

Plan and Control of Soft Projects



软件项目计划与控制

张俊光 著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

Plan and Control of Soft Projects



软件项目计划与控制

张俊光 著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书较为系统地介绍了软件项目计划与控制的管理理论和实践方法，并附有实际的项目案例、翔实的项目数据、易行的操作方式。本书在编写过程中考虑了 PMBOK®和 CMMI1.2版的主要内容，以软件项目计划与控制的实施过程为主线，介绍了软件项目计划基础、软件项目计划方法、软件项目重大偏差标准设定、软件项目监督与控制、第三方监控。作者以其丰富的教学和实践经验，为读者带来了软件项目计划与控制管理中众多问题的全新视角与观点。本书还介绍了大量软件项目计划与控制的实用表单和工具，为读者提供了现实的操作指南。

本书可作为高等院校项目管理专业、软件工程专业及工程管理专业本科和研究生项目管理类课程的教材或参考书。同时，本书对于企业项目管理人员、软件开发人员也是一本重要的参考读物。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目（CIP）数据

软件项目计划与控制 / 张俊光著. —北京：电子工业出版社，2009.5
ISBN 978-7-121-08389-1

I. 软… II. 张… III. 软件开发—项目管理 IV. TP311.52

中国版本图书馆 CIP 数据核字（2009）第 025395 号

责任编辑：韩丽娜

文字编辑：徐 萍

印 刷：北京机工印刷厂

装 订：三河市鹏成印业有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：720×1000 1/16 印张：13 字数：195 千字

印 次：2009 年 5 月第 1 次印刷

定 价：29.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zltz@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。
服务热线：（010）88258888。

前言

与其他学科相比，项目管理是一个比较年轻的学科；与传统行业（如机械行业、建筑行业等）相比，计算机软件行业是一个比较年轻的行业。软件项目管理则是年轻行业中的一门年轻学科。20世纪70年代中期，美国首次提出软件项目管理概念。针对软件开发不能按时提交、预算超支和质量达不到用户要求等原因，美国国防部研究发现，70%的项目出现问题是因为管理不善，而非技术原因。于是，软件开发中的各项管理问题逐渐被软件开发者所重视。到了20世纪90年代中期，软件项目管理不善的问题仍然存在。据专门对IT项目进行调查研究的Standish Group统计，1994—2003年，IT项目的平均成功率（按时、按预算实现项目的质量目标）仅从16%提升到34%，仍有一大半的项目要么超支，要么拖期，要么降低了质量标准，要么“无疾而终”。

软件企业实施项目管理的挑战，贯穿于项目的整个生命周期。本书以当前软件项目管理的实际情况为出发点，结合作者多年的项目管理理论研究、项目管理课程教学和企业实践经验，对软件项目的计划与控制理论体系进行了系统思考并提出了切实可行的操作方法。本书具有很强的系统性、针对性、实效性，对于从事软件项目的各类人员及项目管理者有良好的借鉴作用。

本书按照软件项目管理的基本流程，建立了一个从估计到偏差标准设定再到监控的软件项目计划与控制的模型体系，同时针对体系中的每一个“子系统”（如规模、工作量和进度等）也分别建立

管理模型，最终形成一个集成性的“大中套小”的项目计划与控制模型体系。全书共由导论、软件项目计划基础、软件项目计划方法、软件项目重大偏差标准设定、软件项目监督与控制、第三方监控 6 章组成。

本书的突出特点是具有理论和实践的双重支撑。理论上，本书有 20 余篇（包括一级学报、EI 检索、ISTP 检索和中文核心期刊）论文、文献的支撑；实践上，本书凝聚了作者十余年大型 IT 企业项目管理的工作心得。因此，本书可作为项目管理的教学用书，也可供企业管理界人士和从事软件开发的人士参考。

本书的撰写得到北京科技大学经济管理学院及北京邮电大学经济管理学院领导和多位老师的大力支持与帮助，同时也参考了大量的国内外相关资料及文献，在此一并表示衷心的感谢。

由于作者水平有限，书中难免有不妥之处，希望读者和同行不吝批评指正，以便再版时完善。

张俊光

目 录

第 1 章 导 论	1
1.1 软件项目	1
1.2 软件项目管理	2
1.3 软件项目计划与控制模型体系	6
第 2 章 软件项目计划基础	11
2.1 历史数据的收集	12
2.1.1 需要收集的历史数据	12
2.1.2 历史数据的收集与分析	15
2.2 平均生产率及线性回归方程统计	17
2.2.1 组织平均生产率统计	19
2.2.2 工作量和规模的线性回归方程	24
2.3 软件项目范围管理	31
2.3.1 软件项目范围计划	32
2.3.2 软件项目范围定义及核实	33
2.3.3 工作分解结构	38
2.3.4 范围变更控制	41
2.4 项目范围管理附表	45
第 3 章 软件项目计划方法	51
3.1 软件项目规模估计	52

3.1.1	规模估计方法简介	52
3.1.2	规模估计方法的应用	55
3.2	软件项目工作量估计	59
3.2.1	自上而下的估计方法	59
3.2.2	自下而上的估计方法	63
3.2.3	基于经验的系数估计方法	65
3.2.4	模型估计法	67
3.2.5	工作量估计方法有效性分析	72
3.3	软件项目进度估计	74
3.3.1	构筑进度安排指南	74
3.3.2	进行项目进度估计	78
3.4	其他估计项的估计方法	92
3.4.1	估计项目共利人员	94
3.4.2	估计项目所需知识与技能	96
3.4.3	估计项目资源	98
第 4 章 软件项目重大偏差标准设定		100
4.1	概述	100
4.1.1	几个重要概念	100
4.1.2	重大偏差标准及风险储备时间的定性设定	102
4.2	项目规模重大偏差标准定量设定	104
4.2.1	项目规模偏差的统计与分析	104
4.2.2	项目规模重大偏差标准的设定	106
4.3	项目工作量重大偏差标准定量设定	106
4.3.1	项目工作量偏差的统计与分析	106
4.3.2	项目工作量重大偏差标准及风险储备时间设定	108

4.4	项目进度重大偏差标准定量设定	108
4.4.1	项目进度偏差的统计与分析	108
4.4.2	项目进度重大偏差标准及风险储备时间设定	111
4.4.3	按统计概率进行重大偏差标准设定	115
4.5	一般偏差标准设定	117
第 5 章 软件项目监督与控制		121
5.1	概述	122
5.1.1	监控的原理	123
5.1.2	监控过程中重大偏差标准的使用	126
5.1.3	软件项目监控过程	127
5.2	项目各监控项的监控	129
5.2.1	项目规模的监控	129
5.2.2	项目工作量的监控	133
5.2.3	项目进度的监控	138
5.2.4	承诺的监控	144
5.2.5	其他监控项的监控	147
5.2.6	总结各监控项的监控情况和分析结果	150
5.3	项目执行情况评审	151
5.3.1	项目周例会	151
5.3.2	管理层月会	153
5.3.3	里程碑评审	154
5.4	问题处理	157
第 6 章 第三方监控		161
6.1	启动项目 PPQA 并进行培训	163

6.2	制定 PPQA 计划	165
6.3	执行并维护 PPQA 计划	173
6.4	PPQA 工作有效性验证	175
6.4.1	PPQA 不符合问题验证	175
6.4.2	PPQA 工作情况验证	177
6.4.3	PPQA 工作的监督	180
6.5	PPQA 过程引用的规程	181
6.5.1	过程 PPQA 的评审规程	182
6.5.2	工作产品 PPQA 的审核规程	183
6.5.3	PPQA 不符合问题的处理规程	185
6.6	软件质量保证附表	188
	参考文献	197

第 1 章

导 论

1.1 软件项目

软件是计算机系统中与硬件相互依存的部分，它包括程序、数据及相关文档的完整集合。其中，程序是按事先设计的功能和性能要求执行的指令序列；数据是使程序能正常操纵信息的数据结构；文档是与程序开发、维护和使用有关的图文材料。

软件项目除了具备项目的基本特征之外，还有如下特点。

(1) 软件项目所创造的产品或服务是一种逻辑载体，是无形的，因此它具有抽象性。

(2) 软件项目的开发过程与硬件项目不同。软件项目是一种特殊的项目，只有逻辑的规模和运行的效果。

(3) 软件产品没有硬件的老化和机械磨损问题。然而，软件产品也存在无形磨损的问题。

(4) 软件项目的开发受到计算机系统的限制，对计算机系统有不同程度的依赖。

(5) 软件项目开发至今没有摆脱手工的开发模式，软件产品基本上是定制的。

(6) 软件项目本身是复杂的。它的复杂性源自应用领域实际问题的复杂性和应用软件技术的复杂性。

(7) 软件项目的开发成本比较高。软件项目开发需要投入大量的、复杂的、高强度的脑力劳动，还受到机构、体系和管理方式等问题的限制，因此成本比较高。

软件项目不同于其他项目，不仅是一个新领域，而且涉及的因素比较多，管理也较为复杂。目前，软件项目的开发和运作远远没有其他领域的项目规范，很多理论还不能适应所有的软件项目，经验在软件项目管理中仍起到很大的作用。软件项目是由相互作用的各个系统组成的，系统包括彼此相互作用的部分，软件项目中涉及的因素越多，彼此之间的相互作用就越大。如果项目中存在 n 个元素，则存在 $n(n-1)/2$ 种相互作用的关系。另外，变更在软件项目中也是常见的现象，如需求的变更、设计的变更、技术的变更、环境的变更等，所有这些都说明了软件项目管理的复杂性。

1.2 软件项目管理

1. 国内外软件产业发展概况

近年来，世界软件产业取得了突飞猛进的发展，到 2000 年已成为世界第一大产业，并继续以年均 15%~20% 的速度增长。2007 年全球软件与信息服务业市场规模达到 9 400 亿美元，同比增长 7.4%，2009 年世界软件及信息服务业市场将远远超过 1 万亿美元的规模。在全球软件市

场中，美国软件市场是发展最为成熟的市场，代表了世界软件市场与技术发展的方向。美国已成为世界软件产业发展的龙头，大多数先进的软件技术及产品源于美国。软件业已成为美国的第三大支柱产业。

1990年以后，我国软件产业的增长速度平均每年都在20%以上。工业和信息化部的最新统计显示，我国2007年软件与信息服务业规模为5800亿元，占到全球比重的8%。到2010年，我国软件与信息服务业国内市场规模有望突破1万亿元，其中，国产软件和信息服务业将占国内市场的65%以上。我国软件企业主要集中在北京、上海、深圳、沈阳、武汉、西安等大中城市和沿海地区。我国软件产业在国家各项政策的扶持下迅猛发展，2008年“软件收入规模前100家企业”（入围门槛为2.7亿元）的软件收入达到1127.5亿元，比2007年增长了13.7%。2008年百家企业共实现软件业务收入占我国软件业总收入的28%。在百家企业中，软件收入超过10亿元的企业有28家（2002年仅有17家）；超过50亿元的企业有6家（2002年仅有1家）。排名前10位的企业总收入比2007年增长23.2%，占百家企业收入的59.1%，比2007年提高了4.6个百分点。

2. 软件危机

20世纪60年代末期，软件危机问题日益严重。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。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”这个名词。据美国国家标准和技术研究院的一份报告显示，占世界软件销售额85%的是大型的专用软件，而其开发的失败率却高达70%。在这种情况下，软件工程学科应运而生，其目的就是借鉴传统工业的工程化思想和方法进行软件开发，以提高软件的质量和降低软件的成本。40多年来，软件工程硕果累累但收效甚微。以我国为例，近年来我国引进和开发的各种软件工具、开发环境等种类繁多，花费达几十亿元，但我国软件开发能力依然很差。软

件危机主要表现在以下 5 个方面：软件成本高，且开发成本超出预算；软件开发的进度难以控制，实际进度超出预订计划；软件质量难以保证，软件产品的质量靠不住；软件工作的统计、修正及维护比较困难，工作量较大，用户对软件系统不满意。

软件危机的原因有两种。一方面，与软件本身的特点有关。软件是一种逻辑载体，具有抽象性，人们可以把它记录在纸上、计算机内存、磁盘和光盘上，但却无法看到软件本身的形态，必须通过观察、分析、思考、判断，才能了解它的功能、性能等。软件是人类有史以来生产的复杂度最高的工业产品，软件涉及人类社会的各行各业、方方面面，软件开发常常涉及其他领域的专业知识，这对软件工程师提出了很高的要求。另一方面，由于软件行业是一个相对年轻的行业，缺乏成熟通用的规范、标准。然而，更为重要的是，软件开发项目本身是一个动态的复杂系统，因此，在软件开发中人们面临的不仅是技术问题，更重要的是管理问题。

3. 软件项目管理的产生

为了从根本上解决软件开发过程中的危机问题，专家们开始从软件的生产管理方面寻求解决办法。软件生产管理侧重的是软件企业在软件开发过程中对需求管理、计划安排、合同规范、项目跟踪、资源分配和质量要求等的管理方式。研究表明，大多数软件开发项目的失败，并不是软件开发技术方面的原因，而是由于不适当的管理造成的。遗憾的是，尽管人们对软件开发项目管理重要性的认识有所提高，但在软件开发项目管理方面的进步远比在设计方法学和实现方法学上的进步小，至今还没有一套管理软件开发的通用指导原则。回顾国内外软件产业的发展情况，软件产业经历了 3 个不同的发展时期，软件项目管理也随之产生并逐步发展起来。

20 世纪 60 年代中期以前，软件开发的特点是面向批处理、有限分布及自定义软件。这一阶段并不存在软件产业，在软件项目管理学领

域也是一片空白。20世纪60年代到70年代末期的十余年,开始出现软件作坊,软件开发和使用范围较第一阶段有很大的拓宽。但软件开发仍以个人单打独斗的作坊式开发为主,管理方面往往依靠开发者个人的素质。软件生产过程中尚未自觉地应用组织管理技术。到60年代末期,开始出现软件工程(Software Engineering)的概念,但软件工程的应用仍处于初级阶段。20世纪70年代中期以后的十余年,软件开发开始产业化,软件产业进入第一发展阶段——软件结构化生产阶段。以结构化分析与设计、结构化评审、结构化程序及结构化测试为特征的软件结构化生产开始出现并发展壮大,软件工程得到广泛发展和应用。

软件生产的第二阶段自20世纪80年代中期开始,以过程为中心,以个体软件过程(Personal Software Process, PSP)和群组软件过程(Total Software Process, TSP)的出现为标志。

软件生产第三阶段从1995年开始,以软件过程、面向对象和构件重用三项技术为基础。

第二阶段和第三阶段的软件生产更强调生产过程的标准化,ISO 9000质量保证体系及软件能力成熟度模型(Capability Maturity Model, CMM)得到广泛应用。CMMI是能力成熟度模型集成(Capability Maturity Model Integration),由CMU/SEI(美国卡内基梅隆大学软件工程研究所)于1986年开始研究,并于1991年正式推出了CMM1.0版,1993年推出CMM1.1版,2002年发布CMMI1.1版本,业界使用较多的是SEI于2006年8月推出的CMMI1.2版。为适应我国软件企业改进软件过程和提供软件开发规范水平的需求,工业和信息化部在CMMI的基础上,颁布了《软件过程能力评估模型》(SJ/T 11234—2001)和《软件能力成熟度模型》(SJ/T 11235—2001)两项电子行业标准。CMMI主要是为了指导软件开发组织,通过确定当前过程的成熟度和识别执行软件过程的薄弱环节,以及解决对软件质量和过程改进至关重要的几个问题,来形成对其过程的改进策略;通过关注并认真实施一组有

限的关键过程活动，稳步地改善组织的软件过程，使组织的软件过程能力持续增长。但 CMMI 只告诉了项目应该做什么，没有告诉应该怎么做，而项目管理理论又缺乏与软件项目的有效结合。因此，软件企业界及理论界迫切需要将项目管理理论和 CMMI 结合起来，做出一套软件项目计划与控制的模型。

软件项目管理是软件工程经济学的主要内容之一，包括软件度量、项目估计、项目计划、人员组织、进度控制等。软件工程经济学是 20 世纪 90 年代初由美国人 Boehm 等提出的，作为软件工程的重要代表，近年来受到广大研究者的高度重视并进行了理论和应用的深入研究。软件工程经济学旨在利用微观经济学来研究软件成本，并以此指导软件开发过程中和软件成本相关的需求管理、计划安排、资源分配、合同规范、项目跟踪及质量要求等一系列软件生产决策问题。

1.3 软件项目计划与控制模型体系

一种新技术的出现，必然有其发展的思想来源及历史背景。管理科学的发生和发展由来已久，经历了 19 世纪 20 年代的泰勒制，20 世纪 30 年代的行为科学学派（代表作为工业文明中人的问题），40 年代的数学管理派（代表作为生产组织与计划中的数学方法），50~60 年代的计算机管理学派，70 年代的系统工程学派，以及 80 年代和 90 年代的信息化和集成化。软件项目管理是项目管理学和先进软件开发技术与开发方法的有机结合，是研究如何有效地对软件开发项目进行管理，以便按照预订的成本、进度、质量等完成软件项目开发计划，实现预期的经济效益和社会效益的一门学科。

在软件项目管理领域中，目前还存在许多尚未解决、未能完全解决或解决不太理想的实际问题。例如，最初的估计模型计算出来的费用误差高达 20% 以上，而时间误差更高达 70% 以上。近几年，虽然大

量新技术（如人工神经网络、人工智能与专家系统及 Petri 网技术等）被应用到软件项目管理中，使得一些问题的估计精度大大提高，但尚存的问题依然很多。随着 CMMI 模型的引入，软件项目管理中的许多问题更加突出，并急需解决。本书在软件项目管理领域的若干应用中，主要研究软件项目计划基础、软件项目计划方法及软件项目监督与控制等方面的问题。

1. 软件项目的度量

软件项目的度量是研究软件项目管理所有问题的基础，包括以下一些内容。

（1）面向规模的度量。面向规模的度量采用的指标有每千行代码的错误数、每千行代码的缺陷数、每千行代码的成本、每千行代码的文档页数、每人月错误数、每人月代码千行及每页代码的成本等。

（2）面向功能的度量。功能不能直接测量，必须通过其他直接测量来导出。功能点（FP）是基于软件信息领域可计算的（直接）测量及软件复杂性的评估而导出的。FP 有 5 个信息域特征：用户输入数、用户输出数、用户查询数、文件数、外部接口数。FP 用来测量软件的以下属性：每个 FP 的错误数、每个 FP 的缺陷数、每个 FP 的成本、每个 FP 的文档页数，以及每人每月完成的 FP 数等。

进度、规模、工作量和故障都是项目的基本度量指标，它们形成一个稳定的指标集。

2. 软件项目计划、估计和监控

软件项目计划是软件项目的开始，而估计是制定软件项目计划及规划资源分配的基础。估计一个项目的常用技术既可以基于规模进行，也可以基于使用的过程进行。Putnam 和 Myers 推荐了 4 种估计软件规模的方法：模糊逻辑法、功能点法、标准构件法及修改法。基于过程的估计是将过程分解成相对较小的活动或任务，再估计完成每个任务

所需的工作量。

典型的估计模型是通过以前的软件项目中收集的数据进行回归分析而导出的。目前的估计模型有两大类。

(1) 面向源代码 (KDSI) 的估计模型。

1974 年, Wolverton 提出的 TRW Wolverton 模型; 1977 年, Herd 和其他人提出的 Doty 模型; 1977 年, Walston-Felix 提出的 IBM-FSD 模型; 1977 年, Black 和其他人提出的 Boeing 模型; 1979 年, Freiman-Park 提出的 RCA PRICES 模型; 1979 年, Putnam 和 Fitzsimmons 提出的 Putnam SLIM 模型; 1979 年, Carriere-Thibodeau 提出的 GRC 模型; 1981 年, Bailey-Basili 提出的 Bailey-Basili Meta 模型等。

(2) 面向功能点 (FP) 的估计模型。

$$E = -13.39 + 0.0545 FP \quad \text{Albrecht 和 Gaffney 模型}$$

$$E = 60.62 \times 7.728 \times 10^{-8} FP^3 \quad \text{Kemerer 模型}$$

$$E = 585.7 + 5.12 FP \quad \text{Masten, Barnet 和 Mellichamp 模型}$$

在众多的估计模型中, 最有名的是构造性成本模型 (Constructive Cost Model, COCOMO) 和 Putnam 模型。COCOMO 模型包括基本 COCOMO 模型、中级 COCOMO 模型和高级 COCOMO 模型。

软件成本估计模型估计出来的结果误差相当大, 如果估计的软件开发成本与实际成本相差不到 20%, 时间估计相差不到 70%, 就是不错的结果, 就足够对软件工程经济分析及决策提供很大帮助。目前, 国内外一些学者将神经网络和遗传规划等技术应用于软件成本的估计, 也取得了一定的成绩。例如, 1998 年, Anita Lee, Chun Hun Cheng, Jaydeep Balakrishnan 等人利用 ANN 技术研究了软件开发费用估计的问题, 提高了软件费用的评估精度; 1997 年, Dolad 利用遗传规划 (GP) 技术研究了 Albrecht 和 Mark 11 功能点的关系问题; 1998 年, Dolad 利用 GP 技术、神经网络技术及线性衰退方法研究了软件项目评估问