮助将源自其它项目的信息和技术管理规范联系和集成起来。实际上，它使软件项目经理利用客观的信息来作出决策。

很显然，软件度量提供了客观的信息，帮助项目经理完成下面的工作：

- 有效地交流：度量为整个软件组织提供了客观的信息。这降低了经常困扰复杂和受限的软件项目的不确定性。度量帮助管理者在组织的各个层次上标识、设定优先级、跟踪和交流目标以及相关的问题。它对于供应商和采购商组织之间的交流也是同样重要的。
- 跟踪特定的项目目标：度量可以准确地描述软件项目过程和产品的状态。它对于在项目生命周期中客观地描绘项目活动的进展以及相关软件产品的质量非常关键。度量帮助回答了至关重要的问题，如“项目是按进度进行的吗？”以及“软件是否做好了交付给用户的准备？”
- 在早期标识并纠正问题：度量促进了主动性的管理战略。潜在的问题作为风险客观地标识出来，受到评估和管理。对已存在的问题可以更好地进行评价和设定优先级。度量促进在早期发现并纠正技术和管理上的问题，这些问题如果留待以后解决将会更加困难并且代价昂贵。管理者使用度量这项资源可以预测问题并避免被迫陷入一种消极的、修复失败的

方法上。

- 作出关键的权衡决策：每个软件项目都有约束。费用、进度、能力、技术质量和性能必须相互权衡并进行管理，以满足项目的既定目标。一个领域内的决策通常总会影响到其它的领域，即使它们看起来毫不相关。度量帮助决策制定者客观地评估这些影响，进行适当的权衡，以达到项目的最佳目标以及优化软件项目和产品的性能。
- 调整决策：当前的软件和信息技术商业环境要求有成功的项目性能。业务经理、技术经理和项目经理必须能够使用历史性能数据作为自己估计和计划的基础。然后，还必须能够根据当前的性能数据调整计划。度量

3

正像所有的管理或技术工具都不能保证一个项目将会成功，度量也一样不能保证。然而，它确实可以帮助决策制定者采取一种积极的方法来处理软件项目中固有的关键问题。度量帮助项目、继而帮助组织获得成功。

1.2 度量作为组织的鉴别器

当前的软件商业环境的特征可以由激烈竞争的市场中迅速改变的技术来刻画。无论是在打包软件或是单一应用软件的市场中，客户都要求以更低的费用来获得更多的功能并迅速地实现新的能力，以满足不断变化的商业需求。在当前环境中，要想使一个软件组织成为信息技术市场的领头人越来越困难，而更困难的是维持一个高性能组织的地位，使它无论在技术上还是在商业前景上都优于自己的竞争对手。

经验表明，几乎所有的高性能组织都可以这样描述：

- 所有的决策制定者都可以获取准确的、客观的信息，而且信息的使用已经成为企业文化的一部分。
- 过去、现在和将来的商业与技术前景受到重视，用以帮助定义项目期望的目标和性能。

- 设计了组织级的过程、规程，用以标识、刻画和管理变更。变更处理已经成为组织商业运作的一部分。
- 无论是好消息还是坏消息均可在组织中自由交流。问题的标识和解决是公开的。
- 为基于信息的决策和采取的行动建立了一种文化氛围。

这些特征都是信息相关的，因而也是度量相关的。作为领域内的高性能组织，首先需要定期获取正确的信息，作出正确的决策，使用信息使自己更为高效并生产出更高质量的产品。度量促进和推广了组织的经验，并支持企业在市场中的适应力。度量提供了从每一个项目中学习的结构，无论是成功的经验还是失败的经验。度量同时还帮助组织理解自身性能与市场所需性能之间的差距，使得组织可以在自身商业和技术约束下优化自己的性能。实际上，度量信息成为一项竞争资源，而且一个有效的度量过程也成为组织的鉴别器。

1.3 基础：项目度量

我们已经给出了度量与软件组织性能间的密切关系，其关键的挑战是实施度量的方式，能否使其给组织内的每个项目带来正面的影响。任何组织级的软件度量程序的基础都是建立在项目级。实质上，一个组织的性能是基于其项目的业绩。要想取得成功，项目经理和技术经理必须能够持续不断地就项目资源、进度和功能能力作出关键决策。这些决策必须在组织和市场驱动的约束下带来软件产品的业务和技术性能上的优化，使项目几乎没有返工或重头再来的可能。

软件项目经理必须将软件成功的各个方面结合在一起。决策的制定即使不是每小时一次，至少也是每天都有的。这些决策包括如何开发和管理技术产品，如何分配资源，哪些问题需要以什么优先级解决。可用的数据和信息越客观，决策就越有效；决策越有效，项目就越容易成功。项目经理通常必须在已有的约束下指引着项目朝着计划的目标发展。

5

项目环境通常是要考虑的因素之一，其它因素包括费用、进度和技术等。经验表明，即使是在相当小的信息技术组织中，每一个项目在其领域应用、技术实现过程、与其它系统的相互依赖性以及各种各样的约束方面都是独一无二的。再加上经常和不断的技术和管理变更，每一个项目都显示出自己独特的过程和产品特征。

组织级度量的实现方法应该是可以调整的，以便于有效地满足每一个项目的不同信息需要和特性。本书关注于项目级的度量以及如何裁剪度量来满足每个项目的需要。

在组织内，虽然项目级是实现度量的主要级别，但在较高的组织级管理层也确实有信息需要。几乎在所有的情况中，客观的组织级信息都来源于项目。需要有合适的项目数据，以便对整个组织进行分析。组织级软件度量可以视为跨越多个项目的度量，将每一个项目生成的度量数据有效地整合，并使用不同的分析技术来满足不同的信息需要。例如，组织中不同层次的管理者可能都关心开发和投产一个指定的软件产品要花多长时间。项目经理关心的是实现软件功能和可靠性所需要的时间以及如何在费用目标范围内开发软件产品。而市场或业务经理则可能关心的是一项新的功能投入市场的时间以及可能的延期对市场份额的潜在影响。过程经理可能会关心组织的所有产品的平均开发时间在整体上是增加了还是减少了以及组织级过程变更带来的相关影响。无论在什么情况下，关键的可变因素在于项目级质量数据的可用性，而且，实际上大多数企业的度量活动都发生在项目级。

1.4 使度量有效的因素

6

目前已有许多不同的方法用于实现软件密集型项目的软件度量，但并非所有的度量方法都是有效的，其中的大多数就没有经受住项目的考验。一些方法基于详细定义可应用于组织中所有项目的“最佳”度量集。其它方法依赖于从供应商处购买的自动化度量和分析工具，几乎与已有的组织过程或商业实践无关。但这些方法都只是将度量作为“额外的工作”，而没有在本质上帮助项目达到目标。如何使度量在一个已有的充满挑战的项目环境中发挥作

用？大量软件开发和维护的项目经验表明一个成功的度量程序应具备两个关键特性：

- 度量数据的收集、分析和报告，直接关系到项目决策者的**信息需要**。这可以刻画为信息驱动的度量方法，也就是通过已确立的项目目标和相关问题，度量的定义和实现是为了在设定优先级的基础上解决项目特殊的信息需要。随着项目的进展和信息需要的变化，应用的度量方法也应随之改变。信息驱动的度量方法从本质上来说必须理清三方面的关系：需要的是什么样的信息，实际度量的是什么，以及度量如何定义并组合成可用的结果。
- 结构化的和可重复的**度量过程**，定义了项目的度量活动和相关的信息接口。此过程必须是灵活的和可调整的，以支持现有的软件技术、管理过程和环境，同时支持特定的应用领域特性。度量过程一定是迭代的，持续关注最关键问题的度量成果。度量过程必须贯穿于整个项目的生命周期。它支持对进化的过程和产品属性的度量，例如项目信息需要、相关的目标和问题及其变更。

信息驱动的度量方法和度量过程结合在一起，持续应用于项目的生命周期中，根据项目经理需要的变化，为项目经理提供不同的信息。两者都需要经过裁剪来满足每个项目的专门特性。同时，它们为度量任务和规程提供了基础，支持成功的项目实施、改进的商业和工程性能。

要想获得有效的度量，必须在项目或组织内将度量作为软件工程支持过程来实现。若是这样，度量就必须包括整个项目或组织级结构内策划、执行、评价度量相关任务的所有关联活动。度量并非是孤立的，它是在项目环境内实施的，定义了决策制定者所需要的信息以及这些信息是如何收集、分析、提交和使用的。度量过程将不同的主观和客观数据整合成集成的信息产品，直接用于满足已定义的项目信息需要。

本书对如何为软件相关项目定义和实施可行的信息驱动的度量过程提供了基于经验的指导。PSM使用**度量信息模型**解决了项目度量信息结构的开发问题，同时还使用**度量过程模型**描述了度量活动和任务。