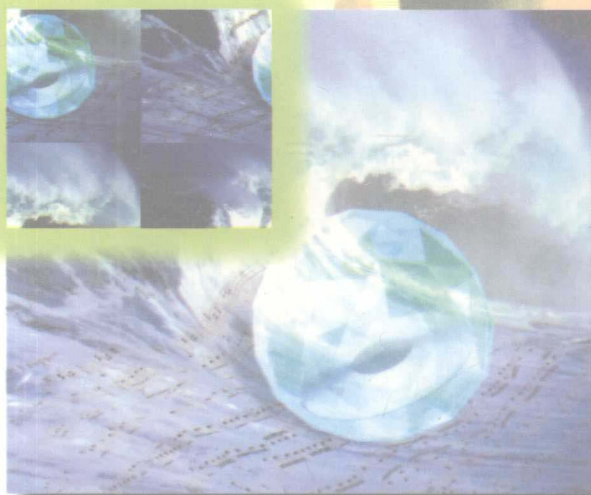


软件工程

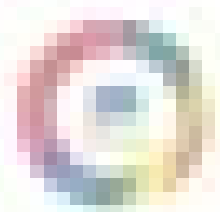
梁昌勇 主编



中国物资出版社

软件工程

（第二版）



中国美术学院出版

软 件 工 程

主审 杨善林
主编 梁昌勇
马溪骏

中国物资出版社

图书在版编目(CIP)数据

软件工程/梁昌勇等主编. —北京:中国物资出版社,
1998. 1

ISBN 7—5047—1350—3

I. 软… I. 梁… III. 软件工程 IV. TP311

中国版本图书馆 CIP 数据核字(97)第 26204 号

责任编辑:王秋萍

特约编辑:沙 金

装帧设计:木 贞

王 磊

责任校对:沙 金

中国物资出版社出版发行

(北京市西城区月坛北街 25 号 邮编:100834)

全国新华书店经销

北京科发文化交流有限公司激光照排

安徽省蚌埠市红旗印刷厂印刷

787×1092 毫米 1/16 印张:13.25 字数:330 千字

1998 年 1 月第 1 版 1998 年 1 月第 1 次印刷

印数:00001—10000 册

ISBN 7—5047—1350—3/TP·0022

定价:17.80 元

总 序

从 1946 年第一台计算机诞生至今已历经半个多世纪。计算机的出现和广泛应用,标志着人类社会的一次大飞跃,信息时代的一次大转折,生存方式的一次大变革,现代文化的一次大融汇。随着计算机技术的飞速发展和广泛普及,其应用已遍及社会生活的各个领域。由于计算机技术已进入到我们生活中的方方面面,人类社会的生活方式、思维方式以及时空观念等各个方面都已经发生了深刻的历史性的巨变。

随着信息化社会的发展,人们对信息交流的要求越来越高。世人已普遍公认:哪个国家的信息化程度高,其经济竞争力就越强,科技发展就越快,办事效率就越高,对下一代的教育条件就越好。信息化的进一步深入需要我们培养一大批高素质人才。当今社会,熟练掌握计算机应用技术已成为高素质人才的必备条件。因此,一个十分艰巨的任务,就是要使受教育者具有使用计算机的能力和与之相适应的计算机文化素质。如果我们的知识结构和文化修养准备不足,就不能适应时代和社会发展的需要。

一本好书,是人生旅途的一掬甘泉;一套好教材,是教学成功的必要条件。广大学子和读者殷殷所望,无非是博得一艺,学有所用。本着对读者负责的精神,我们组织北京电子科技学院、集美大学、合肥工业大学、安徽大学、安徽财贸学院、江苏广播电视大学等高等院校和部分大中专学校教学经验丰富的教师,以及一些具有较高理论基础和软件开发经验的计算机技术人员共同合作编写了这套计算机及应用专业教材。为保证教材的质量,我们还聘请了一批学术造诣较深的专家、教授作为本套教材的主审和顾问。本套教材具有以下几个方面的特点:

首先,作为一套计算机专业教材,必须保证整个计算机知识体系的完整性。本套教材包括必修课 17 种,选修课和配套教材 3 种,基本上涵盖了目前大中专院校计算机及应用专业所必修或选修的课程内容。各种教材在编排上,既注意到内容上的连贯性,又保证了教学上的相对独立性。

其次,在教材内容的组织上,注重介绍和汲取当今计算机领域的一些新技术和新知识,摒弃了传统教材中一些过时的内容,这些变化在各本教材中都得到程度不同的体现。本套教材编写时既参照了有关部委计算机及应用专业教学大纲,又参考了“程序员考试大纲”和“全国计算机水平等级考试大纲”的内容。因此本套教材既适合作为各级各类院校计算机及应用专业教材,亦可作为计算机水平等级考试学习用书。个别教材之间内容上的重复,是为了照顾部分读者单独选用单本教材的需要,敬请广大读者予以谅解。

再次,考虑到各校教学的特点,本着学以致用原则,在本套教材编写中我们始终贯彻“由浅入深,理论与实践相结合”的原则,以阐明要义为主,辅之以必要的例题、习题和上机实习,以便使读者尽快领悟和掌握。

在本套教材编写过程中,各位作者付出了艰辛的劳动,教材编委会的各位专家和教授对各本教材的内容进行了认真的审定和悉心的指导。在本套教材出版过程中我们自始至终得到中国物资出版社领导和编辑以及印制单位的大力支持和帮助。本套教材承蒙中国科学院计算技

术研究所、国家智能计算机研究开发中心王川宝、高文、中国机械科学研究院江波等同志进行了较为细致的终审终校工作。正是由于各方面的通力配合,才使得本套教材得以顺利出版和发行。书中参考、借鉴了国内外同类教材和专著,在此一并表示感谢。

近年来,计算机技术发展日新月异,异彩纷呈,许多新的概念和内容都在不断扩展之中,囿于编者学识和水平,书中疏漏、错误之处还望广大读者不吝批评指正,以便对本套教材不断修订完善。

计算机及应用专业教材编委会

1998 年 1 月

附:计算机及应用专业教材编委会名单

顾 问

(以姓氏笔划为序)

王仲文	北京电子科技学院院长、教授
韦 穗	安徽大学副校长、教授
张全寿	铁道部电子计算中心主任、北方交通大学教授
李文忠	全国计算机基础教育学会副理事长、东南大学教授
杨善林	合肥工业大学副校长、教授、博士生导师
辜建德	集美大学校长、教授
魏余芳	西南交通大学教授

编 委

鄂大伟	集美大学副教授
李树德	北京电子科技学院教授
刘 锋	安徽大学副教授
王川宝	中国科学院计算技术研究所、国家智能计算机研究开发中心 硕士研究生
高 文	中国科学院计算技术研究所、国家智能计算机研究开发中心 博士研究生
江 波	中国机械科学研究院硕士研究生
屈道良	上海铁路局蚌埠分局高级工程师
蒋翠清	上海铁路局蚌埠分局高级工程师

前 言

自从1946年世界上出现第一台电子计算机以来,计算机应用取得了巨大的经济效益和社会效益。计算机应用主要取决于软件,因此软件开发是计算机应用的重要环节。在软件开发早期阶段,由于一些错误的观点和做法,使得软件开发遇到一系列严重的问题,即所谓的“软件危机”。为了解决“软件危机”,在软件开发过程中,人们引用了“工程化”的指导思想,从而产生了一门新的学科——软件工程。

软件工程是在开发软件中因多学科交叉而形成的一门学科,是采用工程的概念、原理、技术和方法来开发和维护软件,把经过时间验证的管理思想、技术方法融合到软件开发过程中去。软件工程与管理信息系统等计算机信息系统是不同的学科,管理信息系统等信息系统是研究在特定应用领域如何实现一个系统功能所涉及的管理、技术和方法,而软件工程则是从抽象的软件开发过程中研究软件开发的工程化问题。软件工程主要研究软件设计方法论,软件工具及其支撑环节,软件管理及软件工程标准和规范等有关理论和方法等。

本书是在我们多年从事软件工程教学和科研的基础上,参考了国内外大量文献资料的基础上编纂而成的。全书全面系统地介绍了软件开发各个阶段的有关理论、方法和工具。本书共分十章内容。第一章简要介绍了软件工程概念、研究内容和面临的问题;第二章至第七章分别讨论了软件生命周期各阶段的任务、实施方法和技术,特别是对有关软件设计方法,如结构化设计方法、Jackson方法、面向对象设计方法等进行了详细的阐述;第八章和第九章还讨论了对软件进行评价和工程管理的內容;第十章对目前软件工程研究热点问题作了简要介绍。本书既可作为大中专学校计算机及应用专业和管理类专业教材,亦可作为软件开发和管理人员业务学习参考。

本书由合肥工业大学梁昌勇(博士研究生)、马溪骏(高级工程师)合作编著。参加编写工作的还有张韶、晏争农、李开忠、蔡红星等同志,由合肥工业大学副校长杨善林教授主审。

本书在编写过程中得到杨善林教授、蒋翠清高级工程师、张翠勤工程师的指导和鼓励,另外在成书过程中还始终得到北京科发文化交流有限公司张辉总经理及其同事的大量帮助。书中参考、引用、借鉴了国内外有关文献资料,在此一并表示感谢!

由于本书编写时间仓促,编者学识水平有限,书中肯定存在许多疏漏和错误之处,恳请有关专家和广大读者不吝批评指正,以便不断修订完善。

编 者

1998年1月

目 录

第一章 绪 论	(1)
§ 1.1 软件和软件工程	(1)
§ 1.2 软件工程原理	(5)
§ 1.3 软件工程的研究内容	(6)
§ 1.4 软件工程开发模型	(8)
§ 1.5 软件开发工具和软件工程环境.....	(11)
§ 1.6 软件工程面临的问题.....	(14)
第二章 可行性研究与软件计划	(16)
§ 2.1 定义问题、确定目标	(16)
§ 2.2 可行性研究.....	(17)
§ 2.3 软件系统边界确定.....	(21)
§ 2.4 资源分析.....	(22)
§ 2.5 软件成本估计.....	(23)
§ 2.6 软件开发进度计划.....	(25)
§ 2.7 软件计划说明书.....	(27)
第三章 软件需求分析	(29)
§ 3.1 需求分析的目标和任务.....	(29)
§ 3.2 需求分析存在的困难.....	(30)
§ 3.3 结构化分析方法.....	(32)
§ 3.4 HIPO 法	(43)
§ 3.5 面向对象的分析方法.....	(48)
§ 3.6 软件需求规格说明书.....	(50)
§ 3.7 审查和复审.....	(52)
§ 3.8 需求分析工具简介.....	(52)
第四章 软件设计	(55)
§ 4.1 软件设计的概念和原理.....	(55)
§ 4.2 软件设计任务和过程.....	(60)
§ 4.3 软件设计准则.....	(64)
§ 4.4 结构化设计方法(SD 方法)	(67)
§ 4.5 Jackson 方法	(84)
§ 4.6 Warnier 设计方法	(93)
§ 4.7 面向对象设计方法.....	(99)
§ 4.8 详细设计	(112)
§ 4.9 设计文档	(122)

第五章 软件编码	(125)
§ 5.1 软件的翻译过程和方法	(125)
§ 5.2 程序设计语言的分类及选择	(125)
§ 5.3 编码风格	(131)
§ 5.4 程序设计功效	(132)
§ 5.5 程序设计途径	(133)
第六章 软件测试	(137)
§ 6.1 基本概念	(137)
§ 6.2 测试过程和策略	(144)
§ 6.3 软件测试技术	(155)
§ 6.4 软件测试规格说明	(158)
§ 6.5 自动测试工具	(159)
第七章 软件维护	(161)
§ 7.1 软件维护的定义及其特点	(161)
§ 7.2 易维护性度量	(164)
§ 7.3 维护过程	(166)
§ 7.4 维护的副作用	(168)
§ 7.5 维护支援工具和技术	(169)
第八章 软件评价	(173)
§ 8.1 软件的质量及其度量模型	(173)
§ 8.2 软件的复杂性和可靠性度量	(175)
§ 8.3 软件的用户评价	(176)
§ 8.4 软件质量保证	(180)
第九章 软件工程管理	(181)
§ 9.1 概 述	(181)
§ 9.2 人员管理	(182)
§ 9.3 组织管理	(183)
§ 9.4 成本管理	(188)
§ 9.5 文档管理	(190)
§ 9.6 软件管理工具	(197)
第十章 软件工程发展展望	(198)
§ 10.1 软件工程发展阶段	(198)
§ 10.2 CASE 工具	(198)
§ 10.3 集成 CASE 环境	(199)
§ 10.4 软件过程研究	(200)
§ 10.5 软件工程方法论研究	(201)
主要参考文献	(202)

第一章

绪 论

自从 1946 年出现第一台电子计算机以来,计算机技术得到了空前的发展,计算机应用已广泛深入到社会生活的方方面面,取得了巨大的经济效益和社会效益。一般说来,计算机是由硬件和软件构成的。硬件是计算机的基础设施;软件是计算机的灵魂,是计算机系统能否充分发挥效能的关键部分。因此,开发研制计算机软件必须遵循一定的目标原则进行设计,才能使最终软件产品能够耦合于计算机系统,以发挥其最大功效。在软件开发早期,由于一些错误的观念和做法,曾使软件开发遇到一系列严重问题,这种现象被一些工业发达国家的计算机科学家称之为“软件危机”,并于 60 年代后期开始认真研究、解决软件开发中存在的问题,从而逐步形成了计算机科学技术领域中一门新学科——计算机软件工程。

本章将就软件、软件工程的有关概念、内容及其发展作一介绍。

§ 1.1

软件和软件工程

1.1.1 软件

什么是软件?人们实际上很难对这一名词作出精确的定义。“软件”这个概念,在不同的时期、不同的人有着不同的理解,随着计算机科学技术的发展和计算机应用的广泛和深入,人们对“软件”的认识也在不断深化。在 70 年代以前,人们普遍认为软件=程序;进入 70 年代,认为软件=程序(主)+文档(从);从 80 年代开始,则认为软件=文档(主)+程序(从)。目前对“软件”的普遍认识是:“软件是计算机系统中与硬件相互依存的另一部分,它是包括程序、过程及其相关文档资料的完整集合,是用户(人)与硬件子系统之间的接口界面。”软件主要由两大部分组成:第一,是可执行部分,即以编码信息形式存放在存贮介质上的程序和过程。程序和过程是一组能被计算机理解和执行指令序列的集合;第二部分,则是与程序和过程有关的文档资料。文档是指用自然语言或者形式化语言编写的文字资料和图表,用以描述程序的内容、组成、设计、功能规格、开发情况及使用方法等。这些文档资料虽然不可执行,但关系到软件能否有效地运行和最佳地维护等一系列问题。对于一个计算机应用系统来说,其软件完整配置由以下四个主要部分构成:

(一)应用程序

又称用户程序,是直接面向用户的、为解决各种特定问题而编写的程序,如物资管理程序、工资核算程序等。

(二) 系统程序

又称支撑环境,它是面向计算机硬件,是将用户程序逐层解释变成计算机可执行程序的程序,如操作系统、数据库系统等。

(三) 面向用户的文档资料

指如何使用、维护应用系统的资料,如用户手册、操作手册等。

(四) 面向开发者的文档资料

指开发者为按期、按质完成软件开发任务而对系统设计的文档资料。如可行性研究报告、程序设计说明书等。

1.1.2 软件的重要性

软件是计算机的灵魂。有人将计算机比喻成一架钢琴,计算机软件就相当于钢琴乐谱,没有乐谱,钢琴是不可能发出优美的旋律的。同样,没有计算机软件,计算机就是一堆废铁,就发挥不了其应有的作用。计算机软件的主要任务是着重解决管好、用好计算机,指挥、监督和维护计算机的正常运行,实现计算机的各种功能和用途,提高计算机的效率,甚至还可以模拟计算机硬件所不具备的某些功能,起到扩充设备的作用。

目前,随着社会向高度信息化方向发展,计算机软件逐渐成为高度信息化社会发展的最重要资源。如果把从 60 年代中期开始计算机带动的社会称为信息化社会,那么当前,由于计算机与通信技术的发展以及新的介质出现,信息化程度已达到了与传统截然不同的新水平,并逐步进入高度信息化社会。今天,通信技术与计算机相互融合形成系统,计算机网络星罗棋布于世界各地,形成全球范围内的 Internet 网络,这一功劳应主要归功于软件。同时,随着社会的发展,人们日益要求信息和通信的适应性和多样性,于是又必然促使信息和通信技术的结合和革新,必然会对软件的需求和开发提出前所未有的新挑战。因此可以说,今后的社会在很大程度上将依赖于软件的发展。现在,不少先进国家把软件开发作为经济、技术、国防竞争的重要手段。例如,海湾战争结束后,美国为保持其综合国力优势,在“美国重要技术报告”中列有六大门类 22 种技术是美国必须依靠的技术,美国政府重点对其支持。其中与计算机科学有关的就有三项,即软件、高性能计算机以及存贮器等外围设备。关于软件技术,报告中指出两个重点:(1)利用计算机设计、开发软件的 CASE 工具;(2)先和用户在短时间内开发出核心部分原型,然后再予扩充。由此可见,选择软件作为重点技术是因为各种高度的自动化系统都是以高可靠性、高性能的软件为基础。

1.1.3 软件危机

60 年代末至 70 年代初期,由于工业发达国家在软件开发上存在着错误的观点和做法,使得软件开发遇到了一系列严重的问题,这种情况被科学界称为“软件危机”。这场软件危机主要表现在以下几个方面:

(一) 软件十分复杂

随着计算机系统的迅猛发展和应用范围的日益广泛,软件的规模越来越大,复杂程度越来越高。工厂管理、银行事务、病人监护、机票预订、飞机模拟系统、飞船控制等各种软件系统常常相当庞大,逻辑上也十分复杂,往往要花几十、几百,甚至几千人年,需要一个小组或集团的共同努力才能完成。例如,美国航空公司 1965 年开始使用的 SABRE 飞机订票系统,1968 年达到

30 万条指令,可同时接 2000 个终端,高峰时间每分钟可处理 3000 件左右的业务。

(二)软件价格昂贵

近几年来软件费用在急剧上升。据美国计算机科学家 Boehm 估计,1980 年美国计算机硬、软件总开支为 479 亿美元,其中软件开支为 402 亿美元。根据美国 TRW 公司的统计预测,1985 年软件开支将占计算机系统总开支的 85%,即硬件和软件的开支比率为 1:9 左右。如 APPLE 公司仅仅为 APPLE 机开发一套完整的应用系列软件产品就曾投资 5000 万美元,开发大的软件系统花钱之多,就更不必说了。软件成为许多计算机系统中花钱最多的项目。

在我国,目前虽然人工工资不高,但从发展趋势来看,软件不值钱的时代将会成为过去。可以预言,随着硬件成本下跌和工时费上升,我国的软件价格将会上升。

(三)软件供需差在不断扩大

由于信息化的迅速发展,对软件需要量大幅度增加,但由于软件生产率很低,软件人员又非常短缺,而且只能提供有限的产品,因此大大限制了软件的供应量,其结果造成软件的供需差有越来越大的趋势。

(四)软件研制过程往往失去控制

在软件研制过程中常常遇到很多困难,有些软件系统的开发彻底失败了;有些系统的开发几经周折虽然最终还是完成了,但比原订计划大大推迟了。如由于 FORTRAN 的 DO 语句中少一个逗号,造成美国第一次飞向金星的试验失败;哥伦比亚号航天飞机推迟首航也是因软件故障造成的;通过反复调试的控制程序仍有可能使导弹发射失败;耗费巨资的超大型软件的开发往往以“难产”或“胎死腹中”而告终;有些系统质量低劣,未能圆满地实现当初的期望,甚至无法修改或维护,最终不得不废弃。此外,我们还看到,开发软件过程中需要投入相当多的组织管理工作,这也是软件本身的特点所决定的,特别是大型软件更是如此。从图 1-1 中可以看出,当所开发的软件具有一定的规模时,需投入的管理工作量几乎和技术工作量相等。因此,软件成为最难设计,最少可能成功(在时间上和成本方面)和管理起来最危险的系统成分。

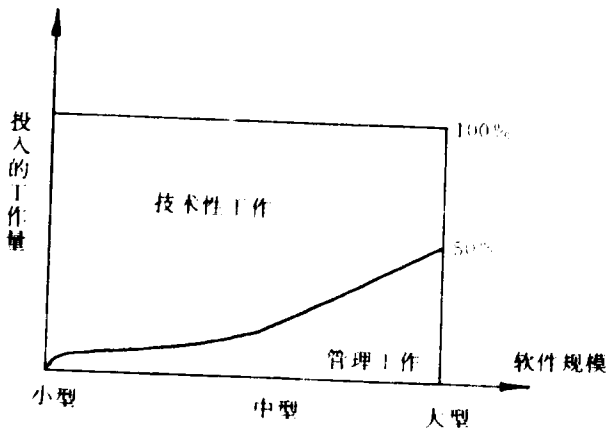


图 1-1 软件开发中管理工作随着软件规模增长而增长

一个典型的例子就是 IBM 公司的 OS/360 系统。这家公司曾经把大量的资金和人力 (5000 人年) 投入到 OS/360 系统的开发工作中去,该系统有 346 万条汇编语句,从 1968 年~1978 年共修改了 21 版,经过了相当长的一段时间,尽管解决了一些问题,但是仍未能取得预

想的结果。摆在设计者面前的是一堆困难和一片混乱,虽然历尽艰辛,作了最大的努力,想摆脱困境却仍不能自拔。

OS/360 系统负责人 BROOKS 曾生动形象地描述过研制过程的混乱和困境:“像巨兽在泥潭中作垂死挣扎,挣扎得越猛,泥浆沾得越紧,最后,没有一个野兽能逃脱淹没在泥潭中的命运……程序设计就像是这样一个泥潭……一批程序员在泥潭中挣扎,没有人料到问题竟会这样棘手……”。

软件十分复杂,价格昂贵,供需差日益增大,软件开发时又常常受挫,质量差,制订的进度表和完成日期很少能够按时实现,研制过程很难管理,即软件的研制往往失去控制。我们称软件开发中遇到的这一系列问题为软件危机。

综上所述,为了扭转软件生产中的这种困难局面,必须对软件开发的方法和理论进行研究,从而产生了软件开发工程化方法——软件工程学。

1.1.4 软件工程

软件工程是指导计算机软件开发和维护的工程科学。软件工程就是采用工程的概念、原理、技术和方法,把经过时间考验而证明有效的管理技术和当前能够得到的最好的技术方法结合起来开发和维护软件。

软件工程强调使用生存周期方法学和各种结构分析及结构设计技术,就是从时间角度对软件开发和维护的复杂问题进行分解,把软件生存的漫长周期依次划分为若干个阶段,每个阶段有相对独立的任务,然后逐步完成每个阶段的任务。目前,一般把软件生存周期划分为:需求分析(系统分析)、软件设计、软件编码、软件测试、软件维护等五个阶段。采用软件工程方法论开发软件的时候,从对任务的抽象逻辑分析开始,一个阶段一个阶段地进行开发。前一阶段任务的完成是开始进行后一阶段工作的前提和基础,而后一阶段任务的完成通常是使前一阶段提出的解法更进一步具体化,并加进了更多的物理细节。每一个阶段的开始和结束都有严格标准,对于任何两个相邻的阶段而言,前一阶段的结束标准就是后一阶段的开始标准。在每一个阶段结束之前都必须进行正式严格的技术审查和管理复审,从技术和管理两方面对这个阶段的开发成果进行检查,通过之后这个阶段才算结束;如果检查通不过,则必须进行必要的返工,并且返工后还要再经过审查。审查的一条主要标准就是每个阶段都应该交出“最新式的”(即和所开发的软件完全一致的)高质量的文档资料,从而保证在软件开发工程结束时有一个完整准确的软件配置交付使用。文档是通信的工具,它们清楚准确地说明了到这个时候为止关于该项工程已经知道了什么,同时确立了下一步工作的基础。此外,文档也起备忘录的作用。如果文档不完整,那么一定是某些工作忘记做了,在进入生存周期的下一阶段之前,必须补足这些遗漏的细节。在完成生存周期每个阶段的任务时,应该采用适合该阶段任务特点的系统化的技术方法——结构分析技术或结构设计技术。

把软件生存周期划分成若干个阶段,每个阶段的任务相对独立而且比较简单,便于不同人员分工协作,从而降低了整个软件开发工程的困难程度;在软件生存周期的每个阶段都采用科学的管理技术和良好的技术方法,而且在每个阶段结束之前都从技术和管理两个角度进行严格的审查,合格之后才开始下一阶段的工作,这就使软件开发工程的全过程以一种有条不紊的方式进行,保证了软件的质量,特别是提高了软件的可维护性。总之,采用软件工程方法论可以大大提高软件开发的成功率,同时,也可明显地提高软件开发的生产率。

目前划分软件生存周期阶段的方法有许多种,软件规模、种类、开发方式、开发环境以及开发时使用的方论都能影响软件生存周期阶段的划分。在划分软件生存周期的阶段时应该遵循的一条基本原则就是:使各阶段的任务彼此间尽可能相对独立,同一阶段各项任务的性质尽可能相同,从而降低每个阶段任务的复杂程度,简化不同阶段之间的联系,有利于软件开发工程的组织和管理。

软件工程就是对软件生命周期进行科学管理的一种系统方法。

§ 1.2

软件工程原理

自 60 年代末期出现“软件工程”概念以来,不少学者和软件工程师从系统工程和工程管理观点出发,根据各国的实践经验,提出了许多软件工程的开发原则。B·W·Boehm 基于众多学者的经验,总结出七条软件工程原理,并认为这些原理是确保软件产品质量和提高开发效率的最小集合。这七条基本原理是:

- (一)严格按照计划进行管理;
- (二)坚持进行阶段评审;
- (三)实行严格的产品控制;
- (四)采用现代化的程序设计技术;
- (五)结果要能清晰地审计;
- (六)开发小组的人员素质要好,数量不宜多;
- (七)要承认不断进行软件工程实践的必要性。

根据 P·W·Motzger 的统计,在不成功的软件项目中有 50% 是由于计划不周所造成的。可见,Boehm 把建立和完善软件计划,严格按计划进行软件工程管理作为第一条基本原理,是吸取了“前车之鉴”的。他认为在软件整个生存期中,要牢牢抓住项目的纲要、基线控制、项目组织、资源控制和产品质量配置控制,以及包括测试验证、运行维护在内的各种计划、各个不同层次的管理人员都必须按计划各尽所能地管理项目的开发工作。

第二条基本原理是坚持进行阶段评审。历史经验告诉我们,软件检验工作不能推迟到编码阶段结束之后再来进行,这是因为:(1)大部分错误是在编码之前造成的。根据 B·W·Boehm 等人的统计,编码之前的错误占 63%,编码错误仅占 37%;(2)错误的检测与修改愈迟,所付出的代价也愈高昂。因此通过阶段评审,以期尽可能在早期发现错误是必须遵循的重要原则,这正是软件工程方法中建立软件开发阶段论的重要依据。

前文已指出,在软件开发和维护过程中,修改是经常发生的现象,除了用户需求经常发生变化外,分析与设计工作也常常会考虑不周。为了使反复修改而形成的智力产品——软件,具有很好的质量特性,严格实行产品控制势在必行,开发阶段的一切中间产品,必须按软件产品的标准规范形成,一切修改必须按严格的规程进行,经过建议、评审、批准(或否决),然后实施。

这七条原理是相互独立、缺一不可的,概括软件的各方面,是相当完备的。

正如 B·W·Boehm 所指出:人们虽不能证明这七条原理是一个完备的集合,但可以说明人们已经提出的超过 100 条的软件工程“信条”,都可以由这七条原理的组合来蕴涵或引申而来。

§ 1.3

软件工程的研究内容

软件的基本特征以及当前软件开发过程中所存在的问题和困难,使得软件工程为了获得优秀的软件产品,满足软件市场日益增长的需要,开始在规范与标准、技术与方法、工具与环境、管理等方面进行研究。

1.3.1 软件的规范与标准

工程不同于科学,科学在于对“未知”的探索,往往是不计时间、成本,不受任何约束去获取某项有益成果;而工程的目的在于生产出有用的产品,这种产品既要求它具有很好的功效,又要求它具有良好的技术经济指标,而且必须在一组约束条件下进行开发。软件工程的生产品对象是“软件”产品,因此必须有一组规范和标准——约束条件,用来限制软件开发中的自由度,以保证软件生产具有高度的可靠性和可维护性,较高的劳动生产率和较低的成本等。不同的国家、不同的软件公司具有不同的软件规范和标准,一般说来,是从以下几个方面来进行规范化和标准化的。

(一)软件开发过程中作业标准化,就是把软件开发过程中需要做的工作分割成若干个作业,确定每一个作业所承担的工作内容,以及相应文档资料的内容;

(二)确定每个作业的代表形式;

(三)确定每个文档资料的格式;

(四)规定一组符号;

(五)依据多年的开发经验,归纳出大家可以接受的开发原则和制度。

1.3.2 软件工程中的方法学

软件工程中的方法学是研究在软件生命周期中,每个阶段进行作业开发时所采用的工作办法的一门科学。由于生命周期中各个阶段的目标与任务均不一样,因而所用的方法也是多种多样的。从生命周期各个阶段来看,有:

(1)系统分析方法。如结构化分析、HIPO 分析方法;面向对象分析方法等;

(2)设计方法。如结构化(按数据流的结构或数据结构等)设计法;面向对象设计方法等;

(3)编程方法。如结构化程序设计,自顶向下程序设计法;面向对象编程方法等;

(4)调试方法。如自顶向下、自底向上、层次结构等调试方法。

就方法自身的特征而言,大体上可分为以下几种:

(一)定义说明方法

其目的是使软件中的操作、操作对象以及其它名词术语都能有一个唯一的确切含意,这种含意要为所有开发者、使用者所理解。一般有两种方法进行定义说明:一种是形式化方法,它建立在严密的数学基础之上,满足逻辑上的完备性和一致性。如程序设计语言的方法,常常使用有限状态自动机的数学形式来定义;另一种是非形式化定义方法,它常常用自然语言来定义。为了尽可能地避免歧义性,必须遵循形式逻辑学关系下的定义规则,必须与用户、共同开发者取得一致的理解。

(二)结构化的构造方法

结构化的概念来源于一般工程学科,基于任何复杂系统总可以分解成一组有限的通用软件,或者说任何一个复杂系统都可以由所选择的一组通用部件来组成的原理。最典型的例子就是积木玩具。软件工程中结构化的概念首先是由 Dijkstra 引入的,他提出了任何一个程序都可以由三种基本结构成分(即顺序、分枝、重复或循环)及一组规则按照层次来构造的。二十多年来的实践证明结构程序设计是保证程序质量的有效途径。与此同时,人们还把结构化的概念引入到软件工程中,目前在软件生命周期的各个阶段几乎都引入了结构化概念,如结构分析法、结构设计法、结构层次调试法等等。软件工程中的结构化方法原理可以归结如下:“用抽象的方法,把事物的固有属性、内部或外部的联系抽象成一种具有层次关系的模型;而后按照规定的约束条件进行分解,使得每个层次只能按照规定的基本结构成分来组成。”不同的结构化方法,只是这个原理的具体化而已。实践证明,结构化方法是行之有效的。

(三)文档资料的描述方法

文档资料是计算机软件产品中不可缺少的一部分,否则系统就不能进行有效地运行、开发和维护。但是人们常常忽视了这一点。因为这是一项艰苦的、枯燥的劳动,开发者往往急于使系统能够迅速运行,精力集中于结构设计和程序设计上。殊不知,文档资料不完备或不合乎要求将会造成以下严重后果:

1. 随着时间的推移,人们的记忆慢慢地会被淡忘。如果文档资料不完备、不精确就会给可靠性带来严重的影响。
2. 不能把自己意图不出差错地传达给对方。
3. 如果使用手册有错误或不好懂,使用就要出错,即使程序编写得再完善精致也不能满足用户的要求。

另一方面,文档资料也不必过分详尽,否则将使开发工作量增加,拖延开发周期,使读者查询困难。应当以能完整地、确切地描述出所需的内容而又不致于繁琐这一总的准则来进行文档资料的编写。其具体原则和方法如下:

必须严格遵守规范和标准。

对于具体的软件开发任务,应制定软件文档资料的种类。在编制时应当考虑到软件任务的规模、性质、复杂性,生产软件的组织形式,开发的成败风险等因素。目前,国外常采用“等级划分法”、“因素求和法”来研究种类的划分。

文档资料的详尽程度取决于任务的规模、性质、复杂性、使用对象、使用环境和管理水平等因素,具体详尽程度往往是由项目负责人根据规范与标准及具体情况来确定的。

4. 凡属能用图形、表格描述的,就不采用文字描述,以保证简明扼要。
5. 对于文中首次应用的专门术语、首字母词组,均必须加以定义说明和列出首字母词组的原词。

1.3.3 软件工具和软件开发支撑环境

“软件工具”是指这样一类程序,它可以用来帮助开发、测试、分析、维护其它计算机程序及其文档资料,实现软件生产过程自动化,提高软件开发的劳动生产率和可靠性,降低软件生产成本。如程序设计语言编译程序、编辑程序、查错程序、诊断程序等一类程序均可称为软件工具。而对于大规模计算机程序,及其文档资料的生产所使用的软件工具,则是指需求分析工具、

设计工具、编码工具、确认工具、维护工具等一类较为复杂的软件工具,是一种自动化系统。在软件开发、维护和管理中所起的作用,常常为人力所不及,这是 70 年代以来,软件工程主要研究的内容之一。

软件工具的种类繁多,且形式多种多样,但都只适用于软件生命周期中的某一阶段或某一环节,而不能对整个生命周期有效。为了能够对软件生命周期全面提供支持,为此提出了建立软件开发支撑环境的课题。

软件开发支撑环境的概念是指在基本硬件与系统软件的基础上,提供一组能支持软件生命周期的软件工具,也就是说,能支持软件开发、维护、管理、质量控制等各个方面,而且能适应多种用户的要求。软件开发支撑环境一般可由数据库、一组工具和一组统一的命令或调用方式三部分组成。其建设规模取决于开发软件项目的一些基本特征(如开发周期、项目经费、系统寿命、开发人员、开发领域、代码行数、可靠性要求)。目前,许多国家和公司已经建立起各种用途的软件开发支撑环境,我国起步较晚,但发展较快。软件工具和软件开发支撑环境是软件 engineering 研究最活跃的领域。下面将单独设一节对其详加介绍。

1.3.4 软件管理

在软件工程中,管理技术是相当重要的。国内外经验都表明,许多软件项目的开发之所以失败,主要是由于管理不善。目前软件工程中维护问题大于开发问题,管理问题大于技术问题。虽然从长远来看,软件开发技术与工具的研究是改善软件生产的根本途径,但是,即使软件生产完全自动化了,管理问题仍不容忽视。

软件管理主要包括软件生命周期中的计划管理、成本管理、质量管理和组织管理等。其目标是合理地使用人力、物力资源,以最少消耗来开发软件,确保软件质量,最有效地组织力量,制订计划,保证各类资源与各种活动协调同步。

由于软件开发在很大程度上取决于每个人的能力,而人的能力差异很大,由 H·Sackman 等人的调查资料表明:“工作能力强的人与工作能力差的人就软件生产量来说,其比例为 10:1;在质量方面,写出程序的时空效率为 5:1。”因此,在软件管理中要想估计软件工时数、掌握软件产品的质量和进度,将比其它产品困难得多。目前主要致力于以下几个方面的研究:

1. 积累软件开发中的统计数据;
2. 探索软件质量的定量标准与质量管理方法;
3. 研究工时估计、进度估计的实用计算方法;
4. 建立软件管理工具;
5. 研究建立软件管理课题小组的组织方法,最大限度地发挥每个成员的才能。

§ 1.4

软件开发模型

软件开发遵循一定的开发方法和模型,在软件工程中,软件开发模型是一个十分重要的课题,软件开发模型是软件开发的行动指南,按照一定的模型来进行开发,可以大大提高软件生产效率。

目前国内外使用的软件开发模式,可归纳为“瀑布”方式、分解开发方式、“清洁房间”方式、原型方式、部件化再利用方式等等,现分别予以简要介绍。