

软件过程与质量焦点的分析

谢安俊

(温州大学城市学院, 浙江温州 325035)

摘要: 软件产业正在走向工业化生产时代, 软件工业化生产是以标准、规范为基础, 以软件质量为保证. 在分析和比较了软件能力成熟度模型 (Capability Maturity Model, CMM)、集成软件能力成熟度模型 (Capability Maturity Model Integration, CMMI) 与 ISO (International Standard Organization) 9000 质量体系的基础上, 结合以极限编程 (Extreme Programming, XP) 为代表的敏捷过程、统一软件过程 (Rational Unified Process, RUP) 及微软过程 (Microsoft Process, MP) 的特点, 对不同流派的软件过程进行了分析和比较, 重点讨论了合理的软件过程是软件质量的关注点.

关键词: 软件质量; 软件过程; 统一软件过程; ISO 9000; 极限编程; 微软过程

中图分类号: TP311.5 **文献标识码:** A **文章编号:** 1006-0375(2007)04-0028-07

当前, 计算机技术的处理能力、可用性快速增长, 使得软件应用的领域越来越广, 软件的规模越来越大, 开发的难度也越来越高, 对软件企业、软件质量以及开发、维护人员都产生了重大的影响. 软件复用和软件构件技术作为未来软件开发的发展方向, 已引起软件产业的深刻变革, 软件设计与生产工厂化和软件工程项目外包日益普遍, 互联网的应用使得大型软件的分布式开发更为有利, 软件产品的国际化程度将不断提高, 促使软件产业全球化发展趋势. 但是传统软件工程方法已不能很好地解决当前软件开发中的问题, 只是为解决软件危机提供了基础, 还不足以从根本上解决软件危机, 我们更需要好的需求去理解技术和管理方法. 建立稳定的、可持续改进的软件过程, 才能充分利用这些新的、有效的技术和方法, 从根本上解决软件危机问题. “好的方法永远无法拯救坏的管理过程, 而好的管理过程可以在多数技术方法中获得成功”^[1].

1 软件工业化生产要以软件过程的合理性来保证软件质量

如今, 软件已经从学术和科研体系中走出来, 正在成为制造业和服务业的一个部分, 成为一个整体产品的一个组成部分, 渗入社会生活的每一个角落. 数字电视、数码相机、汽车、手机等产品中都安装了嵌入式软件, 未来的软件应用范围 and 市场规模还将进一步扩展, 这对软件的质量、成本、完成时间都提出了更高的要求. 软件产业也就是进行软件产品开发、生产、销售、配套和信息技术服务的技术产业, 是信息产业的核心和灵魂, 已发展为独立的产业, 是 21 世纪全球主导产业. 全球竞争和客户对于更高软件质量的需求正推动软件企业进行软件过程的改进活动. 这些活动包括:

- (1) 要求对企业现有的开发软件的质量进行评估;

收稿日期: 2007-01-12

作者简介: 谢安俊(1949-), 男, 重庆人, 教授级高级工程师, 硕士, 研究方向: 软件工程, 电子商务

(2) 要使过程改进的目标作为软件企业长期总体目标的一部分;

(3) 以达到把服务质量和软件生产率同时提高^[2]。

现代质量理论认为:“质量形成于过程”。所以软件过程的管理,也是软件质量控制中非常重要的环节^[3]。但软件的质量标准比其他产品具有更多的复杂性,标准的量化、测试结果的取得、标准值范围等等。将质量的度量分为产品质量度量和过程质量度量两大类,通常产品质量度量依赖于具体的产品标准,通过测量获得产品质量特性,以合适的统计技术以确定产品或同批产品是否满足规定的质量要求。但软件产品质量标准的量化是一个世界性难题,标准不全面,难于量化。为了解决这个问题,人们可以利用另外的办法来弥补这个问题:从过程上解决,软件过程质量的度量目前已取得明显成果,一些实用的统计技术,如规定的 Bug 发生率、千行误码率等等,已经得到比较广泛的应用。由于过程不可见,对软件质量过程的控制远不如制造业成熟。

企业在开发产品的过程中要根据自身实际情况去应用各种量化的质量保证手段,保证企业开发过程符合标准、有能力保证产品的质量(企业是成熟的,过程的合理性保证产品质量)。

软件过程成熟度表示软件企业内部对软件过程的定义、管理、度量和控制的完善程度和有效程度。软件过程成熟度意味着产生于软件企业的软件过程的生产率,质量可以加以控制和不断提高。随着软件过程的成熟,又不断对文档化的软件过程,标准和软件组织进行更新,使其制度化。软件过程是一个持续改进的过程。

2 重量级软件过程——CMM/CMMI 与 ISO 9000 作为软件质量管理标准

由于软件产业已经或正在经历着“软件过程的成熟化”,并“向软件工业化”渐进过渡,所以规范的软件过程是软件工业化的必要条件^[3]。

目前软件过程的主要流派有三个典型的软件质量管理标准有三大流派:一个是美国 Carnegie Mellon 大学软件工程研究所(CMU/SEI)的 CMM(Capability Maturity Model) / PSP(Personal Software Process) / TSP(Team Software Process),它以过程能力为核心;一个是 ISO 9000 质量标准体系,发展到最新的是 2000 版,此体系一开始时以产品控制为核心,后期发展到过程控制,同时,从强调质量保证到质量管理;第三个是 ISO / IEC 15504 (SPICE),这一标准的目的和思路,均与 CMU / SEI 的 CMM 相似,是 ISO 吸收 CMM 思想颁布的标准^[2]。因此,目前在软件企业中采用最多的软件过程或软件质量管理标准。是 CMM/PSP/TSP 和 ISO 9000。尽管以 XP(Extreme Programming)为代表的“敏捷”(Agile)方法作为一种新型的、非常有特点的“轻量级软件开发过程”^[3],体现了通过组织和挖掘个人的丰富经验以实现轻量级软件过程的高质量的软件开发,但只是作为一种软件开发方法提出来的,并且定义了一些原则,但是它还不是公认的软件质量管理标准和主流的软件过程。

2.1 软件能力成熟度模型——CMM 的核心思想

CMM 是专门为软件开发而确定的质量管理规范体系,体现了质量管理思想。CMM 用途广泛,结构严谨,功能强大。CMM 只是规定成熟的软件组织应该达到的关键能力,是一种改进软件过程的策略,是专门针对软件开发而确定的质量管理规范体系,体现了质量管理思想。要求成熟度级别的上升和五级定义的缺陷预防等,其目标是通过改进软件开发过程,实现高效率、高质量、低成本的软件开发,是一个庞大、完整、严密的软件开发过程,已经成为国际认可的、预测企业生成的软件质量的重要标准^[4]。

CMM 是一种软件过程控制和评估框架, 它列出了每个级别需要完成的目标以及判定条件, 但并没有叙述如何实现这些目标. 于是, 卡内基梅隆大学 (CMU) 1989 年开始研究并于 1994 年提出个体软件过程 PSP (Personal Software Process), 1994 年开始研究并于 1998 年由 CMU/SEI 召开的过程工程年会上第一次介绍了群体软件过程 TSP (Team Software Process) 草案, 于 1999 年发表了有关 TSP 的一本书, 使软件过程框架形成一个包含 CMM、PSP 和 TSP 三者的严密的整体 CMM/PSP/TSP, 形成了 CMM/PSP/TSP 体系.

CMM/PSP/TSP 为软件产业提供了一个集成化的、三维的软件过程改革框架能具体指导软件企业如何进行高质量的开发. 应着重指出, 单纯实施能力成熟度模型 CMM, 永远不能真正做到能力成熟度的升级, 而需要将实施 CMM 与实施 PSP 和实施 TSP 有机地结合起来, 才能达到软件过程持续改善的效果^[1].

CMMI (Capability Maturity Model Integratation) 是 SEI 于 2000 年发布的 CMM 的新版本. CMMI 不但包括了软件开发过程改进, 还包含系统集成、软硬件采购等方面的过程改进内容. CMMI 纠正了 CMM 存在的一些缺点, 使其更加适用企业的过程改进实施. CMMI 适用 SCAMPI 评估方法. 需要注意的是, SEI 没有废除 CMM 模型, 只是停止了 CMM 评估方法.

2.2 ISO 9000 质量标准体系

ISO 9000 是典型的质量体系, 关注的全部就在于产品和过程的质量标准. ISO 9000 是国际标准化组织 ISO 在研究了英国、法国、德国、荷兰、加拿大以及美国的量管理标准的基础上, 于 1987 年 3 月公布的质量管理和质量保证标准系列. 从上世纪末欧洲联合会积极促进软件质量的制度化, 制定了 ISO 9000 有关软件标准系列: ISO 9001、ISO 9000-3、ISO 9004-2、ISO 9004-4、ISO 9002. 这一系列现已成为全球软件质量标准^[2], 欧洲通过 ISO 9000 认证的企业数量最多, 约占全世界的一半以上. 受此影响, 相当多的欧洲软件企业选择了 ISO 9000 认证. 软件企业贯彻 ISO 9000 质量标准, 应当选择质量保证模式标准 ISO 9001-2000, 它描述了 20 个质量体系要素, 在建立质量体系时可以适当裁减, 并在质量体系审核时受到验证^[4].

ISO 9000 标准的一个重要科学依据就是“质量形成于全生产过程中”, 具有事前计划、事中严格按计划实施、事后检查、总结分析并采取改进措施的循环模式上. 确保影响软件产品质量的全部因素在生产过程中始终处于受控状态. 不论 ISO 9000 还是 CMM 都强调过程、活动之间的关系, 尽管 ISO 9000 和 CMM/CMMI 对过程 Process 一词所涵盖的范围有所不同, 但他们的本质是一样的. 质量改进是现代质量管理的必然要求, ISO 9000 要求组织定期进行内审和管理评审, 采取有效的纠正预防措施, 保持组织的质量方针和目标持续适合组织的发展. 前面我们定义的组织支持过程中, 内部质量审核和管理评审, 即是质量改进而定进行, 改进过程是重要的质量过程.

2.3 CMM/PSP/TSP 与 ISO 9000 质量标准体系的比较^[5]

美国 Carnegie Mellon 大学软件工程研究所 (CMU/SEI) 的 CMM 是当前最好的软件过程, 已成业界公认的软件过程的工业标准. 而 ISO 9000 侧重于保证生成的规范和质量问题可以追踪, 最新的 ISO 9000 标准已经关注缺陷预防、过程改进等内容, 体现了质量保证思想, 明确提出了过程方法和管理系统方法. 尤其是 ISO 9000:2000 版标准增加了持续改进、质量目标的量化等方面的要求后, 在基本思路上和 CMM 更加接近. 一个符合 ISO 9001 的机构, 处于 CMM 第 1 级的机构的确有可能通过 ISO 9001 的认证; 同时, 该机构又可能具有很强的第 2 级的过程实力和明显的第 3 级的过程实力.

从软件业的质量体系角度看 CMM 中，虽然有的问题描述得还不够充分，但大体上包容了 ISO 9001；而 ISO 9001 却不能包容 CMM。

3 统一软件过程（Rational Unified Process, RUP）

Rational 公司（现已并入 IBM 公司）的 RUP（Rational Unified Process）结合了 UML 语言和面向对象技术提出了统一软件开发过程框架，其原则是以软件是以 Use Case 驱动、以基本框架为核心，强调增量和迭代开发过程。目标是帮助企业有效利用资源，并进行协调，从而更好地进行软件开发。

RUP 把一个项目分成 9 个核心工作流程（Core Workflows）和 4 个阶段 Phases），并以核心工作流程为 Y 轴，阶段为 X 轴建立起一个项目视图^[6]，如图 1 所示。

RUP 把开发中面向过程的方面（例如定义的阶段，技术和实践）和其他开发的组件（例如文档，模型，手册以及代码等等）整合在一个统一的框架内。Rational 是提供基于业界开放标准的工具、最佳方案和服务，用于开发商业应用和构建软件产品及系统，为开发人员和开发团队提供了整个开发生命周期的支持。单独使用时，软件的每种工具在其各自市场领域中都处于领先地位。

RUP 还帮助实施 CMM/CMMI 提供了所需要

了主要工具：软件开发过程框架、需求管理工具、面向对象的分析设计工具、配置管理工具、变更管理工具、项目管理工具和软件测试工具。RUP 及其配套软件工具也是重量级的软件研发管理解决方案，它面向的是高端用户，费用相当高，对用户的财力、开发和管理能力要求都很高。

4 敏捷（Agile）过程——轻量级过程^[1,3]

对软件质量的控制，衡量其效果的标准，归根结底有两点：效率和质量。CMM 与 ISO 9000 以复杂的质量保证过程为关注点，通过复杂的辅助和管理过程来保证质量。这样的方法的成功取决于管理水平，只适合大型软件企业。而 CMM/CMMI 基本上不谈“成本管理”和“人力资源管理”，它先假设机构有充足的资金和人力资源，通常不切合企业实际情况。由于软件业是一个高速变化、新技术层出不穷的行业，同时又是人力资源成本相对较高的行业。

为适应 Internet 时代“按需应变”的软件开发环境，解决重型软件开发过程所遇到的问题，实现对软件需求变化迅速地做出反应，利用可伸缩的业务流程，适应易变的成本结构，开发并同其他系统互操作的系统等目标，一种所谓的“轻量型”或“灵活”的软件过程模型和敏捷方法诞

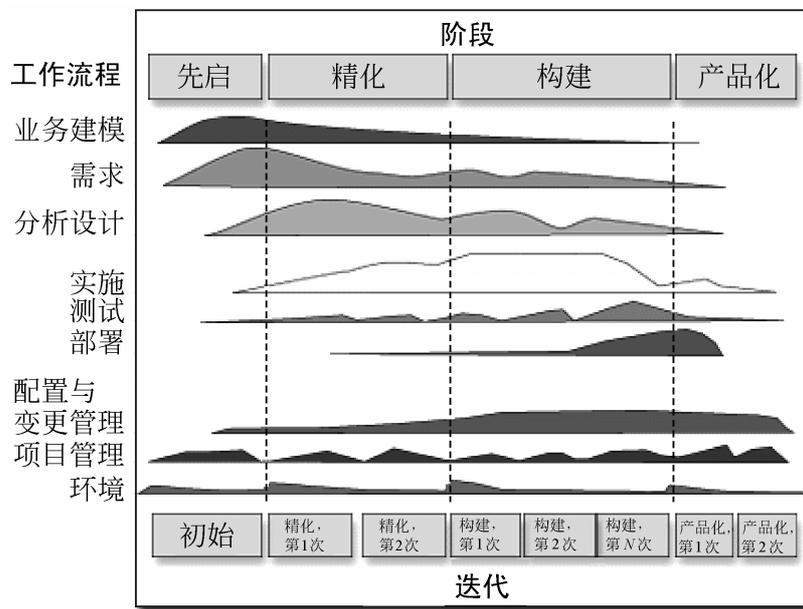


图 1 RUP 软件开发模型

生了, 将面向对象语言的威力和灵活性结合起来, 提供了一种很有特点的软件开发解决方案. 敏捷强调适应变化而非预测; 软件开发要以人为中心, 而不是以过程为中心. 敏捷式开发在质量和效率上有很大的提高, 比如整体的缺陷低 13%, 核心成本降低 62%, 开发时间降低 69%. 其特点如下: (1) 依赖人、人与人的交流而不是过程和工具; (2) 生产可用的软件而不是大量高质量的文档; (3) 与客户协作而不是合同交易; (4) 适应变化而不是紧抱计划不放^[3].

与 CMM 与 ISO 9000 相比, 敏捷过程在过程管理方面开销较低, 减弱了过程对开发人员的约束力. 由程序员驱动, 而不是由管理层和质量部门驱动. 敏捷开发最重要的特点就是“以人为本”, 而不是“以方法为本”. 软件开发是高智力脑力的劳动, 而不是简单的工作. 它把质量与效率的关注点直接放在软件过程中真正产生价值的开发过程上. 同时, 通过加强开发过程本身, 保证产品的质量, 降低辅助过程的开销.

敏捷方法的策略是“用正确的方法做事, 而不是将事情做正确”, 而不是为了遵循某些标准和模型; 采用更好方法和依靠“天才”的人. 在人力资源管理方面的强大投入, 包括招聘高素质的项目管理人员, 让软件项目管理人员具有专业的技能, 并把握顶尖技术人才, 让程序员发挥他们的专长编写代码, 建立有力合理的组织结构. 基于中小型小组(6-20人)同客户紧密合作, 放弃精确、详细、预先的规划, 而通过快速迭代来给客户提供了当、真正的结果, 实现“以客户为中心”的理念.

目前以 XP 为代表的敏捷方法已得到业内专家的积极评价和认同, 已形成包括有以下以不同背景组成的“敏捷联盟”: XP (Extreme Programming, 极限编程); SCURUM (迭代的增量化过程); Crystal Methods (水晶方法族开发过程); ASD (Adaptive Software Development, 自适应、超轻型、可伸缩软件开发过程) 和 FDD (Feature Driven Development, 基于特征驱动开发过程).

目前敏捷方法也以惊人的速度不断夺取传统方法的市场份额. 某著名的咨询机构2002年对全球相关行业IT经理进行的调查表明: CMM/CMMI使用率达到27%; RUP使用率达到51%; ISO 9000使用率达到26%.

今后, 大约50%以上的项目会使用敏捷 (Agile) 式开发方法——轻量级过程 (Lightweight Process), 14%调查者认为所有项目会使用敏捷 (Agile) 开发方法^[3].

5 微软过程 (Microsoft Process, MP)

微软过程是该公司根据自身经验为企业设计的一套有关软件开发的准则. 微软过程是建立在微软解决方案框架 (Microsoft Solution Framework, MSF) 的基础上^[6], 提供了从软件项目组织规划到产品发布管理的一套全方位、高效、灵活、可扩展的软件开发管理体系. 微软过程分为构想、计划、开发、稳定和发布五个阶段, 各阶段都有一个主里程碑 (见表1). 而且每个阶段都涉及到产品管理、程序管理、开发、测试、用户体验、发布管理的角色和活动内容 (图2).

微软过程以体系结构为中心, 面向过程, 提倡团队协作, 强调风险管理, 提供一系列用于关键可交付产品的模板, 可以根据现有的开发风格来定制其实现模型. 微软过程没有CMM/CMMI, ISO 9000那样完整、全面、细致的标准. 但很实用、工程性强, 以不追求理论上的完美, 而追求效率和结果为目的. 以角色为中心分工负责 (见表2), 成员关系平等, 突破传统软件开发的组织形式, 按民主平等组织原则, 这是软件智力劳动的特点, 充分发挥参与者的解决性, 有利于员工快速成长和成熟.

表 1 微软过程的说明

阶段	说明	里程碑
构想	和客户一道了解 and 解决方案，确定系统总目标	认可的预想
计划	制定项目规划，描述任务、依存性以及完成解决方案所需资源	认可的项目规划
开发	开发、编码和测试应用程序	完成范围内产品特性
稳定化	最后验收和提供给客户的发布版本	可发布版本准备就绪
发布	采用递进的版本发布策略(见图 3)	发布完成

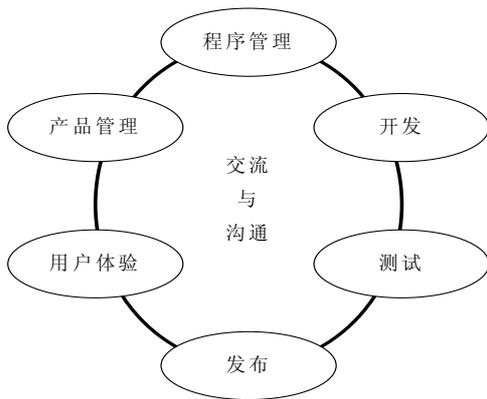


图 2 微软过程项目结构

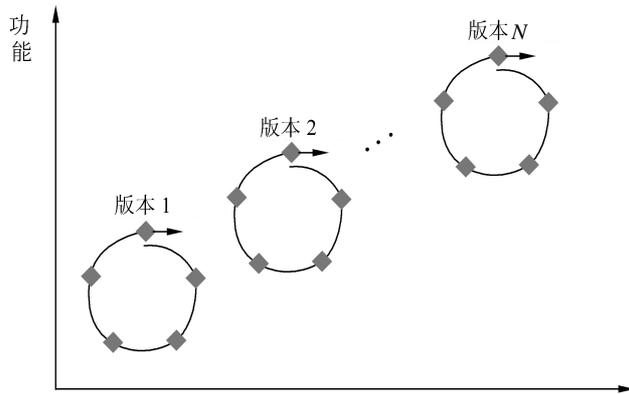


图 3 微软过程的生命周期模型

微软过程、敏捷过程、统一过程三者的相互关系：统一过程作为一种大而全的软件过程可实用各种不同环境类型的项目开发，敏捷过程作为针对商业环境中具有有限资源和时间进度限制的小型项目提出一种软件过程模式，对 RUP 进行了多方面的有益补充和完善；微软过程作为另一种针对资源有限和时间限制的项目的软件过程模式^[6]，综合了 RUP 与敏捷过程诸多优点，并且进一步在人员及其组织、过程中的方法等方面提出了独具特色的、操作很强的实践规范，是一套优秀的成功项目开发实践经验总结。

表 2 微软过程的团队的不同角色

团队角色	目标
产品管理	让客户满意
程序管理	在项目约束条件下交付
开发	按产品规格交付
测试	解决所有已知问题发布
用户教育	提高用户胜任力
发布管理	管理产品部署

6 结束语

CMM/CMMI和ISO 9000有着明显的倾向，即“管理的规范化”重于“开发的规范化”。它们规定了严格的管理制度、文档和评估软件能力、成熟度等级的一套标准，几乎包括所有的软件企业^[7]。在企业里，大部分的工作是成熟的，有成功模式可以借鉴，应当走规范化的路线。对于软件过程改进而言，CMM/CMMI和ISO 9000等都是用来参考的，而不能迷信它作为“万能的方法和过程”。

敏捷开发方法对于提高个人、小型团队的工作效率是很有帮助的，但是企图用它指导大型、

中型软件机构的研发管理是有很高风险的,它的某些主张是局部观点而不是全局观点,如果把握不好分寸的话可能导致整体混乱,而“整体的混乱”会淹没“局部的好处”。微软过程是一套较完整的软件过程模式,是一套操作性很强的成功经验总结,但国内有关文献介绍不多,是一套优秀的成功项目开发实践经验总结,应值得推广。

软件质量的提升,更广义地说是实现高效率、高质量的软件开发,这是人们始终追求目标。无论是标准也好,一些观点、方法也好,都是我们达到目标的手段。在选用过程和标准时,一定要注意合理性,注重分析标准的结构、标准的体系、方法的思想。除此之外,还需要建立以质量为核心的软件工程文化,软件组织才能够保证自身绩效的不断提高^[8]。彻底地理解和选择适合的标准和方法,通过实践,提高软件开发质量和开发效率才是最重要的。软件企业只有结合具体项目的实际情况灵活应用,才能达到良好的效果。

参考文献

- [1] 龚波,刘宝军,刘卫宏,等.软件过程管理[M].北京:中国水利水电出版社,2003.44-214.
- [2] 刘寅琥.系统分析与软件开发过程管理实用案例教程[M].北京:清华大学出版社,2003.184-200.
- [3] 钱岭.敏捷软件开发方法综述[A].见:刘江.Dr. Dobb's 软件开发[C].北京:机械工业出版社,2003.34-44.
- [4] “推进我国软件企业 ISO 9000 质量体系认证的研究”课程组.软件企业 ISO 9000 质量体系的建立和认证[M].北京:清华大学出版社,1999.10-132.
- [5] 王青,刘怀璋,李明树.软件质量管理——标准、技术与实践[M].北京:中国计划出版社,2002.63-133.
- [6] 金敏,周翔.高级软件开发过程[M].北京:清华大学出版社,2005.44-93.
- [7] 赵池龙,姜义平,张建.软件工程实践教程[M].北京:电子工业出版社,2007.352-355.
- [8] 林锐,范同祥.面向企业的软件研发管理解决方案[M].北京:电子工业出版社,2004.57-140.

Analysis of Software Process and Quality Focus

XIE Anjun

(School of City, Wenzhou University, Wenzhou, China 325035)

Abstract: Software industry is developing to the times of the industry production. It is based on standard and the software quality as guarantee. In this paper, after analyzing and comparing CMM (Capability Maturity Model), CMMI (Capability Maturity Model Integration), ISO (International Standard Organization) 9000, Rational Unified Process, Agile Process and Microsoft Process, we will discuss the Software Quality focus.

Key words: Software Quality; Software Process; ISO 9000; RUP; AP; MP

(编辑: 赵肖为)