

武汉新利文化  
6.0X100Y封箱胶

# 科技软件工程 概论

庄建南 唐学勇 编著

南京大学出版社

# 科技软件工程概论

钱为民 主审

庄建南 唐学勇 编著

南京大学出版社

(苏)新登字第 011 号

**科技软件工程概论**

keji Ruanjian Gongcheng Gailun

钱为民 主审

庄建南 唐学勇 编著

\*

南京大学出版社出版

(南京大学校内 邮政编码:210008)

南京豪利电脑照排中心照排

江苏省新华书店发行 淮安市印刷厂印刷

\*

开本:787×1092 1/16 印张:16.375 字数:415千

1993年4月第1版 1993年4月第1次印刷

印数:1—6000

ISBN 7-305-01746-9/TP·52

定价:8.10元

## 前 言

“软件工程”是自 60 年代末期发展起来的一门新兴学科,其发展速度之快,影响之大十分惊人,已成为推动当今软件产业发展的重要支柱。

电子计算机的迅速发展和广泛应用,使软件的社会需求日益迫切,于是大批应用软件应运而生,软件产业相继形成。科技软件是最早发展起来的一类应用软件,其应用范围覆盖了科学研究和工程技术的各个领域,是促进科学和工程技术的新发展、新设计的重要工具。目前,随着巨型计算机、超级微型机和人工智能的发展,科技软件正朝着综合化、集成化、智能化方向迈进,从而使“软件工程”成为大中型科技软件研制开发的必备的理论和方法。

本书是为适应软件产业的迅速发展,以及信息化社会对软件产品和软件开发人员的迫切需求而写的。前几年曾作为南京大学计算数学和应用软件专业学生的选修课,深受学生欢迎,使他们拓宽了知识面,增加了对信息化社会需求的适应性;另外,还由山东大学等兄弟院校试用过,都取得了良好的效果。经过多年教学实践,并吸取江苏省计算技术研究所和江苏软件技术开发中心等单位在软件开发中的经验,在原讲义的基础上,进行了更新和扩充,特别是收入了国家颁布的各种软件开发标准和技术规范,使内容更加充实、新颖,更具有针对性和适用性。

全书共分九章。首先介绍软件产业的形成和发展,接着从科技软件的功能、结构和特点入手,引进了“软件工程”概念和基本理论,然后介绍软件生存期各阶段的主要任务及实施方法与技术,尤其对开发阶段中比较适用的 SA、SD、SP 等方法作了较为详细的介绍,还特别强调了编码风格等问题,最后对软件维护和管理中的有关技术和方法以及文档编制等问题也作了相应的介绍。为便于理解,书中插入了大量的图表和例题,各章后面均附有习题,供练习与思考。

本书可作为大专院校计算数学和应用软件专业、系统科学与信息工程专业、计算机科学系的应用软件专业及其他相关专业的教材,亦可供大专院校的教师、研究生及工矿企业、科研单位从事软件工作的管理人员和技术人员参考。对于广大从事科技应用软件开发的技术与管理人人员,通过自学本书,可以掌握软件开发技术,了解技术规范,牢固树立“工程化”的软件开发思想,革新传统的生产方式,从而会收到有计划、有步骤、高质量、低成本地开发软件产品的效果。

本书由江苏省计算技术研究所所长钱为民教授主审。在编写过程中,也得到了南京大学数学系陈仲副教授的大力支持,地质系徐福林老师在百忙中抽空协助绘制了全书的插图,数学系吴兆金老师为本书的出版也给予积极支持。另外南京大学数学系研究生孔敏、刘卫国、孙德峰、俞斌、张宇飞、骆建文等承担了书稿的抄写工作,在此向他们表示衷心感谢。

限于经验和水平,书中难免有疏漏和不妥之处,希望专家、同行及广大读者不吝指教。

编 者

1992 年 2 月

# 目 录

<b>第一章 软件产业的形成与发展</b> .....	(1)
1.1 软件的概念 .....	(1)
1.1.1 程序与文档 .....	(1)
1.1.2 软件的定义 .....	(1)
1.1.3 软件的地位 .....	(2)
1.2 软件产品的商品化 .....	(3)
1.2.1 软件产品的概念 .....	(3)
1.2.2 软件商品化过程模型 .....	(3)
1.2.3 软件产品的历史 .....	(5)
1.3 软件产业的形成 .....	(7)
1.3.1 软件产业的概念 .....	(7)
1.3.2 软件产业发展过程 .....	(7)
1.3.3 软件产业的业务范围 .....	(8)
1.3.4 软件产业的发展现状 .....	(8)
1.4 振兴软件产业的措施.....	(10)
习题一 .....	(12)
<b>第二章 科技软件及软件工程概述</b> .....	(13)
2.1 科技软件.....	(13)
2.1.1 科技软件的概念.....	(13)
2.1.2 数学软件.....	(14)
2.1.3 辅助性软件.....	(15)
2.1.4 结构工程软件.....	(16)
2.1.5 专业性技术软件.....	(16)
2.1.6 人工智能软件.....	(17)
2.2 科技软件的功能与结构.....	(18)
2.2.1 功能与结构的发展阶段.....	(18)
2.2.2 发展趋势.....	(20)
2.3 科技软件的评价.....	(22)
2.3.1 可移植性.....	(22)
2.3.2 可用性.....	(25)
2.3.3 可维护性.....	(26)
2.3.4 文档的完善性.....	(26)
2.4 大中型科技软件的特点.....	(27)

2.5	软件工程概述	(27)
2.5.1	软件工程登台	(28)
2.5.2	软件工程基础理论	(30)
2.5.3	软件工程原理	(31)
2.5.4	软件工程中的几个基本概念	(32)
2.5.5	软件工程标准化	(34)
2.6	软件工具与软件工程环境	(35)
2.6.1	软件工具	(35)
2.6.2	软件工程环境	(36)
2.7	软件生存周期和软件开发模型	(39)
2.7.1	瀑布式开发模型	(40)
2.7.2	原型化开发模型	(42)
2.7.3	进程组开发模型	(46)
	习题二	(48)
<b>第三章</b>	<b>可行性研究与软件计划</b>	<b>(49)</b>
3.1	问题定义	(49)
3.2	可行性研究	(49)
3.2.1	可行性研究的目的与任务	(49)
3.2.2	可行性研究的步骤	(50)
3.2.3	系统流程图	(51)
3.3	软件作用范围	(55)
3.4	资源分析	(56)
3.4.1	人员资源	(56)
3.4.2	硬件资源	(57)
3.4.3	软件资源	(57)
3.5	软件的成本估算	(58)
3.5.1	估算中的计量单位	(58)
3.5.2	软件成本估算方法	(59)
3.5.3	软件成本估算模型	(60)
3.5.4	语句行成本估算技术	(66)
3.5.5	任务—工作量成本估算技术	(67)
3.5.6	自动化成本估算技术	(68)
3.5.7	成本估算实例	(69)
3.6	软件开发进度安排	(71)
3.7	软件计划说明书	(73)
3.8	软件计划复审	(74)
	习题三	(74)
<b>第四章</b>	<b>软件需求分析</b>	<b>(75)</b>
4.1	需求分析概述	(75)

4.2	需求分析的重要性	(75)
4.3	需求分析的困难	(77)
4.4	需求分析的任务	(78)
4.5	需求定义方法	(79)
4.6	需求分析方法(SA法)	(80)
4.6.1	数据流图	(80)
4.6.2	数据流图的画法	(83)
4.6.3	数据字典	(90)
4.6.4	SA方法的分析步骤	(92)
4.7	软件需求规格说明	(93)
4.7.1	需求规格说明的特征	(93)
4.7.2	需求规格说明的组成	(94)
4.8	复审	(95)
4.9	需求分析工具简介	(95)
	习题四	(97)

<b>第五章</b>	<b>软件设计</b>	<b>(98)</b>
5.1	软件设计任务	(98)
5.2	软件设计步骤	(101)
5.3	软件结构与软件过程	(101)
5.3.1	软件结构	(102)
5.3.2	软件过程	(104)
5.4	软件设计的概念和原理	(105)
5.4.1	模块化	(105)
5.4.2	抽象化	(106)
5.4.3	信息隐蔽与局部化	(107)
5.4.4	模块独立性	(107)
5.5	软件设计指导原则	(112)
5.6	图形工具	(115)
5.6.1	层次图	(115)
5.6.2	HIPO图	(115)
5.6.3	结构图	(118)
5.7	SD方法	(120)
5.7.1	软件结构的标准形式	(120)
5.7.2	SD设计步骤	(123)
5.7.3	实例	(128)
5.8	Jackson方法	(134)
5.8.1	Jackson图	(134)
5.8.2	Jackson方法设计步骤	(136)
5.8.3	一个例子	(137)
5.8.4	结构冲突和回溯	(141)

5.8.5 应用范围 .....	(145)
5.9 详细设计 .....	(145)
5.9.1 结构程序设计方法 .....	(146)
5.9.2 详细设计工具 .....	(149)
5.9.3 PAM .....	(159)
5.10 设计文档 .....	(163)
5.11 设计复审 .....	(164)
习题五 .....	(165)
<b>第六章 编码</b> .....	<b>(167)</b>
6.1 编程语言分类 .....	(167)
6.2 编程语言的特性 .....	(170)
6.3 工程观点 .....	(172)
6.4 编程语言的选择准则 .....	(172)
6.5 编码风格 .....	(173)
6.5.1 代码文件 .....	(173)
6.5.2 数据说明 .....	(176)
6.5.3 语句构造 .....	(177)
6.5.4 输入/输出(I/O) .....	(180)
6.5.5 功效 .....	(182)
习题六 .....	(183)
<b>第七章 软件测试</b> .....	<b>(185)</b>
7.1 测试的复杂性 .....	(185)
7.2 测试阶段的任务 .....	(186)
7.3 测试的基本原则 .....	(188)
7.4 测试方法 .....	(189)
7.4.1 黑箱法 .....	(189)
7.4.2 白箱法 .....	(196)
7.5 测试过程 .....	(199)
7.5.1 单元测试 .....	(200)
7.5.2 集成测试 .....	(202)
7.5.3 验收测试 .....	(205)
7.6 测试规格说明 .....	(206)
7.7 自动测试工具 .....	(207)
习题七 .....	(209)
<b>第八章 软件维护</b> .....	<b>(210)</b>
8.1 软件维护的定义 .....	(210)
8.2 软件维护的特点 .....	(212)



8.2.1	结构化维护与非结构化维护 .....	(212)
8.2.2	维护的代价 .....	(212)
8.2.3	维护中经常遇到的问题 .....	(213)
8.3	可维护性 .....	(214)
8.3.1	影响可维护性的环境因素 .....	(214)
8.3.2	影响维护活动因素的分析 .....	(214)
8.3.3	可维护性的定量评价 .....	(215)
8.3.4	可维护性复审 .....	(216)
8.4	维护技术 .....	(216)
8.5	维护过程 .....	(216)
8.5.1	维护组织 .....	(217)
8.5.2	维护报告 .....	(217)
8.5.3	维护的事件流 .....	(218)
8.5.4	保存维护记录 .....	(219)
8.5.5	评价维护活动 .....	(220)
8.6	维护的副作用 .....	(220)
8.6.1	编码的副作用 .....	(221)
8.6.2	数据的副作用 .....	(221)
8.6.3	文档的副作用 .....	(221)
8.7	维护支援工具和技术 .....	(222)
8.8	小结 .....	(224)
	习题八 .....	(226)

<b>第九章</b>	<b>软件工程管理 .....</b>	<b>(228)</b>
9.1	软件工程管理导论 .....	(228)
9.2	软件的行业管理与企业管理概述 .....	(229)
9.3	软件项目管理 .....	(229)
9.3.1	人员管理 .....	(229)
9.3.2	组织管理 .....	(230)
9.3.3	成本管理 .....	(233)
9.3.4	项目计划的管理 .....	(235)
9.3.5	质量管理 .....	(237)
9.4	文档管理 .....	(238)
9.4.1	文档的重要性 .....	(238)
9.4.2	文档的分类 .....	(239)
9.4.3	文档编制应考虑的因素 .....	(242)
9.4.4	文档管理规范 .....	(243)
9.4.5	文档编制实施规定的实例 .....	(246)
9.5	软件管理工具 .....	(248)
	习题九 .....	(249)
	主要参考文献 .....	(249)

# 第一章 软件产业的形成与发展

软件产业是近 20 年来逐步形成和发展起来的一种知识密集型和技术密集型的产业,是信息产业的核心。它的产生是新技术革命到来和世界跨入信息时代的重要标志之一,是促进各行各业实现电脑化和社会信息化的重要支柱,也是软件技术迅速发展的必然结果。

众所周知,在迎接信息时代,向高度信息化社会迈进的过程中,作为现代信息处理的主要工具电子计算机是信息社会的灵魂。当今计算机不仅从超大型到微小型,种类繁多,而且应用广泛,从产业界到机关、银行、学校、医院、商店等业务部门,甚至进入家庭,渗透到社会和生活的各个方面,给人类的整个生活产生了极大的影响。计算机的应用越广泛,社会对软件产品的需求就越迫切。从 70 年代开始,国外一些发达国家从大量的经验和教训中认识到,必须在计算机硬件发展的基础上,相对独立地发展计算机软件,并使软件开发“工程化”,软件产品“商品化”。于是相继形成了软件产业。目前软件产业正处于蓬勃发展的新时期。

## 1.1 软件的概念

### 1.1.1 程序与文档(program and document)

计算机程序是一串指令序列的集合,它能被计算机理解和执行,是对计算任务的动作和对象的描述。严格地说,计算机程序是指为了得到某种结果而可以由计算机等具有信息处理能力的装置执行的代码化的指令序列,或者可被自动转换成代码化指令序列的符号化指令序列或者符号化语句序列。

文档是指用自然语言或者形式化语言所编写的文字资料和图表,用来描述程序的内容、组成、设计、功能规格、开发情况、测试结果及使用方法,如程序设计说明书、流程图、用户手册等。

### 1.1.2 软件的定义(software definition)

软件是计算机系统中与硬件子系统相互依存的另一子系统,是一个包含程序、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。它主要由两大部分组成:其一是可执行部分,即以编码信息存放在存贮介质上的程序和过程;其二则是与程序和过程有关的文档资料,这些资料虽然不可执行,但关系到软件能否有效地运行、最佳地维护等一系列问题。

随着计算机科学的发展,人们对软件的认识不断深化。在 70 年代以前,普遍认为软件=程序;进入 70 年代,认为软件=程序(主)+文档(从);80 年代开始,则认为软件=文档(主)+程序(从)。今后,文档将会变得越来越重要,并且可以自动化地生成。

### 1.1.3 软件的地位

软件是计算机的灵魂。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,起到扩充设备等作用。没有软件,计算机就无法发挥其作用,犹如失去了灵魂。

软件是打开硬件销路,促进计算机广泛应用的决定因素。自40年代第一台电子计算机问世以来,经过迅速发展,目前计算机已经历过四代演变,正向第五代迈进。在60年代初期,软件开发费用在整个计算机系统中只占很少的百分比,硬件却非常昂贵,因而人们自然把管理工作和规程应用于控制硬件的成本。但随着微电子技术的迅速发展,硬件成本骤降。相比之下,软件成本却像通货膨胀那样持续上涨。进入80年代,软件在以计算机为基础的系统开发中已经变为最主要的成本因素。因而使越来越多的计算机厂商的注意力逐步转向软件,投入大量的人力、财力从事软件的研制和开发,想以此来打开他们的硬件的销路。与此同时,他们还研制所谓“罐头式”的软件,以及各种“软件包”作为单独的产品销售,为大量没有编程能力的用户提供了方便,极大地促进了计算机的应用。

软件是高度信息化社会发展的最重要的资源。当前,社会上出现了一种高度信息化(第二次信息革命)的新潮流。如果把60年代中期开始,将计算机带动的社会称为信息化社会,那末目前的信息化社会,由于计算机与通信的结合以及新介质的出现,已达到了与传统不同的新水平,正向高度信息化社会发展。在计算机与通信结合的背景里,蕴藏着新技术的精华,其中包括硬件和软件的各种努力及技术,包括被喻为人类头脑细胞的LSI及超LSI,喻为人的神经的光纤电缆以及类似于人的感觉的传感器等各种先进电子技术,其中还包括了INS(信息网络系统)、双向CATV(Community Antenna Television)、卫星通信、卫星广播、文字图形信息网络系统、电子邮件等工具的应用。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遍布于世界各地,这一功劳应主要归功于软件。计算机网络从50年代单功能批处理为中心,60~70年代主机集中式联机处理,发展到80年代以来的分布式处理,使计算机处理的思想为之面目一新。而推动这种发展的,是软件、通信技术和计算机技术三位一体相互作用的结果。另一方面,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革新和结合,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。如果把能源看作是工业化社会发展的最主要的资源,那末软件将是高度信息化社会发展的最重要的资源。

软件是国际竞争中能否取胜的关键。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,美国总统布什向国会提出“美国重要技术报告”,指出为了保持美国综合国力优势,有六大门类22种技术是美国必须依靠的技术,美国政府必须重点支持。其中与计算机科学有关的就有3项,即软件、高性能计算机以及存储器等外围设备。关于软件技术,报告中指出有两个重点:(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,然后再予以扩充。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要以软件的可靠性和高性能为前提,且它的价格也越来越受软件的开发、生产费用所左右。正如有人说,自80年代起已进入“软件时代”,究其原因,有两层意思,一是说明计算机在发展高度信息化社会中所起的核心作用;二是认为,在某种意义上软件

决定着未来的方向。

## 1.2 软件产品的商品化

### 1.2.1 软件产品的概念

当软件在市场上作为商品流通时,才有软件产品的概念,这是伴随着信息处理行业在社会中占有一定的比重时而产生的。

社会的需求,是软件产品化的客观因素。最初,大量的软件都是面向特定的用户,为特定的任务而开发的。然而,除极个别之外,应用市场上总存在着需求与之相似的用户,只要这种用户存在,已经开发的软件就可以作为商品出售给这些用户。这种软件,在开发之初可能不是为了商业的目的,但由于客观的需要,最终成了商品,实现了由非商品到商品的转化。少量的软件作为商品进行交换流通,将大大刺激货主市场与需求市场,于是促使更多的商品化软件产品的出现。

制造方法的变革是软件商品化的重要标志。软件通常是由程序课题组研制,而软件产品是在主程序员负责制下的主程序员小组完成研制工作(例如,有一个主程序员负责全面开发工作,五个程序员,还有二个负责文件资料工作的资料员,以这八个人为一单元的层次组织结构)。这种分工和以前的大锅饭相比,优点之一能提高生产率。缺点是单纯作业往返枯燥无味,于是开发使用了各种对话型自动工具,这也是把软件作为产品生产的一个特征,同现代工业中的生产自动化进程如出一辙。例如,软件产品通常有这样的功能,要修改时有立即可以更新的联机文件,用户使用产品无预备知识,通过终端对话,有可用的对话型用户手册,还有编入产品本身中的内部维护功能,就像有一个专门的维护小组自动进行产品的维护一样。因此,软件产品就在于能从研制小组脱手作为独立的商品存在。

### 1.2.2 软件商品化过程模型

一般产品的商品化过程是比较简单的。由开发到投放市场,根据市场反映来确定扩大再生产,根据使用反馈来确定产品市场寿命与改进。这是一个波浪式线性周期过程。

软件产品不同于一般产品的特性,它的市场风险更大,更难以预测。一般产品一旦进入市场,总的来说是个能力衰亡过程,产品的开发集中在商品开始流通之前,也就是说,产品开始生产后,其主要特征是保持很大的稳定性,今后的改进、发展是有限的,且方向是可预测的。而软件产品,更多的再开发工作集中在初次与市场接触之后,开发方向是随机可变的,难以完全预测。软件产品在市场上总的来说是一个生长过程。

软件商品的市场实现是一个复杂的循环过程,各个国家实现商品化的过程和途径也不尽相同,因而对商品化的过程模型也有许多讨论。这里仅介绍如图 1-1 所示的软件商品化模型,该模型是对某些成功商品化软件分析研究后提出的。

#### 1. 概念期

软件在市场生命期之初,一般都是为特定用户开发的,开发工作可以由特定用户自己完成,也可以委托其他开发者完成。特定用户开发该软件的目的可能是供自己使用,也可能直接为商业销售。开发工作以特定的需求被初步提出为标志。需求的提出到初步可行性分析的完

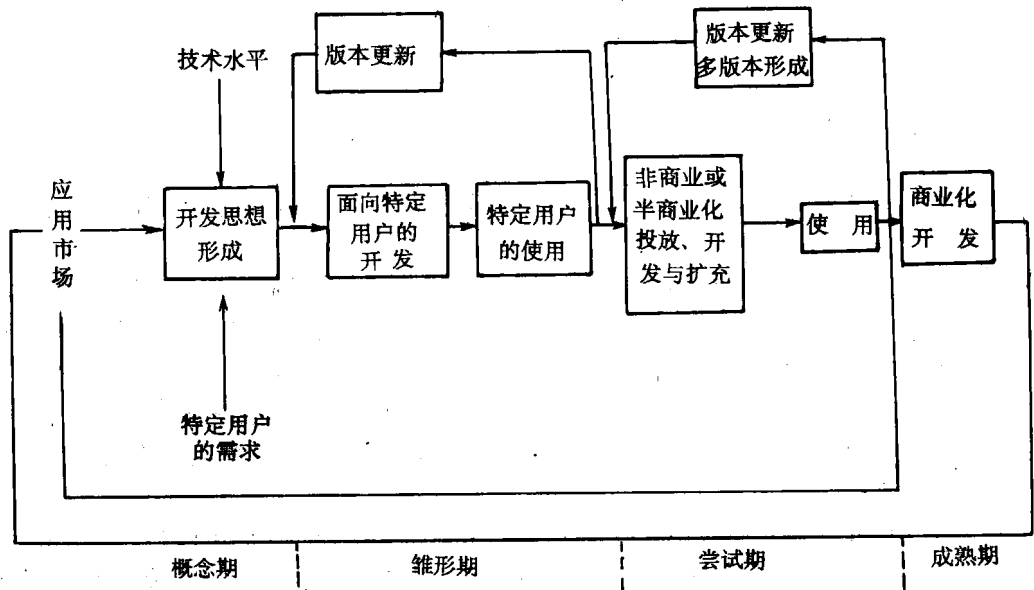


图 1-1 软件商品化过程模型

成，这一段就称为软件商品化的概念期。它的作用可概括为“开发思想的形成”，其中包括两层意思：一是需求的形成；二是实现该需求的方法的形成。用户需求的提出是以目前市场状况为背景，例如需求提出往往会针对市场上的某种缺乏和不足；同时，需求提出将受到目前技术水平的制约，一般总希望需求及实现都是较高水平的，但也不能空想，造成失败。实现思想主要取决于当前技术水平，尤其是开发者的水平，以及技术经济特性。

## 2. 雏形期

一个软件的开发工作一旦被确定下来并开始着手进行，便进入软件商品化过程的雏形期。在这一阶段，开发者将进行详细的可行性分析，详细的需求分析、设计、编码、调试和实现，而后交付特定的用户使用。在使用过程中，对不符合规格要求及不正确的部分加以修改、增补，直至特定用户达到一定程度的满意。绝对的满意是不存在的，因为软件的特性决定了只要它在运行中，它便处在持续的生长中，可以达到的往往是“满意”、“可能”与“值得”等因素的某种综合权衡。在雏形期将形成未来商品化软件的雏形，即主要典型特性。一个软件的雏形基本决定了未来商品化软件的生命力，甚至决定该软件能否商品化，这就要求雏形是可生长的。在开发雏形时，出发点是面向特定的用户，而且必须把主要精力集中在特定用户上，而不要以面向特定用户以外的用户为主导。只有这样才能保障开发工作不会因盲目求全而失败。60年代大型操作系统 MULTICS 起初失败的原因之一，就在于追求“庞大齐全”，因而未能较快地形成商品化软件。现在广泛使用的 UNIX 操作系统吸取了上述教训，先搞一个小而精的系统，逐步形成了商品化软件。当然，同时也要尽可能有意识地考虑到商品化的可能性，为这一目标准备一定的基础。雏形期的工作与一般性软件维护是不尽相同的，它还包括为特定用户向少量其他机器上的移植。

在雏形期内，雏形软件有一个不成熟到成熟的过程，经过在雏形期内的多次循环，最终形

成一个对特定用户来说是成熟的软件。

### 3. 尝试期

从市场角度的整体来看,雏形期成熟的软件在市场上仍是不成熟的。所以,在雏形期内成熟的软件还不能真正为市场所接受,也就是还不能立即“推广”,这一点是以前为人们所忽视的,也是过去软件难以商品化或失败的重要症结所在。它严重忽略了软件对特定应用的强依赖性,把在子域(特定应用)内的适应与子域外的适应混同起来。因此,在雏形期之后必须有一个尝试期。尝试期直接决定了软件商品化的成败。尝试期的实质包括两个方面:一是将子域内的适应性引向子域外的环境,这必须通过修改和增删功能来实现;二是形成多个变种(版本),将大的应用域化成多个各具共通性的子域,这就扩大了软件在更大环境中的适应性。

根据成功系统的经验,尝试期的工作主要有:传播、尝试和筛选。将雏形期已经成熟的软件以完全非商品化或半商品化方式提供给少量几个不同类型的用户去使用(一般是具有相当开发能力的用户),这些用户(包括雏形软件的拥护者)为了使雏形软件能在用户环境下运行,一般要做两项工作:一是往其他机器上移植;二是为软件增加新的功能或删去某些功能。移植的机器都是目前或将来流行的机种,这就为这类机器的大量用户购买该软件成为可能。软件增加或删除某些功能是针对某一子域用户的,于是形成后的软件将能够为这一类用户的大多数所接受。

### 4. 成熟期

经过上述试用工作后,多种经过扩充和删改过的软件能在多类用户中相对稳定地运行,即形成多种机器上的多种版本(同一种机器对不同用户可以是多种版本,不同种机器可以是同一种版本),而后的工作是严格地评价和选择,甚至将其标准化。各类用户都对使用过的软件表示基本满意,而那些尚未使用该软件的用户则表现出浓厚的兴趣,观看后又有直观的满意感,标志着软件在市场上成熟期的到来,即可以完全商品化。经过商品化开发(如加密、标准化等)后,投入应用市场。商品化开发也包括根据市场反馈进行的适应性开发和版本的自然更新。软件以商品形式进入应用市场后,可能会迅速出现兼容系统,这是取得巨大商业成功的显著标志。一定时期后,竞争性的软件开始出现,这是一种扩大商品市场的有利刺激,同时也预示着某个商品软件退出市场的开始。

## 1.2.3 软件产品的历史

1950至1962年是程序设计时代。在此期间,有才能的程序员竞相开发很方便的本人用的程序,除了周围人以外,别人是不用的,因此主要运用的技术是专家个人的手艺或者口传之类的东西。软件由程序员个人编写,由他使用运行,出了故障也由他修补。这是一种私人化的软件环境,设计只是在人们头脑中执行的隐含过程,通常都不编制文件资料。

1963至1970年是软件开发时代。这时期,社会对软件的需求增长,软件在整个计算机系统经费比重增大,适应某项任务的强有力的课题组也组织起来。同时有助于用户灵活使用软件的多种用户手册相继问世,用于排错和维护的内部说明书也纷纷出现。一个课题就成立一个课题组,成员尽可能挑选有能力的人员组成,各人大致均等地承担课题任务,从设计、编码直至编制手册。

从1971年开始,上述软件生产方式在成本上或产品交付期上已不能适应用户需要,不再由几个能力相当的人承担一个课题,而是采取主程序员负责制的组织方式来进行软件生产。首

先找一、二个经过多次训练且精通软件设计的人当领导,封为主程序员,负责全面工作,然后吸收若干编码能手专门从事程序编制,最后物色一、二个熟悉联机资料编制的资料员,把用户要用的,包括维护用的或教育用的各种各样的文件编入联机文件中。这是软件产品生产进行彻底分工的必经之路。

引起软件生产方式发展的原因之一,是使用的软件工具发生了巨大的变化。程序设计时代频繁使用的主要是手工制作的流程图,从流程图到编码就是标准的工作方法。但是,据统计,程序中指令条数超过 5000~10000 时,流程图就不是程序设计的有效工具了。从程序设计时代到软件开发时代,使用的软件工具有编辑程序和多种语言的翻译程序等。在软件开发时代,为了生产大量软件而广泛采用的是联机编辑程序和 HIPO 那样的分析设计工具,也研制了为排错用的排错工具等。

进入软件工程时代,就逐步采用生成程序的生产方式,也就是用设计规范语言写软件设计,用生成程序进行软件生产,如编译程序的生成程序等。更进一步,发展能写用户需求规范

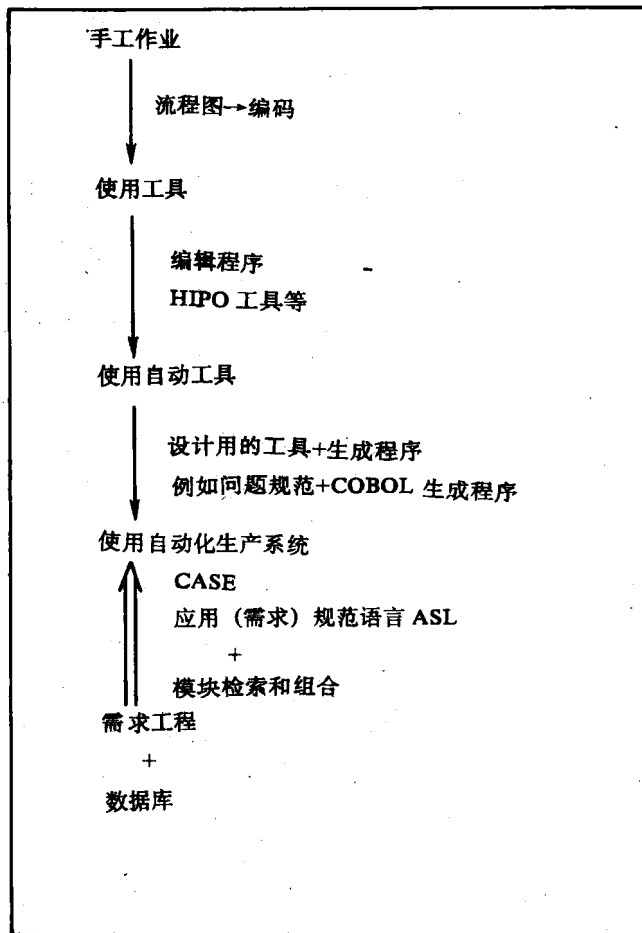


图 1-2 软件工具发展过程

语言,把这种语言书写的需求作为输入和原先存放的程序模块或数据模块加以比较、综合和评论,自动生产出符合用户需求的软件,这一工作正在重点进行试验。这样的需求规范技术及其需求规范语言正在逐渐发展为需求工程(Requirements Engineering)。这也是软件工程追求的目标之一。

80年代以来,软件工具进一步发展,出现了许多支持软件开发方法的综合的、集成化的软件开发环境,如CASE(Computer Aided Software Engineering)类软件产品发展很快,且逐步趋向商品化,大大促进了软件的自动化生产。

关于软件产品史和软件工具发展过程,如表 1-1 和图 1-2 所示。

表 1-1 软件产品史

年代	内容	维护情况
1950~1962 年程序设计时代(个人时代)	程序	不用(只限用本人及其周围)
1963~1970 年软件开发时代(课题小组时代)	软件	为维护准备的内部说明书,外部说明书,维护手册
1971~现在软件工程时代(主程序员负责制时代)	软件产品	机内维护,高可靠性功能,联机对话型手册

## 1.3 软件产业的形成

### 1.3.1 软件产业的概念

关于软件产业的概念,至今没有形成统一的说法。按日本“信息处理振兴事业协会等法律”规定:软件产业系指“根据他人要求而制作程序的事业。”欧美各国以前只提计算机服务业或信息服务业,未有软件产业的单独提法,近来虽有 Software Industry 的提法,但未有专门的定义。我国倾向于把软件产业定义为:“从事软件研究、开发、生产、销售、维护、使用和服务的产业。”有些国家和地区则把信息服务业视为软件产业。

### 1.3.2 软件产业发展过程

自 60 年代开始,随着电子计算机的品种和数量的急剧增加及其广泛使用,必然要有一定数量和质量的系统软件、应用软件来支持,于是就使已有的和正在开发的软件在社会上逐渐流通和使用起来,成为社会上共同享用的财富,由此就逐步出现了软件市场,促进了软件产业的形成。

大批商品化软件以商品形式在流通领域内流通是软件产业形成的最显著的标志。然而,软件商品化只是软件产业形成的前提,有商品化软件的存在,并不意味着已经形成了软件产业。软件产业的形成需要更复杂、更庞大的社会背景作为条件。商品化软件在软件产业形成的过程中是一个活的积极因素,少量软件产品作为商品进行交换只标志着软件产业的萌芽,但是这些软件将大大刺激环境背景——货主市场和需求市场,促进更多的商品化软件出现,从而进一步加快软件产业环境的成熟,最终导致软件产业的确立。从各国软件产业的组建和计算机应用的



情况分析结果来看,一般一个国家计算机应用达到 3000 台左右,即具备了建立软件产业的市场基础,可以考虑建立软件产业。

软件产业的发展至今经历了三个阶段:

#### 1. 萌芽阶段

标志:独立软件公司出现或政府有计划地开发,产业规模不大,商品化未解决,软件工程未提到议事日程上来。

时间:美国始于 1959 年,西德、日本始于 1966~1968 年。一般地讲,整个 60 年代处于软件产业的萌芽阶段。

#### 2. 形成阶段

标志:软件价格分离,成为可独立流通的商品,软件工程已提到议事日程,软件产业也有一定的规模。

时间:美国 1969 年率先实行软件价格分离,1970 年在纽约时报新闻记事信息检索系统软件开发中,首先实现工程化管理,形成软件产业。日本、西欧在 70 年代也陆续形成软件产业。总体看,70 年代处于软件产业形成阶段。

#### 3. 成长阶段

标志:高速增长。如美国 1979 年起进入软件产业高速发展时期,1979~1983 年平均增长率达 80%,其他先进国家一般在 30%以上。从 1980 年起,软件产业一般都进入成长期。

我国及亚洲各国都是 80 年代初才把建设软件产业提到日程上来,目前仍然处于萌芽阶段。

从组建到形成产业,美国从 1959~1970 年费时 11 年,日本从 1966~1978 年,费时 13 年。可见,软件产业的形成大体要 10 年左右的时间,而关键在于软件产品商品化和软件生产工程化的进展程度。

### 1.3.3 软件产业的业务范围

从国内外软件产业的实际情况看,目前软件产业主要由三大类型公司,即计算机生产厂家(软件生产部门)、独立软件公司和集成系统公司(即交钥匙系统公司)构成。而从事的业务主要有专门服务、软件产品和集成系统三大服务。

专门服务主要包括合同程序或定制软件、系统管理设计、与计算机有关的咨询、教育、训练等。

软件产品服务主要搞通用软件包,包括系统软件、应用软件以及软件工具等。

集成系统服务主要为集成系统提供软件。

### 1.3.4 软件产业的发展现状

当前活跃在建设软件产业舞台上的主要有四支力量。它们是美国、西欧(法国、德国、英国)、日本,以及亚洲其他国家和地区,其中包括新加坡、南朝鲜、印度、中国大陆和台湾省。在上述四支力量中,只有前三支是现实力量,亚洲只能算潜在力量。在现实力量中,美国是主导力量,日本已上升到第二位。东欧计算机发展比较缓慢,90 年代由于局势的变化,引起了经济形势的改变,有可能加快信息化进程,但因资金短缺,目前还不可能形成有竞争能力的软件产业。

#### 1. 软件的产值