软件工程方法学及应用

汤 庸 编著

中国三峡出版社

软件工程方法学及应用

汤 庸 编著

中国三峡出版社
1998

内容介绍

软件工程方法学是研究软件设计方法论及工程开发技术的一门新兴学科。本书系统地介绍软件工程与方法学的基本概念、基础理论，并注重讨论软件工程方法学新进展和新技术以及应用开发模型。

全书分三个部分。第一部分《软件方法学》，主要讨论软件设计的基本方法结构化和模块化，介绍程序设计正确性证明、程序形式化推导及变换技术。第二部分《软件工程学》，介绍软件工程学的四个时期（分析、设计、实现、维护）的基本方法和工具以及软件工程管理的基本概念。第三部分《应用开发模型》，介绍两种代表性的应用开发模型：面向对象模型和典型数据库应用模型。

本书是结合作者十多年数据库、软件工程方法学方面的教学和研究成果编写而成。本书构思新颖，结构清晰，逻辑性、系统性强；在选材上，注重基础理论和基本概念，又强调应用技术与实用方法。

本书适合于软件开发人员阅读，也可用做高等院校高年级学生和研究生教材或教学参考书。阅读本书时，不同层次的读者可以做不同的选择。

图书在编目（CIP）数据

软件工程方法学及应用 / 汤庸 编著. —北京：中国三峡出版社，1997.11

ISBN 7-80099-357-4

Ⅰ. 软… Ⅱ. 汤… Ⅲ. 软件工程 Ⅳ. TP311

中国版本图书馆 CIP 数据核字（97）第 23537 号

责任编辑：牛 力

特约编辑：李 龙

出 版 者：中国三峡出版社出版发行（北京市海淀区蔡公庄一号）

印 刷 者：农垦总局印刷厂

版　　次：1998 年 1 月第 1 版 1998 年 1 月第 1 次印刷

书　　号：**ISBN 7-80099-357-4/TP • 3**

开　　本：787×1092 毫米　1/16　印张：14.8　字数：342 千字

定　　价：**19.90** 元

前　言

软件工程学与程序设计方法学是研究软件开发与程序设计的两种途径和方法。随着软件技术的迅速发展，软件工程学和程序设计方法学的研究的内容都在不断发展，研究的内容和方法相互渗透。事实上，人们已经很少区分什么是软件工程学的范畴，什么是程序设计方法学的范畴了。也许采用“软件工程方法学”这个名词更能概括当前软件工程学和方法学研究的内涵。可以这样描述：软件工程方法学是应用计算机科学理论和工程方法相结合的研究方法，研究软件生存周期一切活动（包括软件定义、分析、设计、编码、测试与正确性证明、维护与评价等）的方法、工具和管理的学科。

简单来说，软件工程方法学是研究软件设计方法论及工程开发技术的一门新兴学科，它包括软件工程学和方法学研究的内涵。软件工程方法学最终目的是为经济的、有效的开发软件系统提供科学的方法和工具。软件工程方法学既强调软件（一般指大型软件）开发的工程特征，又强调软件设计方法论的科学性、先进性。

本书的组织与内容

本书系统地介绍软件工程与方法学的基本概念、基础理论，并注重讨论软件工程方法学新进展和新技术以及应用开发模型。全书共十一章，分三个部分。

第一章《绪论》介绍软件工程与方法学的形成，软件生命周期，并提出软件四个活动时期的概念，最后概述当前主要软件开发模型和开发方法。

第一部分《软件方法学》共三章，包括结构化技术和模块化技术，程序设计正确性证明技术，程序形式化推导及程序变换技术。这部分讨论了软件基本方法和基础思想：结构化与模块化原理与技术。形式地定义结构化程序，讨论模块化设计方法与准则；较系统介绍了程序设计方法学基本技术。

第二部分《软件工程学》共五章，前四章分别讨论软件的四个活动时期（软件定义与分析，软件设计，编码与测试，运行维护）的基本要求、基本方法步骤，介绍主要流行技术和工具。最后一章讨论软件工程管理与质量评价。这部分较完整的、系统的讨论了软件开发的全过程、基本方法及主要文档。

第三部分《应用开发模型》共两章，介绍两种典型应用模型：面向对象模型和典型数据库应用模型。前者是目前最流行的软件方法学，后者是软件工程最典型的应用对象。

本书的特点与读者

本书是结合作者十多年数据库、软件工程及方法学方面的教学体会和研究成果编写而成。本书构思新颖，结构清晰，逻辑性、系统性强；在选材上注重基础理论和基本概念，又强调应用技术与方法。主要特点包括：

1）将软件工程与方法学有机结合一起，使软件开发的基本理论方法与工程技术方法

相辅相成。目前还未见这样的专著。

2）提出软件生命周期的四个活动时期概念，并按之组织系统地组织《软件工程学》部分，结构清晰，逻辑性强。

3）本书主要章节都有“小结”，总结该章的主要内容和要求；并留有一定的思考题和综合实践练习。

本书适合于软件开发人员阅读，也可用做高等院校高年级学生和研究生教材和参考书。阅读本书时，不同层次的读者可以做不同的选择。例如，研究生可以主要选择第一部分与第三部分；大学本专科生则以第二部分为主，根据不同的学时数选择其它部分内容。在教学中可以设计相应的课程设计作业。

致　谢

本书的工作得到了广东省自然科学基金、广东省重点学科建设和广东省重点课程建设的资助。在有关教学与科研工作中得到了区益善教授、李显济教授、杜秋虹、李振坤、傅秀芬副教授、印鉴博士、王勇等老师的帮助和支持，研究生魏际洲、苏军根等也做了不少工作，在此一并表示衷心的感谢。

此书谨以献给我最亲爱、最慈祥的祖母！

编　者

1997 年 11 月 13 日

关于作者

汤庸，男，33 岁，分别于 85 年和 90 年获武汉大学计算机科学系学士和硕士学位，95 年破格晋升副教授。从事软件研究与教学十多年，在国内外负责和参加了十多项重要软件项目并取得多项成果。近年以第一作者出版著（译）作 5 部，在国内外发表论文三十多篇。现任广东工业大学计算机系副主任、硕士生导师，是广东省首批“千、百、十工程”培养对象。

目　　录

目　录

第二部分 软件工程学

第三部分 应用开发模型

目　录

第一章 绪 论

1.1 软件工程与方法学的形成

1.1.1 软件危机

通俗来说，计算机系统由硬件和软件两大部分。电子计算机自 1946 年诞生后到 60 年代中期是计算机发展的早期。在早期，计算机系统还是以硬件为主，软件费用是总费用的 20%左右。到了计算机中期（60 年代中期到 80 年代初期），软件费用迅速上升到总费用的 60%，软件不再只是技巧性和高度专业化的神秘机器代码。而到 85 年以后，软件费用已上升到今天 80%以上。软件相对硬件的费用比例在不断提高，图 1.1 是硬件和软件费用比例变化示意图。

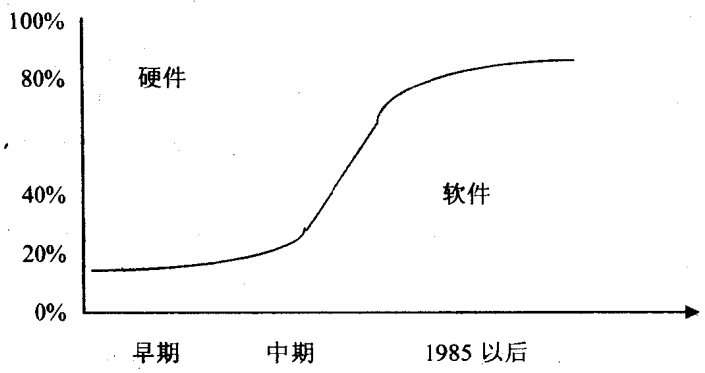

图 1.1 软硬件费用比例

自 60 年代中期以后，随着计算机技术迅速发展和应用领域迅速拓宽，软件需求迅速增长，软件数量急剧膨胀。传统的程序设计方法和软件开始出现了一系列严重的问题。这些严重问题就是所谓的“软件危机”（Software Crisis）。

软件危机主要包含软件开发和维护两方面的问题。具体来说，软件危机主要有如下一些表现（但不是全部）。

1） 计算机应用系统越来越大，软件复杂程度越来越高。以个体为特征的“软件作坊”方式已无法完成大型软件系统的开发。

2） 软件发展历史较短，缺乏软件开发经验和大量的历史数据积累，使得软件开发成本和进度估计常常很不准确，很少有在预定的成本和预定的进度内完成的。

3） 由于没有统一的、科学的开发规范，使得软件常常是不可维护的。

4） 软件质量差，可靠性难以保证。由于没有将软件质量保证技术（审查、复查和测试）应用于软件开发的全过程，因而，所开发软件的质量往往靠不住。

5） 由于没有科学的软件需求分析方法和手段，往往是仓促上阵编写程序，使的用户对“已完成的”软件系统不满意现象经常发生。

6） 软件通常没有适当的文档资料。文档资料是软件的重要组成部分，而不是可有可无的。没有必要的文档资料或文档资料不合格，必然给软件开发和维护带来许多严重的困难和问题。

7） 软件应用系统的需求量日益增长，软件产品“供不应求”现象使人们不能充分利用计算机硬件提供的巨大潜力。软件开发生产率提高的速度远远跟不上计算机应用迅速普及和深入的趋势。

1.1.2 软件工程与方法学的形成

软件危机出现了，如何进行软件开发呢？为了解决这个问题，程序设计方法学和软件工程学就逐渐形成了。

1967 年 Floyd 提出的断言方法证明流程图程序的正确性；1968 年 Dijkstra 的“GOTO”有害论；1969 年 Hoare 在 Folyd 断言法基础上提出的程序公理方法；1971 年 Wirth 的“自顶而下逐步求精”等对软件工程和程序设计方法学的形成和初期的发展有着深刻的影响。1969 年 IFIP（国际信息处理协会）成立了“程序设计方法学工作组”——WG2.3，云集了当时许多著名的计算机科学家，专门研究程序设计方法学，这个国际组织对以后的程序设计方法学的发展起了很大的促进作用。

“软件工程”（Software Engineering）作为一个术语，是在 1968 年北大西洋公约组织的一次计算机学术会议上，正式提出来的。这个会议专门讨论了软件危机问题。这次会议是软件发展史上一个重要的里程碑。

至此，一门新兴的计算机学科——软件工程与方法学就诞生了。

1.1.3 软件工程学与程序设计方法学

事实上，在软件工程和方法学形成以后，研究工作一开始就分成了两种不同的角度和方法，形成了两种紧密相关、相辅相成又各有侧重的学科。一种是以数学理论为基础的理论性学科：程序设计方法学；一种是以工程方法为基础的工程学科：软件工程学。软件工程学和程序设计方法学的研究都是研究软件开发和程序设计的学科，它们的研究对象、研究内容、出发点和目标都是一致的。

软件工程学是主要应用工程的方法和技术研究软件开发与维护的方法、工具和管理的一门计算机科学与工程学交叉的学科。程序设计方法学则是主要运用数学方法研究程序的性质以及程序设计的理论和方法的学科。

软件工程学与程序设计方法学的研究对象是软件和程序。它们的根本目标是以较低的成本开发高质量的软件和程序，具体包括：1）提高软件的质量与可靠性；2）提高软件

的可维护性；3）提高软件生产率，降低软件开发成本等。

但是，软件工程学和程序设计方法学研究的途径和侧重点有所差异，主要差异有：

1） 研究方法和途径不同。软件工程学应用的是工程方法；而程序设计学依据的数学方法。软件工程学注重工程方法与工具研究，程序设计方法学则注重算法与逻辑方法研究。

2） 研究对象有所侧重，软件工程的对象所指的软件，一般是指“大型程序”，是一个系统；而程序设计方法学的研究对象则侧重于一些较小的具体程序模块，早期的程序设计方法学研究重点是某个单独的程序的时空效率、正确性证明等问题。

3） 软件工程学注重“宏观可用性”；程序设计方法学注重“微观正确性”。例如软件工程学研究软件的“可靠性”的方法是“软件测试”，程序设计方法学研究的方法则是程序的“正确性证明”。

1.1.4 软件工程方法学

随着软件技术的迅速发展，软件工程学和程序设计方法学的研究内容也都在不断发展，研究的内容和方法相互渗透。事实上，人们已经很少、也没有必要区分什么是软件工程学的范畴，什么是程序设计方法学的范畴了。逐渐地，这两条研究途径的界限又模糊化、一体化了。

一方面，程序设计方法学研究已发生了较大的变化，逐渐从“纯粹的程序”正确性证明等较老的课题转向“软件”的结构化、正确性、可靠性及软件设计方法方面的研究。例如，现在程序设计方法学及软件工程学都将面向对象的方法做为其重要的新的研究方向，“程序设计方法学”正逐渐发展成“软件设计方法学”。

另一方面，软件工程从一开始就是以程序设计方法学为基础的一门工程学科，而且还在不断吸收程序设计方法学和计算机科学理论新成果和新技术。从某种意义上，可以说，软件工程学实际上就是“应用设计方法学”。

所以实际上“软件工程学”或“程序设计方法学”术语都已难以准确表达它们的研究内涵和含义了。也许采用“软件工程方法学”或“软件工程与方法学”更能概括当前软件工程学和方法学研究的内涵。

可以这样描述：软件工程方法学是指应用计算机科学理论和工程方法相结合的研究方法，研究软件生存周期一切活动（包括软件定义、分析、设计、编码、测试与正确性证明、维护与评价等）的方法、工具和管理的学科。

软件工程方法学既强调软件（一般指大型软件）开发的工程特征，又强调软件设计方法论的科学性、先进性。其的基本内容包括：

1） 结构化理论与方法；

2） 模块技术与数据抽象；

3） 软件测试与程序正确性证明；

4） 软件分析与设计方法、工具及环境；

5） 软件工程管理与质量评价。

1.2 软件与软件生命周期

1.2.1 软件与程序

软件工程与方法学的研究对象是软件和程序，这里软件通常是指大型程序。

一、软件的定义

什么是软件呢？软件有许多定义形式，一种较公认的定义为：

软件由如下三部分组成：

1） 在运行时能够提供所希望的功能和性能的指令集合，即程序；

2） 使得程序能够正确运行的数据结构；

3） 描述程序研制过程、方法和使用的文档资料。

二、软件与程序的区别

在计算机发展早期，软件与程序通常是没有分别的。但是，随着计算机的发展，软件与程序有很大程度的区别，主要区别如下。

1） 由上述定义也可以看出，程序是软件的一个组成部分。

2） 软件工程的所指软件不同于一般程序，规模庞大是它一个显著的特点。

三、软件与硬件的区别

相对硬件而言，软件有如下特点：

1） 软件是计算机系统的一个逻辑部件，而不是一个物理部件。它具有抽象性，虽然可以记录在介质上，但无法看到软件本身的形态。

2） 软件没有“用坏”、“老化”的概念，如果运行发现错误，也是软件开发时期引入没有被测试出来的故障。

3） 软件是在研制开发活动中被创造出来的，其开发成本远远大于生产（拷贝）成本。

4） 软件的开发至今还没有完全摆脱人工开发方式，而且软件开发成本越来越高。

1.2.2 软件生命周期

软件生命周期（Software Life Cycle），也称为软件生存周期，是软件工程最基础的概念。软件工程的方法、工具和管理都是以软件生命周期为基础的活动。换句话说，软件工程强调的是使用软件生命周期方法学和使用成熟的技术和方法来开发软件。

软件生命周期的基本思想是：任何一个软件都是从它的提出开始到最终被淘汰为止，有一个存在期。

软件的“生命周期”概念与前面所提到的软件没有“用坏”和“老化”概念并不矛盾，前者所谓的“生命”是指软件的“应用”寿命，如果一个软件不再具有应用价值，并

不是说该软件被用坏了，而是指该软件不再有用了。

1.2.3 四个活动时期

类似人在生命周期内划分成若干阶段（如幼年、少年、青年、中年、老年等）一样，软件在其生存期内，也可以划分成若干阶段，每个阶段有较明显的特征，有相对独立的任务，有其专门的方法和工具。目前，软件生命周期的阶段划分有多种方法。软件规模、种类、开发方式、开发环境与工具、开发使用模型和方法论都影响软件生命周期阶段的划分。但是，软件生命周期阶段的划分应遵循一条基本原则，即：要使个阶段的任务尽可能相对独立，同一阶段各项任务的性质应尽可能相同。这样降低每个阶段任务的复杂程度，简化不同阶段之间的联系，有利于软件开发的管理。

软件生命周期一种典型的阶段划分为：问题定义、可行性研究、需求分析、概要设计（总体设计）、详细设计、编码与单元测试、综合测试、维护等八个阶段。有些资料也将问题定义与可行性研究合并称为系统分析阶段或软件计划阶段。

但是，软件生命周期内阶段的划分要受软件的规模、性质、种类、开发方法等影响，阶段划分过细还会增加阶段之间联系的复杂性和软件工作量。在实际软件工程项目较难操作。

我们提出软件生命周期内划分成四个活动时期：软件分析时期、软件设计时期、编码与测试时期以及软件运行与维护时期。它们的关系如图 1.2。

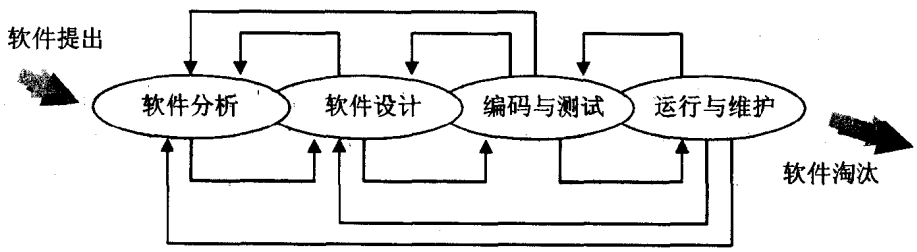

图 1.2 软件的四个时期

一、软件分析时期

软件分析时期，也可称为软件定义与分析时期。这个时期的根本任务是确定软件项目的目标，软件应具备功能和性能，构造软件的逻辑模型，并制定验收标准。在此期间，要进行可行性论证，并做出成本估计和经费预算，制定进度安排。通俗地说，软件定义与分析时期主要解决如下问题：

1） 要做的是什么软件？

2） 有没有可行性？

3） 软件的具体需求是什么？验收标准是什么？