

广东省电子信息系统推广应用办公室
中山大学 广东省计算中心

软件开发规范的实施文档

● 黄思曾 陈仲驹



中山大学出版社

广东省电子信息系统推广应用办公室
中山大学、广东省计算中心

软件开发规范的实施文档

黄思曾 陈仲驹

中山大学出版社

JS202/07

软件开发规范的实施文档

黄思曾 陈仲驹

*

中山大学出版社出版发行

广东省新华书店经销

中山大学印刷厂印刷

850×1168毫米 32开本 10.25印张 1插页 25万字

1991年9月第1版 1997年2月第3次印刷

印数:10001—13000册

ISBN7-306-00464-6

TP·8 定价:13.00元

出版说明

1988年,国家标准局颁布了两个关于软件产品开发规范的国家标准,GB8566—88《计算机软件开发规范》和GB8567—88《计算机软件产品开发文件编制指南》。

1989年,我们在全省范围内对上述两个国家标准组织了广泛的宣传和推广应用。

为了进一步提高计算机软件开发的工程化、规范化和系统化水平,1990年经省科委立项,由省电子信息系统推广应用办公室、省计算中心依据国家标准具体组织制订相应的实施方案和配套的软件开发工具。其中,《广东省计算机软件开发规范(国家标准)实施方案》由中山大学肖金声教授负责,在方案制订的基础上,为了帮助软件技术人员更好地理解、掌握、应用,由黄思曾、陈仲驹编著了《软件开发规范的实施文档》,详细地介绍了“实施方案”。《广东省计算机软件开发规范(国家标准)实施方案计算机辅助工具》、《软件设计图形支持工具FLOWTOOL》由暨南大学吴恭顺教授负责。实施方案制订后,我们广泛听取了省内有关专家和计算机软件开发人员的意见,并在一些软件开发项目中组织试用,反复修改后定稿。希望实施方案的推广应用,能对我省软件产业的发展起到积极的作用。

借此机会,我们对为参与这项工作而付出了大量辛勤劳动的全体同志表示衷心的感谢!对为实施方案的制订给予大力支持与提供了许多宝贵意见的专家和计算机软件开发人员表示衷心的感谢!

广东省电子信息系统推广应用办公室

一九九一年七月

前 言

随着计算机软件开发工作的发展,七十年代建立的软件工程学的观点和方法已经逐渐深入人心。人们认识到用工程化、规范化、系统化的方法来开发计算机软件,是提高软件产品质量,降低开发成本,提高工作效率的正确道路。在国内,一些行业陆续制订了软件开发规范并在本行业内推行,取得了良好的效果。

1988年,国家标准局颁布了两个关于软件产品开发规范的国家标准,即 GB8566—88《计算机软件开发规范》和 GB8567—88《计算机软件产品开发文件编制指南》。两个国家标准可以用“8个阶段 14种文档”一句话来概括,即详细规定了计算机软件开发和使用过程的8个工作阶段,每个阶段的任务、实施步骤、实施要求、完成标志,以及应交付的14种文档的编写内容提要。

1989年,我们曾在广东省内从事宣传和推广上述两个国家标准的工作,在一些软件开发项目中推行国家规范。通过实践,我们体会到文档的编写是执行规范的重要环节,事实上软件文档的完成也就是按照国家标准完成软件开发的过程。GB8567—88中已给定了14种文档自然语言形式的编写纲要。实践表明,由于开发人员的素质以及对规范的理解程度存在差异,编写的文档格式必然难以统一。我们觉得有必要在国家规范的容许范围内制订一个实施方案,帮助软件开发人员在工作中能够更具体、更明确、更清晰、更简单地实施国家的软件开发规范。

1990年完成的广东省软件开发规范(国家标准)实施方案的核心是37份表格(总共67种不同页面格式)的设计。这些表格组成控制和表达软件开发全过程的5种技术性文档,即可行

性分析报告,软件需求说明书,系统设计说明书,模块开发卷宗,测试计划和测试分析报告。应该强调指出,作为国家规范的一种实施方案,我们的设计是符合 GB8567—88“5.3 灵活性”一节中所规定的原则的。

工作完成之后,不少同行向我们索取表格样本,为了推广使用,我们以上面提及的 37 种文档表格为纲编写成书,在实际的开发工作中,完成这些表格的填写,也就是按国家规范的精神完成软件开发的过程。文档首先是工程化软件开发的工具,同时也是开发过程的记录信息。

书中,我们从“HB 厂管理信息系统”的文档中抽出一些页面,作为每份表格的编写例子。由于篇幅所限,这些例子只能作简化,未免不连贯。希望它们对读者编写自己的文档仍能起示例作用。

《广东省软件开发规范(国家标准)实施方案》是经省科委立项,由省电子信息系统推广应用办公室、省计算中心具体组织实施的。方案的技术研制工作由中山大学肖金声和黄思曾完成负责。本书由黄思曾执笔,肖金声教授对全书作了逐字逐句式的审定,省计算中心陈仲驹提供了全书的示例。

希望本书能够给读者带来帮助。恳请读者反馈宝贵意见,尤其是经过软件开发实践之后产生的意见。

编著者

一九九一年七月于康乐园

目 录

第一章	软件工程和软件开发规范	1
§ 1	软件开发的工程方法	1
§ 2	软件生存周期	8
§ 3	软件文档	12
第二章	软件开发规范的国家标准	16
§ 1	计算机软件开发规范(GB8566—88)	16
§ 2	计算机软件产品开发文件编制指南 (GB8567—88)	20
§ 3	有关的其它国家标准	25
第三章	开发规范的一个实施方案	27
§ 1	实施方案的研制目标和准则	27
§ 2	实施方案的设计思想	28
§ 3	文档格式	31
第四章	可行性研究报告	39
§ 1	简要说明(F01)	42
§ 2	现有系统说明表(F02)	54
§ 3	候选系统说明表(F03)	58
§ 4	数据流图(F04)	68
§ 5	数据流/存贮描述表(F05)	73
§ 6	系统流程图(F06)	76
§ 7	可行性研究结论(F07)	79

第五章	软件需求说明书	85
§ 1	系统需求简要说明表(A01)	88
§ 2	数据流图(A02)	97
§ 3	数据流/存贮/组合元素描述表(A03)	108
§ 4	数据元素描述表(A04)	110
§ 5	数据概念结构图(A05)	122
§ 6	基本处理说明(A06)	127
§ 7	系统 I/O 描述(A07)	133
§ 8	数据元素代码表(A08)	143
§ 9	系统流程图(A09)	148
§ 10	软件需求说明书确认书(封底)	152
第六章	系统设计说明书	154
§ 1	系统概况(D01)	158
§ 2	系统流程图(D02)	163
§ 3	模块结构图(D03)	165
§ 4	模块描述表(D04)	172
§ 5	数据结构图(D05)	177
§ 6	数据结构描述表(D06)	182
§ 7	接口设计(D07)	193
§ 8	功能需求—模块关系表(D08)	204
§ 9	数据结构—模块关系表(D09)	208
§ 10	安全性设计(D10)	212
§ 11	系统设计复审报告	221
第七章	模块开发卷宗	223
§ 1	模块内部结构(M01)	226
§ 2	子模块设计说明(M02)	232

§ 3	子模块实现文档(M03)	246
§ 4	模块测试情况(M04)	252
§ 5	模块复审表	255
第八章	测试计划和测试分析报告	257
I	测试计划	260
§ 1	软件概况(T01)	263
§ 2	测试进度表(T02)	269
§ 3	测试设计(T03)	272
§ 4	测试计划评估(T04)	281
II	测试分析报告	284
§ 5	完成测试清单(T05)	286
§ 6	测试情况表(T06)	289
§ 7	结果分析(T07)	299
附录 I	五个有关软件开发管理 和面向用户的文档编写指南	303
附录 II	N—S 图和 PAD 图	318

第一章 软件工程和软件开发规范

§ 1 软件开发的工程方法

一、软件开发方法的演变

众所周知,计算机系统由硬件和软件两部分组成。人们习惯于按照计算机问世 40 多年来,硬件出现的几次重大演变来划分计算机系统的发展阶段。伴随着从第一代电子管计算机到目前第四代大规模集成电路计算机的发展,对软件的研究也有了长足的进步。但是总的看来,软件开发仍然跟不上计算机应用的要求,以至成为限制计算机应用继续发展的主要因素。

60 年代初期以前,是计算机系统发展的早期时代。人们心目中的软件开发就是编写程序。不存在系统化的开发方法。主要的生产方式是程序员个人的手工技艺方式,特别强调编程技巧,以各种不规范的、往往是灵机一动的方式来节省几条指令,或者以二进制位为单位来使用存贮空间。这种个体化的开发环境,使软件的设计分析过程仅仅隐含在程序员的头脑中。

60 年代中期以后,象操作系统这样复杂的系统软件陆续出现,应用软件的规模也日益增大,因而迫使软件开发不得不采取作坊式的合作生产方式。软件开始成为产品,出现了编写程序的结构化方法,强调逐步求精,把一个大型的程序分解为相互之间

存在某种联系模式的程序模块集合,以此来协调和制约程序员们的集体开发工作。

进入 70 年代,软件规模和复杂程度也随之猛增,大型系统的工作量可达上千人一年,指令数目上千万条。一些软件的开发遇到前所未有的困难,有人用“在泥潭中作垂死挣扎的巨兽”来形容这些开发人员。危机触发了软件工程学的诞生,人们意识到应该有新的理论、新的方法来指导软件产品的开发,从此它得到研究和应用。

历史的简单回顾告诉我们:软件工程学不是学者们坐在象牙塔里构筑的纯理论。它是软件开发发展到一定阶段之后,必然呼唤的工程方法。今天,从事软件开发的人员如果不自觉运用软件工程学的成果来指导自己的开发实践的话,将或早或迟地重新踏上那一条导向“危机”的歧路,重蹈前人的覆辙。

从 70 年代以来逐步形成的工程化的软件开发方法,其特征可以简单概括如下:

1. 把软件的生产过程划分为若干个阶段,每个阶段内都采用经过验证的行之有效的开发方法。无论采用哪种方法,软件生产的全过程都要形成或遵循某种标准规程。标准规程明确规定每个阶段的任务和工作步骤,指导和约束开发人员,使他们遵照规定的方式来理解和处理生产进程。

2. 相当于现代工业大生产方式中的产品质量检查制度,对软件开发的每个阶段都进行技术和管理方面的复审。复审的意义在于防止错误流到下一阶段,以保证所开发软件的质量,以及按计划完成开发工作。

3. 对软件开发的每一个阶段都要求产生规定的文档。文档的产生是阶段工作完成的标志,它有利于软件的交流、管理和维

护。

4. 提供从原始计划到最终产品的一条可追溯的途径。因而应推行配置管理,使得能从目标程序追溯到需求定义,又能从每个功能检查到涉及的程序模块。

二、软件开发工程化方法的形成

早期,人们首先致力于把程序设计工作从一门技巧发展为一门科学,研究集中在三个相关的课题,即:数据结构、算法设计和分析、以及程序的结构方法。

数据是描述客观事物的、能输入到计算机中加以处理的所有信息的统称。实质上它是客观事物表示的一种抽象。数据结构这门学科就是研究如何把客观世界中待处理的对象逐层抽象为计算机可以接受的某种表示形式。

算法是指解决某类特定问题的有穷步处理动作的序列。它最终要体现为能在计算机环境中执行的某种程序语言的语句序列。研究算法的最终目的是为了高效地求解问题。

数据结构和算法是互相连系的。其实,所谓程序就是在数据的特定结构形式上用某种程序设计语言表达的算法。所以,不了解施加于数据之上的算法就无法决定如何构造数据;反之,算法的设计和效率又往往依赖于作为其基础的数据结构。

除了研究表达对象和处理对象的规律外,人们发现要设计出一个高质量的程序尚需要考虑程序结构问题。

60年代中期出现了结构化程序设计技术。曾有人证明,只要用三种基本的流程控制结构就可以编写出任何单入口单出口的复杂程序。这三种基本控制结构是:顺序、选择和重复。由于三种基本结构可以交错嵌套,这就启发人们采用自顶向下逐步

求精的程序设计方法,写出由三种结构元素构造而成的结构化程序。

把结构化技术应用在大型程序的结构设计上,导致了模块化原理的产生,即把程序分解为一组尽可能功能独立的模块,程序模块除了和基本控制结构一样具有单入口单出口的构造特征之外,尚具有信息隐蔽和局部化特征,互相关连的模块在接口上显式定义要传递的数据。模块化原理使软件结构清晰,易理解,易调试,易修改,从而提高了软件的可靠性。模块化也有助于程序从个体化的开发方式向集体开发方式转化,有助于开发工程的组织和管理。

1968年,软件工程的观念被提出来了。一个基本认识是,编写程序只是开发软件的一个环节,而且是个相对次要的环节。程序并不是软件的全部内涵。一个和用户需要及开发目标有差距的“完美”程序,无论如何不是成功的软件。一个有效而合用的软件,其开发过程应当包括需求分析、软件设计、编码(写程序)、测试等几个阶段。

工程化的软件开发的核心理念是,把软件看成是和其它工业产品一样的产品,从而研究以及应用工程化的原理、方法和技术来开发和维护软件。它的最终目标是用尽可能少的投资来获得高质量的软件产品。

从软件的开发和使用实践,人们选择了一组属性来刻画软件的质量。如功能的正确性、运行的可靠性、易使用性、时间和资源的效率、可维护性、可移植性等等。随着硬件技术的飞速进步和软件规模的日益庞大,工程化的开发方法更致力于保证软件的正确性、可靠性、可维护性这样一些重要的质量属性,而把软件的时空效率放在相对次要的考虑位置上。

工程化的软件开发方法首先着眼于开发过程的规范化。这就是制订指导和约束开发人员生产软件产品的某种标准规程,用以协调开发全过程,使软件的开发方式从个体技巧式或者作坊合作式改变成为工程化的大生产方式。这是软件优质高产的必要条件。

一种软件开发规范,应该规定开发人员:“什么时候应该做什么以及如何去做”。可以用三 W 来表示软件开发规范的实质,即 WHEN—WHAT—HOW。开发规范越明确、越具体,对实现软件产品优质高产目标的指导作用就越大。但是软件毕竟是一种逻辑性的产品,软件开发规范保留不同程度的灵活性和试探性是难免的。它很难做到像机械操作规程那么明确具体,也很难像机械工艺流程那样要求生产人员一成不变地遵从。

但是也应该看到,过份原则和概略的规范必然会削弱规范的指导作用,尤其是在开发人员的技术素质不太强的环境中。因此有必要在软件开发单位或部门里,根据面临的开发环境或者某项具体的软件开发任务,制订要遵循的开发规范的实施方案。实施方案应根据规范的精神和原则,订出具体而明确、选择自由度相对较小的规程,迫使开发人员的开发行为比较一致而标准,从而保证了规范的指导作用。

一般来说,一个软件开发规范应该规定:

1. 明确的工作阶段

任何一种工业产品的现代生产方式,必然包含或多或少的生产工序。工序划分越精细,各工序加工动作就越明确,因而就越简单化,这是现代化生产方式的显著特征。尽管这种成批生产模式和软件产品的单件生产状况不完全相同,但是足以启发我们应该把软件的生产过程分为若干个工作阶段,每个阶段都应

该有明确的任务和要求、前提和结果、执行步骤、评价标准和审核活动。

2. 具体的文档格式

任何一个现代企业的生产过程中,伴随着从原材料到制成产品的物质流,必然存在一个信息流。信息流中的各种数据对生产过程起着控制和记录的作用。信息流是如此庞杂,作用又如此关键,不可能以生产者和管理者的头脑为载体,而是表现为各种可以阅读的文档记录。它们是加工图纸、工艺技术规范、管理表格、单据、报表等等。文档是各个工序正确操作的依据,是协调、管理、记录生产全过程的手段。所以,为了以工程化的方法开发软件,开发规范应当制订涉及的文档及其格式,包括每个开发阶段涉及的文档种类、内容和格式、表达手段如图形和符号等等。

大体上讲,工程化的软件开发方法把开发过程划分为问题定义、需求分析、设计、编码、测试等工作阶段。每个阶段都研究出一些行之有效的开发方法和表达工具。在形形色色的方法和技术中,结构化方法仍然是特别简明而有效的一种。最早出现的程序设计(编码)结构化原理,不但影响了现代程序设计语言的机制,而且被推广到软件开发的结构设计和需求分析阶段中去。

无论从需求分析—结构设计—程序编码这个软件开发流程,还是从结构化的分析方法、结构化的软件设计方法、结构化的程序编写方法当中,都体现了抽象与分解的基本原则。这是人类认识复杂现象、解决复杂问题时有力的思维方法。

人们认识到,在解决复杂问题时不要一下子就触及问题底层细节,而是在分析和定义了问题的需求之后,先设计解决问题的一个抽象算法,在抽象的数据表达形式上实施一系列抽象操作,它们反映问题求解的实质步骤而舍略细节。然后在进一步的

工作阶段中,再考虑这些抽象数据和操作的具体实现。抽象程度渐次降低,实现方案越来越具体,直到在程序设计的环境中得到问题的解—要开发的程序系统为止。

软件工程化开发方法从分析到设计和实现的过程就是软件开发从抽象到具体的过程。在抽象级上只明确表达“做什么”,在实现级上再回答“怎么做”。而且结构化的方法在表达“做什么”和“怎么做”时,也遵循从抽象到具体的原则,简明的全局描述,逐层分解,不断添加描述细节,直到达到某一工作阶段要表达的具体目标为止。

尽管结构化方法并不是软件开发各工作阶段唯一可用的方法,但是它和人们解决复杂问题的思维方法一致,因此是一种实用而值得推荐的方法。

三、软件工程学

经过近 20 年的努力,软件工程学这个计算机软件的研究分支已经形成。软件工程学的研究大体上是从软件开发技术和软件工程管理两个方面展开的。

对开发技术的研究首先涉及工程化的开发方法和软件工具。上面已经叙述了以开发规范和各种具体方法(如结构化方法)为核心的工程化开发方法的特征。

随着开发方法的发展,软件工具也在不断产生。软件工具是指在开发的各个阶段中支持和辅助开发工作的各类程序系统。传统的早期工具侧重对开发实现阶段的支持,如操作系统、编译程序、解释程序、汇编程序、连结程序、调试查错程序等等。以后,又陆续研制出支持开发全过程各个工作阶段的包括需求分析、设计、编码、测试所使用的开发工具、管理工具、以及软件的维护

工具等。

开发方法和软件工具的发展导致它们的互相结合。按照一定的软件开发方法组织起来的一组软件工具,对开发的全过程提供支持,这样的计算机辅助开发软件的程序系统连同硬设备就被称为软件工程环境。目前这一方面发展仍然是初步的,软件开发方法、工具和工程环境还有待于从理论上、方法上、技术上进一步完善。

软件工程学除了从技术方面展开研究之外,还从管理的角度研究软件的开发。和其它工业生产一样,技术和管理是软件生产不可缺少的两个方面。任何开发技术只有在管理的决策和支持下才能达到目标。

大体上说,软件工程管理包括费用管理、质量管理、配置管理、项目管理等内容。研究内容包括软件成本估算模型和方法、全面质量控制计划、软件的完整性和一致性控制、开发进度的安排控制,开发人员的组织等等。

本书目的并非全面介绍软件工程学所涉及的领域,而是仅仅以软件开发规范为中心介绍工程化开发技术的思想,并提出一个开发规范的实施方案。

§ 2 软件生存周期

软件生存周期是软件工程学最基本的概念之一。各种各样的开发方法都不同程度地承认这个基本概念。

一、含义

软件生存周期是指:从构思软件产品开始到产品不能再使