

软件项目质量控制相关技术研究

任永昌, 彭霞, 常革新

(渤海大学 信息科学与技术学院, 辽宁 锦州 121013)

摘要:软件项目质量管理是贯穿整个软件生命周期的重要工作,有效地实施软件产品的质量控制是提高软件质量、降低质量成本的重要手段。首先,研究质量控制模型,包括产品、过程和资源三大控制要素,用PDCA循环表示的质量控制模型结构,并对四个过程八个阶段分别论述;其次,研究质量控制体系,包括基础、手段、目的三层体系结构;然后,研究质量控制图,包括质量控制图结构,确定控制中心线、控制上线、控制下线的数学公式推导,阐明确定失控点的7条规则;最后,研究质量控制框架,包括控制框架结构和控制框架特点。结果表明,质量控制相关技术,为质量控制提供科学管理方法和技术支持,是实现软件质量控制目标的有效方法和途径。

关键词:软件项目管理;软件质量控制;质量控制模型;质量控制图;质量控制框架

中图分类号:TP311

文献标识码:A

文章编号:1673-629X (2012)10-0143-04

Relevant Techniques Research of Software Project Quality Control

REN Yong-chang, PENG Xia, CHANG Ge-xin

(College of Information Science and Technology, Bohai University, Jinzhou 121013, China)

Abstract:Software project quality management is the important work throughout the entire software life cycle, the effective implementation of software product quality control is an important means of improving software quality, reducing the cost of quality. First, research the quality control model, including products, processes and resources of three control elements, with the quality control model structure that the PDCA cycle expresses, and elaborates separately to four process eight stages. Secondly, research the quality control system, including basis, means and purpose of three-tier architecture. Then, research the quality control chart, including quality control chart structure, to determine the mathematical formula of control center line, upper control line, lower control line, to clarify the 7 rules of the points out of control. Finally, research the quality control framework, including the control framework structure and the characteristics of control framework. The results show that through researching the quality control relevant techniques, provide the scientific management methods and technical support for quality control, is the effective ways and means to realize the goal of software quality control.

Key words:software project management; software quality control; quality control model; quality control chart; quality control framework

0 引言

软件项目质量管理,是贯穿整个软件生命周期的重要工作,是软件项目顺利实施并成功完成的可靠保证。随着软件开发技术的发展和信息技术的广泛应用,软件项目质量管理越来越受到重视。实现软件项目质量管理与国际标准接轨,加强软件管理、改善软件开发过程、提高软件质量,已成为软件行业面临的巨大难题。通过软件质量控制,提高软件产品的生产可靠性、降低软件产品的开发成本^[1]。

高质量的软件离不开有效的管理和控制。质量和

成本,是衡量项目成功与否的两个关键因素,通过质量控制也能降低项目成本^[2]。Donald Reifer 给出软件质量控制的定义:软件质量控制是一系列验证活动,在一系列的控制活动中采取有效措施,在软件开发过程的各个监测点上,评估开发出来的阶段性产品是否符合技术规范^[3]。

质量控制是软件项目管理的重要工作,文中对关键技术进行了研究。

1 质量控制模型

全面质量控制过程,就是质量计划的制定和组织实现过程。由休哈特(Walter A. Shewhart)提出构想,经过著名质量管理专家戴明(Edwards Deming)的深化和发展,总结出管理学的通用模型,称戴明环,在很多资料上也称为PDCA循环。

收稿日期:2012-02-21;修回日期:2012-05-31

基金项目:国家自然科学基金资助项目(70871067);2011辽宁省科学事业公益基金项目

作者简介:任永昌(1969-),男,教授,博士,从事信息处理、软件工程、软件项目管理等研究。

1.1 质量控制要素

软件项目质量控制的三大要素是产品、过程和资源,需要不断进行调整和检查。

三大要素表述如下:

(1)产品(Production)。一个过程的输出产品,不会比输入产品的质量更高,如果输入产品有缺陷,会在后续产品中放大,并影响最终产品质量。软件产品中的各个部件和模块,必须达到预定的质量要求,特别需要保证各模块共用的 API 和基础类库的质量,否则各个模块集成以后的缺陷会成倍放大,并且难以定位,修复成本也会大大增加。

(2)过程(Process)。软件项目过程分为两类。一类是技术过程,包括需求分析、架构设计、编码实现等;另一类是管理过程,包括技术评审、配置管理、软件测试等。技术过程进行质量设计并构造产品,同时会引入缺陷,因此技术过程直接决定了软件质量特性;管理过程对质量过程进行检查和验证,发现问题并进行纠正,间接地决定了最终产品质量。因此,技术过程和管理过程都对软件质量有重要影响。

(3)资源(Resource)。软件项目中的资源包括:人、时间、设备和资金等,资源的数量和质量都影响软件产品质量。软件是智力高度集中的产品,人是决定性因素,软件开发人员的知识、经验、能力、态度,都会对产品质量产生直接影响。在大多数情况下,项目的时间和资金都是有限的,构成了制约软件质量的关键因素。而设备和环境不足也会直接导致软件质量低下。

1.2 模型结构

将 PDCA 循环用于质量控制模型结构如图 1 所示^[4]。

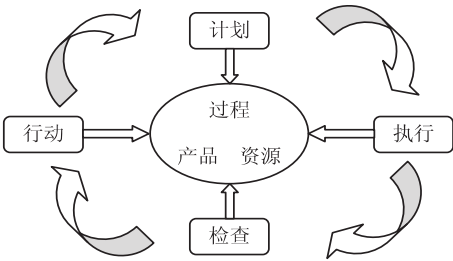


图 1 质量控制模型

1.3 四个过程八个阶段

PDCA 包括四个过程八个阶段:

(1)计划(Plan)。分析现状、发现问题、找出原因,制定相应的质量方针、目标、计划和原则。该阶段包括四个阶段,即:找出问题、找出原因、找出要因、制定计划。

(2)执行(Do)。根据计划实施,执行计划中规定的各项活动。该阶段包括一个阶段,即执行计划。

(3)检查(Check)。对执行的结果进行检查、审核和评估,收集数据并进行分析,度量工作的质量,发现存在的问题。该阶段包括一个阶段,即检查结果。

(4)行动(Action)。针对检查中发现的问题,采取相应的改进措施纠正偏差。总结成功经验,吸取失败教训,形成标准和规范指导以后的工作,通过行动提高并升华。该阶段包括两个阶段,即:总结经验、提出新问题。

2 质量控制体系

大部分软件开发企业在软件质量控制方面都处于相对独立的环节上,并未形成统一、规范的质量控制体系,缺乏质量度量的有效方法。建立质量控制体系的目的,就是通过形成一个有机整体,对软件项目开发的全过程进行有效地控制,需要在完整的软件质量控制体系作用下,软件质量从开始就能够进行预期,并能在软件开发过程中对质量进行控制,确保质量符合要求^[5]。软件质量控制体系结构如图 2 所示。

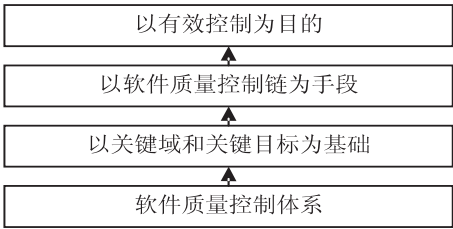


图 2 软件质量控制体系结构

软件质量控制体系不是万能的,一个好的流程和规范能够真正起到作用,还需要人员和技术的支持。在软件质量控制过程中,要不断采用新技术,质量控制人员应保持和开发人员的良好沟通和互动,协助开发项目组完成软件质量目标^[6]。

3 质量控制图

控制图是数据的图形表示,是画有控制界限的一种图表,表明一个过程随时间变化的结果,用来分析质量波动究竟是由于正常原因引起,还是异常原因引起,从而判明生产过程是否处于控制状态。主要用途是预防缺陷,而不是检测或拒绝缺陷,可以帮助人们判断一个过程,是在控制之中还是失去了控制^[7]。

3.1 控制图结构

质量控制图结构如图 3 所示。一般有 3 条控制线,上面一条虚线称为控制上线(UCL),下面一条虚线称为控制下线(LCL),中间一条实线称为控制中心线(CL)。将所控制的质量特性用圆点标记,若圆点全部在控制界限内,且排列无缺陷(如趋势、周期、接近),则可判断项目质量处于受控状态,否则认为项目实施存在异常,必须认真检查并予以消除。

3.2 控制线计算

一般来说,控制图的中心线使用过程的平均值估计来代替;控制上线使用平均值加上 3 倍的标准差估计值;控制下线使用平均值减去 3 倍的标准差估计值。这些计算来自于过程产出,并假设这个过程的产出不存在特殊情况。

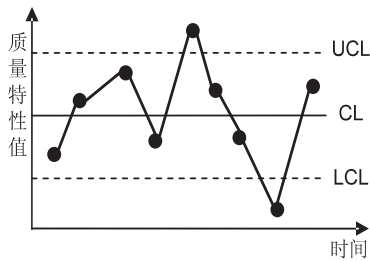


图 3 质量控制的示意图

设取 n 个随机样本,其中不合格数为 D ,则 D 服从参数为 n 和 P 的二项分布^[8],即:

$$P\{D = x\} = C_n^x P^x (1 - P)^{n-x} \quad x = 0, 1, \cdots, n \quad (1)$$

二项分布的均值和方差分别为:

$$\mu = nP \quad (2)$$

$$\sigma^2 = nP(1 - P) \quad (3)$$

样本错误率 p 定义为样本不合格数 D 与样本 n 的比值,即:

$$p = D/n \quad (4)$$

随机变量 p 的均值和方差分别为:

$$\mu_p = P \quad (5)$$

$$\sigma_p^2 = P(1 - P)/n \quad (6)$$

与正态分布不同, μ_p 与 σ_p^2 不是独立的,只需一张控制图对过程进行控制,控制线分别为:

$$CL = P \quad (7)$$

$$UCL = P + 3 \sqrt{P(1 - P)/n} \quad (8)$$

$$LCL = P - 3 \sqrt{P(1 - P)/n} \quad (9)$$

3.3 确定失控点的规则

控制线的区域可以划分为 6 个区域,每个区域是 1 个标准差的宽度,如图 4 所示^[9]。中心线的 1 个标准差范围内为 C 区域;中心线的 1 个和 2 个标准差范围称为 B 区域;最外面的区域,中心线的 2 个和 3 个标准差之间的范围称为 A 区域。

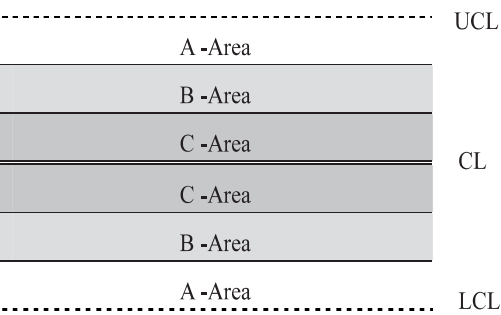


图 4 控制图的 ABC 区域

根据这些区域,将发现的所有失控点都直接标记在控制图上,一般采用 7 个简单的规则来确定过程是否失控。

规则 1,如果任何 1 个错误率统计值落在控制线之外,则软件质量失控;

规则 2,如果中心线同一侧有 2/3 的连续错误率统计值落在 A 区域或超出 A 区域,则软件质量失控;

规则 3,如果中心线同一侧有 5 个连续错误率统计值中的 4 个落入 B 区域之内或超出 B 区域,则软件质量失控;

规则 4,如果 8 个或 8 个以上的连续错误率统计值落在中心线的同一侧,则软件质量失控;

规则 5,如果 8 个或 8 个以上的连续错误率统计值向上运动或向下运动,则软件质量失控;

规则 6,如果一些不寻常点在中心线上下来回跳动(锯齿模式),则软件质量失控;

规则 7,如果 13 个或 13 个以上的连续错误率统计值都落在任意一边的 C 区域里,则软件质量失控。

4 质量控制框架

软件开发机构应用软件质量控制框架(SQCF, Software Quality Control Framework),可以帮助其有效地解决质量控制的问题。

软件质量控制框架结构如图 5 所示^[10]。

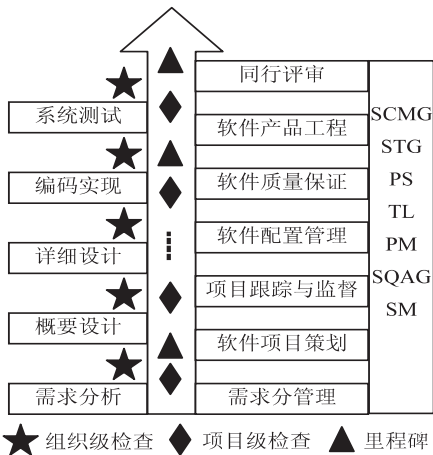


图 5 软件质量控制框架

从软件质量控制框架可以看出:

(1) SQCF 贯穿于整个软件开发过程。包括:需求分析(requirement analysis)、概要设计(summary design)、详细设计(detailed design)、编码实现(coding)、系统测试(system testing)。

(2) SQCF 建立在 CMM 的 7 个关键过程域(KPA, Key Process Area)之上。这 7 个 KPA 分别是:需求管理(requirement management)、软件项目策划(software

project planning)、软件项目跟踪和监督 (software project tracking and oversight)、软件质量保证 (software quality assurance)、软件配置管理 (software configuration management)、软件产品工程和同行评审 (software product engineering and Peer-Reviewed)。

(3) SQCF 定义了 4 个角色和 3 个组。4 个角色分别是:高级经理 SM (Senior Manager)、项目经理 PM (Project Manager)、任务负责人 TL (Task Leader)、项目成员 PS (Project Staff); 3 个组分别是:软件配置管理组 SCMG (Software Configuration Management Group)、软件质量保证组 SQAG (Software Quality Assurance Group)、系统测试组 STG (System Test Group)。这些角色和组贯穿 SQCF 的始终,分别承担和履行着不可或缺的职责,是 SQCF 中必不可少的部分。

(4) SQCF 建立了多个检查点。检查点指在规定的时间间隔内对项目进行检查,比较实际与计划之间的差异,并根据差异进行调整。可将检查点看作是一个固定的“采样”时点,而时间间隔根据项目周期长短不同而不同,如果频度过小会失去意义,如果频度过大会增加管理成本。常见的间隔是每周一次,项目经理需要召开例会并上交周报。

(5) SQCF 中的里程碑。里程碑是完成阶段性工作的标志,不同的项目类型里程碑不同。里程碑在项目管理中具有重要意义。里程碑通常是一些重要活动的完工、重要文档的交付、阶段评审通过等。正式评审合格后里程碑才算完成,正式评审由 SQAG 主导负责,相关负责人员共同参加^[11,12]。

5 结束语

通过一系列的软件质量控制活动来衡量软件质量,对发生的质量问题及时纠正,而不要等到最后无法弥补时才发现问题^[13]。有效地实施软件产品的质量,不但可以解决企业软件交付延期、费用超支、质量无法保证等问题,而目改进开发过程、提高用户满意

度、缩短产品开发周期与投放市场时间和降低质量成本等^[14]。

通过对质量控制相关问题的研究,为质量控制提供科学的管理方法和技术支持。文中的研究内容,对于从事软件项目管理和质量管理的人员具有较高的参考价值。

参考文献:

- [1] 朱一玮,徐 畅,章俊玲,等. 软件开发过程中质量控制的具体改进方法[J]. 计算机与现代化,2007,23(2):4-6.
- [2] 谭守标,徐 超,李正平. 日本软件质量保证实践[J]. 计算机技术与发展,2007,17(6):46-48.
- [3] 于芳民. 软件开发工程项目管理中的质量控制研究[J]. 潍坊教育学院学报,2008,21(3):39-41.
- [4] 任永昌,鄂 旭,李春杰,等. 软件项目开发方法与管理[M]. 北京:清华大学出版社,2010.
- [5] 陈 起. 打造软件质量控制体系[J]. 金融电子化,2008,15(9):43-45.
- [6] Ren Y C, Xing T, Zhu P, et al. Relevant Issue Research on the Quality Control of Software Project Management[C]. [s. l.]: IEEE Press, 2010:313-316.
- [7] 朱少民. 软件质量保证和管理[M]. 北京:清华大学出版社,2008.
- [8] 张公绪,何国伟,郑慧英. 新编质量管理学[M]. 北京:高等教育出版社,1999.
- [9] Gitlow H S. 质量管理[M]. 北京:机械工业出版社,2008.
- [10] 李 涛. 基于 CMM 的软件质量控制框架[D]. 西安:西安建筑科技大学,2004.
- [11] Holmes J S. Software Measurement Using SCM[J]. Software Quality Professional, 2004, 7(1):14-21.
- [12] Voas J. Assuring software quality assurance[J]. IEEE Software, 2003, 20(3):48-49.
- [13] 王如龙,邓子云,罗铁清. IT 项目管理从理论到实践[M]. 北京:清华大学出版社,2008.
- [14] 黄立威,黄 伟,冯 径. 支持软件质量控制的软件配置管理研究[J]. 计算机技术与发展,2011,21(7):50-53.

(上接第 142 页)

- er/controller architecture for organic computing[C]//INFORMATIK 2006 – Informatik für Menschen, Volume P-93 of GI-Edition – Lecture Notes in Informatics (LNI). [s. l.]: Bonner Kollen Verlag, 2006:112-119.
- [6] Prothmann H, Rochner F, Tomforde S. Organic Control of Traffic Lights[C]//Proceedings of the 5th International Conference on Autonomic and Trusted Computing (ATC-08), Volume 5060 of LNCS. [s. l.]: Springer, 2008:219-233.
- [7] 田银花. 网络任务调度模型及算法的 Petri 网模型研究[D]. 济南:山东科技大学,2007.

- [8] 李志清,傅秀芬,蒋明亮,等. 一个基于服务的网格计算模型[J]. 计算机技术与发展,2006,16(6):119-120.
- [9] 陈宇寒. 网格计算技术研究[J]. 计算机技术与发展,2008,18(5):119-120.
- [10] 黄智维,倪子伟. 网格计算环境下资源管理的研究[J]. 计算机技术与发展,2009,19(3):200-203.
- [11] 杨 炼,杨长兴. 基于层次化的网格资源三层调度模型[J]. 计算机技术与发展,2008,18(9):43-45.
- [12] 张建勋,贺毅朝,田俊峰. 基于市场的网格资源分配管理模型研究[J]. 计算机技术与发展,2007,17(2):193-196.

软件项目质量控制相关技术研究

作者: 任永昌, 彭霞, 常革新
作者单位: 渤海大学 信息科学与技术学院, 辽宁 锦州 121013
刊名: 计算机技术与发展
英文刊名: Computer Technology and Development
年, 卷(期): 2012(10)

本文链接: http://d.g.wanfangdata.com.cn/Periodical_wjfz201210038.aspx