

软件过程评估与改进及其标准化进展述评

The Commentary of Standardizing Development of Software Process Assessment and Improvement

王志刚

廖晓峰

(楚雄师范学院计算机科学系 楚雄675000) (重庆大学计算机科学与工程学院 重庆400044)

Abstract The software process assessment and improvement has been one of the main approaches to enhance the quality and success rate of software products. Scheduled to be released in 2002, the ISO/IEC 15504 (SPICE), after integrating the merits of earlier standards such as SEI CMM, will become a better software process assessment standard with improved integrity, rationality and maneuverability and will, when used together with ISO/IEC 12207 and ISO 9000-3, set a new strict standard system in regulating software development and guaranteeing the quality of software. This paper is a presentation of the basic frame and content of ISO/IEC 15504 (SPICE), as well as a contrastive review of other standards such as SEI CMM.

Keywords Software engineering, Software process assessment, ISO/IEC 15504 (SPICE), ISO/IEC 12207, ISO 9000-3, SEI CMM

1 引言

目前,基于 Internet/Intranet 的分布式计算正逐渐成为主流应用技术,与早期的单机计算系统或集中式计算系统相比,计算机应用系统结构和功能日益复杂,而系统实现的技术与管理难度也相应加大,项目的风险越来越高。由美国 The Standish Group 于1997年提供的调查报告表明:有70%左右的软件项目不能完全实现用户所需的功能。又据美国政府近年的调查,在实施的软件工程项目中,约有60%的项目延期交付,约有50%的项目超出预算,约有45%的已交付项目不能使用。这种困境多数是因开发组织的软件过程缺陷及过程能力有限而造成的。

软件过程是指人们用于开发和维护软件及其相关过程的一系列活动,包括软件工程活动和软件管理活动。软件过程能力则是描述(开发组织或项目组)遵循其软件过程标准能够实现预期结果的程度,它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。软件产品能否按时保质地交付,不但依赖于开发团队的技术水平,更取决于开发组织的软件过程能力,取决于科学规范的工程过程管理标准。

一个差的、不成熟的软件过程是混沌的、无组织和无管理的,仿佛一个黑盒(如图1所示),在软件产品最终出炉之前无法预见开发活动的成败。整个开发过程基本上是无定义、不可测量和无法预测的,因此软件开发的质量、进度和成本也常常是失控的,纵容此种软件过程的项目经理往往是不太熟悉软件工程技术而把软件开发看作是不可预知的一种魔术。



图1 差的、不成熟的软件过程

一个好的成熟的软件过程则应是可定义、可测量和可预测的,整个开发过程是可视的(如图2所示),在按阶段划分的各个检查点都能依照目标和定义可量化地测量软件工程的进

展,因此软件开发的质量、进度和成本可以很好地定量控制。

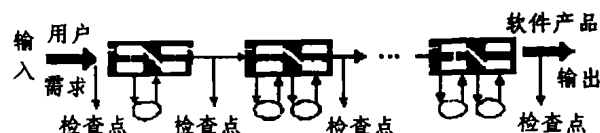


图2 好的成熟的软件过程

开发完成得最早、较有影响的软件过程评估标准是由美国卡内基·梅隆大学软件工程研究所(CMU/SEI)提出 CMM(软件开发能力成熟度模型)。CMU/SEI 的 W. S. Humphrey 等人从1986年开始相关研究,1987年发表研究成果“承制方软件工程能力的评估方法”,并于1991年发表评估标准的最早版本 SEI CMM 1.0,于1993年发表 SEI CMM1.1,于1997年推出 SEI CMM2.0。CMM 标准侧重于对软件开发过程和开发方法的考察评估,以过程成熟度来衡量开发组织的开发能力,过程成熟度是指软件开发过程被定义、管理、测量、控制和有效性的程度。

CMM 的影响之一是促使 ISO 组织开始着手制定有关标准,1991年6月 ISO/IEC 的 JTC1 联合技术委员会(Joint Technical Committee 1)的 SC7 专委(Subcommittee 7)调查制定软件过程评估国际标准的需要,研究结果表明:由于国际贸易的需要以及软件开发与应用市场的国际化,在保证用户需求的基础上,制定一套国际通行的软件过程评估标准已经达成国际共识。于是,ISO 组织于1993年7月决定实施一项研究计划以推动现有软件评估方法的标准化和改进,即 SPICE 计划(Software Process Improvement and Capability dEtermination,软件过程改进与能力测定),该计划由 SC7 的 WG10 工作组(Working Group 10)具体负责,以便制定《ISO/IEC 15504: 信息技术——软件过程评估》标准,该标准预计将于2002年正式发布,1998年发表了最新的进展技术报告 TR。欧洲软件研究所(ESI, European Software Institute)在 SPICE 计划中承担了主要研究和开发工作,另外 ESI 在基于 ISO9000 的软件

王志刚 副教授,主要研究方向为软件工程、容错计算。廖晓峰 博士,教授,博士生导师(电子科技大学),主要研究方向为人工神经网络、计算机网络与通信。

过程改进方面,也做了许多贡献。

2 ISO/IEC 15504评估标准

ISO/IEC 15504提供了一个软件过程评估的框架——软件过程改进与能力测定模型 SPICE。它可以被任何组织用于软件的设计、管理、监督、控制以及提高获得、供应、开发、操作、升级和支持的能力评估。可供软件开发人员了解本组织的过程状态,以便进行过程改进;或确定其开发过程对需方的某种特定要求的适合程度和对某种具体合同的适合程度。

2.1 开发能力等级划分及各等级过程的属性

SPICE 模型将软件能力等级分为6级,从底层的0级不完整级(incomplete)到顶端的5级优化级(optimizing)逐级提高(如图3所示),各能力等级是以必须具备的过程属性来加以度量的,一共9个通有属性。

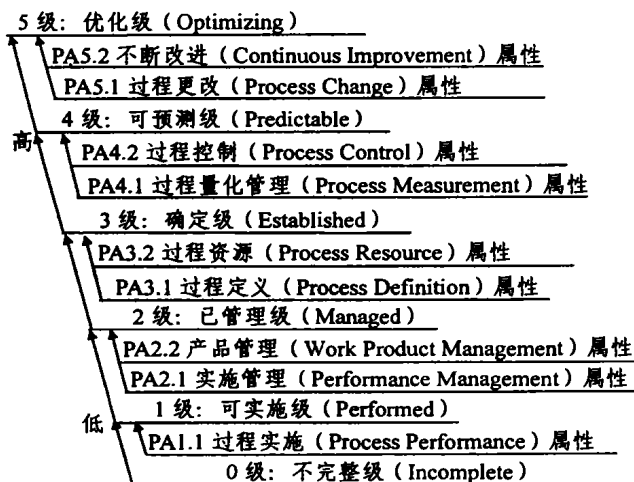


图3 ISO 15504的过程能力等级和各等级的属性要求

第0级——不完整级：在这一级，软件开发过程不能完全实施或不能达到目的。

第1级——可实施级(Performed)：在这一级，当软件开发过程按标准定义时，能够达到预定目的。用以测定该等级按标准实施程度的过程属性是：

PA 1.1 Process Performance Attribute (过程实施属性)——开发过程实施时能开始一组实践活动，然后用可确认的输入而能生成足以达到开发目的工作产品输出的情况。

第2级——已管理级(Managed)：软件开发过程是可实施的，能按时间进度和资源要求交付工作产品，产品的质量是可接受的。这一级有两个过程属性：

PA 2.1 Performance Management Attribute (实施管理属性)——软件开发过程的实施得到有效管理，以按规定的的时间和资源要求生成工作产品的程度。

PA 2.2 Work Product Management Attribute (产品管理属性)——软件开发过程的实施得到有效管理，以生成文档化的、可控制的、符合其功能及非功能需求并与过程质量目标相一致的工作产品的程度。

第3级——确定级(Established)：在第3级，软件开发过程是可实施的、有管理的和确定的，并能采用基于良好的软件工程原则的过程方法。这一级有两个过程属性：

PA 3.1 Process Definition Attribute (过程定义属性)——在软件开发过程的实施中采用基于标准方法的过程定义，使开发过程服从于确定的组织商业目标的程度。

PA 3.2 Process Resource Attribute (过程资源属性)——在软件开发过程的实施中有效地使用合适的熟练人力资源和基础设施，使开发过程服从于确定的组织商业目标的程度。

第4级——可预测级(Predictable)：在第4级，软件开发过程是确定的，并能在确定的控制界限内连续一致地实施以完成目标。这一级有两个过程属性：

PA 4.1 Process Measurement Attribute (过程量化管理属性)——开发过程的实施受到软件目标和量化管理的支持，以确保过程实施服从于开发目标的实现的程度。

PA 4.2 Process Control Attribute (过程控制属性)——通过对软件开发过程的实施量化管理情况的收集和分析，控制过程实施并在必要时予以纠正，以使过程实施得到有效控制而可靠地完成确定目标的程度。

第5级——优化级(Optimizing)：在这一级，软件过程是可预测的，并优化实施以适应当前和未来的商业需要，而且适应商业目标方面实现了可重复性。这一级有两个过程属性：

PA 5.1 Process Change Attribute (过程更改属性)——过程的定义、管理和实施的更改受到有效控制以更好地实现组织商业目标的程度。

PA 5.2 Continuous Improvement Attribute (不断改进属性)——开发过程的更改是确定的和可实现的、可确保在实现组织商业目标方面能够不断改进的程度。

2.2 软件过程改进与能力测定(SPICE)模型

15504标准提供了用于软件过程改进与能力测定(SPICE)的参考模型。SPICE模型中由两组结构成分，一组是按活动类型，把软件生存周期的基本活动分成基本实践、过程、过程类三个层次。一种基本实践是实现某个具体过程必须进行的一种软件工程和管理活动；一个过程是实现某个目的需进行的一组活动；一个过程类是针对同一通用活动领域的一组过程。该模型总共分五类过程，即需方-供方类、工程类、项目管理类、支持类和组织类，可再分为40多个过程。

另一组按形成相应等级能力所需开展的实现方面或制度化方面的活动类型加以组织，也分三个层次：一般实践、共同特征、能力等级。一般实践是提高执行任何过程的能力所必须进行的过程实现或制度化实践；一个共同特征是针对过程实现或制度化的某个侧面进行的一组一般实践；某个能力等级具有相应的一组共同特征，其各项一般实践共同保证执行过程的能力有较大的增强。基本实践对于各能力等级是共同的，一般实践则随各能力等级而不同。

ISO/IEC 15504在制定时力求做到齐全而完备，因此无疑也都是极为复杂和庞大的，在具体应用时必须结合实际项目的特点和需要对标准进行裁剪，从中选取必要的部分形成一个标准的子集版本，而ISO/IEC 15504正是便于裁剪和实用的。

3、ISO 软件标准体系

在ISO/IEC 15504的软件过程或过程能力参考模型中，所涉及的过程和实践、以及过程分类方法，与ISO/IEC 12207标准的规定一致，这两项标准作为ISO9000系列标准的支持标准，在软件质量管理方面明确规定，质量保证体系活动按照合同中规定的ISO9000-3的条款执行。因此，ISO/IEC 15504、12207和ISO 9000-3三者浑然一体，构成了规范、评估、管理、改进软件工程以及保障软件产品质量的完整严密的标准体

系。

3.1 ISO/IEC 12207标准

ISO/IEC 12207标准(信息技术——软件生存周期过程)是ISO组织于1995年8月1日发表的软件生存周期过程标准。该标准为计算机软件产品研制和管理事项提供了基本框架,为软件产业确立了一个软件生存周期过程的通用标准框架,说明需方在获得一个含软件的系统、一个单独的软件和一项软件服务时,以及供方在供给、开发、操作和维护软件产品时,所涉及的各种必要的过程、各过程包含的活动和各活动包含的任务;该标准还为软件组织规定了一个用于定义、控制和改进其软件生存周期过程的标准过程。

该标准的基本原则是使诸如软件开发与维护的活动按工程规范的方式进行。在这个原则下建立起一种与系统工程环境(软件、硬件、人与业务实践的总和)联动的框架。

12207过程可分为基本过程、保障过程与组织过程三大类,每一类又包含一些过程,三大类共定义了软件生存周期的17种过程。基本过程包括那些启动或进行软件产品的开发、操作或维护的过程;保障过程用于控制质量,并提供支持保障;组织过程包括与机构建设、人员培训等有关的管理过程。12207软件生存周期过程的结构如图4所示。

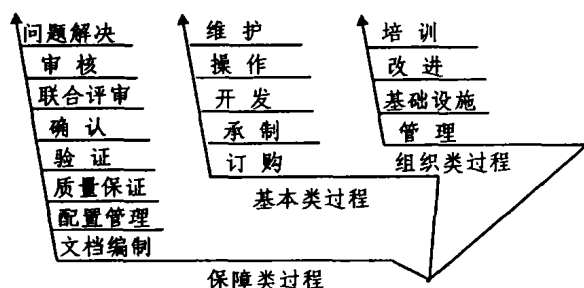


图4 ISO 12207软件生存周期过程

该标准详细规定了各过程的具体活动(总共73项)及每项活动的具体任务(总共233项)。每个过程由其相应的构成活动来定义,而每个活动又通过其组成的任务来定义,过程内的活动为一组内聚的任务集合。任务以需求、自声明、推荐或允许动作的形式表达。例如,开发过程包含下列13种活动:

(1)过程的实现;(2)系统需求分析;(3)系统体系结构设计;(4)软件需求分析;(5)软件体系结构设计;(6)软件详细设计;(7)软件编码和测试;(8)软件集成;(9)软件鉴定测试;(10)系统集成;(11)系统鉴定测试;(12)软件安装;(13)软件验收支持。

12207标准只描述软件生存周期过程的体系结构,而不规定过程中的活动和任务该如何实施。例如它规定一个项目应选定软件开发方法,确定具体的软件生存周期模型,但选用什么方法、确定用哪种生存周期模型,以及如何选用等都留给本标准的使用者自己去决定。

12207标准明确规定了组织的软件生存周期过程。在标准的第三大类过程即组织的软件生存周期过程中明确规定,一个组织应从软件项目中总结经验,建立组织的软件生存周期过程,作为该组织承担具体软件项目、建立具体软件生存周期过程的一个基础,并不断加以改进和提高。这比起ISO9000-3标准有所发展,并最终在15504标准中得以淋漓尽致地充分发挥和体现。

我国的国家标准GB8566—1995的内容与该标准的主要

内容大体相同,可以认为贯彻了GB8566—1995便基本执行了该标准,但在过程的划分以及许多过程的具体活动和任务的规定上有一定差别。

3.2 ISO 9000-3标准

国际标准化组织(ISO)自1987年推出ISO9000系列标准以来,为企业提供了一种科学的质量管理和质量保证机制,负责起草ISO9000的标准技术委员会ISO/TC 176把工业产品分为硬件、软件、流程性材料和服务四种类型,在给出了通用标准的基础上,针对每类产品的生产特点,分别制定了相应的补充和支持标准,ISO9000-3(ISO9001在软件开发、供应和维护中的使用指南)即为软件生产领域的补充性指南,最新版于1997年发表。

ISO9001标准的主要内容有:管理职责,质量体系,合同评审,设计控制,文件和资料控制,采购,顾客提供产品的控制,产品标识和可追溯性,过程控制,检验、测量和试验设备的控制,检验和试验状态,不合格品的控制,纠正和预防措施,搬运、储存、包装、防护和交付,质量记录的控制,内部质量审核,培训,服务和统计技术。

ISO9001确保每一个产品要素和相关服务的质量可重复地被保证。ISO9001针对合同环境下设计、开发、生产、服务等环节,给出了所需要的最基本质量要素。ISO9001根据一个企业的质量体系中是否覆盖了所有要求的质量要素(以文档化的形式),且这些要素是否有效地按定义方式实施来判断该企业是否符合ISO9001要求。

4 其他评估标准及比较

4.1 CMM 标准

SEI的CMM把软件开发能力由低到高划分为五个成熟度等级:(1)初始级;(2)可重复级;(3)已定义级;(4)已管理级;(5)优化级。每一等级都定义了相应的标准,开发的能力越强,开发组织的成熟度等级越高。

CMM描述了每个等级开发过程的基本特点,并以关键过程域KPA(Key Process Areas)明确规定了二级以上各个不同等级应解决的主要问题和形成的过程能力。每个KPA都有明确的目标(goals),而且都包含若干关键实践(key practices)。这些关键实践都是实施该关键过程域并加以制度化所必须开展的活动和必须具备的基础。关键实践包括对关键过程域的实践起关键作用的方针、规程、措施、活动以及相关基础设施的建立。这些关键实践分别组成KPA的五个属性,称为5个共同特征,即实施承诺CO(Commitment to Perform)、实施能力AB(Ability to Perform)、实施活动AC(Activities Performed)、测量和分析ME(Measurement and Analysis)、实施验证VE(Verifying Implementation)。

因此,CMM中的5个成熟度等级、18个KPA、5类公共特征和316个关键实践构成了一个层次化的结构,每个等级的过程能力都是其高一级的基础,形成了一个逐步升级的结构模型(如图5所示)。

CMM自提出之后,首先在美国国防部(DoD)和国家航天局(NASA)范围内得到认可,并逐渐在美国、欧洲、日本和印度等国家和地区得到广泛应用,中国国内有三家软件机构——鼎新公司于1999年7月、东大阿尔派公司于2000年10月、联想软件于2001年1月相继通过了第三方认证机构的CMM2级认证,近年来国内学者对CMM也加强了研究和应用。

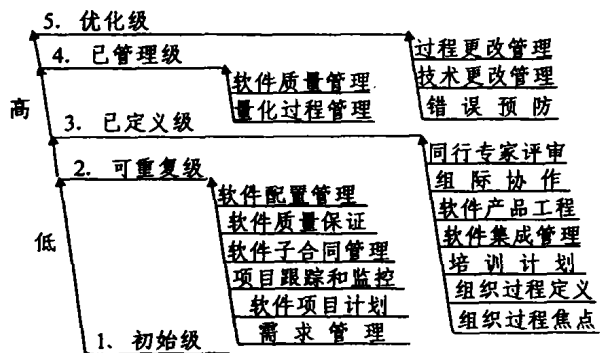


图5 CMM 的成熟度等级和各等级实施的关键过程域

CMM 的缺点之一是从起点级1级“初始级”到2级“可重复级”之间的 KPA(关键过程域)较多,台阶太高,跨度太大,实施周期长而见效慢,在一定程度上挫伤了软件机构的积极性;而 SPICE 模型要求软件机构从起点级0级“不完整级”开始首先建立基本的软件工程过程,树立质量意识,实现 PA 1.1 过程实施属性(Process Performance Attribute)要求后升到1级“可实施级”,然后按照 PA 2.1 实施管理属性(Performance Management Attribute)和 PA 2.2 产品管理属性(Work Product Management Attribute)的要求进一步规范和改进软件过程而达到2级“已管理级”,……软件能力等级划分更为合理,更有利于软件开发组织循序渐进地逐步提升软件能力等级。

CMM 的缺点之二是关于等级4“已管理级”和等级5“优化级”的阐述显得较为空泛,与等级2“可重复级”和等级3“已定义级”相比,有关的 KPA 及其所包含的关键实践定义得还不够完整和具体,增加了操作歧义和困难。这一问题在 ISO/IEC 15504 中有了较大的改善。

从逻辑修辞上看,ISO/IEC 15504 在软件能力等级、过程、实践等方面的定义和描述用语比 CMM 要更为准确和恰当。另外,SPICE 的结构层次划分也比 CMM 更为合理和完整。除此之外,CMM 中所涉及的软件生存周期过程描述与国际标准 ISO/IEC 12207 不太一致,也是 CMM 不利于国际化推广的一个缺点。

4.2 Bootstrap 和 Trillium 评估标准

除此而外,还有一些影响较为有限的评估标准,如欧洲 Bootstrap 和加拿大 Trillium 等。Bootstrap 标准主要为欧洲的某些组织所采用,Bootstrap 3.0 由 Bootstrap Institute 于 1995 年发表,其评估方法趋向于符合 ISO 15504 标准要求并借鉴 CMM 对过程能力和成熟度的定义,其特点是较为灵活实用,专门设计了适用于不同情况的三种方式:自我评估、完全评估和选择性评估。

(上接第82页)

需要指出的是,以上所给出的等价变换规则、扩充的代数规则及一些相应的性质,不仅可以用来在全局查询转换为中间查询的过程中进行代数优化,它们同样适用于中间查询向局部查询的转换,从而实现全局查询的优化处理。

参考文献

- Sheth A P, Larson J A. Federated Database System for Managing Distributed, Heterogeneous, and Autonomous Database, ACM

结束语 ISO/IEC 15504 吸收了 SEI CMM 等早期标准的优点,在完整性、合理性、可操作性等方面有较大改进和发展,是一个集大成的软件过程评估标准。ISO/IEC 15504 评估标准不仅可供用户(需方)对软件开发组织(供方)的开发能力及水平进行考察评估,同时也为开发机构提升软件过程能力水平提供了依据。许多软件开发组织多年的运用经验表明,SPICE 模型及相应的一套评估方法的确能帮助软件开发组织迅速改进软件开发过程,提高软件工程能力。统计数据说明随着软件开发过程的改进,所开发的软件产品中所含缺陷数明显减少;费用进度控制情况也随过程改进而改进。

在软件开发组织按有关标准对本组织的软件过程进行考察和改进时,可以使用美国 Rational 公司的 RUP、Intersolv 公司的 PVCS 和 CA 公司的 ADvantage 等软件开发管理工具来辅助和加强软件开发管理,营造一个规范而高效的管理机制,许多开发组织成功地采用上述工具而通过了 ISO 或 CMM 的评估认证。

可以预见随着 ISO/IEC 15504 标准的完成,SPICE 模型及其方法将为软件开发组织不断改进软件工程过程、提高过程能力和产品质量指明可行的科学途径。

参考文献

- International Organization for Standardisation (ISO). Information technology - Software process assessment: [Technical Report]. ISO/IEC TR 15504[S]:1998
- <http://www.software.org>: ISO 15504 (SPICE) Description [EB/OL], 2001. 8
- Rational Software Corporation. Assessing the Rational Unified Process against ISO/IEC 15504 [R], 2000
- International Organization for Standardisation (ISO). Information technology - Software Life Cycle Process, ISO/IEC 12207 [S]: 1995
- International Organization for Standardisation (ISO). Quality management and quality assurance standards Part 3: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software, ISO 9000-3 [S]: 1997
- Magnani G, Garro I. Software Process Improvement towards Business Improvement [J]. METHODS & TOOLS, ISSN 1023-4918, Fall 1999 (Volume 7 - number 3)
- Paulk Mark C. Capability Maturity Model for Software, Version 1.1 [S]. Carnegie Mellon University, Software Engineering Institute, CMU/SEI-93-TR-25, Feb. 1993
- <http://www.sei.cmu.edu>: Capability Maturity Model (SW-CMM) for Software [EB/OL], 2001. 8
- 何新贵. 软件能力成熟度模型 CMM 的框架与内容 [J]. 计算机应用, 2001 (3)
- Herron D, Garmus D. Estimating Software Earlier and More Accurately [J]. METHODS & TOOLS, ISSN 1023-4918, Fall 1999 (Volume 7 - number 3)

Computing Surveys, 1990, 22(3): 183~236

- Chen M S, Yu P S. Using combining join and semijoin operations for distributed query processing. IEEE Trans. Knowledge and Data Engineering, 1993, 5(3): 534~542
- Lee C, Chen C J. Query Optimization in Multidatabase Systems Considering Schema Conflicts. IEEE Trans. on Knowledge and Data Eng., 1997, 9(6): 941~955
- Zhu Q, Larson P. -A. Solving local cost estimation problem for global query optimization in multidatabase systems. Distributed and Parallel Databases, 1998, 6(4): 373~420