

# 基于 EVMS 的软件过程改进

赵昊翔 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** 软件项目通常会遇到项目花费超出预算、项目完工日期拖延等问题而导致软件质量下降甚至项目失败,为了提高软件质量、减少项目风险,人们提出了一系列的项目管理方法,实绩价值管理系统 EVMS(Earned Value Management System)就是由美国国防部和国家标准学会建立的一个项目管理系统。本文介绍了 EVMS 的作用、历史、概念以及计算方法,最后结合软件过程改进的自底向上模型提出了软件企业应用 EVMS 管理项目的基本步骤以及组织关系。

**关键词** 挣得值(EV),实绩价值管理系统(EVMS),自底向上软件过程改进模型

## Using EVMS to Improve Software Process

ZHAO Hao-Xiang PAN Jin-Gui

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

**Abstract** Software projects are often faced with these situations such as cost exceeding budget, time beyond deadline, and so on. It always leads to the failure of the projects. To avoid them, a lot of project management approaches are provided and EVMS (Earned Value Management System) is one of them. This paper introduces EVMS's conception, history, benefits and calculation models. After all, it provides several steps to manage software project combined with Bottom-Up Software Process Improvement Model, and also sets up organizations for software enterprise supporting EVMS.

**Keywords** Earned Value (EV), Earned Value Management System (EVMS), Bottom-up software process improvement model

## 1 引言

高效率、高质量地开发软件一直是软件企业所追求的目标,但是软件企业所开发的众多项目中,真正能够成功的软件其实并不多,美国政府软件开发项目统计数据能够清楚地说明这一点(见表1)。

表1 软件开发应用比例

软件最终状态	所占比例
软件提交后未得到成功的应用	45%
软件提交后得到成功的应用	1.77%
经过修改后软件得到应用	2.95%
经过大量的改造后软件得到使用或是废弃	19.2%
软件开发未提交成果	28.8%

从表1中可以看出真正得到成功应用的软件所占比例不到5%。为什么软件项目的失败率如此之高呢?经研究发现,软件项目失败主要有三个原因:

- (1) 软件开发的时间超出客户所能容忍的最后期限;
- (2) 软件开发的费用超出预算;
- (3) 软件开发遇到某种困难(比如技术上无法克服的障碍,bug太多难以修正等)无法继续进行下去。

以上原因归根结底就是软件过程进行的不合理性所造成的,如果要提高软件质量,软件过程的改进是关键。软件过程

改进有其自身的特点:

- (1) 软件过程的过程性。表现为软件过程的建立、进行以及改善是一个延续的过程;
- (2) 软件过程的持续性。表现为该过程改进是一个循环持续的过程;
- (3) 软件过程的特殊性。表现为每个企业因存在着人员、资源、技术、规模等种种差异,而有不同的过程改进方法;
- (4) 软件过程的动态性。表现为对于每个企业来说,软件过程改进的方法是通过不断摸索、不断总结得到的,而且还会根据自身的发展不断地变化。

传统的软件企业通常是把项目实际费用和项目预期费用结合起来作为衡量软件过程的标准,但这是很不充分的,它有以下几个不足:

- (1) 软件开发者无法预测项目结果;
- (2) 软件开发者对于项目进度的认识还只是停留在一个感性的基础上,并没有上升到理性的高度,即该认识只是一个没有充足根据的臆测;
- (3) 软件过程中发生了新的事件(比如引入了新技术、遇到了某种困难等),其结果对过程产生的影响对于软件开发者来说还是一片茫然,根本无法反馈到软件过程中;
- (4) 软件开发者不能有效地评估软件过程。

针对以上情况,美国国防部和国家标准学会建立了实绩价值管理系统 EVMS(Earned Value Management System)。

赵昊翔 硕士研究生,研究方向为软件工程、软件过程改进等。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的

它在传统衡量软件过程标准的基础上,引入了挣得值 EV (Earned Value) 的概念。它综合了项目范围、进度计划和资源,能够测量项目绩效;它比较了计划工作量、挣得值与实际花费成本,以决定成本和进度绩效是否符合原定计划。EVMS 的使用能够减少项目风险,改进软件过程,提高软件质量。

## 2 EVMS 简介

### 2.1 沿革

100多年前工业工程师们首先用 EV 来确定生产的费用绩效,20世纪60年代初,美国空军把 EV 概念用于测定导弹的研发进展,制定了一个被称作“CSPPCS”的费用进度计划与控制规程,这个规程包含35条实施准则。1967年“CSPPCS”演变为“C/SCSC”费用进度控制系统准则,作为美国国防部的采购管理文件。经过1994年的改革,该系统划归新组建的整体项目管理机构(IPMI)来实施,大大提高了项目监控和管理的效率。1996年,美国国防部将 C/SCSC 准则修改为32条,建立了 EVMS。经过30多年上千个项目的验证,1998年美国国防部和国家标准学会将“实绩价值管理系统”颁布为标准,以 EVMS 命名<sup>[3]</sup>。

### 2.2 基本概念

EVMS 能够针对项目的基线,准确提供项目的当前进度信息,也能够预测项目的结束状态,对于超出预算、开发期限等情况提出实时的警告信息。项目经理可以根据这些信息了解项目的进展情况,判断计划安排的合理性,必要时可做出适当的调整,以保证项目能够顺利完成。

技术、费用和计划是影响项目风险的三个关键要素。EVMS 的主要目标就是建立一个将这三者结合起来管理项目的系统,并使客户信赖该系统周期性产生的数据。同时,这些数据还能够辅助项目经理进行决策分析,降低项目风险<sup>[4]</sup>。

简单来说 EVMS 有三点性质:

(1)EVMS 建立了一种对软件过程进行衡量的单位。这个单位可以是费用、时间或其他,但是一旦该过程分析决定采用了某一种单位,则在它的生存周期内,该单位是不能发生变化的,否则就无法有效地衡量软件过程。

(2)EVMS 是一个能对项目进行持续衡量的方法。它并不是只在某一特定时段、特定状态起作用,而是能够持续地在

项目进行的任意时刻实时汇报项目的进展情况,具体的衡量周期是根据项目经理的需要而指定。

(3)EVMS 提供了对项目费用分析的基础。如果在项目进行的过程中想实时了解项目开支的具体情况,以及想预测一下按照当前进度,该项目完成后所需要的费用,通过项目开支计划表和实际支出表,并结合 EV 所提供的一系列估算方法,就可以知道该项目结束时的状态,并能以此判断计划制订的得失,必要时做适当的调整。所有的这些决策都能减少项目进行的风险,提高决策者的决策水平和管理质量,并将损失降低到最小<sup>[6]</sup>。

### 2.3 计算方法

在使用 EVMS 之前,首先要建立项目状态基线,它包括范围、计划和费用三个要素(如图1所示)<sup>[1]</sup>。这三者的结合使对过程的跟踪和预测成为可能,而项目状态基线的建立也是对软件过程进行自底向上改进的基础。

#### 1. 范围



#### 2. 计划



#### 3. 费用

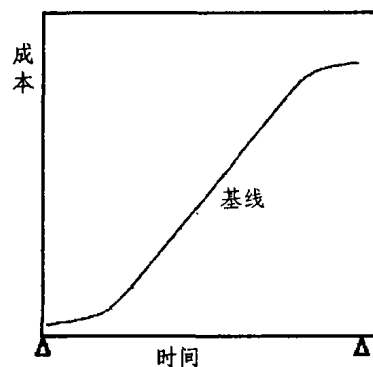


图1 状态基线图

工作的范围通常用工作分解结构 WBS (Work Breakdown Structure) 来表示。WBS 是根据树形图将一个功能实体(项目)先分解为子项目,再逐级分解成若干个相对独立的工作单元,并确定每个工作单元的任务及其从属的工作(或称之为活动);以便更有效地组织项目的进行。

当项目状态基线建立后, EV 的计算模型就可以用图2的 S 曲线来表示<sup>[2]</sup>。

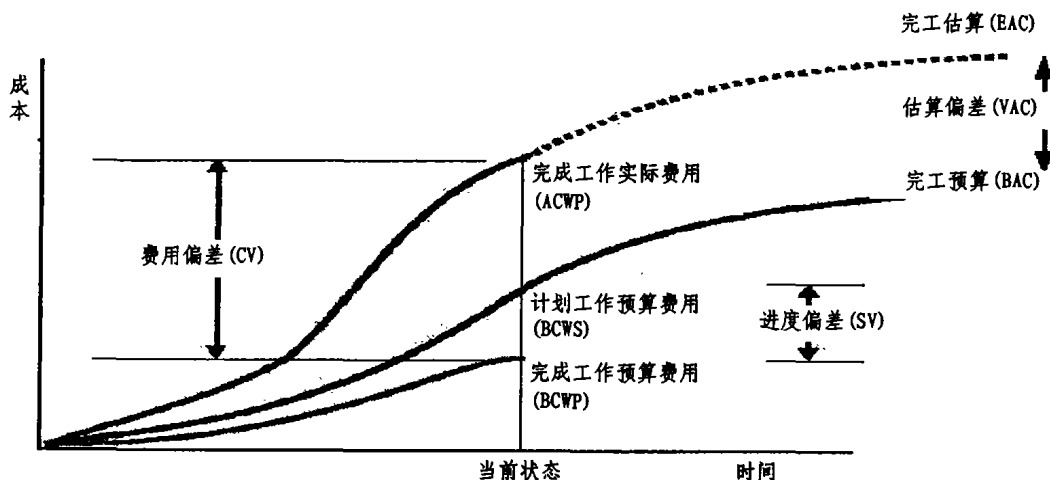


图2 S 曲线图

图2包含了3条曲线:

- 计划工作预算费用 BCWS (Budgeted Cost for Work

Scheduled):指根据批准认可的进度计划和预算到某一时点应当完成的工作所需投入的资金。俗称“计划投资额”或“计划

值”。这个值对衡量项目进度和费用都是一个标尺或基准。一般来说,BCWS 在项目实施过程中应保持不变,除非预算、计划或合同有变更,如果这些变更影响了工作的进度和费用,经过批准认可,BCWS 基准也应作相应的更改。

- 完成工作实际费用 ACWP (Actual Cost of Work Performed): 指到某一时点已完成的工作所实际花费或消耗的金额。俗称“消耗投资额”或“消耗值”。进度、费用、质量是工程项目管理的三个重要目标,以往是对这三个目标分别管理,相互之间缺乏紧密的联系,这带来了许多的问题。例如:项目进度因资金不到位而拖延;由于赶工或资金短缺引起潜在的质量问题;当成本结算发现严重超支时,已为时过晚等。

- 完成工作预算费用 BCWP (Budgeted Cost of Work Performed): 指根据批准认可的预算,到某一时点已经完成的工作应当投入的资金。当然,已完成工作必须经过验收,符合质量要求。这个值反映了满足质量标准的工作实际进展,实现了投资额到项目成果的转化。俗称“实现投资额”或“实绩值”。在很多情况下,由于业主正是根据这个值对承包商完成的工作量进行支付,也就是承包商挣得的金额,故常称为“挣得值”(Earned Value)。这个值在 EVMS 中其实只是一个监控和管理的数据或信息,而并不是支付金额或挣得金额的标准。

进行进度、费用、质量的联合监控是项目管理需要解决的问题,而实施进度、费用的联合管理,要求在项目进展过程中即时获得上述三个投资额基本值数据,利用计算机技术把进度计划和项目预算有机地结合起来,绘制出时间的费用流曲线,包括计划费用流曲线、实际消耗费用流曲线等,这样就可以对工程进度和费用进行跟踪监控。

这3个值的综合使用可以提供评价工作是否按照计划完成的尺度,以下几个概念是项目衡量的基础:

- 进度偏差 SV (Schedule Variance) = BCWP - BCWS。SV > 0,表示项目进度提前;SV = 0,表示进度符合计划;SV < 0,表示进度拖后。对于进度拖后的项目要及时采取适当措施。

- 费用偏差 CV (Cost Variance) = BCWP - ACWP。CV > 0,表示项目费用在预算之内;CV = 0,表示项目费用符合预算;CV < 0,表示费用超出预算。对于超支的项目要及时采取适当措施。

- 进度绩效指数 SPI (Schedule Performance Index) = BCWP / BCWS。SPI > 1.0表示进度超前;SPI = 1.0表示进度符合计划;SPI < 1.0表示进度滞后。它被广泛用于预测项目完工日期。

- 费用绩效指数 CPI (Cost Performance Index) = BCWP / ACWP。CPI > 1.0表示费用结余;CPI = 1.0表示费用符合预算;CPI < 1.0表示费用超支。它被广泛用于预测完工时的项目费用。

- 完工预算 BAC (Budget At Completion)。是各活动计划消耗的总和,即项目完成的预算。

- 完工尚需估算 ETC (Estimate-to-complete) = 1 / CPI \* (BAC - EV)。

- 完工估算 EAC (Estimate At Completion) = ACWP + ETC。指在项目进行过程中,运用 EV 预测出的项目完成后的花费。通过与 BAC 的比较,我们就能科学地衡量出项目进行的状态,合理地判断出计划制订的成败。

### 3 利用 EVMS 进行软件过程改进

#### 3.1 自底向上软件过程改进模型

• 156 •

软件过程改进的最终目的是提高软件组织的生产力和软件产品的质量。它主要有两种模型,自顶向下软件过程改进模型和自底向上软件过程改进模型。其中自底向上过程改进方法是以最终产品为出发点,根据产品生产的具体需求改善过程,过程改善的动因来源于产品。

自底向上软件改进模型的目标是持续的过程和产品改善。为达到此目标,把软件过程改进分为3个阶段:理解阶段、评估阶段和打包阶段。软件过程改进以这3个阶段为一个循环,上一次循环的结束作为下一次循环的开始,循环往复达到持续的关键过程改进。

(1)理解阶段:该阶段的目标是建立过程改进基线,内容包括过程、产品和环境三个方面。过程信息包括项目组成员工作投入的分布情况和资源使用情况;产品信息包括各个软件阶段产品开发的花费、产品规模和软件问题报告;环境信息包括软件信息、开发技术等。

(2)评估阶段:该阶段的目的是实施过程变更,有三个子阶段——建立过程改进目标、引入过程变更、评价过程变更。它们都是依据“理解阶段”收集的数据量化目标,在软件基线的基础上根据部门或项目组的需要制订软件过程、产品改善的目标,进而改进过程并评价改进的得失。

(3)打包阶段:该阶段的目的是把成功的经验包装成可重用的模型和规范。打包内容包括两个方面:一是通过评价将成功的经验拟定标准、制订规范、进行推广;一是将不成功的经验放入经验库内,为下一次过程改进提供负面信息<sup>[8]</sup>。

#### 3.2 EVMS 的实施步骤

根据自底向上软件过程改进模型,EVMS 的实施从理解阶段、评估阶段和打包阶段三个层次可以分为以下10个主要步骤。

##### (1)理解阶段

① 建立 WBS,将项目分解成若干个可供管理的工作单元。

② 确定工作单元的任务,并将其分解为活动,活动是衡量项目的最小单元,是不能够再次分割的。

③ 估算出每个活动的预期花费和活动完成的最后期限。

④ 利用列表、制图和分析数据等一系列方法来确定这个工作计划是否可行。如果可行则跳到步骤⑤,否则回到步骤①。

##### (2)评估阶段

⑤ 根据活动进程的状态,更新计划表。

⑥ 输入活动完成的实际花费。

⑦ 执行 EV 计算,做运算完成后的报告和报表。它包括三个方面:

a) 计算进度,包括 SV、SPI;

b) 计算费用,包括 CV、CPI;

c) 计算 EAC。

⑧ 分析这些数据,它包括三个方面:

a) 根据 SV、SPI 评估进度;

b) 根据 CV、CPI 评估费用;

c) 根据 EAC 和 BAC 的比较评估软件结束状态。

根据这些分析结果评估软件过程,预测项目前景以及将经验置于经验库中。如果项目完成、需求变更或者计划调整,则跳到步骤⑨,否则转至步骤⑤。

评估阶段具体的工作流程图可以用图3来说明。

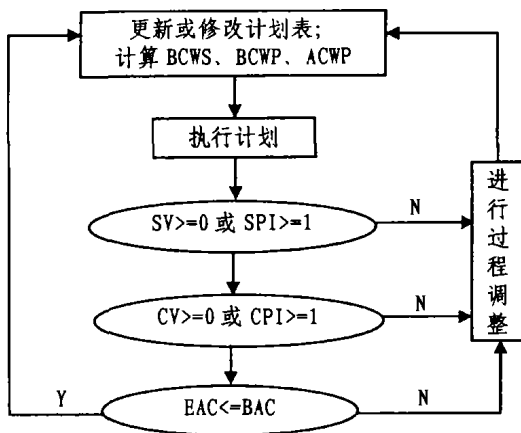


图3 评估阶段工作流程图

(3)打包阶段

⑨ 从经验库中分析过程产生的经验。将成功的经验制订规范和标准予以推广，将失败的经验做成反面教材予以警示。

⑩ 根据经验库的经验改进软件过程，返回到步骤①。

3.3 EVMS 的组织关系

根据 EVMS 的实施步骤，考虑在软件企业内建立三种组织：开发组、过程组和支持组。其功能分别是：

(1)开发组负责软件开发、维护以及应用有效的软件工程技术方法于软件开发过程中。其目标是开发高质量的软件产品，充分满足客户需求。

(2)过程组负责评价软件过程，将软件过程中得到的经验进行分类以及将成功经验打包予以推广。其目标是充分支持开发组的工作，帮助开发组避免以前的错误，吸取成功的经验更好地进行开发工作。

(3)支持组负责配置软件开发环境、平台，管理经验库以及有效地收集、存储、跟踪信息等。其目标是为开发组和过程组做好后勤工作，保证它们的工作能够有效地进行。

三个组织的关系如图4所示。

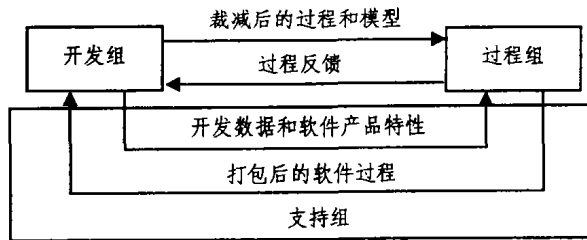


图4 EVMS 的组织关系

**结束语** 软件企业进行软件过程改进的最终目的是为了高质量、高效率、低投入地生产软件。自底向上模型是软件过程改进的一种方式，它以最终产品为出发点，根据产品生产的具体需求改善过程；EVMS 是项目管理的一种先进方法，它能够在项目进行过程中随时报告项目进展，并能预测项目完成后的状态，协助项目经理提高决策质量，帮助项目沿着一个良性的轨道发展。而将两者结合起来进行软件过程改进，则能够帮助软件企业降低项目风险，提高开发效率和软件质量，保证企业获得最大的利润。

参考文献

- 1 Staley M J, Oberndorf P, Sledge C A. Using EVMS with COTS-Based System. CMU/SEI-2002-TR-022, ESC-TR-2002-022, 2002
- 2 Banerjee G, Das R. Earned Value Management System - for IT Projects, May 2003
- 3 何轶彬, 陈晓彬, 金和平, 何文. 工程项目成本/进度综合控制方法及应用, 2001, URL: <http://www.cws.net.cn/Journal/Three-Gorges/200107/14.html>
- 4 What Is Earned Value Management. URL: <http://evm.nasa.gov/definition1a.html>
- 5 Fleming Q W, Koppelman J M. Earned Value Project Management - A Powerful Tool for Software Projects. The Journal of Defense Software Engineering, July 1998. 19~23
- 6 Wilkens T T. Earned Value, Clear and Simple, April 1999
- 7 Boehm B. Value-Based Feedback in Software/IT Systems, 2000
- 8 李健, 金茂忠. 有效改善软件过程方法研究. 计算机研究与发展, 2001, 38(1): 26~35

(上接第139页)

性检测增加的通信量 = 正常情况下的概率性检测消息 + 商家转发给银行的服务请求消息 + 银行给商家的应答消息 + 超额花费者引发的警告消息 + 诚实用户引发的警告消息 + 超额花费者引发的概率性检测消息 =  $c|U| + (1-c/N)|U|W + |U|W + 2t|U|W + 3d|U|W + t|U|M$ 。

所以新协议给每个用户每天增加的通信量 =  $(c+t)M + W(2+3d+2t-c/N)$ 。

Payword 协议每个用户每天的通信量 =  $2N$ 。

在实际电子交易中检测次数  $c$ 、每天每个用户参与交易的商家的平均数  $W$  以及错误门限  $M$  和  $k$  可控制在一定范围内，从而可把概率检测机制增加的通信量控制在实际交易可接受的限度。

**小结** 在分析现有小额电子支付协议的基础上，提出了一个新的小额电子支付协议，新协议可以防止用户超额花费，同时具有特点：(1) 具有较好的公平性。(2) 记费的双向不可否认功能。(3) 信息产品的保密传输功能。(4) 双向认证功能。(5) 适当的运行效率。(6) 出错处理功能。

参考文献

- 1 Glassman S, et al. The Millicent protocol for inexpensive electronic commerce. In: Proc. 4th Int. World Wide Web Conf. Boston,

- MA, Dec. 1995. 603~618
- 2 Manasse M. The Millicent protocols for electronic commerce. In: Proc. 1st USENIX workshop on electronic commerce, New York, USA, July 1995. <http://www.research.digital.com/SRC/millicent/>
- 3 Furche A, Wrightson G. SubScript-An efficient protocol for pay-per-view payments on the internet. In: Proc. 5th Int. Conf. on computer communications and networks (ICCCN'96), Rockville, MD, Oct. 1996. <http://www.cs.newcastle.edu.au/Research/a-furche/subscrip.ps>
- 4 Rivest R, Shamir A. Payword and Micromint: Two simple micropayment protocols. In: Proc. Security Protocols 1996, Springer LNCS 1189, 69~88
- 5 Anderson R, Manifavas C, Southerland C. NetCard-A practical electronic cash system. In: Proc. Security Protocols 1996, Springer LNCS 1189, 49~57
- 6 Hauser R, Steiner M, Waidner M. Micropayment based on iKP: [Technical report RZ 2791]. IBM Research, Feb. 1996
- 7 Pederson T. Electronic payment of small amounts. In: Proc. Security Protocols 1996, Springer LNCS 1189, 56~68
- 8 Damgard I. A design principle for hash functions. Advances in Cryptology-CRYPTO'89, Springer-Verlag, 1990. 416~427
- 9 Even S, Goldreich O, Micali S. On-line/off-line digital signatures. J. Cryptology, 1996, 9: 35~67