



软件能力成熟度模型

何新贵 王 伟 王方德 崔宗学 余林生 范如鹰 周伯生 蔡愉祖 编 著



清华大学出版社

<http://www.tup.tsinghua.edu.cn>



软件能力成熟度模型

何新贵 王 伟 王方德 崔宗学

余林生 范如鹰 周伯生 蔡愉祖

编 著

清华大学出版社

(京)新登字 158 号

内 容 简 介

软件能力成熟度模型被国际软件界公认为是软件工程学的一项重大成果。它既可用于科学地评价软件开发单位的软件能力成熟度等级,也可用来帮助软件开发单位进行自检,了解自己的强项和弱项,从而不断完善单位的软件开发过程,确保软件质量,提高软件开发效率。

本书系统而全面地介绍和论述了几种能力成熟度模型的概念、总体框架、具体内容及其思想精粹。全书共分两部分 11 章。第一部分共 8 章,详细论述了软件能力成熟度模型的概念、总体框架以及与各个成熟度等级相关的关键过程域的目标和关键实践。第二部分共 3 章,介绍了国际上有关软件能力成熟度模型的新进展,其中第九章介绍集成的能力成熟度模型 CMMI,第十章介绍个体软件过程 PSP,第十一章是有关软件过程的国际标准 ISO 9001、ISO/IEC TR 15504-98 与 CMM 的异同比较,其中包括 ISO/IEC TR 15504-98 较详细的介绍。

本书不但能起到介绍和引入 CMM 的作用,而且可作为在我国推广应用 CMM 的重要参考资料,也可供高等院校软件工程专业的高年级学生、研究生和教师使用。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名: 软件能力成熟度模型

作 者: 何新贵 王 伟 王方德 等编著

出版者: 清华大学出版社(北京清华大学学研大厦,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印刷者: 北京密云胶印厂

发行者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 16.5 字数: 401 千字

版 次: 2000 年 11 月第 1 版 2000 年 11 月第 1 次印刷

书 号: ISBN 7-302-03811-2/TP·2419

印 数: 0001-6000

定 价: 28.00 元

7-302-03811-2

在 计算机出现以后的几十年中,探索新的软件开发方法和技术以提高计算机软件的
生产率和质量一直是软件工程领域研究的一个焦点。直到 20 世纪 80 年代中,大多
数研究人员都有一个信念,认为“只要有好的软件开发方法和工具就可开发出高质量的软
件,并且就能提高软件生产率”。但是,很多大型软件系统的开发经验表明,事情并不那么简
单。大约到 1987 年前后,美国工业界和政府部门特别是 DoD 和军方开始认识到,在软件开
发中,如果软件开发组织不能良好地定义和管理其软件过程,开发组织往往不能从软件工程
的研究成果即良好的软件开发方法和工具中充分获益,从而得不到期望的结果。软件过程
(包括各种工程过程和管理过程)的不断改进才是增进软件开发组织的能力和
提高软件质量的第一要素。在无纪律的、混乱的软件过程中开发软件往往不是开发进度经常严重滞后、就是经费预算大大超支乃至翻番。在这种背景下,美国卡内基-梅隆大学软件工程研究所 (CMU/SEI) 的亨弗雷 (W. S. Humphrey) 等人在 1987 年前后,提出了**软件过程、软件能力成熟度和成熟度等级**等概念,并正式形成了**软件能力成熟度模型 CMM**,至今已有十多年的历史。

一个软件开发组织的软件能力成熟度是对其定义并使用的各种软件过程成熟度的一种总的度量,它指明组织的软件过程被明确而有效地定义、管理、测量和控制的程度。基于这种概念建立的能力成熟度模型 CMM 既可供用户评价软件开发组织的软件过程能力,也可供开发组织本身进行自检,以便认识其软件过程的强项和弱项,从而指明其过程改进的努力方向。经验表明,只有通过建立全组织的有效软件过程,采用严格的软件工程和管理的办法,并且坚持不懈地付诸实践,才能取得全组织的软件过程能力的不断提高。

本书的大部分作者是×××工程软件专家组的成员,4 年多前为了给工程寻找一种有效地指导软件系统开发的途径和进行全程软件质量管理的方法,开始系统地跟踪国际上软件能力成熟度模型的发展,并具体研究如何将其中国化的问题。

根据我们当时的调查,我国的软件开发现状十分不容乐观,一些软件开发组织之所以能生产出个别优秀软件产品,往往归功于软件开发组的一些杰出个人或小组的努力,而不是通过成功的软件过程的重复使用,开发组织大都没有总结出经过自身验证有效的开发方法和软件过程。

CMM 是一件舶来品,通过研究就可发现,除非全盘改革我国的软件研制和管理体制,要把 CMM 原封不动地搬来中国使用是困难的。因此,为了使 CMM 符合中国国情,必须对它做一些必要的适应性改造。经过近 4 年来的研究,产生了我们的软件能力成熟度模型

CSCMM。应该说明,CSCMM除了上述必要的适应性改造之外,在总体框架和具体内容方面是完全与国际接轨的,可以保证与CMM的成熟度等级之间的对应关系。因此,本书将采用CSCMM取代CMM作为本书的第一部分,用以介绍软件能力成熟度模型的宗旨、概念、总体框架和具体内容等。

1993年2月CMU/SEI发布的CMM 1.1版,在美国DoD和NASA等部门得到了广泛应用。此后,在政府和工业部门许多人的帮助下,CMU/SEI基于几年来在将框架运用到软件过程改进方面所取得的经验,进一步扩展和精炼了该模型,并经在Internet上进行广泛讨论和征求意见,形成了SW-CMM 2.0版的草案C(SW-CMM v. 2.0 Draft C)。草案C采纳了ISO/IEC TR 15504-98, Information Technology-Software Process Assessment 软件过程评估国际标准的一些内容和方法,因而保证了CMM与国际标准的一致性。

与此同时,从CMM的实际应用中发现,纯粹的软件开发组织其实是不多的,较多组织的主要业务是开发各种类型的系统工程产品,其中包括软件工程和软件采办等业务。因此,客观上很需要一种将系统工程、软件工程乃至软件采办等集成在一起的能力成熟度模型。近年来CMM Integration计划就是为了适应这种迫切需要而提出的。CMM Integration计划的产品CMMI将把系统工程能力成熟度模型、软件能力成熟度模型、集成的产品和过程开发(IPPD)以及软件采办能力成熟度模型等集成在一起,形成一个集成的CMM。CMMI强调模型的集成性和完整性,以求更好发挥模型集成后的整体效益。

SW-CMM和CMMI都将注意力集中于软件开发组织(或单位)的软件过程的改进,致力于软件开发组织或软件开发项目的软件过程能力和软件能力成熟度的提高。但是,实践表明,软件开发组织或软件项目的软件能力成熟度的提高,如果没有组成该组织的众多个体的过程意识和过程能力的支持将是无济于事的。针对上述问题,1995年亨弗雷等人又提出了“个体软件过程(Personal Software Process,PSP)”框架。PSP是一种可用于控制、管理和改进个人软件工作方式的自我改善过程,是一个包括软件开发表格、指南和规程的结构化框架。使用PSP方法,软件开发人员可以减少软件缺陷,提高计划能力,增加生产效率。CMM并未提供有关人员实现CMM所需要的具体知识和技能,PSP的开发弥补了这个缺陷。总之,CMM侧重于软件开发组织中有关软件过程的宏观管理,面向软件开发单位;PSP则侧重于组织中个体软件过程的微观优化,面向软件开发人员。二者互相支持,互相补充,使得过程的改进更加完善。

上述内容体现了国际上CMM的最新发展状况,本书的第二部分将着重介绍国际上有关软件过程理论和方法的新进展,包括ISO/IEC TR 15504-98、CMMI和PSP。此外,由于ISO 9001也是评价软件组织的一项国际标准,因此,在第二部分中也对ISO 9001、ISO/IEC TR 15504-98和CMM的异同进行了比较。

本书的目的在于比较系统而全面地介绍和论述各种能力成熟度模型的概念、框架、具体内容及其思想精粹,特别,在CSCMM中包含了作者们的很多创造性。本书是一个集体创造的成果。第一部分CSCMM共八章,是由×××工程软件专家组的何新贵、王纬、王方德、崔宗学、余林生和范如鹰共同研究并编著的,研究初期,蔡愉祖参与了部分工作,最后曾由余林生进行统稿;第二部分的第九章CMMI由王纬编写,第十章PSP由周伯生、车向东和吴超

英合作编写,第十一章“软件过程评估方法的比较”由王伟和何新贵合作编写。最后,何新贵对全部书稿进行了审核和统稿,李梅对全书进行了编排和整理。在本书出版之际,我们要衷心感谢在研究过程中曾给我们大力支持和帮助的陈炳忠、王黎明、宋太亮和牛爱民等同志。我们希望本书的出版不但能起到介绍和引入 CMM 的作用,而且可作为在我国推广应用 CMM 的重要参考资料,希望能不断得到广大读者和具体应用 CMM 的工作者对本书的批评和反馈意见。

何新贵

2000 年夏于北京

第一部分 软件能力成熟度模型 CSCMM/1

第一章 CSCMM 概论/3

- 1.1 基本概念/3
- 1.2 过程成熟度框架/4
 - 1.2.1 软件能力成熟度模型/5
 - 1.2.2 软件过程成熟度等级/6
 - 1.2.3 成熟度等级的行为特征/7
- 1.3 成熟度等级间的关系/9
- 1.4 CSCMM 的可操作定义/10
 - 1.4.1 内部结构的层次/10
 - 1.4.2 成熟度等级/11
 - 1.4.3 关键过程域/11
 - 1.4.4 关键实践类/15
 - 1.4.5 关键实践/15
- 1.5 CSCMM 的应用/16
- 1.6 软件过程评估和软件能力评价方法/17
 - 1.6.1 软件过程能力评估评价组/17
 - 1.6.2 评估评价过程/17
- 1.7 过程评估或能力评价的结果之间的差异/18
- 1.8 软件过程评估和软件能力评价之间的区别/19
- 1.9 CSCMM 在过程改进方面的其他用途/19

第二章 初始级/20

第三章 基本级/22

- 3.1 过程基本特征/22
- 3.2 关键过程域/23
 - 3.2.1 基本软件工程/23
 - 3.2.2 软件需求管理/27
 - 3.2.3 软件计划管理/30
 - 3.2.4 软件版本管理/36
 - 3.2.5 软件外协管理/38

第四章 可重复级/43

- 4.1 过程基本特征/43
- 4.2 关键过程域/43
 - 4.2.1 软件项目跟踪和监督/44
 - 4.2.2 软件子合同管理/48
 - 4.2.3 软件质量保证/54
 - 4.2.4 软件配置管理/58

第五章 已定义级/64

- 5.1 过程基本特征/64
- 5.2 关键过程域/64
 - 5.2.1 组织过程焦点/65
 - 5.2.2 组织过程定义/68
 - 5.2.3 培训大纲/74
 - 5.2.4 集成软件管理/78
 - 5.2.5 软件产品工程/86
 - 5.2.6 组间协调/96
 - 5.2.7 同行评审/100

第六章 定量管理级/104

- 6.1 过程基本特征/104
- 6.2 关键过程域/104
 - 6.2.1 定量过程管理/104
 - 6.2.2 软件质量管理/110

第七章 优化级/116

- 7.1 过程基本特征/116
- 7.2 关键过程域/116
 - 7.2.1 缺陷预防/116
 - 7.2.2 技术革新管理/122
 - 7.2.3 过程更动管理/127

第八章 CSCMM 提问单/134

- 8.1 等级 1: 基本级/135
 - 8.1.1 基本软件工程/135
 - 8.1.2 软件需求管理/137
 - 8.1.3 软件计划管理/138
 - 8.1.4 软件版本管理/139
 - 8.1.5 软件外协管理/140
- 8.2 等级 2: 可重复级/141
 - 8.2.1 软件项目跟踪和监督/141
 - 8.2.2 软件子合同管理/143
 - 8.2.3 软件质量保证/144

- 8.2.4 软件配置管理/145
- 8.3 等级 3: 已定义级/147
 - 8.3.1 组织过程焦点/147
 - 8.3.2 组织过程定义/148
 - 8.3.3 培训大纲/150
 - 8.3.4 集成软件管理/151
 - 8.3.5 软件产品工程/152
 - 8.3.6 组间协调/153
 - 8.3.7 同行评审/154
- 8.4 等级 4: 定量管理级/156
 - 8.4.1 定量过程管理/156
 - 8.4.2 软件质量管理/157
- 8.5 等级 5: 优化级/158
 - 8.5.1 缺陷预防/158
 - 8.5.2 技术革新管理/160
 - 8.5.3 过程更动管理/161

第二部分 国外的发展状况/163

第九章 集成的能力成熟度模型 CMMI/165

- 9.1 CMMI 发展概况/165
 - 9.1.1 CMMI 项目的启动/165
 - 9.1.2 CMMI 的发展背景/165
 - 9.1.3 CMMI 的目标/167
 - 9.1.4 CMMI 的现状/167
- 9.2 CMMI 模型/168
 - 9.2.1 CMMI 的连续表示/168
 - 9.2.2 CMMI 分级表示/191
- 9.3 模型的应用/198
 - 9.3.1 基本用途/198
 - 9.3.2 过程评估/198
- 9.4 结束语/202

第十章 个体软件过程 PSP/203

- 10.1 概述/203
- 10.2 个体软件过程框架/204
 - 10.2.1 个体度量过程 PSP0 和 PSP0.1/205
 - 10.2.2 个体规划过程 PSP1 和 PSP1.1/205
 - 10.2.3 个体质量管理过程 PSP2 和 PSP2.1/206
 - 10.2.4 个体循环过程 PSP3/206
- 10.3 时间管理和项目规划/207

10.3.1	时间管理/207
10.3.2	产品计划/208
10.3.3	产品规模/210
10.3.4	进度管理/213
10.3.5	项目计划总结表/214
10.4	产品质量/217
10.4.1	缺陷及其管理/217
10.4.2	缺陷查找技术/221
10.4.3	缺陷估计/224
10.4.4	缺陷排除经济学/225
10.4.5	设计缺陷/226
10.4.6	产品质量/227
10.4.7	过程质量/229
10.5	总结/231
第十一章	软件过程评估方法的比较/232
11.1	前言/232
11.1.1	软件过程评估的必要性/232
11.1.2	现状/232
11.2	软件过程评估方法/233
11.2.1	软件能力成熟度模型 CMM 概述/234
11.2.2	ISO 标准“软件过程评估”(SPA)概述/238
11.3	过程评估方法的比较/246
11.3.1	CMM 1.1 与 ISO 9001 的比较/246
11.3.2	CMM 1.1 与 ISO/IEC 15504 标准的比较/249
	参考文献/253

自从人类发明数字计算机以后,计算机软件的开发一直是广泛应用计算机的瓶颈。研究新的软件开发方法和技术以提高计算机软件的生产率和质量一直是软件工程研究的焦点。这个问题可以说至今也没得到彻底解决,但经过几十年的努力,情况已经大为改观。大约在 1987 年前后,美国工业界和政府部门开始认识到,在软件开发中,最关键的问题在于软件开发组织不能很好地定义和管理其软件过程,从而使一些好的开发方法和技术都起不到所期望的作用[DoD87]。在无纪律的、混乱的软件项目开发状态中,开发组织不可能从软件工程的研究成果,即较好的软件方法和工具中获益。在很多组织中,软件项目开发进度经常严重滞后、经费预算往往超支乃至翻番[Siegel90]。在这种情况下,软件开发组织一般都不能提供使开发项目避免这些问题所需要的基础设施和资金。根据我们的调查,我国的现状更不容乐观。情况几乎与 10 多年以前美国的一样,我国的一些软件开发组织之所以能生产出个别优秀软件产品,成功往往应该归功于软件开发组的一些杰出个人或小组的努力,而并不是通过成功的软件过程的重复使用,开发组织并没有总结出经过验证有效的开发方法和软件过程。

美国 SEI 提出软件能力成熟度模型 SW-CMM,试图用来解决上述问题。CMM 是一个分级的模型,其分层结构是基于已有 60 多年历史的产品质量原理建立的。Walter Shewart 在 20 世纪 30 年代发表了统计质量控制原理。W. Edwards Deming [Deming86]和 Joseph Juran [Juran88, Juran89]的著作又进一步发展和论证了该原理。SEI 的研究人员将这些原理应用于软件开发,改变成为软件过程成熟度框架。该框架为软件过程的定量控制建立了项目管理和项目工程的基本原则,这是软件过程得以不断改进的基础。

实际上,将质量原理改变为成熟度框架的思想是 ITT 的 Philip Crosby 在其书“Quality is Free”[Crosby79]中首先提出的。Crosby 的质量管理成熟度网格描述了采用质量管理实践时的五个进化阶段。该成熟度框架后来在 Watts Humphrey 指导下,由 IBM 的 Ron Radice 及其同事们作了进一步的改进,使之适应软件过程的需要[Radice85]。1986 年, Humphrey 将此成熟度框架带到了软件工程研究所(SEI),并增加了成熟度等级的概念,形成了当前已在整个软件产业界使用的软件能力成熟度框架的基础。

Humphrey 的成熟度框架早期版本发表在 SEI 技术报告 [Humphrey87b]、文章 [Humphrey88]和书《Managing the Software Process》[Humphrey89]中。1987 年发表了初步的成熟度提问单 [Humphrey 87b],它作为工具给软件开发组织提供了软件过程评估的一种方法。1987 年又进一步提出了软件过程评估和软件能力评价的方法,以便估计软件开发组织的软件过程成熟度。自 1990 年以来,在政府和工业部门许多人的帮助下,SEI 基于几年

来在将框架运用到软件过程改进方面所取得的经验,进一步扩展和精炼了该模型,提出了 CMM 2.0 版的草案 C(SW-CMM v. 2.0 Draft C)。草案 C 采纳了 ISO/IEC 的软件过程评估 (SPA)标准 SPICE 的一些方法和内容,发放以进一步征求意见。

在 CMM 的实际应用中发现,社会中纯粹的软件开发组织其实是不多的,较多的组织是开发各种类型的系统工程产品,其中包括软件工程和软件采办等业务。因此,客观上需要一种将系统工程、软件工程乃至软件采办等集成在一起的能力成熟度模型。近年来,CMM Integration Project 就是为了适应这种迫切需要而提出的[CMM Integration Project, 1999]。CMM Integration 产品包括:CMM SW(相当于 SW-CMM)、CMM SE(相当于 SE-CMM)、CMM SE/SW、CMM SE/SW with IPPD(集成的产品和过程开发)以及 CMM SA(软件采办)。CMMI 强调模型的集成性,以求更好发挥模型集成后的整体效益。

根据对目前我国国情的分析,我们认为,从 SW-CMM 开始,逐步开展自己的工作可能是更恰当的。因为,在没有整个组织范围的软件过程的条件下,软件开发过程并没有明确的定义,文档不完整,也不规范,要重复成功的经验很困难。一般必须让同样的人员去做一个相同(或极类似)的项目才有可能成功。显然,这种依赖于个别人员上的成功并不能为全组织的软件生产率和质量的提高打下长期基础。经验表明,只有通过建立全组织的有效的软件过程,采用严格的软件工程方法和管理,并且坚持不懈地付诸实践,才能取得全组织的软件过程能力的不断改进。

根据我国一些大型工程项目的需求,例如×××工程的需求,在对国内外情况和技术资料充分调查研究和分析的基础上,参照原 SW-CMM 的总体框架,并吸取其对我有用的内容,提出了一个符合我国国情和软件开发水平的软件能力成熟度模型 CSCMM。本书的第一部分将主要描述 CSCMM 以及可作为其评估评价工具的“提问单”。

1.1 基本概念

为了叙述的方便和使概念的含义得到准确的理解,本节首先定义一些基本概念。

软件过程: 韦氏字典把一个过程定义为“生产某物的操作体系……能导致结束或得到结果的一系列活动、变更或操作”。IEEE 定义过程为“为实现给定目标所执行的一系列步骤”[IEEE-STD-610]。所以,一个软件过程可以定义为,“人们用以开发和维护软件及其相关产品(例如,项目计划、设计文档、代码、测试用例等等,在模型中往往称这些为软件工作产品)的一系列活动,包括软件工程活动和软件管理活动。自然,其中一定会涉及有关的各种方法和技术等”。

软件过程能力: 描述(开发组织或项目组)通过执行其软件过程能够实现预期结果的程度。一个软件开发组织或项目组的软件过程能力,提供一种预测该组织或项目组承担下一个软件项目时最可能的预期结果的方法。软件过程能力既可对整个软件开发组织而言,也可对一个软件项目组而言。

软件过程性能: 表示(开发组织或项目组)遵循其软件过程所得到的实际结果。软件过程性能既可对整个软件开发组织或项目组而言,也可对一个特定软件项目而言。可见,软件过程性能描述已得到的实际结果,而软件过程能力则描述最可能的预期结果。由于一个特定软件项目的具体属性和执行该项目的的环境所限,该项目实际的过程性能可能并不充分反映其所在组织的整个过程能力。例如,由于应用领域或所用技术的根本改变,可能使开发组织的职工对此不适应,正处于一种学习状态,使得他们的项目组的过程能力和项目的过程性能不一致,远达不到其所在组织的整体过程能力。

软件过程成熟度: 一个特定软件过程被明确和有效地定义、管理、测量和控制的程度。成熟度可指明一个软件开发组织软件过程能力的增长潜力和可改进的方面,也可表明一个组织的软件过程的丰富多样性,及其各个开发项目所遵循的软件过程的一致性。

在一个成熟的组织中,通常是通过正式文档和培训使全组织的有关人员对组织规定的软件过程都能很好地了解,并且使过程的执行始终能得到用户的监控和不断的返回意见。一个软件过程的成熟意味着,由于开发组织运用该软件过程,使得各个开发项目执行该软件过程的纪律性一致地增强,并导致软件生产率和质量随着时间的推移不断得到改进。

随着软件组织的软件过程成熟度的提高,开发组织通过其方针政策、标准规范和组织机构等的制定和建立,将软件过程规范化和具体化,使得开发组织明确定义的一套有关管理和工程的方法、实践和规程等在现有人员离去后仍能继续下去。因为,在现代社会中,各种组织间人员的流动和交换不但是正常的,而且是值得鼓励的。在人员变动迅速的情况下,提高软

件过程成熟度尤为重要。

软件能力成熟度等级：软件开发组织在走向成熟的途中几个具有明确定义的、表征软件过程能力成熟度的平台。每一个成熟度等级为其软件过程继续改进达到下一更高的等级提供基础。每一等级包含一组过程目标,当其中一个目标被达到时,就表明软件过程的一个(或几个)重要成分或方面得到了实现,从而导致软件开发组织的软件过程能力的进一步增长。

关键过程域：互相关联的若干软件实践活动和有关基础设施的集合。每个软件能力成熟度等级包含若干个对该成熟度等级至关重要的过程域,它们的实施对达到该成熟度等级的目标起保证作用,这些过程域就称为该成熟度等级的关键过程域。

顾名思义,既然有关键过程域就有非关键过程域。由于非关键过程域对达到相应软件成熟度等级的目标不起关键作用,所以在定义软件成熟度等级时不叙述它们。

关键实践：对关键过程域的实施起关键作用的方针、规程、措施、活动以及相关基础设施的建立、实施和检查。关键实践一般只描述“做什么”,而不强制规定“如何做”。关键过程域的目标是通过其包含的关键实践来达到的。

国际上有一个已取得共识的基本观点是:整个软件过程的改进是基于许多小的、进化的步骤,而不是通过一次革命性的创新而实现的[Imai86]。这些小的进化步骤就是通过一些关键实践来实现的。

1.2 过程成熟度框架

一个组织的软件成熟度一方面指明其有效地定义、管理、测量和控制软件过程的程度,另一方面也可帮助该组织找到差距,指出其努力的方向和改进的目标。为了设置合理的过程改进目标,必须了解不成熟的软件开发组织和成熟的软件开发组织间的区别。在不成熟的软件开发组织中,软件过程没有一个明确且稳定的定义,一般由开发者及其管理者在项目进行中临时拼凑而成。有时在开发组织内虽已规定了一些软件过程,但各项目在软件开发实践中却不能严格地遵守和实施它。不成熟的软件开发组织实施软件过程的管理方式往往是反应式的,即在遇到问题以后才不得不作出反应。通常,经理们要把精力集中于解决随时可能发生的危机(即所谓“消防模式”)。由于制定进度和预算时,并非基于现实和历史经验进行估计,因而在开发过程中进度滞后和预算超支成为常事。当不得已而硬性规定完成时限(即所谓“后墙不倒”),为了勉强满足进度要求,只好在产品功能和质量上作出牺牲,从而酿成大祸。

在不成熟的软件开发组织中,不存在判断产品质量,以及解决产品质量问题和软件过程问题的客观基础。因此,产品质量难以预测。当项目进度滞后时,常常不得不缩减乃至取消诸如评审和测试这些旨在提高质量的活动。

可是,一个成熟的软件开发组织具有全组织范围的控制软件开发和维护过程的能力。在这种组织中,其软件过程被正确无误地告知所有职工,包括对新职工的特别培训,从而一切工作和活动均强制按照已规定的过程进行。只有当十分必要时,才对这些已定义的过程进行更新,并且通过可控的先导性试点和(或)费效分析使其得到有效地改进。在已明确定义的软件过程中,整个软件开发组织和所有软件开发项目内的各种岗位明确,职责分明,纪律严明。

在成熟的软件开发组织中,经理们时刻监控着软件产品的质量和顾客的满意程度。在判断产品质量和分析产品及过程问题方面有客观的、定量的依据。进度和预算是基于以前的性能数据和当前的实际情况进行的,因而是现实可行的。通常,都能达到产品的成本、进度、功能和质量的预期结果。一般而言,成熟的组织的各项目组都能有纪律地、一致地遵循一个预先已定义的软件过程,因为所有参加者在主观上都了解这样做的价值,而且在客观上也存在支持该软件过程实施所必需的基础设施。

为了能从这些有关不成熟和成熟的软件开发组织的观察资料中获益,需要构造一个**软件过程成熟度框架**。因为,尽管软件工程师和经理们通常都非常清楚地了解实际存在的问题,但他们对在软件过程中改进什么最为重要很可能意见不一致。如果对过程的改进工作没有一个有效的改进策略,那么,要在管理者和专业人员之间就首先进行哪些改进活动形成一致意见是很困难的。因此,为了使软件过程改进工作能不断取得成效,必须设计一条**进化路径**,使软件开发组织的软件过程成熟度循此路径分阶段逐步提高。过程成熟度框架就是要描述一条从无序的、混乱的过程达到成熟的、有纪律的过程的进化途径,要把软件过程、软件过程能力、软件过程性能和软件过程成熟度等概念集为一体。

1.2.1 软件能力成熟度模型

软件过程成熟度框架的基础是软件能力成熟度模型,该模型把软件过程从无序到有序的进化过程分成几个阶段,并将这些阶段排序,形成一个逐层提高的平台,使每个平台的过程能力能为达到下一更高级的平台打下基础。所以,从软件过程成熟度框架导出的改进策略,将对软件过程不断改进的历程提供一份“引导图”。它指导软件开发组织不断识别出其软件过程的缺陷,引导开发组织或项目组在各个平台上“做什么”改进,但它并不提供“如何做”的具体措施。

软件能力成熟度模型是为了指导软件开发组织,通过确定当前过程的成熟度和识别出执行软件过程的薄弱环节,通过解决对软件质量和过程改进至关重要的几个问题,来形成对其过程的改进策略;通过关注并认真实施一组有限的关键实践活动,稳步地改善其全组织的软件过程,使全组织的软件过程能力持续增长。

我们基于CMM 1.1版,参照CMM 2.0版和许多有关的技术资料,进行了多年的调查、分析和研究,提出并逐步形成了符合我国国情的一个软件能力成熟度模型CSCMM。本书的第一部分将主要介绍CSCMM及其相应的提问单等内容。

CSCMM与CMM的主要区别在于,它把软件过程的进化步骤分成6个成熟度等级,比CMM多了1个等级。这6个成熟度等级定义了一个循序渐进的度量标准,用以测量软件开发组织的软件过程成熟度和评价其软件过程能力。它为软件过程的不断改进给出了一个循序渐进的进化途径,帮助软件开发组织对其各项过程改进工作排出优先次序。这将是本章要叙述的主要内容。关于软件过程成熟度等级以及各个等级相应关键过程域和关键实践的详细定义,将在其后面的章节中叙述。

CSCMM是一个6层模型,在原CMM的过程成熟度等级1和等级2之间插入了一个基本级,使得成熟度等级的划分更加均匀,便于软件开发组织更容易地升级。CSCMM如图1.1表示。

CSCMM适当细分成熟度等级的原因主要是:当前我国多数软件开发组织没有适当实

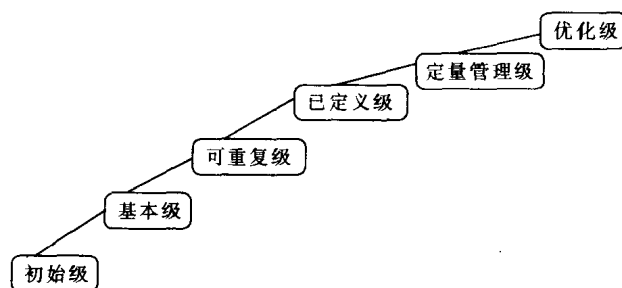


图 1.1 CSCMM 图

施软件工程基本过程,如果等级划得较粗,则一些软件开发组织很可能长期达不到 2 级而一直停留在 1 级上。这样做容易使他们失去信心,对软件开发组织过程能力的提高不利。其次,经过分析发现,把原 CMM 的可重复级适当改造和细分不但十分有利,而且是完全可能的。用 CSCMM 指导我国某大型工程项目的软件过程实践表明,这种决策是正确的。

1.2.2 软件过程成熟度等级

按照软件成熟度等级的定义,各个等级都是经过妥善定义的在向成熟软件组织前进途中的平台。每一个成熟度等级应为软件过程提高到下一更高等级提供基础。

将 CSCMM 组织成如图 1.1 所示的 6 个等级,旨在将软件过程成熟度的改进行动按优先级排序。

下面列出 6 个成熟度等级的特性,突出说明在每个等级上软件过程的主要变化:

(1) **初始级(0 级)**: 软件过程的特点是无秩序的,甚至是混乱的。几乎没有什么过程是经过妥善定义的,成功往往依赖于个人或小组的努力。

(2) **基本级(1 级)**: 已建立一些初级的项目管理过程来管理项目的成本(包括人、财、物、信息等)、进度和功能实现。但项目关于进度和成本的计划是较粗略的,而且计划与执行情况有时可能不一致。为了监督项目功能和性能按用户需求实现,已建立了简单的问题报告制度,但问题的报告和解决有时不够及时。软件工程的基本要求已经建立,且已在执行,但尚未成为自觉行为。

(3) **可重复级(2 级)**: 已建立基本的项目管理过程来策划和跟踪项目的成本(包括人、财、物或信息)、进度和功能的实现,而且计划与执行情况基本一致。软件工程的要求已经建立,且作为纪律严格执行,使得今后完成类似的应用项目能重复以前的成功,并不受人员流动的影响。

(4) **已定义级(3 级)**: 有关软件项目的管理过程和工程过程均已明确妥善定义,且已文档化和标准化,并已将它们集成为软件开发组织统一的标准软件过程。组织内的所有软件开发项目均采用该标准软件过程的一个经批准的剪裁版本,来进行软件的管理、开发和维护等活动。

(5) **定量管理级(4 级)**: 在工作过程中,能详细采集有关软件过程和产品质量的度量数据,使软件过程和产品质量均得到定量地了解和控制。因此,软件开发的成本、进度和软件质量等都是定量地可预测的。

(6) **优化级(5 级)**: 能自觉利用在执行过程中总结出的经验以及来自新思想和新技术

的先导性试点的定量反馈信息,来持续不断地改进组织的标准软件过程,从而使组织的软件过程能力能不断地得到增强和优化。

1.2.3 成熟度等级的行为特征

本节叙述软件成熟度各等级的行为特征,将通过以下两方面加以描述:

- (1) 软件开发组织为建立或改进软件过程所进行的活动及其状态。
- (2) 对每个软件项目组所进行的活动及其产生的跨项目的过程能力。

下面之所以列出等级 0(初始级)的行为特性,是为了在过程改进上给较高的成熟度等级提供一个比较的基础。

1.2.3.1 等级 0——初始级

在初始级上,软件开发组织一般不能提供开发和维护软件的稳定环境。不适当的规划和应急式的管理体系使良好的软件工程方法和实践不能产生良好的效益。

在开发中遇到危机时,一般项目组就抛弃临时拼凑的软件开发过程,回复到仅作编码和代码调试工作的状态。软件项目的成功完全依赖于一个杰出的管理人员或一个有经验的、有战斗力的软件开发队伍。偶尔,一个有能力的、坚强的软件管理人员能经受住要他们在软件过程中走捷径的压力,例如要求他们牺牲测试和其他软件质量保证措施以赶进度。但是,当他们离开该项目后,他们的这种能使过程稳定的影响力也就随之消失。在此要特别强调,关键在于管理。一个很好的软件工程过程并不能克服由于缺乏健全的管理实践所造成的不稳定性。

处于等级 0 的开发组织的过程能力是不可预测的,因为随着工作进展,软件过程经常被改变或修订。进度、预算、资源消耗和产品质量一般是不可预测的。性能依赖于个人的能力,且随个人具有的技能、知识和动机的不同而变化。所以,只能通过个人的能力而不是组织的能力去预测软件项目的性能。

1.2.3.2 等级 1——基本级

在基本级上,已建立一些初级的项目管理过程来管理项目成本(包括人、财、物、信息等)、进度和功能实现。但关于项目进度和成本的计划是较粗略的,而且计划与执行情况有时不一致。为了监督项目的功能和性能按用户需求实现,已建立了问题报告制度,但有时问题的报告和解决不够及时。已建立软件版本管理来控制软件源程序及相应文档版本的生成和更改,但还不是全生命周期的软件配置管理。一些软件工程的基本要求(例如软件生存周期模型,文档标准以及软件工程化方法和技术等)已经建立,但实施有时偏离要求。已初步建立了管理软件项目的方针和实施这些方针的规程,但往往由于主观上或客观上的各种原因,规程实际上不能真正实施。

如果有分承制方的话,能以子合同的形式与分承制方建立一种较严格的子承包关系,从进度和质量上控制分承制方任务的完成,而且分承制方也应处于等级 1。

处于等级 1 的软件开发组织的过程能力可概括为已基本上能按软件工程方法进行软件开发的。所谓“基本上”是指开发软件和管理软件项目的软件过程已基本建立,在项目管理和项目工程两方面都已制定了明确的规定,但有时不能严格执行,因此还不能保证可重复以前