

实绩价值管理成熟度模型及其支持工具 SPIF

赵昊翔 屈庆明 潘金贵

(南京大学软件新技术国家重点实验室 南京大学计算机科学与技术系 南京 210093)

摘要 EVM 是一种能够帮助软件组织降低项目风险、提高开发效率、减少项目开支的行之有效的软件过程改进方法, EVM³ 是一个提高软件组织使用 EVM 管理项目能力的模型。本文介绍了 EVM³ 的基本概念、与 CMMI 的关系, 以及一个支持 EVM³ 进行软件过程改进的工具 SPIF。

关键词 实绩价值管理(EVM), 实绩价值管理成熟度模型(EVM³), 能力成熟度模型集成(CMMI), 软件过程改进(SPI), SPIF

Introduction and Tool for Earned Value Management Maturity Model

ZHAO Hao-Xiang QU Qing-Ming PAN Jin-Gui

(National Key Laboratory of New Software Technology, Computer Science of Nanjing University, Nanjing 210093)

Abstract EVM is a software process improvement method that can decrease risk, improve efficiency and reduce cost. EVM³ is a model that can improve the capability of using EVM. This paper introduces the basic concept of EVM³, its relationship with CMMI and a tool named SPIF based on EVM³ which can help software organizations to improve software processes.

Keywords Earned value management, Earned value management maturity model, Capability maturity model integration, Software process improvement, SPIF

1 引言

上个世纪中后期,随着软件规模的日益增大,软件业爆发了软件危机,大量的软件项目费用严重超支,进度不断拖延,很多项目因此而搁浅,造成了大量的人力、物力损失。人们多方寻求解决之道,其中实绩价值管理 EVM(Earned Value Management)就是其中之一。

EVM 是一种能将项目费用、计划以及状态三者有效结合起来的管理方法,能够在任意时刻根据项目的当前进度对未来走势(包括预期费用、完工日程等)做一个大致的预测,有助于软件组织降低项目风险、提高开发效率、减少项目开支,是管理项目进度的一个行之有效的方法。

EVM 由美国国防部下属的软件组织于上个世纪 70 年代率先使用,并取得了很好的效果,欧美各大软件企业随即纷纷将其纳入自己的软件过程改进范畴之中。随着 EVM 的大量使用,美国卡内基梅隆大学软件工程研究院 SEI(Software Engineering Institution)在 2000 年推出的个体软件过程 PSP(Personal Software Process)和群组软件过程 TSP(Team Software Process)中,将 EVM 的使用纳入了团队建设的关键要素之一^[10]。

2 EVM³ 概述

2000 年 9 月,项目管理协会 PMI(Project Management Institute)在年度会议上推出了实绩价值管理成熟度模型 EVM³(Earned Value Management Maturity Model),历经多次讨论和修订,如今最新的版本是 2.0。该模型为软件开发

组织使用 EVM 的能力提供了一个阶梯式的进化框架,它吸取了以往软件工程的经验教训,指明了一个成功的软件组织如何利用 EVM 以达到预期效果、在软件开发方面应该做哪些工作、这些工作的关系,以及该以怎样的次序按部就班地做好这些工作,使软件组织走向成功。

2.1 结构

EVM³ 将软件过程改进的进化步骤组成 5 个等级,除第一级外,每个等级都包含一组关键过程域,而每个关键过程域则包含一组过程域目标,通过实施相应的一组关键实践来达到这些目标。这些关键实践全部按 4 个共同特征(执行约定、执行能力、执行活动、验证实施)加以组织。这些共同特征是表明一个关键过程域的实施和规范化是否有效、可重复且持久的一些属性。EVM³ 结构如图 1 所示。

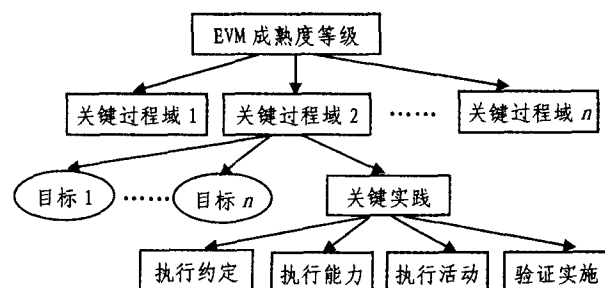


图 1 EVM³ 的结构

2.2 EVM³ 5 级模型

文[3]中将软件组织使用 EVM 的能力分为 5 个等级,分别是初始级、局部化级、ANSI 748-98 标准级、已管理级和优

赵昊翔 硕士研究生,研究方向为软件工程、软件过程改进等;屈庆明 硕士研究生,研究方向为软件过程改进;潘金贵 教授,博士生导师,从事中间件、Agent 技术及多媒体远程教学系统等研究。

硕士研究生,研究方向为软件过程改进;潘金贵 教授,博士生导师,从

化级(如图 2 所示)。

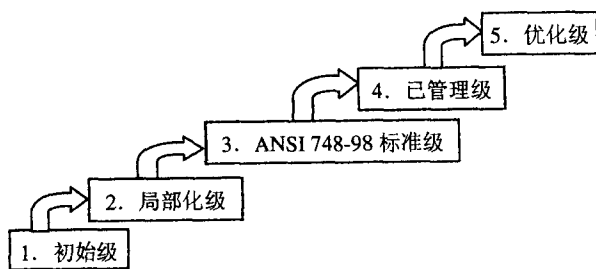


图 2 EVM³ 的 5 个 EVM 成熟度等级

每个等级的定义和特征如下。

(1) 初始级(等级 1)

在初始级上,软件组织没有使用(或者很少使用)EVM。就算是使用 EVM,也只是利用挣得值 EV(Earned Value)计算进度绩效指数 SPI(Schedule Performance Index)和费用绩效指数 CPI(Cost Performance Index)而已,缺少精确的 EV 计划和费用分析数据。在这个等级上,不存在工作分解结构 WBS(Work Breakdown Structure),得到的计划工作预算费用 BCWS(Budgeted Cost for Work Scheduled)和完成工作实际费用 ACWP(Actual Cost of Work Performed)的数据很不精确, EV 完全靠主观臆测。

(2) 局部化级(等级 2)

在局部化级上, EVM 得到了一定的科学使用,且部分满足 ANSI 748 标准的要求。软件组织符合 5 个最基本的标准,分别是:

- 用 WBS 定义和分解工作范围;
- 基于上面的工作范围制定计划;
- 对于每个工作单元估算费用;
- 在项目进行过程中,坚持按照项目基线从事开发活动;
- 掌握项目状态并且预测项目结果。

处于这个等级的软件组织已经意识到利用 WBS 分配工作的重要性并尝试实行,但是该等级缺少一个贯穿项目全程的项目基线和工作时间记录日志, WBS 制定得不是很完全,从而导致 ACWP 数据不够准确,而 EV 也更多是靠主观臆测。这个等级对于小型软件企业已经合适,但对于大中型软件企业来说,该等级还远远不够。

(3) ANSI 748-98 标准级(等级 3)

在该等级上,软件组织使用 EVM 的能力已经符合并且

超越了 ANSI 标准的要求。他们有充足财力和人力所支撑的正式的 EVM 管理程序,建立了实绩价值过程小组 EVPG(Earned Value Process Group)来制定、修正和监控 EVM 过程,以提高 EVM 系统的正确性和效率。同时拥有实绩价值管理系统 EVMS(Earned Value Management System)知识库,该知识库能够捕捉软件过程中的经验教训。软件组织在实行 EVM 时有一整套规范,该组织内的各个小组都是按照规范从事软件开发活动。

(4) 已管理级(等级 4)

在已管理级上,软件组织开始收集软件过程中的度量数据,这个活动独立于利用 EV 制定项目状态报告书活动。度量是检测实际性能与可接受性能之间偏差的基础,同时也是获得过程改进机会的基础。Ray W. Stratton 在文[2]中提出了在软件过程中可以考虑收集的 7 项度量数据,分别是:每个费用记帐员 CAM(Cost Account Manager)负责的工作单元数、软件过程中潜在数据、重新制定的计划和费用变更的数目、子活动变更引起的上溯活动调整数目、从订货到交货时间内计划内的工作单元数、计划内连续的工作单元比例和每一次重新更新计划的所得数据。

该等级的一个重要特征是软件组织已经从收集 EV 数据发展为收集从事软件过程中的过程数据。而且它对于 EV 的应用也提出了更高的要求, EV 数据同时也成为了项目与项目之间比较的依据。

(5) 优化级(等级 5)

优化级代表了使用 EVM 能力的最高等级,在这个等级上,软件开发计划制定得很科学详细,软件组织有明确的 EVM 改进目标,同时也采集了充分的度量数据以跟踪 EVM 的改进。

3 EVM 和 CMMI 的异同

3.1 CMMI 介绍

能力成熟度模型集成 CMMI(Capability Maturity Model Integration)是 SEI 于 2000 年在能力成熟度模型 CMM(Capability Maturity Model)的基础上提出的一个更加先进的软件过程改进模型。它有阶段表述和连续表述两种表述方式。其中,阶段表述采用成熟度等级模型(共 5 个等级,与 CMM 兼容),分别是初始级、已管理级、已定义级、已定量管理级和优化级。

表 1 CMMI 与 EVM³ 共同点

过程域 (Process Areas)	明确目标 SG (Specific Goal)	明确实践 SP (Specific Practice)
项目规划	SG1 建立估算	SP1.1 估算项目范围
		SP1.2 建立工作产物和任务品质估算
	SG2 制定项目计划	SP1.4 确定估算效果和费用
		SP2.1 建立预算和计划
项目监控	SG1 根据计划监控项目	SP2.4 为项目资源制定计划
		SP1.1 监控项目计划参数
	SG2 管理并实施措施改进	SP1.6 管理进度复核
		SP2.1 分析结果
		SP2.2 实施措施改进
		SP2.3 管理措施改进
供应商合约管理	SG2 使供应商满意	SP2.2 执行供应合约
集成化项目管理	SG1 使用项目已定义过程	SP1.3 将所有能够影响项目的计划集成入项目已定义过程
		SP1.4 用集成后的计划管理项目
		SP2.2 管理影响项目的关键因素
度量与分析	SG1 综合度量与分析活动	SP1.1 建立度量标准
		SP1.2 详化度量

3.2 EVM³ 和 CMMI 的共同点

CMMI 强调对项目的定量和集成管理,并将项目的费用、计划和状态三者纳入项目计划的制定范畴之内。它有 5 个过程域是和 EVM³ 紧密相连的^[5],它们分别是项目规划(Project Planning)、供应商合约管理(Supplier Agreement Management)、项目监控(Project Monitoring and Control)、度量与分析(Measurement and Analysis)和集成化项目管理(Integrated Project Management)。Paul Solomon 在文[5]中详细列举了 EVM³ 和 CMMI 的对应关系。综合来说,CMMI 在表 1 所示方面与 EVM³ 完全吻合。

3.3 EVM³ 对软件组织的意义

由于 CMMI 与 CMM 的一个最大区别是 CMMI 更强调定量和集成的项目管理,且提供了 3 个过程域(度量与分析、项目监控和集成化项目管理)予以支持,而这些过程域恰恰是与 EVM³ 兼容的,因此对于已经取得 CMM 等级认证并计划升级到 CMMI 的软件组织来说,可以使用 EVM³ 模型,以减少升级过程中的盲目性、降低风险、减少费用。

对于其他的软件组织,也可以考虑使用 EVM³ 作为软件过程改进模型。因为在 CMMI 一系列的准则中,EVM 能够将大多数准则串连起来并最终达到集成的目的,是基于 CMMI 进行软件过程改进的主线。采用 EVM³ 进行软件过程改进,能够取得事半功倍的效果。

4 SPIF 系统简介

SPIF(Software Process Improvement Framework)是南京大学研发的支持 CMM 和 EVM³ 模型,适合中小软件企业过程改进的一个基于 Web 的框架。SPIF 将软件组织划分为 4 个角色,组织结构如图 3 所示。

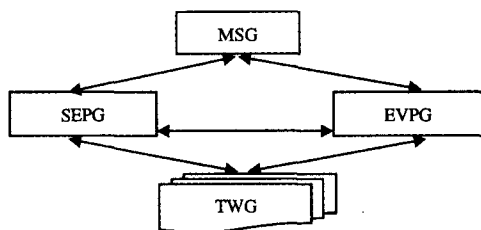


图 3 SPIF 组织结构图

各部分说明如下:

(1)管理操作组 MSG(Management Steering Group)。它是软件组织的最高管理者,指引软件项目实施流程和软件过程改进的方向。

(2)软件工程过程组 SEPG(Software Engineering Process Group)。它是软件过程改进的管理者,负责实施、监控软件过程改进。它由 MSG 任命,并定期向 MSG 汇报过程改进情况。

(3)实绩价值过程小组 EVPG(Earned Value Process Group)。它是 EVM 管理者,负责制定、修正和监控 EVM 过程的执行。它由 MSG 任命,并定期向 MSG 汇报 EVM 的执行情况,是 SEPG 进行过程监控和评估的基础。

(4)技术工作组 TWG(Technical Working Group)。它是过程中每个子活动的执行者,负责子活动的实施、监控。它由 SEPG 任命,并定期向 SEPG 和 EVPG 汇报工作结果,是 SEPG 和 EVPG 进行工作的基础。

SPIF 同时提供了一组工具和知识库,用来对软件开发的

各个过程进行管理,并提供必要的支持。其框架如图 4 所示。

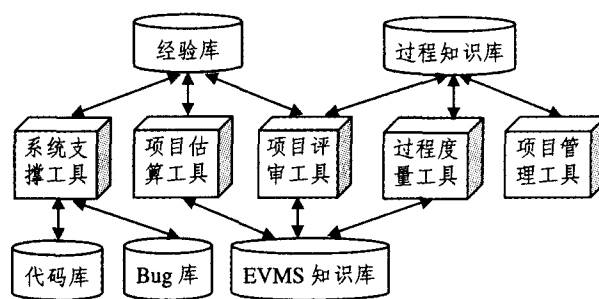


图 4 SPIF 体系结构

SPIF 提供了一系列的知识库,用来存放数据采集信息,各库的说明如下:

(1)代码库。也是组件库,保存项目开发过程当中产生的功能相对独立的某些模块,可以供本项目或其他相关项目复用。

(2)Bug 库。保存项目开发过程当中出现过的所有 Bug,以项目类别与 Bug 类别进行分类,使之在后续的开发过程当中可以用作参考。

(3)经验库。这是一个大的概念,泛指保存开发文档、技术文档的资料库,保存项目统计分析信息的知识库,保存描述项目开发过程经验的文档库等。

(4)过程知识库。这是专门为 TWG、SEPG 提供过程相关知识的库,保存组织开发模型、过程模型等信息。

(5)EVMS 知识库。EVMS 是实绩价值管理系统(Earned Value Management System)的缩写,是专门为 EVPG 提供的 EVM 相关知识的库,保存 EVM 过程中采集的数据及分析报告等。

在此基础之上,系统还提供了如下的一组实用工具:

(1)系统支撑工具。对这些知识库进行组织和维护。

(2)项目估算工具。提供一个计算模型,更新经验库中知识与现实情形,准确估计项目工作量等基本数据。它是一个自学习的智能化工具。

(3)项目评审工具。在每个项目结束之后,对项目的完成情况进行评审,给出评审报告,为后续项目提供参考。

(4)项目管理工具。提供对项目管理的支持,并且在管理过程当中采集大量的项目相关数据。一方面作为项目评审的基础,另一方面也是项目改进的基础。

(5)过程度量工具。收集软件过程中产生的度量数据,分析本次软件过程的得失,为下次改进打下基础。

SPIF 已在南大富士通软件公司正式使用,并帮助该公司于 2003 年底顺利通过了 CMM3 级认证,它的目标是在支持 CMM 的基础上逐步达到 EVM³ 的标准,并最终能够全面支持 CMMI。这是个循序渐进的过程,SPIF 也会在实践中不断地完善。

结束语 软件过程改进是一项复杂的工作,讲究循序渐进、稳扎稳打的原则,切不可一步登天。EVM³ 不但能够满足 CMMI 的大部分要求,而且实施起来简单高效、费用低廉(相对于 CMMI 而言),是软件组织进行过程改进一个不错的选择。SPIF 是一个 Web 化的软件过程改进支持工具,能够协助软件组织达到 CMM 和 EVM³ 的要求,并争取将来能够达到全面支持 CMMI 的目标。实践证明,使用 SPIF 的软件组织能够有效提高生产效率、改善软件过程。

参考文献

- 1 殷新春, 严芬, 汤克明, 等. 基于 CMM 的软件质量保障平台定义框架. 小型微型计算机系统, 2002, 23(11)
- 2 Stratton R W. The EVM Maturity Model-EVM³TM. 2002
- 3 Management Technologies. The Earned Value Management Maturity Model. 2002
- 4 赵昊翔, 潘金贵. 基于 EVMS 的软件过程改进. 计算机科学, 2004
- 5 Solomon P. Using CMMI to Improve Earned Value Management. CMU/SEI-2002-TN-016

- 6 Solomon P. Going From PBEV to CMMI. The Magazine of the Project Management Institute's College of Performance Management. March 2002
- 7 林锐, 王慧文, 董军. CMMI 3 级软件过程改进方法与规范. 北京: 电子工业出版社, 2003
- 8 McFeeley B. IDEAL: A User's Guide for Software Process Improvement. CMU/SEI-96-HB-001
- 9 万江平, 郭荷清, 夏志忠. IDEAL 模型的研究. 计算机应用研究, 2001
- 10 Humphrey W S. The Team Software ProcessSM(TSPSM); [CMU-SEI-2000-TR-023]

(上接第 130 页)

模型根据控件的方向属性和切换界面风格的参数, 采用不同的逻辑层坐标系, 选择不同的算法, 从设计层的设计时数据计算得到逻辑层的运行时数据, 再转换到显示层, 调用绘图函数绘制最终显示结果。控件对事件的响应, 则先将事件信息转换成逻辑层或设计层信息, 运算得出响应行为, 最终由显示层表现出来。

5.4 布局管理和控件风格

模型的核心是逻辑层的布局管理和控件风格的运算, 它们产生最终的用户界面。

现在的许多 GUI 图形库中都引入了布局管理的概念。在容器中, 布局管理器负责动态地调整容器中控件的位置和大小。

控件内部的界面元素的位置则不是由布局管理器定位的, 完全是依赖坐标系。布局管理器负责定位控件, 控件负责绘制自身。实现动态的界面风格就是采用不同的绘制算法绘制控件。

虽然布局管理器也能完成类似坐标变换的功能, 例如用右对齐的布局就可以模拟 RTL 风格。但是, 由于动态界面风格的本质是坐标系的变换问题, 而且布局管理器也无法管理控件内部的界面元素, 因此层次模型采用坐标系变换的原理是从本质上解决动态风格的实现问题。

另外, 基于坐标系变换带来的优点是层次模型也适用于没有布局管理机制的 GUI 图形系统。

5.5 实现

本文提出的层次模型已应用于扩充 Qt3.3 图形库, 实现支持少数民族语言的 GUI 系统。在层次模型的实现过程中, 我们构建新的设计结构, 改变了原有设计中结构和数据定义的含义, 在不同函数中体现层次模型的作用, 修改和添加新的算法实现动态界面的计算和显示。

图 4 展示了 Qt 库的例子程序 smalltable 在扩充后的 GUI 库的支持下, 动态切换为符合蒙文的 TTB-LTR 风格。实现动态界面的算法重新计算了 Table 中的 TableHeader、TableWidgetItem 等界面元素的位置和大小, 修改了交互行为, 以符合 TTB-LTR 界面风格的要求。

原始的 Qt 图形库没有完整地考虑国际化 UI 风格设计, 内部许多算法仅仅对 LTR 风格是有效的, 也缺乏层次概念, 实现镜像功能的代码更多地体现为实现的技巧。相比而言, 改进后的图形库保留了自适应的特点, 同时实现了模型中的多个层次, 形成一个完整的结构。

总结与展望 本文提出的层次模型结合坐标变换和自适应的布局管理, 动态实现不同的界面风格。建立在控件层的

方向属性使得程序可以呈现丰富的国际化特性。运行时改变界面风格的能力, 使得可以快速地发布应用程序的国际化版本, 这一点对于少数民族的信息化工作具有重大的意义。

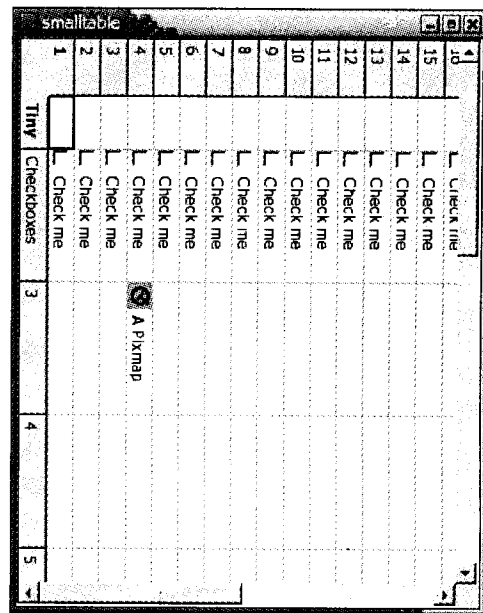


图 4 TTB-LTR 风格的 Table

现有的实现是从实用性的角度出发, 考虑兼容性, 在原来的 Qt 图形库的基础上扩充代码实现层次模型。于是, 不可避免地要部分受制于原有的框架, 代码的复杂程度较高, 这是目前模型实现的不足之处。进一步的研究和开发工作将是规范应用程序图形编程接口, 清晰化内部层次结构, 降低实现的复杂度, 推动图形库向有利于模型实现的方向进化。

参考文献

- 1 Dr. International. Developing International Software. Second Edition. Microsoft Press, 2003
- 2 Sun Microsystems, Inc. JavaTM 2 Platform, Standard Edition, v 1.4.2 API Specification
- 3 Charles P. Introduction to bidirectional languages; Bidi and AIX. IBM Corporate Globalization Team, Globalizing your e-business Newsletter - Issue X, June 2004. <http://www-306.ibm.com/software/globalization/topics/bidiaix/index.jsp>
- 4 Kurlow S. A new object-oriented technique for building a dynamic Graphical User Interface. In: Proc. of the 1995 Conf. of the Centre for Advanced Studies on Collaborative research, Toronto, Ontario, Canada, Nov. 1995
- 5 Landau J. Testing for Globalization. In: PSQT & PSTT East Conference on Washington, 2004
- 6 Stephanidis C. Toward an Information Society for All; An International R&D Agenda. International Journal of Human-Computer Interaction. 1998, 10(2): 107~134