

软件工程化实施问题与对策

左永义, 唐 帅, 易大尚

武汉东湖学院 湖北武汉

【摘要】在调研航空制导武器系统软件开发现状的基础上,对软件工程化实施中的问题进行分析,并提出相应的对策与改进建议,期望提高软件工程化水平和软件产品质量。要加大软件测试技术的研发力度,从而让软件测试技术得到迅猛发展,为我国科学技术的进步提供更多的帮助。

【关键词】软件工程; 软件质量; 正向设计; 软件过程改进

Problems and countermeasures of software engineering implementation

Yongyi Zuo, Shuai Tang, Dashang Yi

Guodian Electric Power Development Co., LTD. And Yu Hydropower Development Co., LTD., Benxi, Liaoning
China Water Northeast Survey, Design and Research Co., LTD., Changchun, Jilin

【Abstract】On the basis of investigating the current situation of the software development of the aviation guidance weapon software system, the problems in the implementation of the software engineering are analyzed, and the corresponding countermeasures and improvement suggestions are put forward to improve the level of the software engineering and the software product quality. We should strengthen the research and development of software testing technology, so that the software testing technology can develop rapidly, and provide more help for the progress of China's science and technology.

【Keywords】software engineering; software quality; forward design; software process improvement

随着软件越来越多地被用于航载兵器的开发,软件的品质对其性能的影响越来越大,因而,软件工程的开发受到了广大的客户和工程承包商的关注;在软件定价、软件贯标、软件定型和软件工程检验等方面,使用者对软件开发的需求越来越大,软件开发企业也在继续按照软件开发的实际需求进行软件开发。通过对国内外专家学者的深入调查和专家们的讨论,总结出在软件工程应用方面存在的问题,并提出解决方案和改进措施。

1 应用工程软件的情况

在 GJB5000A 项目中,系统承接单位以 GJB5000A 为例,在规范、流程、模板等多个层面上,构建了较为完备的软件工程管理制度。在特定的模型开发流程中进行工程管理:开发产品的模型工程概要,以清楚说明产品的设计需求;采取软件开发进度跟踪,软件工程专项审查,进场软件状态清查,外协软件的实施;通过对模型软件工程化的分析和整改,持续完善软件工程系统文档;以“P-D-C-A”

为循环,实现对软件项目的不断改善。

2 问题及成因

通过不断的完善,软件开发企业的软件工程管理已初见端倪,但是软件正向设计、文档质量、测试效果以及算法设计管理等问题仍未得到有效的改善。

2.1 没有高效地进行前向软件的开发

根据软件工程的需要,编写程序时要遵循程序的需求和程序的最终结果;但是,当前软件开发中的软件代码和后补设计文件的问题依然很多,软件开发并没有遵循“先设计后确认”的正向流程,而是通过代码调试和不断试错来确定设计要求。造成这一问题的原因是:(1)没有在系统层面上进行积极的开发,没有对软件进行具体的任务,没有清晰的任务,没有清晰的开发信息;(2)模型线强调结果,轻过程,只注重软件的发布,而忽略了开发过程;(3)软件开发中的“三分离”的基本原理还没有实现,软件需求分析、软件设计、软件编码等工

作往往都是一个人来做, 软件开发人员可以省略分析设计, 然后编写相应的文件。(4) 项目进度不够科学, 软件的分析 and 开发周期短, 难以进行相关工作; 五、不能标准化的复用, 参考其它机型开发时, 只对程序进行代码的更改, 而忽视了对程序的解析和开发。

2.2 一般的软件文件品质不佳

软件文件记录了软件的需求和设计成果, 是软件开发、测试和更新的基础, 是决策评审中的一个重要环节。在三个方面进行了内部和外部三个方面的测试之后, 文件问题占据了全部问题的大多数, 约为 65%。由于对软件的要求和设计的不完整和不细致, 会对软件的质量和测试的结果产生很大的负面作用, 而且在进行定型评审时要耗费很多的时间来完成。软件文档粗糙, 软件产品文实不统一是软件开发和维护的潜在风险。导致产品文档品质差的主要因素有: (1) 模型线重代码、轻文档, 注重文档的存在而忽略文档的品质, 而忽略了文档的需求粒度、文档页数等相关指标; (2) 开发人员兼顾代码编写、调试试验和文件编写, 没有足够的时间和资源; (3) 评审工作的不严谨, 主要体现在评审前没有及时下发被评文档、同时突击评审多个文档、评审未对照检查单有针对性地进行。

2.3 在软件中的性能不佳

软件试验部进行了单元、组件、配置和系统层次的内部测试, 使软件的性能得到了极大的提高。但是, 在内部检测之后, 三个部门和决策部门仍然可以发现更多的问题, 这表明了系统开发部门在内部测试方面还有很大的改进余地。造成企业内部检测结果不佳的主要因素是: (1) 检测人员的素质还不完善: 大部分从事计算机技术工作的测试者, 缺乏对武器和嵌入式软件的开发实践; (2) 测试项目较多, 测试类型较多, 测试人员的工作能力较差; (3) 模型生产线对软件的试验工作所需的工作时长非常少, 试验时间不足, 使试验工作无法全面、深入地进行。

2.4 不合理的算法开发与管理

当前, 软件开发部门都有专业的算法工程师进行算法研发, 该软件具有很高的依赖度, 在开发过程中, 大部分修改都与其有关。但是, 现有的算法在开发过程中没有明确的位置, 没有统一的标准和

模板, 也没有将其融入到技术的管理之中, 使得算法的设计理念很难在软件文件中得到很好的表达, 使得开发和测试的人无法完全了解它的概念, 从而给系统的性能和性能造成了负面影响。

3 对策与改善意见

本文认为, 在今后的工程应用中, 应采取以下措施逐步改进。

3.1 向前开发的软件

(1) 推进开发中的积极设计

提出了建立以模型为核心的系统工程, 以提升系统的正向设计水平, 为今后的系统正向开发打下坚实的基础。提出了在开发过程中, 要对各个开发期进行适当的划分, 在开发过程中采用数码原型进行系统的需求分析与体系结构, 明确系统的功能指标, 明确系统的功能、性能、界面关系, 明确软件所要完成的功能, 明确各个模块的接口关系。

(2) 全力推动 GJB5000A3 次标准的实施

GJB5000A 三级工艺文档对开发的程序进行了详细的说明, 并对开发过程的标准化进行了严格的监督和检查。在 GJB5000A 软件开发过程中, 严格执行 GJB5000A 软件开发“三分离”的基本要求, 并对软件开发周期进行科学的规划, 确保软件需求的获取、需求分析、软件设计等方面的有效实施, 从而推动软件开发人员按规范的程序进行软件开发, 提高软件的质量。

(3) 软件重用库的持续丰富和软件重用的标准化

在软件复用过程中, 可以降低重复劳动, 增加生产力, 缩短软件的研发时间。根据 GJB5000A 的标准, 开发单位已经制定了重用库和重用库的管理标准, 并按照规定对复用模块进行需求、设计和代码的梳理, 确保复用模块的元素的完整性; 在进行软件复用的过程中, 需求、设计和代码的充分复用应该遵循前向设计流程。

3.2 改进软件文件的品质

要加强对设计审核的管理, 不断改进审核的品质, 以提高系统的文件品质。评审委员会应当邀请同行评审之前, 应当将受评文件预先分发, 评审工作要严格按照评审表执行, 确保评审时间, 杜绝一次性突击评审多份文件的现象, 评审发现的问题应如实记录并逐一纠正, 评审组长应严把评审质量关,

软件质量保障人员应对评审的规范性和有效性进行监督检查。

3.3 提高软件的性能

(1) 将试验融入设计过程, 提前进行可测度的研究

在 GJB5000A 标准的基础上, 采用 V 模式作为标准进行软件开发, 将测试融合到整个系统的开发中, 同时进行测试、规划和测试用例的开发; 将软件的检测从过去的“事后检查”转变为“预先防范”, 及早地检测出“软件的要求”和“软件设计”中存在的问题, 及早地进行“试验计划”和“试验用例”的开发工作, 及早地找到“问题”, 减少“试验”的准备, 提高“测试”的有效性和有效性。

(2) 软件测试流程的优化和团队测试的有效性

软件单元、部件测试工作向前推进, 软件开发人员在代码结束后进行软件单元、部件的测试, 从而及早地找到并修正软件的问题, 从而缩短软件开发的时间; 软件开发人员的单元级自测结束后, 软件测试部将主要精力放在了组态性和系统性的试验上, 着重于软件的功能、体系结构、软件设计等方面的问题。

(3) 人员知识结构的优化和软件测试的提高

本课题旨在让软件研发部门的单元和部件测试人员全面了解软件测试的原理、方法和测试手段; 采用人员轮换、产品设计人员进入软件试验部门等方式, 以加深对产品的了解; 通过与同行的专家进行经常性的沟通, 持续改进软件的测试技术; 通过专家评审、抽查互检等手段, 提高软件的性能和性能。

3.4 标准的规则设计与控制

通过对产品开发过程中的各个环节进行分析, 确定算法的开发过程, 确定算法和软件开发之间的联系, 确定算法的编制标准和范本, 并将其应用于技术状况的管理; 以上的要素被确定后, 将其作为公司的规范, 并在开发中贯彻执行。提出了一种新的方法, 提出了一种新的基于该方法的方法, 该方法开发人员将其应用于该方法的开发过程中, 并对其进行了相应的研究。

3.5 优化人员的分配

软件工程界的“三分离”的基本原理是: 要推进正向设计, 确保软件质量, 因此, 使用者和相关

部门一再重申: “要把软件设计队伍和程序设计队伍分开, 编程队伍和测试队伍分开”。但是, 现行的“三分离”制度并没有得到有效执行, 一个或两个人就能开发出一个完整的软件, 由于缺乏人手, 导致了“三分离”的实施。另外, 软件工程需求的落实还需做很多的工作, 如软件文件质量不高、测试效果差、全面贯标难以执行等。所以, 公司的人事管理部门应该对目前的软件行业进行调研和分析, 并在必要时对其进行适当的补充。近年来, 随着 GJB5000A 系统的全面推行, 很多军事部门都把软件开发的零散的研发人员进行了统一的组织, 成立了软件需求分析、软件设计、软件编码和软件试验小组, 使软件“三分离”, 促进软件正向设计, 推进软件工程化的实施, 取得了很好的成效。

3.6 强化外部协作的工程管理

随着企业间的合作日益增多, 企业必须强化外部合作的技术支持。样板生产线要对外部软件清单进行彻底的整理, 明确管理对象, 按照规定选用符合 GJB5000A 评价的、具备软件承制资质的外协企业, 在签署外协软件时, 要在技术文件中明确软件工程管理的具体规定, 在技术附录中要向外协企业提供软件三方测试要求, 型号线应对外协软件技术协议进行审查, 在软件开发过程中应定期开展外协软件的工程化检查, 并参加任务书、需求等重要环节的文件审核, 在外协软件交货之前, 要组织软件验收审查, 保证承制单位将软件工程化要求落在实处, 确保外协软件质量满足需求。

4 总结

随着计算机技术的不断发展, 计算机技术的发展也日趋成熟, 其性能也日趋完善, 而随着计算机技术的不断发展, 其故障对军事装备造成的冲击也日趋严重。目前, 我国研制的飞机导航武器装备的工程应用中, 还面临着许多亟待解决的问题。

参考文献

- [1] 秦静, 周东生, 赵宏伟, 李长涛. 立体化软件工程专业课思政教育案例设计与实施[J]. 大连大学学报, 2022, 43(02): 113-118.
- [2] 王勇. 软件工程化实施中的问题与对策[J]. 电子技术与软件工程, 2018(19): 58-59.
- [3] 井涛. 面向特定领域的软件工程管理集成平台的研究与

- 实践[D].中国石油大学(华东),2018.
- [4] 黄洋.复杂航空软件工程化管理研究[J].航空标准化与质量,2017(06):35-38.
- [5] 井涛.代码审查在软件工程实施中的重要性[J].电子技术与软件工程,2017(21):43-44.
- [6] 孙琦龙.面向工程化的软件开发人才培养体系研究与实践[J].微型电脑应用,2017,33(04):28-30.
- [7] 缪万胜,周磊,李园. 流程再造,基于信息化的软件工程化实施与改进[C]//2013年中国航空学会管理科学分会学术会议论文集.,2013:728-735.
- [8] 王伟.软件工程与软件可靠性——第六讲 软件可靠性工程实施中的几个技术问题[J].质量与可靠性,2001(06):40-43.
- [9] 李旭亮.关于软件工程标准的实施问题[J].航天标准化,199

6(05):14-18.

收稿日期: 2022年10月12日**出刊日期:** 2022年11月16日**引用本文:** 左永义, 唐帅, 易大尚, 软件工程化实施问题与对策[J]. 工程学研究, 2022, 1(5): 31-34

DOI: 10.12208/j.jer.20220158

检索信息: RCCSE 权威核心学术期刊数据库、中国知网 (CNKI Scholar)、万方数据 (WANFANG DATA)、Google Scholar 等数据库收录期刊**版权声明:** ©2022 作者与开放获取期刊研究中心 (OAJRC) 所有。本文章按照知识共享署名许可条款发表。<https://creativecommons.org/licenses/by/4.0/>**OPEN ACCESS**