

软件质量保证应用软件自动化技术研究

杨勋姮, 段明璐

(中国电子科技集团公司第十五研究所软件测评中心, 北京 100083)

✉yangxunheng116@163.com; 693407919@qq.com



摘要: 针对软件质量保证(Software Quality Assurance, SQA)技术在实际应用过程中任务量大、方法效率低下、容易受主观判断干扰等问题, 探讨了质量管理体系(Quality Management System, QMS)和能力成熟度模型集成(Capability Maturity Model Integration, CMMI)的概念, 对软件质量保证技术进行研究, 包括传统的软件质量保证技术PDCA循环以及七种统计技术; 设计了辅助实施质量保证(Quality Assurance)的自动化软件管理工具, 分析了软件质量保证应用的现状, 总结了质量管理工具的应用要点。结果表明, 质量保证自动化技术的应用研究对改善软件质量、提高工作效率具有实践价值。

关键词: 质量保证; 质量管理体系; 软件质量管理工具

中图分类号: TP311.56 **文献标识码:** A

Research on Automation Technology of Software Quality Assurance Application

YANG Xunheng, DUAN Minglu

(Department of Software Testing Center, the 15th Research Institute of
China Electronics Technology Group Corporation, Beijing 100083, China)

✉yangxunheng116@163.com; 693407919@qq.com

Abstract: In the application of Software Quality Assurance (SQA) technology, there are problems of large numbers of tasks, low efficiency and being easily disturbed by subjective judgment. This paper proposes to discuss the concept of Quality Management System (QMS) and Capability Maturity Model Integration (CMMI), and do research on Software Quality Assurance technology which includes typical Software Quality Assurance technology PDCA Cycle and seven statistical techniques. Automated software management tools are designed to assist in implementing Quality Assurance (QA). Finally, the application status of Software Quality Assurance is analyzed, and the application points of quality management tools are summarized. Experiment results show that the application research of quality assurance automation technology has practical value to improve software quality and work efficiency.

Keywords: quality assurance; quality management system; software quality management tools

1 引言(Introduction)

计算机技术的迅猛发展促进了计算机应用的日新月异, 尤其在武器装备系统中, 软件所占比重逐年提高, 因此对软件程序的质量提出了更高的要求。美国政府机关的数据曾显示, 美国每年由于软件质量问题造成的经济损失高达600亿美元。

近十年来, 基于CMMI的软件工程化被广泛应用于中国各大企事业单位。实施CMMI的本质是帮助组织有效地改善软件工程管理的核心问题, 进而提高软件质量^[1]。软件质量保证作为确保软件项目管理实施规范化的一种技术, 其监督对象

是在组织级定义的项目层面的过程活动和工作产品。

本文明确了质量保证在质量管理中的作用, 探讨了依据国际标准和军用标准建立质量管理体系中的软件质量保证的实施过程及技术方法, 以及在运用软件质量保证进行项目监督评价时所使用的自动化工具的设计实现。

2 软件质量保证的相关概念(Related concepts of SQA)

任何组织都需要管理, 当管理的对象是产品的质量时, 则为质量管理。质量管理是关于质量的一切管理活动, 例如确定质量方针、质量目标, 建立质量管理体系, 具体包括

质量策划、质量控制、质量保证和质量改进。20世纪60年代初,美国的A.V.费根鲍姆提出了全面质量管理TQM(Total Quality Management)的概念,将企业各部门在研制质量、维持质量和提高质量方面的活动构成一种有效体系。中国自20世纪70年代末开始推行全面质量管理,并取得了一定成效。

国际标准和国家标准对质量管理是这样定义的:质量管理是在质量方面指挥和控制组织的协调活动。《装备质量管理术语》(GJB 1405A—2006)中规定了装备科研生产使用中涉及的装备质量管理术语^[2]。

质量管理体系作为在质量方面指挥和控制组织的管理体系,是组织内部建立的、系统的质量管理模式,为实现质量目标,将资源与过程管理方法相结合,涵盖了确定需求、生产研制、检验交付等全过程的策划、实施、监控与改进活动。质量保证是质量管理的一部分,按照质量管理体系的规定实施有计划的系统活动,以保证产品质量为基础,使用户确信产品或服务满足规定的质量要求,致力于获得用户的信任。

2.1 质量管理体系的概念

20世纪60年代,美军制定了MIL-Q-8958A等系列军用质量管理标准,在标准MIL-Q-9858A中提出了“质量保证”的概念。20世纪70年代末,英国将军方在制定合同时使用的质量保证方法引入市场环境,制定了英国国家质量管理标准BS 5750。20世纪80年代末,国际标准化组织制定的国际质量管理标准ISO 9000系列开始在世界范围内推行质量管理与质量保证,该系列标准适用于不同类型产品与性质的组织。

质量保证在ISO 9000:2005中被定义为质量管理的一部分,目的在于通过进行严格的试验来获取顾客信任,满足最终顾客使用的环境、寿命以及产品的相关标准要求。质量保证关注的是与质量活动相关的政策、制度、流程、规范,通过质量保证人员在软件全生命周期的质量评价活动,确保软件项目的实施符合相关标准、制度和规范。

20世纪80年代,美国卡耐基梅隆大学软件工程研究所(Software Engineering Institute, SEI)组织软件管理方面的专家研发了一种软件能力成熟度评估标准CMM(Capability Maturity Model),用于指导软件开发过程的改进和软件开发能力的评估。CMMI是CMM模型的更新版本。2018年,CMMI研究院正式发布了CMMI 2.0中文版,该版本是针对软件、产品和系统的优良实践发布的过程改进模型,致力于利用当前的优良实践,改善业务绩效,以业务绩效为焦点,为使用Scrum的敏捷项目过程提供直接指导。中国人民解放军总装备部参考CMMI最新版本也开始进行GJB 5000A的改版工作,并将在2022年底完成军工企事业单位依据GJB 5000B的体系换版工作。

CMMI为组织提供了一个单一的集成框架,一共包括五个级别,不同的级别包含的过程域也不尽相同。所有过程域分为四大类:过程管理类、项目管理类、工程管理类、支持管理类。支持管理类中包含“过程与产品质量保证”过程域,该过程域为项目管理层提供了项目过程和相关工作产品的客观信息,如图1所示。

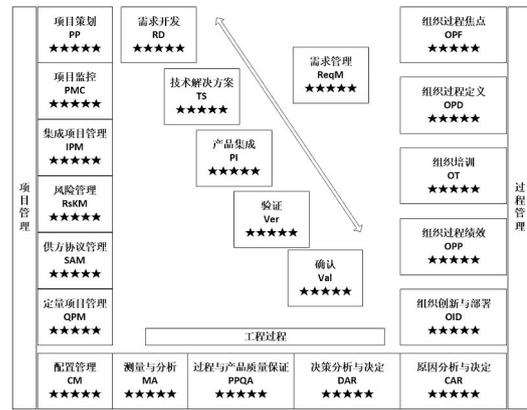


图1 成熟度等级的过程域分类

Fig.1 The process area classification of maturity level

软件质量保证属于支持管理类过程域中的过程和产品质量保证,用以在项目整个生命周期提供对所有过程域中所描述的过程和工作产品的客观评价,并确保这些评价提出的所有问题得到解决,从而确保交付高质量的产品和服务。

2.2 软件质量保证的应用现状

随着移动互联网的蓬勃发展以及云计算、大数据技术的更新换代,软件开发模式也正在发生着适应性变革。敏捷开发作为一种灵活的开发方式正被越来越多的企业所采纳,其面向用户预期的迭代开发方法使软件从需求分析到设计完成的周期变短,更适用于需求变动频繁、快速应对风险、交付周期短的项目。但这种开发模式也使软件质量管理工作的难度更大,更难以建立行之有效的软件质量管理体系。CMMI 2.0则可以辅助使用诸如Scrum这样的敏捷模式的企业更好地实现其管理价值,尤其能够在跨项目的组织层面实现敏捷的规模化,提升组织的敏捷开发能力。

一个组织的质量管理体系通常是国家标准以及军用标准的本地化,在遵守法律法规要求的前提下,所制定规范的适宜性对企业来说至关重要。而对于军用软件,一个完备的武器装备系统可能由多个承制单位合作研发,项目管理难度普遍较大,而其涉及的软件质量关系到国家安全,复杂的管理过程更需要适宜的质量保证技术和高效的软件管理工具的辅助。

国家标准要求研制单位必须采用评审、审计和测试等具体手段来控制质量。在软件开发的每个阶段结束时,都应组织正式的技术评审,即由软件技术人员按照软件需求规格说明和设计文档,对软件产品进行严格的评审。大多数情况下,评审和测试能有效地发现文档以及软件中的缺陷和错误,但这两种技术方法关注的是产品本身的质量,不包括研制过程的质量,而软件质量保证则可以通过监督技术评审等过程的规范性来保证评审的有效性。

软件质量保证的应用需要一个统一的认知环境和价值观,对于企业来说,适宜有效的质量管理体系下的软件质量保证活动可以发挥杠杆作用,帮助软件研发工作提高协调性,提高生产率,改善产品质量,提升竞争优势。

3 软件质量保证的技术方法(Technical method of SQA)

软件质量保证是一套有计划的、系统的技术方法,其所

制定的标准、步骤、规程和方法应该保证能够正确地被客户所采用。软件质量保证是对软件研制过程的一种管理，以独立于项目组的角色履行独立审查职责。通过对软件产品和过程活动的评审和评价来验证软件研制过程是否正确遵循已制定的计划、标准和规程，同时为开发人员和管理层提供反映产品和过程质量的信息和数据，辅助项目取得高质量的软件产品。

3.1 软件质量保证的传统技术

全面质量控制(Total Quality Control, TQC)理论起源于美国，20世纪60年代，为了在一定经济条件下生产尽可能高质量的产品，朱兰、费根堡姆提出质量活动策划应该覆盖所有职能部门。该理论在发达工业国家开始推行，尤其在日本应用广泛，并于20世纪80年代逐渐发展为全面质量管理的概念。

全面质量管理的基本方法是PDCA(Plan-Do-Check-Act)循环，又称戴明环。PDCA循环将质量管理分为四个阶段，即计划(Plan)、执行(Do)、检查(Check)、处理(Act)。该工作方法是企业管理各项活动的一般规律，软件质量保证实施的一般步骤也基本遵从该管理方法。

在执行和检查阶段，为了更好地分析问题，全面质量管理可使用14种质量管理工具。其中，被称为“老七种”的方法运用了统计技术，包括“因果图”“流程图”“直方图”“检查单”“散点图”“排列图”“控制图”，这些方法普遍用于质量改进，其概念如下：

(1)因果图(Fish Bone Diagram)：也称鱼骨图，它的表示方法与鱼骨的形态类似，鱼骨代表问题、缺陷或后果，鱼骨上的鱼刺代表产生问题可能的原因。因果图能够表现各个原因之间的关系。在问题处理过程中，可使用因果图对产生问题的原因进行分析，便于在日后组织层面进行过程改进时，针对某类问题制定纠正措施，对质量管理体系的相关内容进行调整。

(2)流程图(Flow Chart)：主要描述一个工作任务的具体步骤以及输入输出，有助于了解任务的运行结构并用于跟踪。

(3)直方图(Histogram)：用于表示某类事物测量结果的实际分布，可通过图形的分布形状判断事物的整体情况。直方图可用于呈现里程碑节点处一段时间内问题的处理情况，使质量管理人员直观清晰地了解项目现阶段的整体情况，预测未来的质量趋势。

(4)检查单(Checklist)：是预先准备用于记录目标对象行为特征的表格，表格中按行描述特定的“行”为特征，并在每个行为特征后设置描述符合性的“列”，通常可用预置的值表示，如“是”“否”或“不适用”，抑或描述整体情况的“好”“中”“差”^[3]。检查单是软件质量保证人员实施评价所使用的主要技术方法，不同的软件工作产品应使用组织检查单汇总表中的相应的检查单，且可对具体的检查内容进行裁减。

(5)散点图(Scatter Plot)：图形表示为数据点在直角坐标系上的分布，可用于观察随自变量变化的因变量的趋势，数据点可以是科学数据、统计数据或工程数据，主要用于回归分析。

(6)排列图(Pareto Chart)：也称帕累托图，用于统计和显示一定时间内各种类型缺陷或问题的数目，有助于确定造成大多数问题的关键原因，也可以用于分析生产过程中最可能产生某些缺陷的部分。

(7)控制图(Control Chart)：是用于分析和判断过程是否处于稳定状态所使用的带有控制界限的图，可区分正常波动和异常波动，是对过程质量特性进行测定、记录、评估，从而监察过程是否处于控制状态的一种图^[4]。

3.2 软件质量保证的实施过程

软件质量保证人员在项目一开始时就应该与软件项目经理共同建立计划、标准和过程。软件质量保证人员的工作贯穿于整个软件生命周期的全过程活动中，关注过程和中间产品，而不仅仅是最终产品^[5]。软件质量保证实施的一般步骤为：

(1)设定质量目标。根据组织质量管理体系中规定的项目层面的质量目标，以项目需求和开发任务为依据，制定本项目有关活动的质量准则，为实施评价设定依据的质量目标。

(2)制定质量保证计划(Quality Assurance Plan)，又称质量保证大纲(Quality Assurance Program)。质量保证计划通常是质量策划的结果之一，它规定了特定项目中的产品和过程，以及使用相关程序和文件资料的人员和时机，通常包括以下内容：

①支持软件项目质量保证工作的软硬件环境、人员等资源，包括实施评价使用的辅助软件管理工具。

②明确质量保证评价所采用的技术方法，根据项目的具体活动裁减或细化过程活动和工作产品检查单，根据抽检原则确定抽检频次、时机和覆盖率。

③依据软件开发计划制定过程活动和工作产品检查计划，如表1所示，可在计划中确定评价的内容、时机和使用的检查单。

表1 软件质量保证计划

Tab.1 The plan of software quality assurance

评价产品/过程	评价时机	检查单
软件研制任务书	评审前3天	任务单检查表
软件需求规格说明	评审前6天	软件需求规格说明检查单
软件概要设计说明	评审前6天	软件概要说明检查单
软件详细设计说明	评审前6天	软件详细说明检查单
软件单元测试计划	评审前3天	软件测试计划检查单
软件单元测试说明	评审前3天	软件测试说明检查单
软件单元测试报告	评审前3天	软件测试报告检查单
软件验收测试计划	评审前3天	软件测试计划检查单
软件验收测试说明	评审前3天	软件测试说明检查单
软件验收测试报告	评审前3天	软件测试报告检查单
项目监控过程	每周	项目跟踪检查单
同行评审过程	同行评审结束	同行评审检查单
里程碑评审	阶段结束	项目监控检查单
配置管理过程	配置管理活动结束	配置管理检查单
需求管理过程	需求管理活动结束	需求管理检查单

④识别项目利益相关方，描述问题的上报解决渠道。

(3)实施质量评价。在质量管理体系的质量保证程序和质量评价准则的指导下，依据质量保证计划中规定的时间节点，使用相应的检查单对定义的工作产品和过程进行评价，并记录不符合项，确定修改人员与预期修改时间。按阶段统计质量保证活动情况，对质量问题进行趋势分析，并生成阶段性报告，定期将阶段总体情况通报给项目组相关成员和质量管理人员。

(4)质量问题处理。与质量保证评价问题的相关负责人进行沟通确认，在预期修改的时间段内对问题进行跟踪验证；对于有争议的问题，按照组织建立的上报机制，采取逐级上报的方式，力求问题最终得到妥善处理。

(5)过程改进。在组织质量管理体系层面，综合考虑评价过程中发现的问题，对过程和产品质量保证进行改进，包括对评价所使用检查单的制定和修订等。组织可指定软件质量主管对软件质量保证人员的工作情况进行定期审核，并记录纠正审核发现的问题，确保软件质量保证人员评价的客观性与符合性^[6]。

4 软件质量管理工具(The management tools of software quality)

现阶段市场上有许多项目管理工具提供软件工程方面的支持，辅助企业进行项目管理。这类工具软件的功能类别涵盖了软件整个生命周期的过程活动，从研制开发、测试分析到售后服务都有所涉及，有的甚至为软件需求分析人员、设计人员、测试人员等各类角色提供了一个系统化的平台^[8]。

对于辅助完成软件质量保证工作的软件质量管理工具，其功能一般不仅包含质量保证过程的管理，还包括软件测试、评审等子功能，乃至对整个软件项目的策划、监控、风险管理、需求管理、配置管理等。

4.1 软件质量保证功能分析

软件质量保证人员按照检查单的检查项进行过程活动和产品的评价，对检查单的理解不可避免地会受主观因素的影响。为减少由于主观判断造成质量评价的偏差，同时提高项目评价效率，在使用质量管理工具进行辅助评价时，应尽可能地将软件质量保证的检查项中可借助程序自动评价的部分识别出来，在软件质量保证功能定制开发时实现评价的自动化，并根据需要不断完善项目管理工具辅助质量检查的功能。

以使用自研的软件项目管理工具为例，其功能模块包括项目策划、项目跟踪、质量保证、需求管理、配置管理、风险管理和评审。其中，软件质量保证依据3.2部分中软件质量保证实施的一般步骤，对软件将要实现的功能分析如下：

(1)维护质量计划：该功能是根据软件开发计划中的时间节点对待评价的过程活动和工作产品制定进度计划。开发计划中的时间节点在进度计划中描述，在管理工具中应事先在相应模块对其进行维护，程序可以根据进度计划评审任务的

WBS节点的时间生成相应的过程活动评价计划和工作产品评价计划。

(2)管理检查单：该功能用于编制组织质量管理体系层面的检查表单，描述所有过程和活动检查单包含的检查项，以检查条目的方式进行存储，包括组织已识别为可进行程序自动评价的检查项。

(3)实施质量评价：按质量保证计划中的时间节点，对项目的过程活动和工作产品进行评价，使用相应的检查单记录评价情况，将与检查项不相符的问题描述为不符合项，并确定问题修改人员和预期修改时间。对于包含程序自评价检查项的检查单，软件质量保证人员可以对自评价的结果进行修改，或直接将描述问题的评价结果生成质量问题。

(4)处理质量问题：对评价过程中产生的不符合项进行跟踪处理，确认问题的当前状态，记录问题的处置情况，在预期修改时间内回归问题，直至问题关闭。

综上所述，为实现质量保证评价的自动化，重点应关注检查单管理、质量评价实施模块的功能实现，结合企业项目特点，根据项目管理软件提供的功能，前期对各类检查单的检查项进行梳理和识别。软件质量保证的主要功能如图2所示。

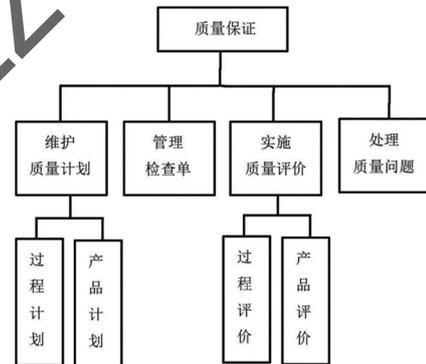


图2 软件质量保证功能图

Fig.2 The function diagram of software quality assurance

为了从使用角色的角度更直观地分析功能用例，使用用例图描述项目角色的工作内容，初步定义功能需求。以“维护质量计划”子功能为例，项目角色软件质量保证人员的功能用例如图3所示。

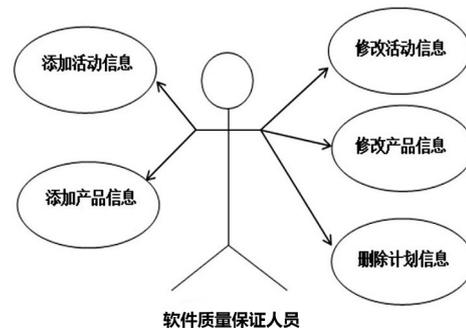


图3 维护质量计划功能用例图

Fig.3 The use case diagram of maintaining quality plan function

由图3可知，软件质量保证人员在维护质量计划基本信息时可以完成添加、修改和删除过程活动及工作产品信息的功能。为进一步分析功能所包含的使用前提、场景以及正常、异常操作，可将功能整理为用例建模，描述更为详细的信息项和操作步骤。

4.2 软件质量保证功能的设计实现

在功能需求明确后，应考虑软件架构设计，确定开发平台、使用语言、数据库，并进行界面设计和数据库的表设计。将满足功能需求的信息项抽象为数据库表，以表为单位来存储功能信息项逻辑字段，并预留支撑软件运行的系统信息字段。根据设计合理表关联的原则，尽量降低表之间的直接耦合度，表之间的内在关系也是程序中子功能关系的体现。以管理检查单功能为例，所管理的检查单分为自定义检查单和自评价检查单两类。如表2所示，表中定义了检查单管理模块各个显示字段的名称、标识，数据库中存储字段的数据类型和属性，以及界面所使用的控件。

表2 检查单界面模型

Tab.2 The interface model of checklist

显示名称	标识符	类型	控件
项目名称	ProjectName	Char(60)	TextBox
检查单名称	ChecklistName	Char(60)	TextBox
所属类别	TypeName	Char(60)	TextBox
建立时间	EstablishTime	DateTime	WebDateChooser
说明	Description	Char(250)	TextBox
评价方法	Estimate	Char(250)	TextBox
参与人员	Estimateperson	Char(250)	TextBox
检查项列表	Checktable	-	UltraWebGrid

程序界面示意图如图4所示，当需要根据项目情况自定义某个检查单的检查项时，选择进入管理检查单页面，界面显示出项目名称和检查单名称，可以维护检查单的相关信息，并在列表中对检查项进行增删改查。

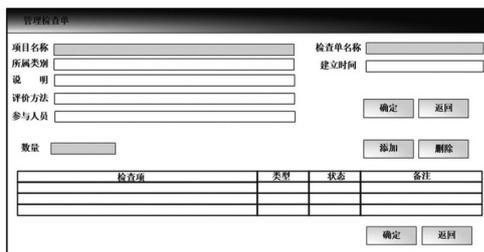


图4 检查单管理信息界面图

Fig.4 The interface diagram of checklist management information

软件质量保证人员对项目跟踪情况例行检查时，对于项目监控过程检查单的部分检查项可采取程序自动判断的方式进行检查。例如，判断项目组每周是否跟踪了任务的进度和工作量，这条检查项可通过检索数据库中项目跟踪功能对应表单是否存在待检查时间段内的记录进行判断，诸如此类的

检查项都可以梳理为程序自动检查的评价内容。

4.3 软件质量保证自动化评价效率分析

以项目监控过程为例，每周监控内容的例行检查项约有30条，每条检查项人工检查需要平均耗时3分钟，也就是说单个项目的单个过程检查单的耗时为1.5小时。而这样的检查单还包括需求管理过程、配置管理过程、评审过程等近10个检查单。假设每周将10个检查单的内容通过人工平均实现，需要15个小时。以一个软件质量保证人员同一时期平均管理的项目数为4个计算，每天8个工时的情况下，需要7.5个工作日才能完成项目的评价。

而根据辅助工具中其他管理模块的关联进行自动化评价的检查项粗略估计可以达到一半甚至以上，因此一个软件质量保证人员只需要不到4天就可以完成假设的工作任务，再加上对自动化评价的人为检查与判断，一周即可管理至少4个项目，工作效率提高了33%。

质量保证管理工具可将功能进一步扩展为自动生成质量评价问题，并推送给问题责任人；定期判断问题修改与否；生成阶段性质量保证报告等。自动化的评价管理将软件质量保证人员从繁重的重复性劳动中解放出来，从而关注软件程序不能判断的、有待人为决策的检查项。同时，项目质量数据在软件中的存储积累也为组织进一步进行数据分析挖掘提供了基础，从而有利于组织过程的改进。

5 结论(Conclusion)

本文针对目前软件质量保证的技术方法进行了研究，重点探讨了软件项目管理工具辅助质量保证实施监督的必要性和关注点，通过程序设计阐述了质量保证实施自动化评价的思路和方法。该系统的应用有利于减少项目管理受主观因素影响导致的评价结果不客观，提高了工作效率和评价质量。对企事业单位实施软件质量保证以及组织的体系建设有一定的实用价值，同时对软件质量的改进工作有借鉴作用。

参考文献(References)

- [1] 万融.商品学概论[M].北京:中国人民大学出版社,2013: 155-160.
- [2] 中国人民解放军总装备部.装备质量管理术语:GJB 1405A—2006[S].北京:中国人民解放军总装备部军标出版发行部,2006:3-4.
- [3] 全国信息技术标准化技术委员会.系统与软件工程 系统与软件质量要求和评价(SQuaRE)第40部分:评价过程:GB/T 25000.40—2018[S].北京:中国标准出版社,2019:9-11.
- [4] 黄小路.蓝领质量素质提升[M].北京:中国质检出版社,2014: 85-121.
- [5] 宁涛,金花.软件项目管理[M].北京:中国铁道出版社,2016: 54-76.