

文章编号:1001-9081(2008)01-0229-03

面向 CMMI 的软件度量分析过程研究与应用

张旭¹, 郭树行², 兰雨晴²

(1. 上海中标软件有限公司 质量管理部, 北京 100080; 2. 北京航空航天大学 软件工程研究所, 北京 100083)
(miaozhang2000@hotmail.com; guoshuhang@buaa.edu.cn; lanyuqing@buaa.edu.cn)

摘要:为了辅助软件组织有效实施符合 CMMI 模型的软件度量过程, 提出了一种 GREDR 过程框架模型, 以指导软件度量的目标策划与过程实施。在实际软件项目过程中的应用结果表明: GREDR 模型能够提高软件度量过程的系统性与效用。

关键词:软件度量; 能力成熟度模型集成; 度量计划; 度量实施; 度量元
中图分类号: TP311.5 **文献标志码:** A

Research and application of software measurement process for CMMI

ZHANG Xu¹, GUO Shu-hang², LAN Yu-qing²

(1. Department of Management, China Standard Software Co., Ltd., Beijing 100080, China;
2. Institute of Software Engineering, Beihang University, Beijing 100083, China)

Abstract: This paper introduced a Goal-Regulations-Data-Result (GREDR) process framework of software measurement for Capability Maturity Model Integration (CMMI) in order to help the organization implement the software measurement that conforms to the CMMI model. From the result of the trial in our organization, the software measurement process can be accomplished through the framework systematically. At the same time, it can assist on the plan for measurement and improve the final effect.

Key words: software measurement; Capability Maturity Model Integration (CMMI); measure plan; measure implementation; measure

度量是软件跟踪和监控的基础。不合适的度量方法不能满足项目跟踪监控的需求, 还会造成资源、人力及其成本浪费。本文面向 CMMI 度量分析过程域, 提出一种指导软件度量的 GREDR 过程框架, 辅助软件组织实现从度量目标到度量结果的转换, 有目的、有计划、有系统地完成度量的实施, 达到预期的目的。

1 CMMI 与软件度量

1.1 CMMI 的度量与分析过程域

相对于 CMM 模型, CMMI^{[1]19} 模型中更加强调了量化管理。管理的透明度和软件开发的透明度得到了升级。度量和分析已经成为 CMMI2 级中的一个单独的过程域, 而不是包括

在其他过程域中。

CMMI2 级中的度量与分析过程域 (Measurement and Analysis, MA) 共有 2 个具体目标和 8 个具体实践, 全面概括了度量的策划、实施、存储和分析报告的各项要求, 为组织的软件度量过程提供了一个准确清晰的指导思路。具体如图 1 所示^{[1]120}。

MA 过程域除作为一个单独的过程域在 CMMI 模型中占有特殊作用外, 还与高成熟等级 (CMMI4/CMMI5) 的 4 个过程域 (OPP/QPM/CAR/OID) 都有着直接的关系。可见, 高成熟度等级的实现是以度量与分析活动作为基础和支持的。另外, 作为支持过程域, 在客观评价过程的执行情况, 工作产品和服务与过程描述、标准和规程的符合情况上对所有的过程

表 1 四种软件度量过程模型和标准的比较^[2,6]

模型标准	主要内容	优点	缺点
G-Q-M 模型	从度量的目标和问题驱动软件度量	符合从上到下、逐步细化的思维习惯	主要集中于度量目标和度量计划的开发, 而没有给出一个完整的度量过程定义
PSM 模型	计划度量、执行度量、评价度量、建立和维持承诺	比较完整的度量过程模型, 它既实现了 ISO/IEC 15939 国际标准, 又与 CMMI 度量分析过程域兼容	缺乏信息需求与度量之间的映射关系以及开发度量计划的具体指导方法
统计过程控制模型	度量定义、过程度量执行、过程度量分析和过程评价	成功的引入了统计过程控制原理和方法	对于过程活动内部没有进行具体的定义
ISO/IEC 15939	度量活动、需要的信息、度量分析结果的应用及确定分析结果是否有效的方法	软件开发和系统工程定义了度量过程模型、度量信息模型和相关的术语	没有软件度量过程具体实施方面没有给出详细的指导

收稿日期: 2007-08-10; 修回日期: 2007-10-30。

基金项目: 国家 863 计划项目 (2006AA01Z186); 上海科技攻关计划项目 (06HX11808); 上海市徐汇科委资助科技创新项目 (200708007)。

作者简介: 张旭 (1974-), 女, 北京人, 工程师, 硕士, 主要研究方向: 软件度量、过程改进; 郭树行 (1978-), 男, 河北人, 博士研究生, 主要研究方向: 需求工程过程; 兰雨晴 (1969-), 男, 内蒙古人, 副教授, 主要研究方向: 软件质量保证、软件测试、软件系统分析、软件项目管理。

域都提供了支持^{[1]127}。

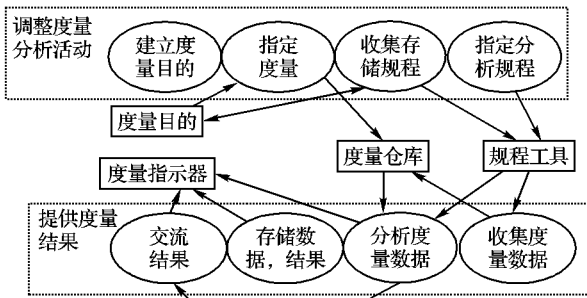


图1 度量和分析语境图

1.2 四种软件度量过程模型和标准

目前比较成熟的软件度量模型有:目标/问题/度量模型(G-Q-M 模型),实用软件度量模型(PSM 模型),统计过程控制模型,ISO/IEC 15939 等。

通过表 1 的比较可知,目前的几种主流软件度量过程模型,在软件度量过程的具体策划和实施方面都没有给出完善、全过程的指导说明。结合 CMMI 模型的要求,以指导过程具体实施为目标,本文提出面向 CMMI 的 GREDR 软件度量过程框架,为中小企业在进行软件度量过程的策划和实施提供具体详细的参考。

2 GREDR 度量过程模型框架

2.1 GREDR 框架概述

GREDR 模型主要涉及软件度量过程中的四个关键步骤:目标的设立,规程的制定,数据收集、分析、存储,结果的分析利用。其中每个关键步骤都包含多个子步骤,它们之间相互依赖,彼此支持,构成了整个软件度量过程的框架(如图 2)。

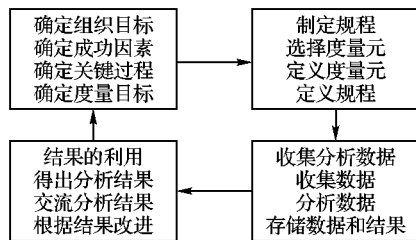


图2 GREDR 框架

2.2 GREDR 模型定义

2.2.1 确定度量目标

1) 确定企业的商业目标、部门目标或项目目标

CMMI 的 MA 过程中明确要求度量的行为要与所需要的目标相一致。所进行的度量,最终都要为实现组织的商业目标、部门目标或项目目标服务。所以,在进行度量定义和实施前,首先要明确组织或项目的目标,并对其含义和要求达成一致的理解,以便将它们作为确定度量目标的基础。例如,组织目标是“打造优质产品”,那么,就需要对“优质产品”的内容进行详细的定义。通过编写《组织/项目目标详解》,保证所有人员对目标有清楚一致的认识。

组织目标或项目目标往往比较概要和宽泛,还需要进行进一步分析和判断才能准确地推导出低层次的度量目标。确定关键成功因素和关键过程是不可缺少的两个步骤。

2) 确定关键成功因素和关键过程

关键成功因素是决定项目或组织达到既定目标的因素。获知组织的关键成功因素,是确定哪些过程活动将给组织带来最大利益的基础。影响组织商业目标和项目目标的成功因素可能会有很多,确定成功因素可用 H-W 原则找出最重要的

3~5 个。通过 How(H)来保证关键成功因素来源于组织或项目目标,通过 W(what)来演化、推导出适宜的关键成功要素。例如:针对于“打造优质产品”的组织目标,我们可以提出这样的问题,“H:怎样可以打造优质产品?”“W:影响产品质量的因素是什么?客户需要什么?”通过对问题的回答,确定关键成功因素为“提高需求分析能力”,“减少产品中的缺陷”。

通过一个关键成功因素可以直接找到与之相关的一个或多个关键过程。如与产品缺陷相关的过程有缺陷发现过程,缺陷修复过程等。关键过程不是一成不变的,可根据情况进行调整。

3) 确定度量目标

确定关键过程后,接着要确定关键过程目标。然后,通过对过程目标提问的形式,来分析哪些内容是我们最需要了解和掌握的。通过对问题的回答,就可以清楚地得出所需要度量的目标。如图 3 所示。

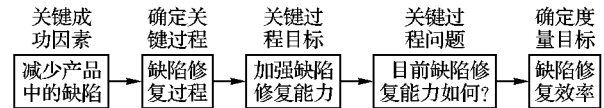


图3 度量目标确定过程

2.2.2 制定规程

规程是指导度量具体实施的标准和依据,涉及人员的职责,资源的使用和度量活动的周期等关键信息。在定义度量规程之前,首先要明确需要度量的对象,即度量元。

1) 度量元设计

度量元可分为基础度量元和衍生度量元。基础度量元的数据来自直接的度量,如代码行、工作量。衍生度量元通过两个以上的基础度量元数据计算得出,如生产率是由代码行和工作量数据计算得出。选择度量元是指选择过程或产品在某些方面的特征化的度量元。这一点可通过 GQM(Goal/Question/Measure)方法开展。GQM 的基本理念是从度量的目标和问题驱动软件的度量,首先定义需要度量的目标,然后通过回答问题的形式来衡量这些目标是否被实现,将模糊的、抽象的目标分解成具体、可测量的问题,通过问题最后确定度量的对象,即度量元。如:

- 度量目标(G): 缺陷修复效率。
- 问题(Q): 平均每修复一个 BUG 需要多长时间。
- 度量元(M): 缺陷修复效率 = 代码返工时间/被修复 BUG 数。

评价所选择的度量元是否合适,可以从以下方面考虑^{[3]18}:度量与问题相关程度、度量元对预期结果的反映程度、数据收集难度,以及数据偏差对有关过程信息的反映程度。

2) 定义度量元

为了确保不同的人、不同时间都能对所使用的度量元有着一致的理解,需要定义组织或项目的《度量元操作定义表》,用来反映度量元中各数值的含义、数目、得到方式、收集人员、产生频率、使用工具等。定义度量元要遵循以下几种原则^{[3]18}:

- (1) 可传达性: 度量元的定义是否准确地表明了度量的对象;
- (2) 可重复性: 当重复执行这个度量时候,要得到相同的结果;
- (3) 可跟踪性: 数据源按照时间、活动、产品、报告机制、

状态等可以被清楚地识别,提供方便的度量工具。

3) 确定数据收集存储、分析规程

度量过程的一个关键是让度量成为过程中例行和完整的元素,因此需要将定义的度量融入软件生命周期恰当的各个阶段和过程中。组织可制定专门的《度量数据收集存储分析规程》,明确规定:

- 1) 收集、存储、分析和报告数据的相关负责人;
- 2) 何时、何处收集数据;
- 3) 数据收集频度、度量的时间点;
- 4) 数据分析和报告的方法;
- 5) 分析结果使用的对象;
- 6) 用于收集、存储、分析和报告分析数据的工具;
- 7) 数据存储的需求、数据的大小和格式、访问权限。

《度量数据收集存储分析规程》既是 G-Q-M 自底向上执行过程的具体体现,也为度量活动提供了统一的制度,保证度量的有序执行。

2.2.3 度量元数据采集

1) 收集数据

数据收集要体现简单、方便、明确原则。所使用的方法和渠道可以是多种多样的。最简单的方法是利用表格形式进行填写。但这种方法存在表格不同版本的一致性等问题,大量表格的统计、管理、保存等问题。工具的使用往往会解决上述问题,并使数据的收集和统计更加规范化、简单化,自动化。如果数据的收集工具能和其他管理工具结合使用,则会起到事半功倍的效果,如:软件缺陷管理工具 (BUGZILLA、ClearQuest) 或项目管理工具。

数据收集过程是否合理可以遵照以下准则进行判断^[4]:

- (1) 过程数据的收集过程应该尽可能地自动化。
- (2) 收集到的过程数据应该达到度量的目的。
- (3) 过程数据应该具有易交换性。软件过程数据一般由多种工具收集,并且在多种工具之间传递,这就要求数据的格式具有易交换性。
- (4) 过程数据应该具有开放性、易访问性、易检索性。
- (5) 过程数据应该具有及时性。
- (6) 过程数据应该具有一致性:同一度量值的定义在同类记录当中的一致。
- (7) 过程数据应具有语义完整性:包含的信息应尽可能丰富。例如,工作量的数据,它所对应的活动、涉及的人员、开始和结束时间等,而不单单是工作量本身的数据和对应阶段的数据。

2) 存储分析数据

CMMI 中要求组织和项目级都要建立度量数据库。度量数据库可以根据度量元操作定义建立,支持对数据的各种检索。度量数据库的建立为组织数据的统一存储、管理和利用提供了方便,并为数据的利用提供了唯一的出口。在进入组织度量库之前,所有数据必须经过严格验证,以确保其正确性和可靠性。

2.2.4 度量数据的可视化分析

1) 分析数据并交流结果

在度量数据分析时,应该紧密围绕目标进行,使用适当的分析方法。常用度量分析技术包括^[5]:散点图、趋势图、直方图、Pareto 图、条形图、控制图等。通过这些图表可以直观地反映过程的稳定性、变化趋势、影响过程性能的因素之间的关联等。表 2 显示某个项目前 10 周的缺陷修复效率。

项目经理希望知道,第 10 周的缺陷修复效率是否低于正常修复水平,是否有异常问题出现。此时,可以使用控制图分析过程的稳定性(如图 4 所示)。

表 2 某个项目的缺陷修复效率

周次	每个缺陷修复所耗时间/h	周次	每个缺陷修复所耗时间/h
1	5.0	6	5.8
2	6.0	7	6.9
3	5.5	8	6.2
4	6.5	9	7.1
5	7.0	10	9.0

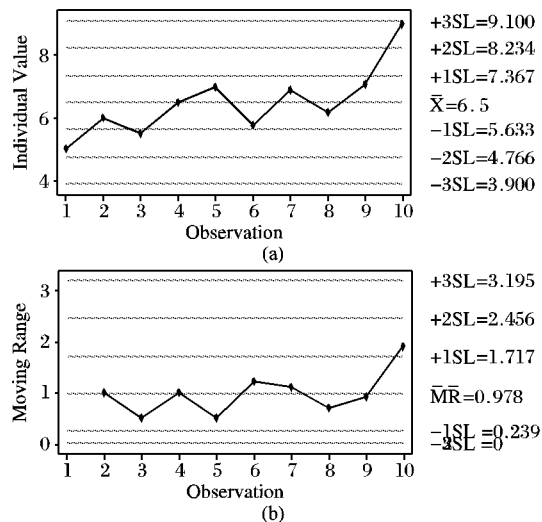


图 4 缺陷分析效率 XMR 图

通过分析发现,第 10 周的缺陷修复效率仍在控制范围之内,暂不需要采取措施对过程进行纠正。

在项目结束的时候,可以分析本项目缺陷修复过程的性能,之后将项目的过程性能反馈到组织度量库。通过分析组织中各个项目的过程性能来确定组织缺陷修复过程的能力。

2) 使用度量结果进行过程改进

如果过程是满足目标的,则需要考虑优化过程以实现更高的目标。如果过程还不能充分满足目标,可以利用 CMMI5 中的 CAR 过程域进行原因分析,找出主要的因素,并将其作为过程改进的需求纳入组织过程改进计划中,使组织有针对性地对过程实施改进,并对改进结果做出量化评价。

3 结语

软件过程管理或软件质量管理需求的不断衍化,为软件度量带来了挑战。基于 CMMI 体系,制定有效的软件度量过程模型是实施软件管理的重要基础。GREDR 模型框架能够满足软件组织在软件度量方面的系统化和具体化需求。并已经在公司实施 CMMI 过程中体现出了其作用,使公司具备了系统的度量体系,在公司的过程改进中发挥了积极的作用。

参考文献:

- [1] 周伯生. CMMI 精粹——集成化过程改进实用导论[M]. 北京: 中信出版社, 2002.
- [2] 任发科, 周伯生, 吴超英. 软件度量过程的研究与实施[J]. 北京航空航天大学学报, 2003, 29(10): 1-2.
- [3] 王志浩. 基于 CMMI 的 6Sigma 软件度量分析系统的研究与设计[D]. 湖南: 湖南大学软件学院, 2005.
- [4] 周宇, 钱红兵. 软件过程中的度量数据收集方法研究[J]. 计算机应用研究, 2005, 22(1): 1-2.
- [5] 孙晶. 基于 CMMI 的软件质量度量模型及工具原型[D]. 大连: 大连海事大学, 2006.
- [6] 钱云峰, 刘宽宏, 覃征. CMMI 度量分析过程域在中小企业的研究与实践[J]. 计算机工程与科学, 2006, 28(11): 1-2.