

中国造船工程学会人才与教育学术委员会教材建设学组推荐

舰船现代化丛书

软件工程管理实践

杨世林 编著



哈尔滨工程大学出版社

中国造船工程学会人才与教育学术委员会教材建设学组推荐

舰船现代化丛书

软件工程管理实践

杨世林 编著

哈尔滨工程大学出版社

内 容 简 介

本书比较全面地介绍了软件工程管理理论、方法和实践。全书共 10 章,首先讨论了软件开发的现状和软件工程管理要求;其次详细介绍了软件过程建立、分配需求管理、软件开发过程管理、软件测试过程管理、软件评审过程管理、软件质量管理、软件开发过程管理和软件配置管理的方法与实践,最后简要介绍了基于 SW-CMM 等级 2 的要求实施软件过程改进的基本方法与要点。

本书强调理论与实践相结合,以全面、实用和可操作为宗旨,可作为软件工程管理培训教材和研究生、本科生教材,也可作为软件工程技术人员和管理人员的参考书。

图书在版编目(CIP)数据

软件工程管理实践/杨世林编著. —哈尔滨:哈尔滨
工程大学出版社,2006

ISBN 7-81073-897-6

I. 软… II. 杨… III. 软件工程-管理
IV. TP311.5

中国版本图书馆 CIP 数据核字(2006)第 092432 号

出版发行 哈尔滨工程大学出版社
社 址 哈尔滨市南岗区东大直街 124 号
邮政编码 150001
发行电话 0451-82519328
传 真 0451-82519699
经 销 新华书店
印 刷 黑龙江省教育厅印刷厂
开 本 787mm×1 092mm 1/16
印 张 15
字 数 355 千字
版 次 2006 年 10 月第 1 版
印 次 2006 年 10 月第 1 次印刷
印 数 1—1 000 册
定 价 55.00 元

<http://press.hrbeu.edu.cn>

E-mail: heupress@hrbeu.edu.cn

前 言

进入 21 世纪的 IT 产业,软件工程倍受关注。软件因其特殊性和复杂性,使软件项目较之其他项目而言难以管理。由于软件是知识产品,进度、成本和质量难以控制,因此,软件项目的成功率很低。

国内外的软件工程经验表明,实现软件工程化是软件组织提高软件过程管理和软件产品质量的必由之路。软件工程化包括软件工程化开发(即按照软件工程方法开发软件)和软件工程管理(即按照软件工程方法管理软件开发)。

从我国软件开发的现状来看,大约有 70% 的软件组织不成熟(即处于 CMM 初始级)。不成熟的软件组织的软件开发很难称得上是“工程”,因为软件组织常常会由于软件项目的管理不善而导致失败或失误。实际上,有许多软件项目在软件定义阶段就存在着失败的风险,一些软件项目往往在软件开发过程中一直就没有稳定的分配需求,甚至有些软件项目的顾客在验收软件产品时才学会提出分配需求,从而使软件产品的更改没完没了;加之,软件项目策划结果与实际偏差太大,又增加了软件项目失败的因素。由于软件组织的软件工程管理薄弱和软件本身的知识密集性,使软件开发过程很难控制,个人英雄主义普遍存在,致使软件项目的成败仅仅把握在个别人手里……

软件项目为何屡屡失败?美国国防部曾立题专门研究软件项目成功率低的原因,发现有 70% 的失败项目是因为管理不善引起的,而不是因为技术实力不够。他们进而得出一个结论,即管理是影响软件项目全局的因素,而技术只影响局部。因此,软件工程管理至关重要。对于较为复杂的软件系统而言,软件工程管理已经成为决定软件项目成败的关键。

软件项目的成功率与软件组织的成熟性和软件工程化密切相关,而软件项目的成败则更多依赖于项目管理者的管理能力和软件项目团队的协同工作能力。

作者根据我国目前软件工程管理和软件过程改进的实际需要,在参考国内外相关标准、著作和资料的基础上,总结了作者多年来的软件开发和管理的实践经验,编写了本书。

本书共分 10 章,第 1 章简要地描述了软件开发的现状和软件工程管理要求,第 2 章至第 9 章分别描述了软件过程建立、需求管理、软件开发过程管理、软件测试过程管理、软件评审过程管理、软件质量管理、软件可靠性管理和软件配置管理的方法与实践,第 10 章描述了软件组织基于 CMM2 进行软件过程改进的方法与要点,以使组织逐步走向成熟。

全书贯穿了作者的全面、实用和可操作的写作宗旨。然而,作者虽辛勤笔耕,但毕竟时间仓促,水平有限,加之篇目繁多,书中难免存在不妥和错误之处,恳请广大读者批评指正。

在本书的写作和出版过程中,得到了中国船舶重工集团公司第七〇九研究所和哈尔滨工程大学出版社的大力支持,同时得到了七〇九所培训中心顾红主任的大力支持和帮助,在本书初稿完成之后,七〇九所王振宇研究员(博导)和武汉科技大学计算机学院院长陈建勋教授审阅了本书全部初稿,并提出了许多宝贵意见,在此谨向上述单位和个人以及对本书出版给予支持和帮助的人们表示衷心的感谢!

杨世林
2006 年 5 月

目 录

第 1 章 绪论	1
1.1 概述	1
1.2 相关术语	3
1.3 软件的特点	4
1.4 软件工程化	5
1.5 软件工程管理的要求	7
1.6 软件工程管理的主要内容	11
第 2 章 建立软件过程	14
2.1 概述	14
2.2 相关术语	14
2.3 软件生存周期过程	16
2.4 软件生存周期模型	26
2.5 建立软件过程	38
第 3 章 需求管理	40
3.1 概述	40
3.2 相关术语	41
3.3 确定需求	42
3.4 管理需求	44
第 4 章 软件开发过程的管理	50
4.1 概述	50
4.2 相关术语	50
4.3 软件项目策划	52
4.4 软件需求分析管理	56
4.5 软件概要设计管理	60
4.6 软件详细设计管理	63
4.7 软件实现管理	66
4.8 软件测试管理	71
4.9 软件重用管理	72
第 5 章 软件测试管理	74
5.1 概述	74
5.2 相关术语	74
5.3 软件测试的 V 模型	76
5.4 软件测试管理	78
第 6 章 软件评审管理	90
6.1 概述	90

6.2	相关术语	91
6.3	软件评审的一般要求	92
6.4	软件研制各阶段的评审	96
6.5	提高软件评审效果	111
第 7 章	软件质量管理	112
7.1	概述	112
7.2	相关术语	114
7.3	软件质量管理	115
7.4	改变软件开发现状,提高软件质量	126
7.5	软件质量管理与软件项目的关系	127
第 8 章	软件可靠性管理	129
8.1	概述	129
8.2	相关术语	130
8.3	软件可靠性基础	131
8.4	软件可靠性模型	132
8.5	软件失效分类与软件错误分类	135
8.6	软件安全关键等级	136
8.7	软件开发过程的可靠性管理	136
8.8	软件失效案例分析	155
第 9 章	软件配置管理	156
9.1	概述	156
9.2	相关术语	158
9.3	基线技术	159
9.4	软件配置管理的任务	161
9.5	软件配置管理要求	162
9.6	实施软件配置管理	164
9.7	软件研制过程的配置控制	170
第 10 章	基于 SW - CMM 的软件过程改进	173
10.1	概述	173
10.2	相关术语	174
10.3	软件组织的成熟性特征	176
10.4	软件过程改进方法	177
10.5	软件过程改进准备	179
10.6	SW - CMM 简介	183
10.7	SW - CMM 在我国的应用	189
10.8	SW - CMM 等级 2 的实施要点	190
10.9	从 CMM 到 CMMI	222
缩略语表		228
参考文献		230

第 1 章 绪 论

1.1 概 述

软件工程管理是软件组织按照软件工程方法对软件研制全过程进行的管理,其目的是为了使软件项目获得成功。软件工程管理任务是对软件项目的分配需求、软件项目策划、软件需求分析、软件设计、软件实现、软件测试、软件验收与交付的全过程,进行一系列有组织的、有计划的、协调一致的管理和监督工作,从而提高软件项目开发的成功率。

1.1.1 软件开发的现状

在软件开发的实践中,人们常常会发现大多数软件项目实际的开发周期要比预期的长,实际的费用要比预算的多,实际的质量要比预期的差。虽然软件工程理论和方法在以超乎想象的速度发展,但是我们仍然不能有效地解决软件研制过程中遇到的各种问题,导致软件项目的成功率很低。

软件项目成功的标准是软件组织在预期的进度和经费内开发出符合分配需求的软件产品。但遗憾的是,这样的软件组织只是极少数,多数软件组织往往因软件项目管理不善而使软件开发的成功率很低。据有关资料报道,美国 1995 年在 IT 上投入 250 亿美元用于 175 000 个软件项目开发,其结果有 31% 的软件项目在项目完成前被取消,有 53% 的软件项目的成本为最初估计成本的 189%,在取消的软件项目上花费了 8 亿美元,最后只有 16% 的软件项目能按时、按预算完成。

我国软件开发的成功率比美国低,软件质量比硬件质量低得多,软件的失效率往往比硬件失效率高一个数量级。如某系统在联试中发现了 243 个故障,其中软件故障占 91%,硬件故障占 5%,工艺与人为因素造成的故障占 4%。这一事实说明,软件故障是导致系统故障的主要因素。

由于系统中的软件规模越来越大,软件种类日益增多,软件工程管理技术滞后于软件开发技术的发展,加之计算机软件本身的特点,使得软件工程管理非常困难,其原因如下。

1) 大多数软件组织没有将软件当作产品来进行开发和管理,只把软件当着硬件产品的一个附属品。

2) 在签订产品合同时,软件未作为产品单独签订合同,费用与进度往往未纳入系统或设备中统一安排,也未单独考虑软件的费用,在进度上未充分考虑软件开发周期的特殊性,往往未对软件测试安排足够的时间。

3) 对于嵌入式软件,更是存在一个误区,认为嵌入式软件最终是要嵌入到目标计算机中去的,因此,将嵌入式软件的开发过程完全与硬件研制混为一同,主要表现在以下几个方面:

- (1) 软件没有单独的研制任务书;
- (2) 软件不按软件过程而按硬件研制过程开发;
- (3) 软件没有单独的测试而用软件与硬件的联试来代替软件测试;

(4)软件没有单独的评审;

(5)有的软件甚至连文档也是与硬件合在一起编写的,严重影响了软件工程方法的应用。

4)软件没有单独的可靠性指标,产品的可靠性指标往往 100% 地分配给了硬件,设计者潜在地认为软件可靠度为 1。

5)系统的总体单位往往不抓配套单位分系统的软件质量,总体单位只抓本单位软件开发,对配套单位承担的分系统软件开发过程监控不力,对配套单位交付给总体单位的软件普遍不进行软件验收测试和评审。

6)软件设计文档的编制不规范,不符合软件工程标准的要求,文档编制水平和文档质量参差不齐,特别严重的是有些软件文档是在软件实现(编码与单元测试)之后补写的,严重违背了软件工程的原理。

7)软件配置管理不严格,不到位,有的软件组织未指定软件配置管理员,对软件配置管理的概念(如:“软件三库”、“配置标识”、“配置控制”等)认识模糊,缺乏合适的软件配置管理工具,对软件开发库、受控库和产品库的设置和管理不当,导致软件在交付之前基本不受控。

8)对软件测试的概念和过程认识模糊,普遍存在将系统联试等同于软件系统测试的错误认识,专职的软件测试人员不足、经验不足,不能满足工程实际需要,测试环境、工具和手段不够完备,现有测试工具无法覆盖被测试软件的所有需求,担心因软件测试导致经费不足和工作量增加而延误工程进度。

9)在软件开发过程中对软件的资源(包括人员、设备、工具和经费等)投入不足,难以做到软件设计、实现、测试三分开。

1.1.2 管理是软件项目成功的关键

据有关文献报告,美国国防部曾组织软件工程专家研究软件项目失败的原因,发现有 70% 的失败项目是因为管理不善引起的,而不是因为技术实力不够。但迄今为止,美国的软件项目开发的状况仍未得到明显改变,软件项目的成败仍然难以预测,软件工程管理仍然是软件开发成败的主要因素。

由于软件产品的特殊性,其进度和质量很难控制,生产效率也很难以保证,有些软件系统的复杂程度也是超乎想象的。例如,宇宙飞船的软件系统源程序代码多达 2 000 万行,如果按过去的生产效率每人每年只能写 1 万行代码的话,将需要 2 000 人/年才能完成代码编写,这是非常惊人的。

软件开发是工程,不是艺术。很多人将软件开发当成艺术创作来充分展示个人的技巧,艺术创作固然能使心灵手巧的能工巧匠们发挥其个体特长,但是,如果这些能工巧匠们只专注于个人的小天地,而忽视“大局面”,则团队中各成员将无法形成合力,这样的团队通常会出现“ $1+1<2$ ”的情况,有时甚至会出现“ $1+1<1$ ”的惨况。现代软件系统大多是复杂的系统,需要由软件开发团队来承担,虽然软件开发团队中的各成员都尽可能地发挥其各自的艺术才能进行软件艺术创作,但如果不充分发挥团队的力量,有谁能相信这个团队将会创作出使顾客满意的软件产品,即使偶尔有个别软件项目获得了成功,又有谁能保证在今后的软件项目创作中能重复以前的成功?也就是说,靠个人技巧和艺术创作方式来开发软件的风险难以预测,其开发的成功率极低。因此,我们必须在充分发挥软件人员的技术优势的同时,按照软件工程的方法来管理软件项目,才能提高软件开发的成功率。

任何工程的成败,都与管理的好坏密切相关,软件工程也不例外,尤其是对于软件这种特殊产品,管理就显得更为重要。软件项目的成败和软件产品质量的好坏,完全取决于软件组织的管理水平和软件工程化程度。

软件组织为了提高软件开发的成功率,就必须充分发挥团队的协作精神,依靠团队各成员的共同努力和密切配合,在采用适宜的软件开发方法、技术和工具的基础上,按照软件工程方法管理软件开发,实现软件组织的软件工程化,才能真正提高软件组织能力的成熟度和软件产品质量,从而使软件组织在激烈的市场竞争中立于不败之地。

1.2 相关术语

顾客(Customer)

接受产品的组织或个人。

最终顾客(End User)

使用软件在运行环境下实现其预定用途的个人或组织。

最终顾客代表(End User Representatives)

代表最终顾客的个人或组织。

组织(Organization)

职责、权限和相互关系得到安排的一组人员和设施。

过程(Process)

一组将输入转化为输出的相互关联或相互作用的活动。

产品(Product)

过程的结果。

项目(Project)

需要协同工作的一组任务,其目的在于开发和(或)维护一个具体的产品。产品可以包括硬件、软件或其他成分。一般项目有自己的经费、成本核算和交付进度。

软件工程(Software Engineering)

软件开发,运行,维护和引退的系统方法。

软件工程环境(Software Engineering Environment)

软件工程项目所需的一组自动工具、固件和硬件的集合。自动工具可包括(但不局限于)编译程序、汇编程序、连接程序、装入程序、操作规程、排错器、模拟器、仿真器、测试工具、文档编制工具和数据库管理系统等。

软件生存周期(Software Life Cycle)

从构思软件产品开始到软件不再使用时为止的时间周期。软件生存周期一般包括定义阶段、需求阶段、设计阶段、实现阶段、测试、验收和交付过程、运行和维护阶段,有时还包括退役阶段。

软件维护(Software Maintenance)

修改交付后的软件系统或成分,以纠正其差错、改进其性能或其他属性或者使其适应改变后的环境的过程。

供方(Supplier)

提供产品的组织或个人。

1.3 软件的特点

1.3.1 软件的共同特点

由于各类软件的用途不同,运行的环境也不相同,各类软件具有的特点也不尽完全相同,但各类软件有以下共同特点。

1)软件是一种无形的逻辑实体。

2)软件的生产与硬件的不同,软件产品的生产过程就是软件开发过程,软件产品不存在原材料的问题,其生产成本主要体现在软件的开发费用上,软件开发完成后,通过复制就能产生批量软件产品。

3)软件一旦成为产品,它具有“千篇一律,一成不变”的特点,在未经人为地改动之前,它不会改变原来的状态(如果软件没有遭到病毒、黑客等人为因素和环境因素破坏的话)。

4)在软件的运行和使用期间,软件产品不存在磨损和消耗问题,它不会因为使用时间过长而老化或损坏。

5)软件产品的生产主要是靠脑力劳动来完成的,难以摆脱手工作坊式的生产方式。

6)软件具有易改变性。如果软件管理不善,软件开发者随时都会轻易地通过修改软件来改变其技术状态。

7)软件的开发和运行常常受到计算机系统的限制,对计算机系统有着不同程度的依赖性。

1.3.2 嵌入式软件的特点

随着信息技术的飞速发展,嵌入式软件已广泛地应用于单片机、单板机等各种嵌入式系统中,这一特点在起控制作用的设备中更为突出,特别是在传感器和数据采集用的 A/D、D/A 接口以及其他控制信号的接口;嵌入式软件除了数学模型的计算与控制外,还大量用于对被控制对象信号的接收与发送,以及大量的数据采集。被控对象既有电子设备,也有机械设备。

嵌入式软件除了具有 1.3.1 所述的软件共同特点外,还具有以下特点。

1)实时性

有的控制系统对实时性要求很高,而给出的时限要求又苛刻,在运行过程中要大量采集各种数据,在确保被采集的数据正确有效的前提下,还需要进行实时加工处理,并按照处理结果去控制被控制部件而作出实时反应。这样的软件既要保证被采集的数据不丢失,又要保证其正确有效,还要进行复杂的加工处理;既要保证安全可靠,又要保证及时有效。

2)时序性

有的系统在接收或发出控制信号时有严格的时序关系,不仅要求产生的信号能发出去,而且还要满足严格的时序关系。如果不能满足时序关系,则容易出现信号产生错误,可能会因此出现控制信号混乱而导致不可预测的后果。

3)适应性

软件一般都是在实验室开发的,各种调试、测试和试验大多是在实验室环境下进行的,而实验室环境与实际环境有很大差别,这就需要在软件开发过程中充分考虑软件的使用环境,使软件产品适应于软件的实际运行(目标)环境。

1.4 软件工程化

由于软件产品的特殊性,使得其质量是由软件开发过程来决定的。软件工程技术发展的历史表明:软件工程化是提高软件产品质量的有效方法。软件工程化包括用软件工程方法开发软件(即软件开发工程化)和用软件工程方法管理软件开发(即软件工程化管理,也称“软件工程管理”)。

为了确保软件产品符合顾客的需求,确保软件项目获得成功,我们必须一手抓软件开发工程化,一手抓软件工程管理。两者不可或缺,软件产品与其他产品一样,其质量是开发出来的,也是管理出来的。

1.4.1 软件开发工程化

软件开发工程化是指按照软件工程方法开发软件。软件工程方法学来源于工程学的工程规范,诸如土木工程和机械工程那样,需要按照策划的安排有计划地分阶段完成每一项工程任务,才能最终完成整个工程。在软件项目启动时,软件组织需要根据软件的特点对软件开发过程进行策划,策划时需要选择软件工程方法(如:结构化方法、面向对象的方法、构件化方法、形式化方法等),在其后的软件开发过程中按照策划的安排有计划地完成每一项软件开发任务,最终使软件项目按照计划安排输出符合需求的软件产品。

在软件开发过程中,软件开发团队需要熟练运用软件定义、软件需求分析、软件设计、软件编码和软件测试等技术和方法。

1.4.2 软件工程管理

1.4.2.1 软件管理的意义

随着现代高新技术在产品中的广泛应用,计算机软件在产品中发挥着越来越重要的作用,软件工程管理已受到顾客和组织的高度重视。目前,有些软件组织已制定了软件工程管理文件,成立了软件工程管理组织,对软件开发人员和管理人员进行了培训,软件测试工作也已开始独立于软件设计,使得软件开发在逐步摆脱个体作坊式的生产方式的基础上朝着软件工程化的方向发展。

软件工程管理是保证软件项目成功的基础性工作,通过这些基础性工作可以提高软件项目的成功率。软件工程管理的实践表明:

- 1)没有适当的管理,就无法使软件工程化;
- 2)软件工程项目越大、越复杂,软件工程管理的工作量就越大,其管理的作用也就越大;
- 3)只有通过有效的管理才能使软件组织以最小代价达到工程项目预定的目标,从而确保软件组织在预定费用和进度内开发出符合分配需求的软件产品。

一般而言,成熟的软件组织能较好地实施软件工程管理,软件项目的进度、成本和质量都能控制在其预期范围内。

1.4.2.2 软件组织的成熟与不成熟

不同的软件组织的软件工程化程度不同,其成熟程度也不一样。软件项目成功率较高的软件组织属于成熟的软件组织;反之,属于不成熟的软件组织。

成熟的软件组织有明确的和稳定的软件过程,软件项目管理者时刻监控着软件产品的质量和顾客的满意程度。在判断产品质量和分析产品及过程问题方面有客观的、定量的依据。进度和预算是基于以前的性能数据和当前的实际情况进行的,因而是现实可行的。成熟的软件组织通常都能达到产品的成本、进度和质量的预期结果。

成熟的软件组织的各软件项目组都能有纪律地、一致地遵循一个预先定义的软件过程,因为所有参加者在主观上都了解这样做的价值,而且在客观上也存在支持软件过程实施所必需的基础设施。

不成熟的软件组织没有明确的和稳定的软件过程,其过程一般是由开发者及其管理者在项目进行中临时拼凑而成的。有的软件组织虽然已定义了软件过程,但各项目在软件开发实践中却不能得到严格地遵守和实施。不成熟的软件组织实施软件过程的管理方式往往是反应式的,即在遇到问题以后才作出反应,其管理模式是“消防式模式”;不成熟的软件组织的软件项目管理者通常把精力集中于解决随时可能发生的危机上。由于在制定进度和预算时,并非基于现实和历史经验进行估计,而软件项目管理者却认为在开发过程中出现进度滞后、预算超支等情况是正常现象。当不得已而硬性规定完成时限(即所谓的“后墙不倒”),为了勉强满足进度要求,只好牺牲产品的功能和质量,从而酿成大祸。

1.4.2.3 软件组织逐步走向成熟

进入 21 世纪后,软件产业界已普遍达成了“软件开发是工程,不是艺术”的共识,并根据软件规模将软件项目划分成个体软件项目和团队软件项目,分别建立了个体软件过程(Personal Software Process,简称 PSP)和团队软件过程(Team Software Process,简称 TSP),为软件工程管理提供了依据。

但是,目前我国多数软件组织仍然处于不成熟状态,主要表现在以下几方面。

- 1)需求定义不明确,在软件开发全过程缺乏对需求进行严格管理。
- 2)缺乏一个良好的软件过程。
- 3)软件项目策划不充分,往往未经项目估计就直接编制软件开发计划,修改软件开发计划时也不进行重新策划和重新估计。
- 4)软件项目团队重技术而轻管理,软件项目管理者始终关注软件开发技术而忽视管理的作用,他们往往未依据软件开发计划跟踪和监督软件项目的活动,导致软件项目的跟踪和监督不力。
- 5)对于软件风险,只识别,不控制。
- 6)子合同管理不严格。
- 7)没有经常注意改进软件过程。
- 8)软件质量保证工作不到位,失去过程监控的作用。
- 9)软件配置管理不严格,导致软件更改失控。

面临着的诸多不成熟因素,软件组织必须改进组织的软件过程,加强软件工程管理,更加严格地实施软件工程,确保软件组织在规定的进度和成本内完成符合顾客需求的软件产品,从而使软件组织逐步走向成熟。

1.5 软件工程管理的要求

1.5.1 软件工程标准推动软件工程化

成熟的软件组织的软件开发、维护和管理是遵循组织定义的标准软件过程。标准软件过程是软件组织根据软件工程标准的要求,结合组织的软件能力成熟度和软件产品的特点制定的,因此,软件工程标准对软件组织的软件工程管理起到极大的推动作用。

我国从 1987 起,先后发布了 40 余个国家软件工程标准和 50 多个军用软件工程标准。随着我国软件工程的快速发展,我国每年都制定和发布一些新的软件工程标准,将最新最好的软件工程技术和方法奉献给软件组织,实时地反映了我国软件产业的发展。软件工程标准的广泛应用加快了软件组织实现软件工程化的进程。

进入 21 世纪以后,我国掀起了以软件过程改进为核心的软件工程管理热潮。我国目前已有几十个软件组织通过了 CMM(能力成熟度模型)评估,有些软件组织已通过了 CMM5 级评估。我国军用软件也于 2003 年启动了 CMM 评价工作,发布了 GJB5000-2003《军用软件能力成熟度模型》,并在军用软件组织中开始实施。CMM 为软件组织提供了软件过程改进的框架,以实现不同成熟度的软件组织的软件工程管理目标,软件组织可以根据自身的特点确定软件过程改进目标。CMM 在我国的广泛应用,进一步推动了我国软件工程化进程,为软件组织与国际接轨打下了良好的基础。

1.5.2 软件工程管理的基本要求

1.5.2.1 对分配需求控制的要求

分配需求(或软件研制任务书)是系统分配给软件的系统需求,分配需求包括技术需求和非技术需求,它是系统需求的一部分。软件组织应主动识别、确定或获取分配需求,按照软件工程标准(如:GJB438A-97《武器系统软件开发文档》)的有关要求形成文档化的分配需求,经顾客认可或联合评审后,建立功能基线,纳入软件配置管理。

分配需求的确定和更改时应进行联合(顾客与组织)评审,形成评审报告并予以保存。

1.5.2.2 对软件开发策划的要求

软件组织应对软件开发进行策划和控制,其控制内容应在策划中作出明确规定,策划的结果应形成软件开发计划,软件开发计划的内容主要包括确定项目软件过程、项目目标、软件项目的各类角色及其职责、软件工程技术要求、进度安排、软件开发阶段划分,以及各阶段的输入、输出、验证、风险识别等工作的具体安排。策划时应确定:

- 1)软件项目的用途、范围、目标和目的;
- 2)软件生存周期的选择;
- 3)开发和维护软件所选用的规程、方法和标准的标识;
- 4)待开发软件工作产品的标识;
- 5)软件工作产品及其任何更改的规模估计;
- 6)软件项目的工作量和成本估计;
- 7)使用关键计算机资源的估计;

- 8)软件项目的进度,包括里程碑和评审的标识;
- 9)项目软件风险的标识和评估;
- 10)软件的可靠性、可维护性和安全性;
- 11)对复杂软件的特性分析;
- 12)项目软件工程的设施计划和支持工具计划。

软件组织应将软件开发纳入相应产品的研制生产计划中,确保软件项目的研制按照已定义的软件过程规范有计划地进行。

1.5.2.3 对软件开发的控制要求

软件组织应在受控条件下进行软件开发,按照策划的安排进行控制,其受控条件包括:按照规定的要求编制软件开发文档,在适当时获取软件特性、编程规则、编程语言、命名约定、编码和注释规则等信息;使用适宜的软件开发环境和软件测试环境进行软件开发和软件测试;使用适宜的开发工具、技术和方法实施监视和测量,在交付和交付后活动中实施监视和测量等。软件的开发文档应按照软件工程标准(如:GJB438A)的有关要求进行编制。

1.5.2.3.1 对软件需求分析的要求

软件项目组应依据分配需求细化软件的功能和接口要求,确定软件的适应性、容量、时间、安全、保密、质量、人员因素等要求以及设计约束,完成计算机软件配置项目分解,确定软件的工程要求和接口要求,编制软件需求规格说明、接口需求规格说明,并对软件需求分析阶段进行评审后,建立分配基线。

1.5.2.3.2 对软件设计的要求

软件设计包括软件概要设计和详细设计,软件组织可根据项目的特点在软件项目策划时确定其阶段划分。

软件概要设计的依据是软件开发计划和软件需求分析文档(如:软件需求规格说明和接口需求规格说明)。软件设计者把每个计算机软件配置项的功能分解为若干计算机软件部件,确定软件共享数据,明确计算机软件部件的设计要求,确定其用途和相互间的接口关系,提出计算机软件部件的测试要求,详细记载那些对理解和设计有用的补充工程信息;编写软件概要设计说明、接口概要设计说明和软件测试计划。

对于较小规模的软件,软件项目组可在软件项目策划时将软件概要设计阶段和软件详细设计阶段合并为软件设计阶段,并将软件概要设计文档与软件详细设计文档合并为软件设计文档。

对于极小的软件项目,软件设计文档与接口设计说明也可合并为软件设计文档。

在软件概要设计和详细设计阶段,软件组织应按照策划的安排进行软件概要设计评审和详细设计评审,并形成评审报告,经评审的软件工作产品和评审报告应纳入软件配置管理。

对于较小或极小项目,软件概要设计评审与详细设计评审合并为软件设计评审,并形成评审报告,经评审的软件工作产品和评审报告应纳入软件配置管理。

当软件需求更改时,应按软件配置管理的有关要求对软件设计文档作相应更改。

1.5.2.3.3 对软件实现的要求

软件项目组应按照软件项目策划的安排,依据软件设计文档和软件编码规范将软件详细设计转化为程序代码(源代码),对计算机软件单元进行调试后提交测试;软件测试组依据软件测试计划和软件测试说明实施计算机软件单元测试,确保软件编码符合软件编码规范

和软件详细说明。

为了确保软件实现工作符合软件设计的要求,软件组织应使参与编码的人员参加软件设计评审,以确保影响编码的各种问题得到识别和解决;软件实现人员应使用适宜的编程方法编制程序代码,当程序代码通过测试机构(或人员)的测试时,应按照软件配置管理要求进行控制。当前期(如:软件定义、需求分析、设计等阶段)的软件工作产品需要更改时,软件实现人员应按照软件配置管理的要求作相应更改。

1.5.2.3.4 对软件测试的要求

为了尽可能发现软件的设计缺陷,软件组织应按照策划的安排和软件工程标准的要求进行计算机软件单元测试、计算机软件部件测试、计算机软件配置项测试和系统测试。

在软件测试过程中应制订软件测试计划,设计软件测试用例,编制软件测试说明,形成软件测试记录,编制软件测试报告和软件问题报告,必要时进行回归测试。软件组织应对各阶段的各类测试结果予以记录,并妥善保存。软件测试过程中形成的所有文档和记录均应纳入软件配置管理。

当合同有要求时,软件组织应按照合同的要求和策划的安排将经过自测试的计算机软件配置项(CSCI)提交第三方软件测试机构进行确认测试。

1.5.2.4 对软件质量管理体系的要求

软件组织应按照 GB/T 19001 - 2000 或 GJB9001A - 2001 标准要求建立质量管理体系。软件组织应根据软件项目的特点,策划和实施与质量管理体系有关的活动。

软件质量策划应与产品实现策划相结合,其策划的内容应符合 GB/T 19001 - 2000 或 GJB9001A - 2001 的要求,策划中还应将风险分析与评估工作(在软件开发的各阶段进行风险分析和评估,形成各阶段的风险分析文件,并提供给顾客)纳入计划中,策划的输出形式可以按组织的运作方式给出,并经顾客会签。

1.5.2.4.1 对软件开发过程的质量管理要求

软件组织应针对软件开发过程进行明确规定。

(1)输入:软件开发各阶段的输入应是软件开发相应阶段软件开发的依据。质量管理体系要求每一开发阶段的输入应形成文档,并要求对输入进行评审,确保软件开发各阶段的输入是充分的和适宜的。

(2)输出:输出应是由输入转化而得到的软件工作产品,这些工作产品应与其输入相一致。

(3)验证:软件开发的每一阶段的输出都应能够针对其输入进行验证。

(4)确认:软件的开发确认应在成功地进行验证后,按照软件项目策划安排的时机和方式进行。

当确认需要在现场进行测试时,应明确测试特性、顾客和组织的职责、测试后的现场使用环境的恢复等事宜。

当需要通过定型(鉴定)进行软件的确认时,软件组织应按有关规定完成定型(鉴定)准备工作。

(5)软件更改控制:当软件需要进行更改时,应在更改前对需要更改的内容进行识别,并予以记录,在适当时间应对更改内容进行评审、批准和验证,必要时进行确认,软件更改的评审、验证、确认、批准情况以及更改实施均应形成记录,以便追溯。

1.5.2.4.2 对软件检验与测试的要求

软件组织的检验和测试部门(或人员)应得到组织最高管理者授权,开展软件检验与测试时不受其他部门和人员的干扰,能独立行使其职权,其检验与测试活动直接对最高管理者负责,检验与测试结果不受任何人干扰。

1.5.2.4.3 对软件开发环境和验证环境的要求

为了保证软件的开发质量,软件组织应确定、提供和维护软件开发环境和软件验证环境。软件开发环境是根据软件的特点选择的软件开发工具、开发平台、开发技术和开发方法。软件验证环境是根据软件的特点确定的软件的检验与测试工具与测试平台、软件测试技术与方法。软件验证环境也可包括供方提供的环境。软件开发环境和软件验证环境都是组织为保证软件质量而必须具备的资源,组织必须提供足够的软件开发和验证资源,并按照质量管理体系的要求进行控制。

1.5.2.4.4 对软件产品采购的要求

由供方提供的(包括外购或分包的)软件产品(包括软件工具和与软件配套的软件)应按照 GB/T 19001 - 2000 或 GJB9001A - 2001 的采购要求进行控制。采购的软件产品可能是构成最终软件的一部分(计算机软件配置项、计算机软件部件或计算机软件单元),也可能是开发工具或测试工具,要求把软件产品的采购视同于其他产品的采购进行质量控制。

1.5.2.4.5 对软件一致性的要求

软件组织应对分配需求的获取与管理、软件项目策划、软件需求分析、设计、软件实现和软件测试等过程自始至终地进行管理和控制,对软件工作产品的更改应按照规定的程序办理更改手续,并得到批准后才能实施更改,以确保分配需求、软件需求分析文档、软件设计文档和代码的一致性。

1.5.2.4.6 对软件复制的要求

对于形成批量的软件,软件组织应根据合同的要求和策划的安排进行软件复制,软件复制应考虑:拷贝数量、介质类型、软件的格式和版本、应交付的文档种类、必要时的主拷贝和备份拷贝的保管、提供拷贝的责任和期限以及灾难性故障的恢复方案等等。

1.5.2.5 对软件交付的要求

软件交付时应遵循软件交付准则,在交付前应按照规定的要求作好交付准备。在交付过程中由顾客按照协商确定的软件交付要求进行软件验收,当软件验收结果满足顾客要求时,顾客应出具软件验收合格证明,方可完成软件的交付工作。在软件交付后应做好培训、技术咨询、现场安装或维护等售后服务工作。

1.5.2.6 对软件运行和维护的要求

软件组织在软件开发过程中应按照策划的安排编写用于软件运行和维护的文档。

软件组织应根据合同的要求和策划的安排制定软件的维护计划,成立软件维护组织,确定软件维护活动的类型,确保软件维护结束时仍能保持分配需求、软件需求、软件设计、代码和相关文档的一致性。

1.5.2.7 对软件配置管理的要求

软件组织应策划和建立软件配置管理系统,明确软件配置管理组织及其成员职责,制订软件配置管理计划,落实软件配置管理的任务,按计划安排开展软件配置管理活动。通过建立和实施软件开发库、受控库和产品库(简称“软件三库”)的控制来实现软件配置管理。

软件组织应按照文档化的规程对配置项的状态、更改申请和已批准更改的实施情况,进行记录、管理和报告,并形成文档化的配置状态报告,若顾客有要求时,组织应将配置状态报

告提供给顾客。

1.6 软件工程管理的主要内容

软件工程管理主要包括对软件研制过程的全过程控制和全方位管理。

软件研制过程的全过程控制包括对软件定义过程、软件开发过程(软件需求分析过程、软件概要设计过程、软件详细设计过程、软件实现过程、软件测试过程、软件评审过程和软件验收过程)的控制,并实施对软件工作产品的管理。

软件的全方位管理包括的内容很多,处于不同成熟度的软件组织其管理的内容有很大区别。如:对于已达到定量管理级的软件组织应按照 CMM 的 15 个关键过程域进行管理,并力求达到优化级的要求;而对于不太成熟(未达到可重复级)的软件组织来说,其软件工程管理的主要内容是着重抓好软件需求管理、软件项目策划管理、软件开发管理、软件评审与审查管理、软件项目跟踪和监督管理、子合同管理、软件质量管理、软件可靠性管理、软件配置管理等,力求达到可重复级的要求。

1.6.1 需求管理

需求在软件工程中通常指分配需求或软件研制任务书, GJB5000 - 2003《软件能力成熟度模型》中对分配需求是这样定义的:分配给软件的系统需求。即拟由系统的软件成分实现的系统需求的子集,它是估计、策划、执行和跟踪整个软件生存周期内软件项目活动的依据。

由于需求是正在构建的系统必须满足的要求,而需求是决定项目成败的要素,因此,找出需求是什么,将它们记下来,形成文档并实施管理是项目软件负责人的首要任务。当需求发生变化时应对其进行控制和追踪,对需求变更进行管理的活动就是需求管理。换句话说,需求管理就是一种获取、组织并记录和控制分配需求变更的过程,这一过程应使顾客与项目团队对不断变更的需求达成一致意见并予以保持。

有关需求管理的详细内容将在本书第 3 章讨论。

1.6.2 软件开发过程管理

软件开发过程管理是指使用软件工程的方法管理软件开发过程。

软件开发过程是随着软件开发技术的发展而不断改进的。从早期的瀑布式(Waterfall)开发模型到后来出现的螺旋式和迭代(Spiral)式开发模型,以至最近开始兴起的构件模型,它们展示出了在不同的时代软件产业对于开发过程的不同认识 and 不同类型项目的理解方法。

注意区分软件开发过程和软件过程改进之间的重要区别。诸如像 ISO 15504, ISO 9000, CMM, CMMI 这些标准描述的是一些软件过程改进框架,它们提供了一系列的标准和策略来指导软件组织实施软件过程改进,从而不断提高软件组织能力的成熟度。而软件开发过程包括从软件项目的分配需求经过批准后到软件产品交付给顾客的所有软件工程活动,这些工程活动的管理已经成为软件项目成功的关键。有关软件开发过程管理的详细内容将在本书第 4 章讨论。