

软件工程的三段论起源及发展趋势

作者：韩万江 整理：Maxwell

软件工程(Software Engineering, 简称为 SE)是针对软件这一具有特殊性质的产品的工程化方法。软件工程涵盖了软件生存周期的所有阶段,并提供了一整套工程化的方法,来指导软件人员的工作。

一. 软件工程定义

“工程”是科学和数学的某种应用,通过这一应用,使自然界的物质和能源的特性能够通过各种结构、机器、产品、系统和过程,成为对人类有用的东西。因而,“软件工程”就是科学和数学的某种应用,通过这一应用,使计算机设备的能力借助于计算机程序、过程和有关文档成为对人类有用的东西。

软件工程的成果是为软件设计和开发人员提供思想方法和工具,而软件开发是一项需要良好组织、严密管理且各方面人员配合协作的复杂工作。软件工程正是指导这项工程的一门科学。软件工程在过去一段时间内已经取得了长足的进展,可以说在软件的开发和应用中起到了其应有的作用。

高质量的软件工程可以保证软件工业中生产的软件是高质量的产品、用户满意的产品。但是,对软件工程的界定,总是存在一定的差异。软件工程应该包括哪些知识?这里我们引用 IEEE 在软件工程知识体系指南(SWEBOK: Guide to the Software Engineering Body of Knowledge 2004Version)中的定义,它是这样定义软件工程的:①软件开发、实施、维护的系统化、规范化、质量化的方法的应用,也就是软件的应用工程;②对上述方法的研究。

IEEE 的软件工程知识体系指南(SWEBOK)中界定了软件工程的 10 个知识领域(KAs: Knowledge Areas),即:软件需求(Software requirements)、软件设计(Software design)、软件构建(Software construction)、软件测试(Software testing)、软件维护(Software maintenance)、软件配置管理(Software configuration management)、软件工程管理(Software engineering management)、软件工程过程(Software engineering process)、软件工程工具和方法(Software engineering tools and methods)和软件质量(Software quality)。这 10 个知识领域的每个知识领域还包括很多子领域。

二. 软件工程起源的三段论

这里的三段论是指从时间角度看软件工程起源的三个阶段。本世纪中期软件产业从零开始起步,在短短的 50 年的时间里迅速发展成为推动人类社会发展的龙头产业。随着信息产业的发展,软件对人类社会越来越重要。软件发展的五十年历史中,人们对软件的认识经历了一个由浅到深的过程。

第一个写软件的人是 Ada(Augusta Ada Lovelace),在 1860 年代他尝试为 Babbage(Charles Babbage)的机械式计算机写软件。尽管失败了,但他将永远载入计算机发展的史册。1950 年代,软件伴随着第一台电子计算机的问世诞生了。以写软件为职业的人也开始出现,他们多是经过训练的数学家和电子工程师。

1960 年代美国大学里开始授予计算机专业的学位，教人们写软件。

软件发展的历史可以大致分为如下的三个阶段：

第一个阶段是 1950 年代到 1960 年代，是程序设计阶段，基本是个体手工劳动的生产方式。这个时期，一个程序是为一个特定的目的而编制的，软件的通用性是很有限的，软件往往带有强烈的个人色彩。早期的软件开发没有什么系统的方法可以遵循，软件设计是在某个人的头脑中完成的一个隐藏的过程。而且，除了源代码往往没有软件说明书等文档，因此这个时期尚无软件的概念，基本上只有程序、程序设计概念，不重视程序设计方法，主要是用于科学计算，规模很小，采用简单的工具(基本上采用低级语言)，硬件的存储容量小，运行可靠性差。

第二阶段是 1960 年代到 1970 年代，是软件设计阶段，小组合作生产方式。在这一时期软件开始作为一种产品被广泛使用，出现了“软件作坊”。这个阶段基本采用高级语言开发工具，开始提出结构化方法。硬件的速度、容量、工作可靠性有明显提高，而且硬件的价格降低。人们开始使用产品软件(可购买)，从而建立了软件的概念。程序员数量猛增，但是开发技术没有新的突破，软件开发的方法基本上仍然沿用早期的个体化软件开发方式，软件需求日趋复杂，维护的难度越来越大，开发成本令人吃惊的高，开发人员的开发技术不适应规模大、结构复杂的软件开发，失败的项目越来越多。

第三个阶段是从 1970 年代至今，为软件工程时代，是工程化的生产方式。这个阶段的硬件向超高速、大容量、微型化以及网络化方向发展，第三、四代语言出现。数据库、开发工具、开发环境、网络、分布式、面向对象技术等工具方法都得到应用。软件开发技术有很大进步，但未能获得突破性进展，软件开发技术的进步一直未能满足发展的要求。软件的数量急剧膨胀，一些复杂的、大型的软件开发项目被提出来了，但很多软件最后都得到了一个悲惨的结局。在那个时代，很多的软件最后都得到了一个悲惨的结局。很多软件项目的开发时间大大超出了规划的时间表，一些项目导致了财产的流失，甚至某些软件导致了人员伤亡。同时软件开发人员也发现软件开发的难度越来越大，在软件开发中遇到的问题找不到解决的办法，使问题积累起来，形成了尖锐的矛盾，失败的软件开发项目屡见不鲜，因而导致了软件危机。

软件危机指的是在计算机软件的开发和维护过程中所遇到的一系列严重问题。概括来说，软件危机包含两方面问题：①如何开发软件，以满足不断增长，日趋复杂的需求；②如何维护数量不断膨胀的软件产品。落后的软件生产方式无法满足迅速增长的计算机软件需求，从而导致软件开发与维护过程中出现一系列严重问题。

最为突出的例子是美国 IBM 公司于 1963 年~1966 年开发的 IBM360 系列机的操作系统。难怪该项目的负责人 Fred Brooks(F·D·希罗克斯)在总结该项目时无比沉痛地说：“……正像一只逃亡的野兽落到泥潭中作垂死挣扎，越是挣扎，陷得越深，最后无法逃脱灭顶的灾难，……程序设计工作正像这样一个泥潭……，一批批程序员被迫在泥潭中拼命挣扎，……，谁也没有料到问题竟会陷入这样的困境……。”IBM360 操作系统的历史教训已成为软件开发项目中的典型事例被记入史册。由于软件危机的产生，迫使人们不得不研究、改变软件开发的技术手段和管理方法。从此软件生产进入软件工程时代。

1968 年北大西洋公约组织的计算机科学家在联邦德国召开的国际学术会议上第一次提出了“软件危机”(Software crisis)这个名词。同时，讨论和制定摆脱“软件危机”的对策。在那次会议上第一次提出了软件工程(Software

engineering)这个概念，从此一门新兴的工程学科——软件工程学——为研究和克服软件危机应运而生。

“软件工程”的概念是为了有效地控制软件危机的发生而被提出来的，它的中心目标就是把软件作为一种物理的工业产品来开发，要求“采用工程化的原理与方法对软件进行计划、开发和维护”。软件工程是一门旨在开发满足用户需求、及时交付、不超过预算和无故障的软件的学科。软件工程的主要对象是大型软件。它的最终目的是摆脱手工生产软件的状况，逐步实现软件开发和维护的自动化。

我们要求工程目标能在一定的时间、一定的预算之内完成。软件工程是针对软件危机提出来的，从微观上看，软件危机的特征正是表现在完工日期一再拖后、经费一再超支，甚至工程最终宣告失败等方面；而从宏观上看，软件危机的实质是软件产品的供应赶不上需求的增长。

自从软件工程概念提出以来，经过几十年的研究与实践，虽然“软件危机”没得到彻底解决，但在软件开发方法和技术方面已经有了很大的进步。尤其应该指出的是，自80年代中期，美国工业界和政府部门开始认识到，在软件开发中，最关键的问题是软件开发组织不能很好地定义和管理其软件过程，从而使一些好的开发方法和技术都起不到所期望的作用。也就是说，在没有很好地定义和管理软件过程的软件开发中，开发组织不可能在好的软件方法和工具中获益。

三. 软件工程内容的三段论

这里的三段论是指从内容角度看软件工程的三段理论。软件工程是为克服软件危机而提出的一种概念，并在实践中不断地探索它的原理，技术和方法。在此过程中，人们研究和借鉴了工程学的某些原理和方法，并形成了软件工程学。软件工程的目的是提高软件的质量与生产率，最终实现软件的工业化生产。既然软件工程是“工程”，那么我们从工程的角度看一下软件项目的实施过程，如图1所示：

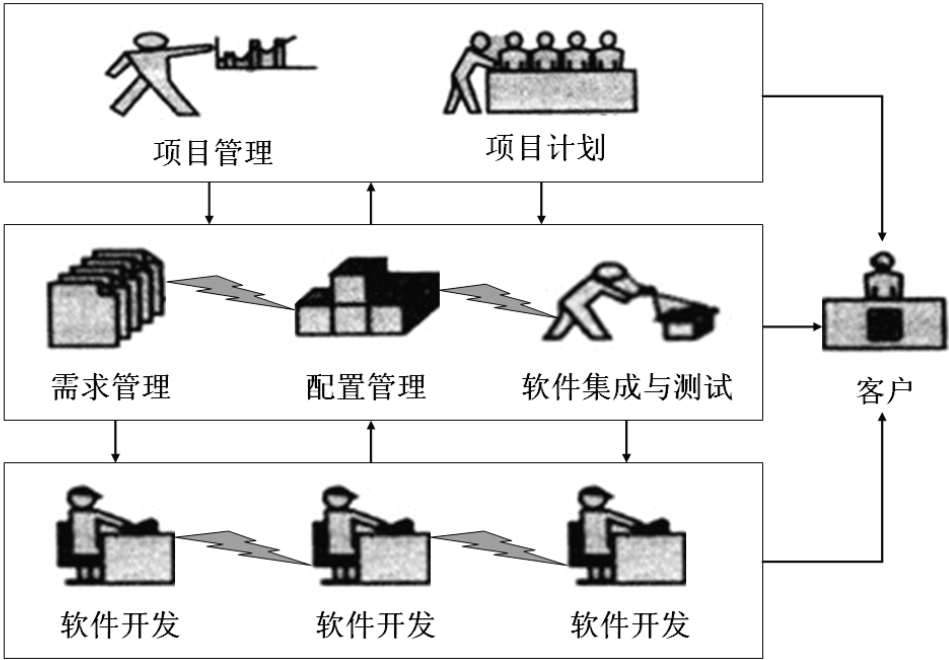


图 1：软件项目的实施过程

客户的需求启动了一个软件项目，为此我们需要先规划这个项目，即完成项目计划，然后根据这个项目计划实施项目。项目实施的依据是需求，这个需求类似工程项目的图纸，开发人员按照这个图纸生产软件，即设计、编码。在开发生产线上，将开发过程的半成品，通过配置管理存储和管理，然后进行必要的集成和测试，直到最后提交给客户。

在整个开发过程中需要进行项目跟踪管理。软件工程活动是“生产一个最终满足需求且达到工程目标的软件产品所需要的步骤”。这些活动主要包括开发类活动、管理类活动和过程类活动，这里将它定义为“软件工程的三段论”，或者“软件工程的三线索”。一段论是“软件项目管理论”，二段论是“软件项目开发论”，三段论是“软件过程改进论”。这个三段论可以用一个三角型表示，如图2所示：

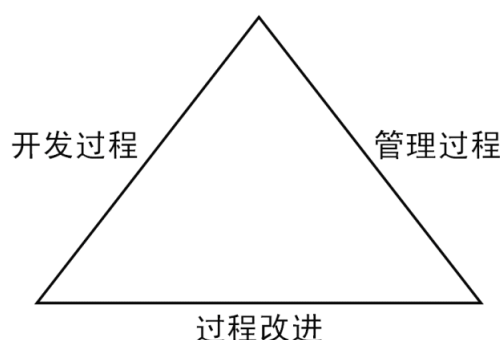


图2：软件工程的三段论

它们类似于相互支撑的三角形的三个边。我们知道三角形是最稳定的，要保证三角形的稳定性，三角形的三个边必不可少，而且要保持一定的相互关系。

其中：

开发过程是软件人员生产软件的过程，例如需求分析、设计、编码、测试等，相当于生产线上的生产过程。管理过程是项目管理者规划软件开发、控制软件开发的过程，相当于生产线上的管理过程，管理过程是伴随开发过程进行的过程。过程改进相当于对软件开发过程和软件管理过程的“工艺流程”进行管理和改进，如果没有好的工艺就生产不出好的产品，它包括对开发过程和管理过程的定义和改进。

为了保证软件开发、软件管理过程的有效性，应该保证这些过程的高质量和过程的持续改进。为此我在软件工程的系列教程中分三册以案例贯穿讲述了软件项目开发过程、软件项目管理过程和软件过程改进。

传统工业中知名的生产方式可算是“手工作坊式”生产方式。过去的一段时间里，甚至到现在，中国软件业中还有一部分公司的开发方式与之类似。公正地说，以此方式还是成就了好多成功的应用开发项目，甚至可以说此法支撑起了软件开发的初期事业。但是，在我们的“作坊”里还有太多的项目失败，例如开发周期的不可控制、项目结果为用户所不认可、项目最终的严重亏损等。这些失败的惨痛出乎我们的意料，以至于我们无所适从。再加上技术人员、资金严重匮乏的困扰，软件开发管理举步维艰。

作坊式的软件企业中，很多方法、规则都是装在开发人员的脑子里面的，往往会因为一两个开发骨干走了，就造成整个公司的瘫痪。赌注完全押在这一两个人的身上，资本投入风险很大，如果研发骨干另谋高就，公司投资就将全部付之流水，作坊式的运作模式严重阻碍了软件企业的成长。

让软件工程成为真正的工程，就需要软件项目的开发、管理、过程等方面规范化、工程化、工艺化、机械化。软件开发过程中脑力活动的“不可见性”大大增加了过程管理上的困难，因此软件工程管理中的一项目指导思想就是千方百计地使这些过程变为“可见的”以及事后可以检查的记录。只有从一开始就在开发过程中严格贯彻质量管理，软件产品的质量才有保证。否则，开发工作一旦进行到后期，无论怎样测试和补漏洞，都会无济于事。

四. 软件工程的发展状况和趋势

软件工程的研究热点是随着软件技术的发展而不断变化的。即便在软件工程的领域内，研究热点也在不断转移。以往软件工程一直不能像其他产品一样，做到标准化，但是，随着技术条件的不断成熟和相应标准的出台，软件人员已经开始重视这方面的工作。实际上可以将许多软件工作分成许多部件去构造。很有可能今后的软件队伍会分为两个部分，一部分专门从事评估，另一部分专门从事集成，集成的对象是软构件。

软构件的开发与运用刚刚开始。在一些公共领域，例如软件的用户界面，通用软构件的使用已经屡见不鲜。然而，对于各行各业的专业领域来说，领域构件的开发和使用还基本处于空白状态。这一工作的进行，意味着各行各业要对本专业领域内的知识形态加以归纳整理，然后以最新的软件形式表达出来。如果全面铺开，这就是一件规模浩大的社会工程，需要各行各业的领域专家和软件专家通力合作才能完成。如果软件生产的“构件-集成”格局的趋势成为现实，各种应用领域里的构件的设计与生产将开辟出一个十分广阔的新天地，产生出巨大的市场需求，而且软构件的使用可以渗透到符合软构件标准规范的所有系统中。

在软件开发过程中人们开始研制和使用软件工具，用以辅助进行软件项目管理与技术生产，人们还将软件生命周期各阶段使用的软件工具有机地集合成为一个整体，形成能够连续支持软件开发与维护全过程的集成化软件支援环境，以期从技术和管理两方面解决软件危机问题。

此外，人工智能与软件工程的结合成为 80 年代末期活跃的研究领域。基于程序变换、自动生成和可重用软件等软件新技术的研究也已取得一定的进展，把程序设计自动化的进程向前推进了一步。在软件工程理论的指导下，发达国家已经建立起较为完备的软件工业化生产体系，形成了强大的软件生产能力。软件标准化与可重用性得到了工业界的高度重视，在避免重复劳动、缓解软件危机方面起到了重要作用。

五. 结束语

在现代社会中，软件应用于多个方面。同时，各个行业几乎都有计算机软件的应用，比如工业、农业、银行、航空、政府部门等。这些应用促进了经济和社会的发展，使得人们的工作更加高效，同时提高了生活质量。随着现代信息技术的飞速发展，软件产品的规模也越来越庞大，而软件开发规模也导致开发队伍的逐渐增大。如果没有规则，每个人都有自己的一套方式，甚至有的人有几套方式，这样，当几个人在一起开发软件的时候，最终的结果就只能是一片混乱。这就意味着我们迫切地需要一种管理规范来规范每个软件人员的工作，使每个项目组成

员按照约定的规则准时完成自己的工作。而软件工程就是这样一套用于软件的团队开发、以提高软件质量和程序员工作效率为目的的规范。

首先，我们要注意到，其他工业领域里的工程的建设目标（比如建设一座桥梁）以及工程在整个工期内所处的环境是相对固定的，而软件工程则不然。众所周知，许多软件项目（例如 MIS）的用户往往说不清楚自己的需要是什么。不仅如此，由于技术进步，由于软件的使用改变了用户的工作环境，由于用户周围环境的变迁，由于用户自身对软件的功能和使用软件带来益处的认识的加深，软件工程建设目标在工程进行期间就会不断地变更。

另外，传统工业能够在相当短的时期内建立起一套与供应商无关的部件分解体系以及与之相应的、受到全社会承认的工业标准，从而形成了严密而有效的社会分工体系。而对于软件产业来说，在很长的时期里，每一项开发工作几乎都要从头做起。软件部件的重复利用处于很低的水平，开发者很少能够“从不同厂商采购软部件，再加上自己的东西，迅速形成一个系统”。这种情况只是到近几年才开始改变。

很多国内搞计算机的专家都认为，国内的软件研发过程个人色彩比较浓。过分地依靠个人无法形成产业规模，而没有规模就谈不上产业化了。实际上，不管有没有软件工程，不管是否存在一个万能的框架系统，我们的应用系统还是要做，各种各样的软件还是要开发。说到底，软件系统是因为有需求才存在的，有应用域才有软件存在的意义。很多时候，我们可以看到国外有各种各样的软件和创新，而我们没有，我们还面临很多的问题，我们不能说，只要解决软件工程问题就能推进软件产业。