

基于缺陷预防的软件过程改进

侯东鸽, 侯 红, 郝克刚

(西北大学 信息科学与技术学院, 陕西 西安 710127)

摘 要:在软件开发过程中,软件缺陷不但影响软件的质量,而且影响软件的开发进度及开发成本,可见必须采取某种方法避免缺陷的产生。而缺陷是由错误造成的,因而应从控制错误的产生入手。缺陷预防方法能够确保错误在标识并解决后不会再次发生,将缺陷预防方法引入软件过程中,目的是控制缺陷的产生,从而优化生产过程,提高软件质量。介绍了缺陷预防的方法,然后将缺陷预防与软件过程改进相结合,提出基于缺陷预防的软件过程框架,最后通过实验进行验证。

关键词:缺陷;缺陷预防;过程改进

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2008)07-0021-03

Defect Prevention - Based Software Process Improvement

HOU Dong-ge, HOU Hong, HAO Ke-gang

(School of Information Science and Technology, Northwest University, Xi'an 710127, China)

Abstract: In the process of software development, software defects not only affect the software quality, but also affect the software development progress and costs, so a method that must be taken to avoid the defects formation. The defect is caused by the error, which should start by controlling the errors. Defect prevention can ensure that errors will not happen again after identification and solution. The defect prevention method is imported in software process in this paper, the aim is control defects, thereby optimizing the production process, improving software quality. In this paper, defect prevention methods are introduced firstly, and then software defect prevention and software process improvement are combined, finally, defect prevention-based process improvement framework is put forward, and carried through validation by experiment.

Key words: defect; defect prevention; process improvement

0 引言

软件质量是软件生产过程的重要因素,在与进度、成本发生矛盾时,软件质量经常作为第一要素考虑。为保证软件质量,必须对软件缺陷这个造成软件质量低下的原凶加以控制。软件缺陷是软件工作产品中不满足指定要求的成分,它是静态的,如果不将其消除,它将永远存在^[1,2],并且软件缺陷是软件产品的固有成份^[3]。为保证软件质量,传统的方法是采用相关技术发现缺陷,并将其纠正,采用这种策略,类似的缺陷会不断地出现、修复,再出现、再修复,将是个无休无止的过程。这不但浪费软件生产人员的工作时间和生产

成本,而且需要承受它们所带来的影响。缺陷预防方法可以防止这种无尽的反复,其原理是分析了解产生缺陷的原因,并采取行动在根源处预防它们。进行缺陷预防,首先必须熟识缺陷和产生缺陷的原因,然后才能制定相应的措施在软件开发过程中实施,并且需要对实施状况进行跟踪以确定下一轮的缺陷预防。缺陷预防使过程改进时刻处于中心位置^[4],促进软件过程的改进。

1 缺陷预防方法

1.1 缺陷预防的概念、意义

缺陷预防是指在软件过程中识别出产生缺陷的原因,并且以此采取防范措施,防止它们再次发生^[5,6]。缺陷预防需要实现三个目标:计划缺陷预防活动;找出和识别缺陷的共有原因;将缺陷共有原因按优先级排序,并系统地排除。只有达到这三个目标,缺陷预防才能真正地发挥作用,软件质量才能得以保证。

软件缺陷是软件生产过程的副产品,正如 Crosby 所说:“要花钱的是‘非质量’的东西,也就是那些一开

收稿日期:2007-10-12

基金项目:国家 863 计划资助项目(2004AA115090)

作者简介:侯东鸽(1981-),女,河南郑州人,硕士研究生,主要研究方向为软件工程;侯 红,高级工程师,博士,主要研究方向为软件过程和软件度量;郝克刚,教授,博士生导师,主要研究方向为软件工程、软件理论、形式化方法。

始没把工作作对,而必须采取的所有的补救措施^[7]”。查找并修正缺陷,需要付出一定的代价,而且随着发现时间的推移,成本会随指数型增长^[8]。采用缺陷预防的方法,可以大规模地减少寻找和修复缺陷的成本,例如:惠普公司通过建立大量的原型和设计评审计划,极大地减少了发布产品后发现的缺陷数^[4];某公司在开发过程中使用了缺陷预防方法,使开发期间所发现的缺陷数降低了 50%,在交付后的产品中所发现的错误数降低了 78%^[2]。并且,采用缺陷预防的方法,使人们时刻关注于容易出错的地方,及时发现问题并纠正。

1.2 缺陷预防的实施步骤

缺陷预防与传统的“救火”方式不同,它追求“第一次就把事情作对”,不是部分人的事情,而是全体成员的职责。因而实施缺陷预防,必须保证每个人都参与其中,尤其需要管理层的参与,只有这样,才能带动每一个人,才能打破传统,逐渐形成人人预防的理念。如果缺少管理层的参与,缺陷预防可能会在实施过程中不了了之。

缺陷预防方法有其具体的实施步骤(如图 1 所示),按此步骤实施,才能保证有效性,解释如下:

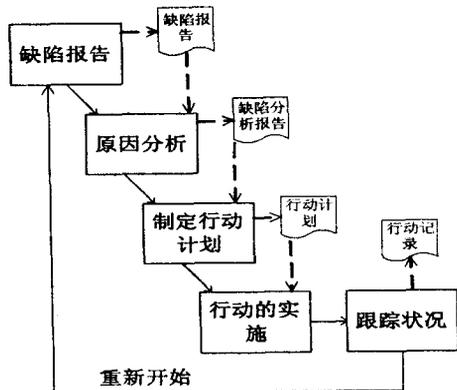


图 1 缺陷预防计划的实施步骤

1) 缺陷报告:将采集的缺陷信息进行记录,包括缺陷的类型、缺陷的引入阶段、排除阶段、修改时间、缺陷描述等。缺陷报告包含尽可能多的信息,从而保证覆盖所有问题。缺陷报告是进行缺陷预防必不可少的。

2) 原因分析:定期召开原因分析会议进行原因分析,可以采用因果关系图等作为工具。原因分析会议的主要目的是找出产生缺陷的原因,并将其分类,然后确定实施哪些措施,以保证这类缺陷不会再次出现。原因分析会议结束后需要撰写缺陷分析报告。

3) 制定行动计划:建立行动组,根据缺陷分析报告制定针对普遍问题制定行动计划。将行动计划文档化,方便行动的实施。

4) 行动的实施:根据行动计划,实施预防,需要全体人员参与其中。

5) 跟踪状况:对行动进行跟踪,直至完成,收集行动中的数据。

6) 重新开始:行动结束后,针对剩余缺陷重新开始。

2 基于缺陷预防的软件过程改进

2.1 基于缺陷预防的软件过程改进组织框架

实施缺陷预防,需要从两个层面上组织人员:一个是要在组织级别上协调缺陷预防活动;另一个是要在具体的项目上协调缺陷预防活动。在组织级别上,负责人通常由 SEPG(Software Engineering Process Group)的负责人担任,同时需要质量保证、配置管理、培训部门、项目组等相关人员参加,缺陷预防小组(Defect Prevention Group,下文简称 DPG)可以是 SEPG 的一部分,也可以是单独的一个部门。在项目级别上,通常由项目负责人协调缺陷预防活动,选取部分成员收集数据和进行原因分析,每个成员都是缺陷预防的执行人。基于缺陷预防的软件过程改进组织框架如图 2 所示。

2.2 基于缺陷预防的软件过程改进框架

在实施缺陷预防时,首先需要发现缺陷,并找出产生缺陷的原因。然后确定预防措施,实施预防,避免它的再次出现。项目结束后或者一个阶段结束后,需要对缺陷预防的实施进行总结,分析预防的效果及有利于产品质量的措施,并且产生过程改进建议。DPG 小组需要将改进建议等信息交付 SEPG,由 SEPG 小组进行过程改进。因而,可以说缺陷预防促进了过程改进。实施缺陷预防需要在原有软件过程中添加特定内容,包括开工会议、数据库建设、原因分析、效果分析和反馈系统(如图 3 所示)。

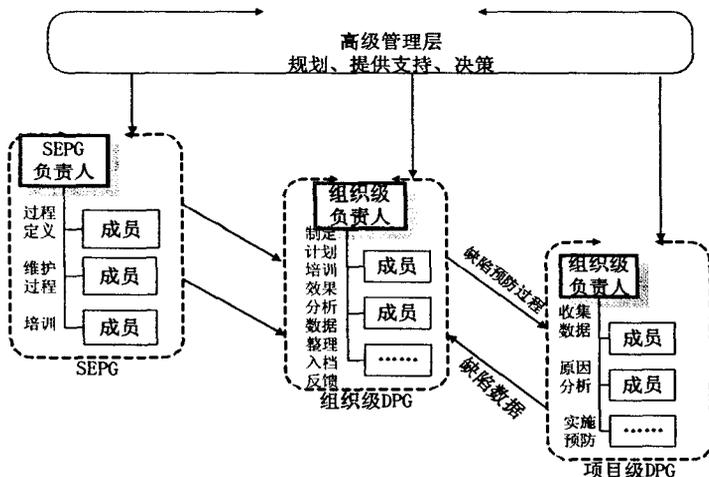


图 2 基于缺陷预防的软件过程改进组织框架

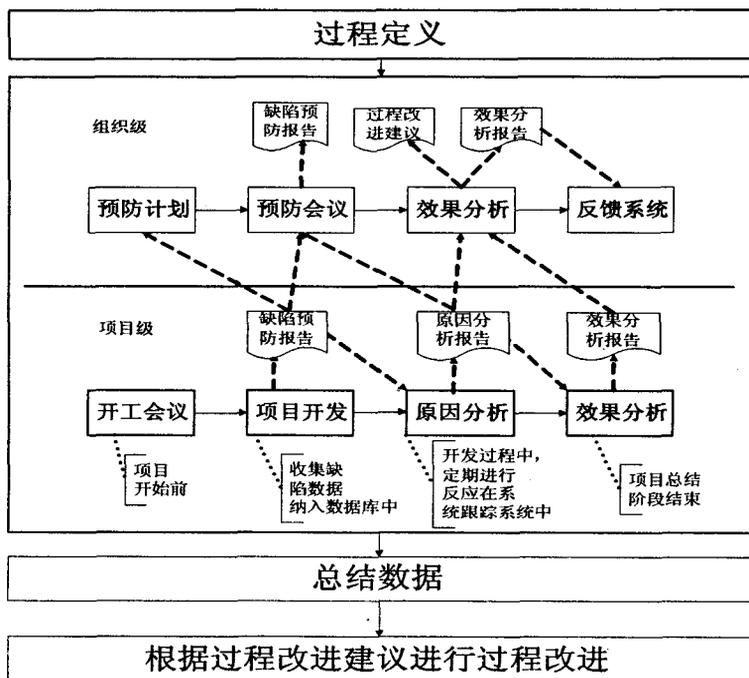


图 3 加入缺陷预防的软件过程改进框架

1) 开工会议。在项目开始时, 召开开工会议, 目的是确定项目和缺陷预防目标, 对需要的过程、工具、方法等相关的方法进行评审和裁剪。

2) 数据库建设。建立相应的数据库, 用来存储开发过程中发现缺陷的相关信息。

3) 原因分析。按照计划, 进行原因分析, 可以定期举行, 也可以在各个阶段结束后进行。分析产生缺陷的原因, 制定预防措施, 对结果进行存档, 供组织和其他项目使用。原因分析可分为项目级别和组织级别两种, 其中项目级别的原因分析只对本项目产生的缺陷进行分类分析, 组织级别的原因分析是将所有从项目组得来的数据统一进行分析、整理。

4) 效果分析。在项目结束后或者阶段结束后, 分析采用缺陷预防带来的成本和收益, 形成效果分析报告。效果分析也分为组织级和项目级, 项目级是分析在项目中采用缺陷预防方法所花费的成本和收益。组织级是从组织的角度分析成本和收益。通过分析成本和收益情况, 撰写效果分析报告, 并总结过程改进的建议。过程改进建议的内容包括问题、解决途径、在软件开发过程中的对策等。

5) 反馈系统。反馈系统的目的是将软件开发过程中的成本、经验、收益等信息及时反应给团队成员。通过反馈系统发布信息, 不但为团队成员提供了一个学习平台, 而且能够调动团队成员的积极性。

2.3 案例研究

为验证采用缺陷预防的软件过程的效果, 进行了案例研究。选取软件工程专业两个班级的六十个本

科生作为实验主体, 并且为了保证实验的有效性, 将这些学生随机分为二个组。同时选择难易程度相当的五个项目作为实验客体, 分别为: 医院管理系统、酒店管理系统、学生管理系统、图书馆管理系统、网上购物管理系统, 并且规定统一采用 Java 作为开发语言。将每个组的成员再随机分为五组, 每组抽取一个项目。实验开始前, 对其中的一个组进行培训, 学习缺陷预防的方法, 以便于将缺陷预防用于软件开发过程中, 另一组则采用传统的软件开发过程。为了数据分析的需要, 在每个小组中选取一人作为缺陷数据收集人员, 由他记录项目中出现的缺陷及相关信息。在项目结束后, 将各小组的缺陷记录进行汇总、分析, 得到如图 4 所示的各班级的平均缺陷引入率。从图中可以看出, 使用缺陷预防方法的班级的平均缺陷引入率低于未使用缺陷

预防方法的班级。从而说明, 采用加入缺陷预防的软件开发过程开发软件能够有效地控制缺陷的产生, 保证生产出高质量的软件产品。

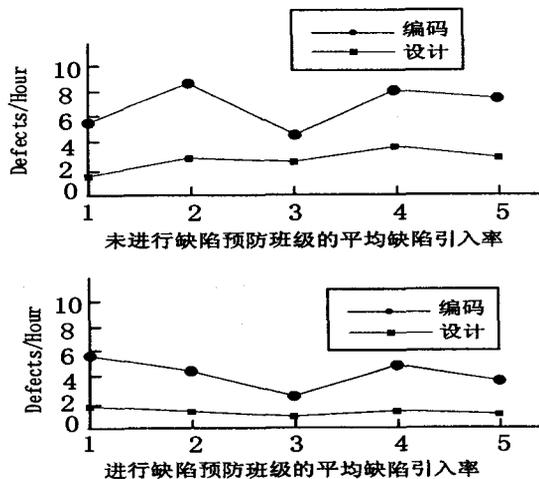


图 4 平均缺陷引入率

3 结束语

软件缺陷是由于人的错误所引入的, 而缺陷预防方法是找出缺陷的共性原因, 也就是错误, 从此入手, 防止它的再次出现。但是, 缺陷预防并不是一个立竿见影的方法, 在实施的初期可能不会得到期望的收益, 但一定要在软件开发过程中长期贯彻执行, 同时寻求专家的帮助, 就一定能够得到丰厚的回报。实施缺陷预防必须确保全员的参与, 并且离不开管理层的支持和带动, 如果人人预防的文化在软件组织中形成, 在积

- [2] Perkins C E, Bhagwat P. Highly dynamic destination sequenced distance - vector routing (DSDV) for mobile computers[C]// Proceedings of the SIGCOMM'94. [s. l.]: [s. n.], 1994: 234 - 244.
- [3] Perkins C E, Royer E M. Ad hoc On - Demand Distance Vector Routing[C]//IEEE Workshop on Mobile Computing Systems and Applications. New Orleans, LA, USA: [s. n.], 1999: 90 - 100.
- [4] Johnson D B, Maltz D. Dynamic Source Routing in Ad hoc Wireless Networks[M]. New York: Kluwer Academic Publishers, 1996: 153 - 181.
- [5] Chlamtac I, Conti M. Mobile Ad hoc Networking: Imperatives and Challenges[J]. Ad hoc Networks, 2003, 1(1): 13 - 64.
- [6] Argyroudis P G, Donal M. Secure routing for mobile ad hoc networks[J]. IEEE Communication Survery & Tutorials, 2005, 7(3): 2 - 21.
- [7] Hu Y C, Perrig A, Johnson D B. Wormhole attacks in wireless networks[J]. Selected Areas in Communications, 2006, 24(2): 370 - 380.
- [8] Zapata M G. Secure ad hoc on - demand distance vector (SAODV) routing[EB/OL]. 2005 - 07. <http://www.ietf-report.isoc.org/idref/draft-guerrero-manet-saodv/>.
- [9] Hu Y C, Perrig A. Ariadne: A Secure On - demand Routing Protocol for Ad hoc Networks[C]//Proceedings of the 8th ACM International Conference on Mobile Computing and Networking. Atlanta, GA, USA: [s. n.], 2002: 12 - 23.
- [10] Padimitrators P, Haas Z. Secure Routing for Mobile Ad hoc Networks[C]//SCS Communication Networks and Distributed Systems Modeling and Simulation Conference. San Antonio, TX, USA: [s. n.], 2002: 27 - 31.
- [11] Yi S, Naldurg P, Kravets R. Security - aware Ad hoc Routing for Wireless Networks[C]//Proc. of ACM MOB IHOC'01. Long Beach, CA, USA: [s. n.], 2001: 299 - 302.
- [12] Li H Z, Singhal M. A Secure Routing Protocol for Wireless Ad Hoc Networks[C]//Proceedings of the 39th Annual Hawaii International Conference on System Science. Hawaii, USA: [s. n.], 2006.
- [13] Helena R F, Jordi H J. Secure Dynamic MANET On - demand (SEDYMO) Routing Protocol[C]//Proceedings of the Fifth Annual Conference on Communication Networks and Services Research (CNSR'07). Fredericton, Canada: [s. n.], 2007: 372 - 380.
- [14] Chakeres I, Perkins C. Dynamic MANET on - demand Routing [EB/OL]. 2006 - 04. <http://tools.ietf.org/html/draft-ietf-manet-dymo>.
- [15] Macropodi R. Secure Multipath Routing Protocol for Mobile Ad hoc Networks[C]//Proceedings of the Second Annual Conference on Wireless On - demand Network Systems and Services (WONS'05). St Moritz, Switzerland: [s. n.], 2005: 89 - 96.
- [16] Hu Y C, Perrig A, Johnson D B. PacketLeashes: A Defense against Wormhole Attacks in Wireless Ad Hoc Networks[C]//INFOCOM '2003. San Francisco, California, USA: [s. n.], 2003: 1976 - 1986.
- [17] Hu Y C, Perrig A, Johnson D B. Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols[C]//Proceedings of the ACM Workshop on Wireless Security (WiSe'2003). Rome, Italy: [s. n.], 2003: 30 - 40.
- [18] Capkun S, Buttyan L, Hubax J. Sector: Secure tracking of node encounters in multi - hop wireless networks[C]//Proceedings of the ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN'03). Washington, D. C, USA: [s. n.], 2003: 21 - 32.

(上接第 23 页)

累的过程中,就会由量变达到质变,整个组织必定会从中受益无穷,成为一个高效、焕然一新的组织。

缺陷预防是一个渐进的过程,每次只是针对优先级最高的缺陷加以预防,然后就是次高的缺陷,不断地重复下去。每次的缺陷预防都将产生对过程的改进建议,因而过程改进也是逐渐进行的,正如 Watts Humphrey 常说的一句话是:“软件过程改进不是目标,而是一条漫漫长路”^[8]。

参考文献:

- [1] 梁成才,章代雨,林海静. 软件缺陷的综合研究[J]. 计算机工程, 2006, 32(19): 88 - 90.
- [2] Chang Ching - Pao, Chu Pchih - Ping. Defect Prevention in Software Processes: An Action - Based Approach[J]. Journal of Systems and Software, 2007, 80(4): 559 - 570.
- [3] Song Qinbao, Shepperd M, Cartwright M, et al. Software Defect Association Mining and Defect Correction Effort Prediction[J]. IEEE Transactions on Software Engineering, 2006, 32(2): 69 - 82.
- [4] Humphrey W S. 软件过程管理[M]. 高书敬,顾铁成,胡寅译. 北京:清华大学出版社, 2003: 342 - 366.
- [5] Biffi S, Halling M. Investigating the Defect Detection Effectiveness and Cost Benefit of Nominal Inspection Teams[J]. IEEE Transactions on Software Engineering, 2003, 29(5): 385 - 397.
- [6] Wagner S, Seifert T. Software quality economics for defect - detection techniques using failure prediction[J]. ACM SIGSOFT Software Engineering Notes, 2005, 30(4): 1 - 6.
- [7] Crosby P B. 质量免费: 确定质量的艺术[M]. 北京: 中国人民大学出版社, 2006: 2 - 15.
- [8] Zahran S. 软件过程改进[M]. 陈新, 罗劲枫, 等译. 北京: 机械工业出版社, 中信出版社, 2002.