

软件成本估算

COCOMO II模型方法

总 论

(美) Barry W. Boehm 等著
李师贤 杜云梅 李卫华 孙 恒 等译



*Software Cost
Estimation
with COCOMO II*



机械工业出版社
China Machine Press

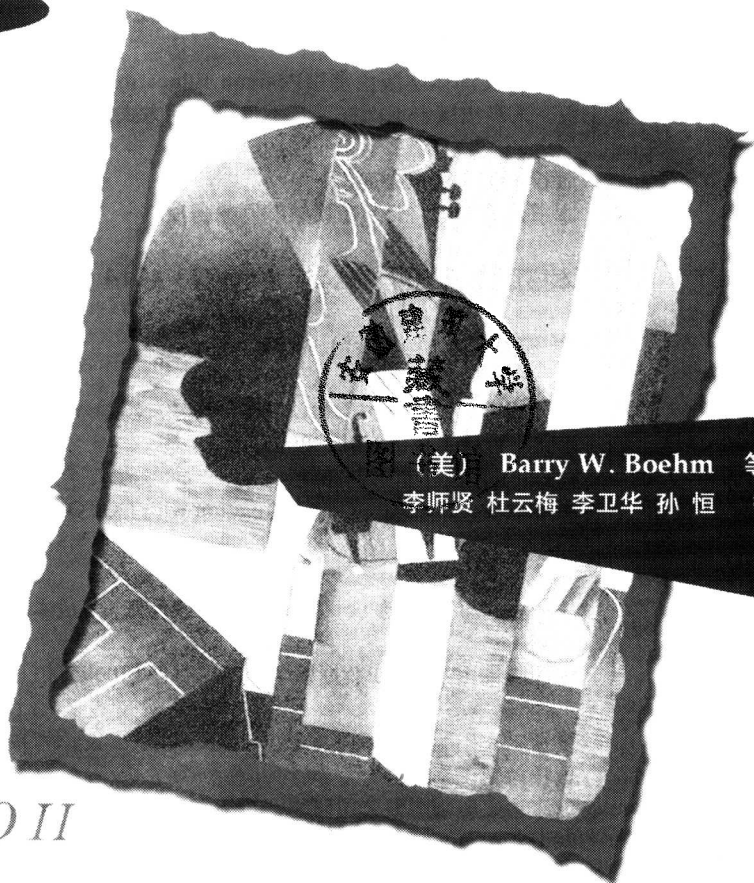
软件工程技术丛书

软件成本估算

COCOMO II模型方法

总论

*Software Cost
Estimation
with COCOMO II*



(美) Barry W. Boehm 等著
李师贤 杜云梅 李卫华 孙恒 等译



机械工业出版社
China Machine Press

COCOMO (CONstructive COst MOdel, 构造性成本模型) 是一种精确的、易于使用的基于模型的成本估算方法。而COCOMO II是对经典COCOMO模型的彻底更新, 反映了现代软件过程与构造方法。本书详细讲解如何利用COCOMO II进行软件开发项目管理中的有效估算。书中用大量实例说明COCOMO II的使用方法, 涉及软件成本估算的各个方面。适合软件开发人员、管理人员、在校学生参考。

Authorized translation from the English language edition entitled *Software Cost Estimation with COCOMO II* by Barry W. Boehm et al., published by Pearson Education, Inc, publishing as Prentice Hall PTR, Copyright © 2000 by Prentice Hall PTR.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanic, including photocopying, recording, or by any information storage retrieval system, without permission of Pearson Education, Inc.

CHINESE SIMPLIFIED language edition published by China Machine Press.

Copyright © 2005 by China Machine Press.

本书中文简体字版由美国Pearson Education培生教育出版集团授权机械工业出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

本书法律顾问 北京市展达律师事务所

本书版权登记号: 图字: 01-2003-4914

图书在版编目 (CIP) 数据

软件成本估算: COCOMO II模型方法/ (美) 勃姆 (Boehm, B. W.) 等著; 李师贤等译. —北京: 机械工业出版社, 2005.4

(软件工程技术丛书·总论)

书名原文: Software Cost Estimation with COCOMO II

ISBN 7-111-15777-X

I. 软… II. ① 勃… ② 李… III. 软件开发-成本计算-数学模型 IV. TP311.52

中国版本图书馆CIP数据核字 (2004) 第130695号

机械工业出版社 (北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 李云静

北京中兴印刷有限公司印刷·新华书店北京发行所发行

2005年4月第1版第1次印刷

787mm × 1092mm 1/16 · 27印张

印数: 0 001-4000册

定价: 55.00元 (附光盘)

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换
本社购书热线: (010) 68326294

译者序

有效的估算是软件开发项目管理中最具挑战性也是最重要的活动。只有使用科学的方法对软件项目的规模、工作量、进度与成本做出合理可靠的估算，才能实施良好的项目计划与控制。然而由于各种原因，目前的软件行业常常做不好项目估算，我们因此而遭受了很多本不应该承担的损害。

《软件成本估算：COCOMO II模型方法》和《软件工程经济学》是Barry W. Boehm博士所著的姐妹篇，介绍了世界上应用范围最为广泛的成本估算模型——COCOMO模型，这个模型由Barry W. Boehm博士开创并加以发展。《软件工程经济学》以经济学的观点来研究和分析软件的开发，以及探讨软件成本估算技术与成本估算模型。《软件工程经济学》是原版COCOMO模型（COCOMO 81）最完整的信息资源，提供了帮助我们理解与应用COCOMO II的大量素材；本书详细介绍了COCOMO II模型及其使用，它对COCOMO 81做了重要更新，以改进它对新的开发过程、方法、工具、技术和企业文化的适用性，并提高了估算的准确性。

我们衷心希望这两本书能帮助软件组织做出更好的软件估算，进而提高软件项目管理水平。

参与这两本书翻译及校对工作的有李师贤、杜云梅、李卫华、孙恒、唐培和、孙念、游梦良、蒋敏迪、崔茂林、李皓、李智、蔡树彬、徐晶、熊春玲等，由李师贤统一审校。另外，本书的翻译工作得到了广东省自然科学基金项目的资助（批准号：04009863）。

由于时间仓促，译者水平有限，译稿难免存在错误和疏漏，欢迎读者批评指正。

译者

2004年8月广州康乐园

原 序

我预计本书将在软件行业某一类人中非常畅销，但他们不会是编程能手、市场天才、技术专家或软件架构师，也不会是过程组成员、工具使用者、质量评估者或项目经理。它将在那些在大多数机构中都默默无闻的人群中最为流行：“专业”软件工程师——那些工作在幕后的人，他们关心的是如何在有利可图的商业环境下构造有用的产品。

COCOMO II是一个计划和执行软件项目的目标成本模型。它是管理软件项目或商业软件生产线的重要组成部分。成本模型提供了软件开发干系人之间交流商务决策的框架。COCOMO II支持合同协商、过程改进分析、工具购买、体系结构变更、组件开发或购买权衡，以及其他一些有可信估算基础的投资收益决策。本书阐述了如何使用COCOMO II进行这些分析以及所给出的结果有怎样的可信度。

Barry W. Boehm博士是软件业中常常被人提及的、最有影响的专家之一，他开创并发展了COCOMO II。早在1981年出版的经典著作《软件工程经济学》(Software Engineering Economics)^①中他就介绍了COCOMO，它为推断软件解决方案的成本和进度隐含关系提供了一个良好定义的、开放的“工程”基础，并在软件行业中取得了一定的突破。COCOMO迅速成为最流行的软件成本模型。

在过去几年中，COCOMO进行了一些调整与改进，以适应在软件生命周期、技术、组件、工具、表示法及企业文化方面的明显变化。Boehm博士和USC软件工程中心已投入了四年多的时间，进行COCOMO的改进，以使其能更好地适应现代软件工程的发展趋势。本书正是这些努力的成果。COCOMO II包括若干实地测试的改进，以拓展其能力，并提高对现代软件开发方法估算的准确性。它是许多软件经济学专家在很多软件领域和机构实验室的实际应用中长期使用COCOMO的结果。

COCOMO II包括两个底层信息模型。第一个是用于描述软件项目的框架，包括过程模型、文化、干系人、方法、工具、开发团队以及软件产品的规模或复杂性。第二个就是经验库，可以从历史案例估算出项目可能需要的相关资源（工作量与时间）。COCOMO II对COCOMO做了重要更新，以改进它对现代过程、方法、工具和技术的适用性。它也提供了更大、更恰当的现代案例数据库，并改进了模型的适应性，从而可以在更多的领域及其项目环境中进行优化。本书收集了这些工作的成果，文笔流畅，适合许多读者，如COCOMO专家、成本模型新手、COCOMO用户和COCOMO工具开发者阅读。本书通过详细的例子和富有哲理的评论给出了多个观点。

在过去的十多年中，盛行的浮躁之风已影响了软件行业。对各种过程改进、新方法和新技

^① 《软件工程经济学》一书已由机械工业出版社引进出版。

术存有偏见的介绍，是一个行业带有盲目性和不负责任的征兆。随着产品进入市场的时间越来越短，先简化开发以后加以完善的方法被采用得过于频繁。本书介绍的COCOMO II产品是人们急于进入市场思想的一个极好反例。COCOMO II基于成熟的数学、优秀的行业专业知识和广泛的现代项目经验，是一个严密的工程模型。作为USC会员，Rational软件公司公开支持并投资于COCOMO II的开发。软件成本建模，尤其是COCOMO II与现代迭代过程、体系结构先行的生命周期（architecture-first life cycles）和基于UML的分析与设计方法等的结合，已成为在软件开发项目中取得成功所必需的显著优势。

20年前，我有幸参加了UCLA计算机科学系首开的软件工程经济学研究生课程，由Barry W. Boehm执教。包括我在内的大多数学员，对如何成功提交软件产品的理解极其有限。我们多半也体会不到大多数软件开发项目的背景和决策主要是受商业因素而非技术因素的制约。这门课程对我的项目管理观点，甚至整个软件行业都有深远的影响。虽然原始COCOMO的大部分材料仍然适合于今天的软件管理学科，但这本新书包括了根据过去20年所得经验进行的重要更新。软件项目的决策比过去任何时候都更受商业因素的影响，COCOMO II模型为现代软件项目管理制定了一个新的标准，为软件行业中下一代的过程、方法和工具改进的评价提供了一个经济学的框架。

Walker Royce

Rational软件公司副总裁

前言

“我们都快成软件公司了。”越来越多的机构——诸如财务、交通、航空、电子和制造公司——如是说。竞争的优势日益依赖于智能的、可裁剪产品和服务软件的开发，以及比竞争对手更快捷地开发和调整这些产品与服务的能力。这些趋势突出表明了需要准确地估算软件成本和进度，并支持软件决策的权衡、风险、灵敏性和商业实例分析的强大能力。

同时，新一代软件过程和产品正在改变机构开发软件的方式。这些新方法——不断发展的、风险驱动的和协作的软件过程，第四代语言 and 应用程序生成器，商业成品（COTS）和复用驱动的软件方法，快速跟踪软件开发方法，软件过程成熟度的倡导——带来了提高软件质量和降低软件成本、风险和周期时间的重大收益。

然而，现有的一些软件成本模型虽然初步解决了这些问题的各个方面，但至今仍没有一个能很好地匹配这些新方法的软件成本和进度估算模型。这使机构很难使用这些新方法进行有效的的项目计划、分析和控制。

正是基于这些因素，本书作者提出了构造性成本模型（CONstructive COst MODEL, COCOMO）的新版本，用于软件工作量、成本和进度估算。原始的COCOMO 81[Boehm 1981]以及后来的专用Ada COCOMO[Boehm and Royce 1989]比较好地匹配了它们所建模的一类软件项目：主要是由客户定制、按说明书构造的软件[Miyazaki and Mori 1985; Goudy 1987]。虽然Ada COCOMO增加了估算增量软件开发的成本和进度的能力，但在面对由螺旋或进化开发模型创建的软件或主要通过商业成品（COTS）应用组装能力开发的软件时，COCOMO 81成本估算遇到了越来越多的困难。

软件估算的实践状况

许多商业软件估算模型因为对用户增加了价值而通过了市场的考验。也有相当一些机构在软件项目计划、估算和控制方面做得不错。然而，大量的机构却做不好：1995年Standish Group研究发现，超过53%的软件项目在预算和进度两方面都超支和逾期50%以上[Standish 1995]。

对一些近来收集的软件失败实例进行研究，可获知这些逾期和超支的详细原因。表P1概述六个这样的软件项目情况，前三个来自[Flowers 1996]，后三个来自[Glass 1998]。

模型冲突和MBASE

我们分析了表P1中的项目，发现它们的许多问题来自模型冲突（model clash）[Boehm-Port 1999b]。模型冲突源于定义和管理项目的基本模型间的不兼容性，包括产品模型（需求、体系结构、设计等）、过程模型（瀑布、螺旋、成熟度模型等）、属性模型（成本、进度、性能等）和成功模型[商业实例、干系人双赢、IKIWISI（I'll know it when I see it）等]。

表P1 软件超支案例研究

项 目	最初; 最终估算		完成状态
	成本(百万美元)	进度(月)	
PROMS (Royalty collection)	12; 21+	22; 46	取消, 28个月
London Ambulance	1.5; 6+	7; 17+	取消, 17个月
London Stock Exchange	60 ~ 75; 150	19; 70	取消, 36个月
Confirm(Travel reservations)	56; 160+	45; 60+	取消, 48个月
FAA Advanced Automation System	3700;7000+	48; 96	取消, 70个月
Master Net(Banking)	22; 80+	9; 48+	取消, 48个月

表P1的项目中大量的模型冲突导致了它们的失败。有几种模型冲突涉及到成本和进度属性模型。例如在London Ambulance项目中, 顾问报告指出: 如果能找到(事实上找不到)成品方案, 那么最佳成本为2 250 000美元, 而项目却将它误解为成本基线。有人给出1 500 000美元的低价, 显然是基于非常不彻底的、从零开始构造的产品模型。

在USC软件工程中心(USC Center for Software Engineering, USC-CSE), 我们正在研究一种命名为基于模型的(系统)体系结构和软件工程的方法(MBASE), 以期为项目提供一组一致的且互相支持的产品、过程、属性和成功模型[Boehm-Port 1999a; 也见<http://sunset.usc.edu/research/MBASE>]。它用于干系人成功模型作为主要的项目驱动因子, 用螺旋模型的干系人双赢变量为该系统确定一个兼容的产品、过程、属性和成功模型集。它也将软件工程与系统工程结合起来, 为过渡到新的集成化能力成熟度模型(CMM I)提供强有力的框架。

MBASE和COCOMO II

我们对MBASE的研究涉及成功模型(双赢、期望管理); 过程模型(双赢螺旋模型、锚点、需求协商); 产品模型(领域模型、软件体系结构)和属性模型(基本的COCOMO II和它的扩展)。并行研究这些组件模型有助于加强它们的关系, 使它们建立在软件实践的基础上。例如, 我们的锚点过程研究始于定义一组公用生命周期里程碑的工作量, 它是COCOMO II成本和进度估算的基础。与Rational公司合作时, 我们把MBASE与Rational统一过程(RUP)的阶段和里程碑集成在一起, 为COCOMO II提供了MBASE/RUP阶段和活动分布估算量(见附录A)。

还有, 我们试图把MBASE应用到我们自己的项目中。这样, 在确定COCOMO II范围时, 我们利用了MBASE原则来识别干系人、收集他们使用软件估算模型的目标, 并确定他们改进现有COCOMO 81性能的主要需求。因此, 本书第1章从总结COCOMO II的用户目标开始, 接着介绍一组根据对COCOMO 81所希望做的改进而得到的COCOMO II模型目标, 最后介绍为达到这些目标而建立的一组COCOMO II开发和改进策略。

干系人的目标之一是避免把COCOMO II建立在单个过程模型基础上, 比如COCOMO 81中所假定的瀑布模型。因而, 我们为瀑布模型、MBASE/RUP和增量开发都设计了COCOMO II的解释程序。

本书各章的内容

第1章给出了COCOMO II总的脉络和框架,包括未来软件实践的模型,以及对未来主要软件实践可进行裁剪的三阶段模型的最终选择过程。第2章给出了COCOMO II量化、估算公式、成本驱动因子与比例因子定义和等级量表,以及在特殊情况下解释这些定义的指导原则。

第2章的最后一节给出了如何用COCOMO II对第1章所建立的一些目标做出快速的手工分析:进行投资决策、权衡与风险分析、针对项目实践裁剪模型。

第3章用两个代表性的例子:事务处理系统和航空雷达系统,说明如何用USC COCOMO II工具去完成更广泛的成本估算和比较分析。

第4章概述了COCOMO II贝叶斯校准过程及其结果,并为机构产生自己的校准模型提供了指导原则。

第5章描述了对核心COCOMO II模型的一些新扩展。仍然把应用组装模型看作是一个正在形成的扩展,因为它的校准和计算规则仍未成熟。其他模型扩展解决了下列问题:估算软件COTS集成的成本(COCOTS),进度和工作量的阶段分布(COPSEMO),快速应用开发的工作量和进度调整(CORADMO),根据已提交的缺陷率确定质量(COQUALMO),以及应用软件生产率策略的效果(COPROMO)。

第6章重点介绍软件发展的趋势,以及它们可能对软件成本估算和COCOMO II造成的影响。

六个附录提供了:A) COCOMO II定义、假设和阶段/活动分布估算;B) 增量开发估算模型;C) 用于COCOMO II和新扩展的数据收集表单,以便更好地校准模型以适合机构的需求;D) USC-CSE COCOMO II的信息和其他会员章程;E) 关于USC COCOMO II工具的软件参考手册和COCOMO II数据收集的表单和指南;F) 介绍随书附带光盘的内容和使用。

该光盘提供了USC-CSE开发的、最新版的COCOMO II估算软件包,以及三个供演示的商用COCOMO II软件包。我们也把书中使用的例子作为文件放入光盘中,这样你也可以生成我们用这个软件包得到的结果。这一点很重要,因为它告诉你如何有效地使用软件包和COCOMO II模型,以便研究估算,并比较和完成各种成本分析。

该光盘也提供了一些附加材料,包含关于COCOMO II软件包的简介、报告、手册和详细的细节,使你能了解它的历史和由来。我们的目标是每隔两年当出版本书的更新版本时也更新该光盘。

不同读者感兴趣的章节

如果你是软件经理、分析员或开发者,希望进行和使用软件估算,则请阅读第1~3章以及附录E,并相应地使用USC COCOMO II工具。

如果你是机构的软件估算专业人员,则应研读全书。

如果你是软件度量和模型研究者,则请阅读第1、2、4、5、6章和附录A、附录B、附录C。

如果你只想对COCOMO II和它的使用有一般的理解, 则请阅读第1~3章。

与1981年版《软件工程经济学》一书的关联

本书替换了1981年版的《软件工程经济学》[Boehm 1981]中30%的内容, 这里逐章介绍1981年的版本与这本COCOMO II书内容上的联系——注意, 下面列出的各章是指原1981年书的章节:

第1章和第2章, 作为驱动软件工程发展的人类经济学方法仍是适时和有用的。

第3章, GOALS方法已演化为MBASE和干系人双赢方法。这一章和附录B的材料是很好的背景和例子, 但不是操作指导。

第4章, 已构造出COCOMO II用于产生成本和进度估算, 其与该章中的瀑布模型里程碑一致。工作分解结构框架仍然可以采用。

第5~9章, 基本和中等COCOMO 81的材料已由COCOMO II应用组装、早期设计和后体系结构模型系列所代替, 第6章和第7章关于瀑布模型阶段和活动分布的那些部分仍然与本书相关。

第10~20章, 作为使用COCOMO II估算框架的软件微观经济学的材料仍然有效。

第21章和第22章, “七步方法”和软件估算技术的比较仍然有效。COCOMO II一书通过实例为使用COCOMO II进行各种估算、比较分析和生命周期估算补充了相关方面的材料(在第2章和第3章)。

第23章, 该内容专用于COCOMO 81。所包括的阶段敏感性成本驱动因子是COCOMO II的间接目标之一。

第24~27章, 对部分成本驱动因子经验的讨论已经过时了(如周转时间、工具使用、现代编程规范、进度延长的影响), 但其他仍然有用。我们计划在未来的版本中更新并加进新的成本驱动因子。

第28章, 现在人员连续性和文档化是COCOMO II成本驱动因子, 它们在最近的COCOMO II数据分析中有重大影响。需求易变性现在通过损耗参数REVL作为规模修改量处理, 本章的其他部分很大程度上专用于COCOMO 81。

第29章, 该章的大部分内容由COCOMO II一书的第4章代替。

第30章, 维护数据的趋势仍然与本书相关, COCOMO II的维护模型已经升级, 包括了软件理解和人员不熟悉因子。

第31章, 大部分估算变换、计算机、安装、培训和文档化成本的内容已经过时, 但31.8节的商业案例仍然非常有用。

第32章, 计划和控制技术仍然有效, 目前挣值技术更频繁地用于软件。尚未把握的主要趋势是其他进展跟踪度量的使用, 如在《Practical Software Measurement》手册中收集的那些材料[McGarry et al. 1998]。

第33章, COTS的覆盖范围和超高级语言在1981年版的书中进行了较好的介绍, 但现在我们了解得更多了, 详见本书5.4节中的COCOTS讨论。对产品、平台、人员生产率因子的讨论仍然

有用。COCOMO II还在它的比例因子中有更多的过程生产率选项，且有更强的复用处理。

附录A，由本书的附录C代替。

附录B，能作为很好的背景，如第3章所讨论的那样。

概要

1981年版《软件工程经济学》的大部分内容仍有用，也有意义，过时的或被本书替代的部分是第5~9章、第23章、第28~29章、第31章（31.8节除外），以及附录A和附录B。

作者简介

Barry W. Boehm获哈佛大学学士学位，获洛杉矶加州大学博士学位，并获马萨诸塞州大学名誉博士学位。他目前是南加州大学计算机科学系软件工程TRW教授和软件工程中心主任。

Chris Abts在佐治亚理工学院获学士学位，在南加州大学获工业和系统工程硕士学位，并有15年软件工程师和顾问的经验。

A. Winsor Brown在Rensselaer技术学院获工程科学学士学位，在加州理工学院获电气工程硕士学位。他目前是南加州大学软件工程中心的副主任，有超过30年的顾问和软件工程师经验。

Sunita Chulani获印度孟买大学学士学位，在南加州大学获计算机科学博士学位。她目前供职于加州圣何塞的IBM Almaden研究中心。

Brad Clark在佛罗里达大学获学士学位，在南加州大学获计算机科学博士学位。他目前是卡内基-梅隆大学软件工程研究所的访问学者，是华盛顿特区附近的软件度量咨询公司的负责人。

Ellis Horowitz在布鲁克林学院获学士学位，在麦迪逊的威斯康星大学获博士学位。他目前是南加州大学工程学院的远程教育和信息技术主任，计算机科学系教授和荣誉退休主席。

Ray Madachy在代顿大学获机械工程学士学位，在南加州大学获工业和系统工程博士学位。他是C-bridge因特网解决方案的管理负责人（之前他是Litton向导和控制系统软件工程过程组的经理），是南加州大学计算机科学系助理教授。

Don Reifer在新泽西理工学院获学士学位，在南加州大学获得硕士学位。他是加州托兰斯城的Reifer顾问公司的创始人和总裁，并且有30年软件工程师、程序经理和管理顾问的经验。

Bert Steece在南加州大学获学士学位和博士学位。他目前是南加州大学马歇尔商学院的信息和运营管理系统专业教授和研究院的副院长。

目 录

译者序

原序

前言

作者简介

第1章 COCOMO II介绍.....1

1.1 COCOMO II用户目标1

1.2 COCOMO II模型目标2

1.3 COCOMO II开发和发展策略3

1.4 未来软件实践市场模型4

1.4.1 中间部分5

1.4.2 1999年模型评估5

1.5 最终的COCOMO II模型系列6

1.5.1 针对不同软件市场部分的COCOMO
II模型6

1.5.2 根据过程策略裁剪COCOMO II估算
模型7

第2章 COCOMO II模型定义.....9

2.1 引言.....9

2.1.1 概述9

2.1.2 标称进度估算公式9

2.2 规模估算10

2.2.1 源代码行 (SLOC) 计算11

2.2.2 未调整功能点 (UFP) 计算11

2.2.3 UFP与SLOC关联14

2.2.4 累加新的、改编的和复用的代码15

2.2.5 需求演进和易变性 (REVL)19

2.2.6 自动转换的代码20

2.2.7 计算软件维护的规模21

2.3 工作量估算22

2.3.1 比例因子23

2.3.2 工作量乘数30

2.3.3 多模块的工作量估算41

2.4 进度估算42

2.5 软件维护42

2.6 应用COCOMO II进行软件决策43

2.6.1 投资决策和商业案例分析43

2.6.2 设定项目预算和进度44

2.6.3 权衡分析45

2.6.4 成本风险管理45

2.6.5 开发与复用决策46

2.6.6 遗留软件逐步淘汰决策47

2.6.7 软件复用和产品线决策47

2.6.8 过程改进决策49

2.6.9 决策分析总结50

2.7 COCOMO II模型总结和版本50

2.7.1 模型公式、表和驱动因子等级量
表51

2.7.2 COCOMO II版本参数值54

2.7.3 源代码逻辑行计数规则56

2.7.4 COCOMO模型比较61

第3章 应用实例.....63

3.1 引言63

3.2 事务处理系统 (TPS) 概述63

3.2.1 事务处理系统描述63

3.2.2 事务处理系统的软件功能64

3.2.3 事务处理系统的软件开发机构65

3.2.4 事务处理系统的软件开发估算66

3.2.5 划定风险的边界78

3.2.6 执行权衡研究81

3.2.7 评估生命周期成本85

3.3 机载雷达系统 (ARS) 概述87

3.3.1 ARS描述88

3.3.2 原型演示 (起始阶段)89

3.3.3 实验模型系统 (细化阶段)90

3.3.4 完全开发——顶层估算94

3.3.5 完全开发——详细的组件估算	98	5.3.2 与COCOMO II的关系	172
3.3.6 增量开发实例	102	5.3.3 模型概况	172
第4章 校准	107	5.3.4 模型细节	174
4.1 贝叶斯校准和COCOMO II建模方法学	107	5.3.5 处理的范围和生命周期	182
4.1.1 贝叶斯校准	107	5.3.6 电子表格模型实现	182
4.1.2 COCOMO II建模方法学	108	5.3.7 应用实例	186
4.2 讲述的主题	110	5.3.8 结论	187
4.3 COCOMO II模型的数据收集方法	110	5.3.9 未来工作	187
4.3.1 获得一致数据	110	5.4 COCOTS: COTS 集成估算	188
4.3.2 Rosetta Stone	110	5.4.1 背景和基本原理	188
4.4 模型建造	115	5.4.2 与COCOMO II的关系	190
4.4.1 统计的建模过程	116	5.4.3 模型概况	190
4.4.2 观测数据的分析	117	5.4.4 目前已处理的范围和生命周期	191
4.5 COCOMO II校准	120	5.4.5 成本来源	192
4.5.1 COCOMO II.1997	120	5.4.6 四个子模型	193
4.5.2 COCOMO II.2000	125	5.4.7 评估	193
4.6 针对特定机构裁剪COCOMO II模型	135	5.4.8 裁剪	196
4.6.1 用现有项目数据校准模型	135	5.4.9 连接代码	197
4.6.2 合并或消除冗余参数	141	5.4.10 系统易变性	200
4.6.3 在模型中增加不明显但重要的成本驱动因子	141	5.4.11 总的COTS集成工作量	201
4.7 COCOMO II数据总结	141	5.4.12 结论	201
4.8 结论	149	5.5 COQUALMO: 质量估算	201
第5章 新扩展	151	5.5.1 引言	202
5.1 应用组装: 应用点模型	151	5.5.2 背景模型	202
5.1.1 对象点数据和实验	151	5.5.3 软件缺陷引入 (DI) 模型	203
5.1.2 应用点估算过程	152	5.5.4 软件缺陷消除模型	207
5.1.3 应用点估算的准确性和成熟度	153	5.5.5 COQUALMO与COCOMO II的集成	212
5.2 COPSEMO: 阶段进度与工作量估算	155	5.5.6 结论和进行中的研究	213
5.2.1 背景	155	5.6 COPROMO: 生产率估算	213
5.2.2 模型概况	160	5.6.1 背景和基本原理	214
5.2.3 模型实现	162	5.6.2 与COCOMO II的关系	215
5.2.4 应用示例	163	5.6.3 模型概况	215
5.2.5 动态COCOMO	163	5.6.4 目前包括的范围和生命周期	215
5.3 CORADMO: 快速应用开发估算	170	5.6.5 模型细节	216
5.3.1 背景和基本原理	170	5.6.6 电子表格模型概况	219
		5.6.7 使用实例	219

5.6.8 COPROMO 0.3文档	223	6.5.2 处理项目实施中的变更	237
5.6.9 结论和未来工作	225	6.5.3 处理COCOMO II模型所需要的 变更	237
5.7 专家COCOMO: 风险评估	227	6.5.4 主动的机构变更管理	238
5.7.1 引言和背景	227	附录A COCOMO II: 假设条件和阶段 /活动分布	241
5.7.2 风险描述	227	附录B COCOMO II: 估算增量开发	263
5.7.3 风险分类学和规则库	228	附录C COCOMO 套件: 数据收集表单 和指南	271
5.7.4 风险量化	229	附录D COCOMO II和USC-CSE会员 章程	307
5.7.5 输入异常	230	附录E USC COCOMO II. 2000软件 参考手册	309
5.7.6 实现	230	附录F 附赠光盘的内容	381
5.7.7 当前状态和进一步的参考	231	词汇表	385
第6章 未来发展的趋势	233	参考文献	393
6.1 在软件生产率与估算准确性方面的 趋势	233	索引	399
6.2 对应用领域增加理解带来的影响	233		
6.3 创新与变化的影响	234		
6.4 处理变化: COCOMO II	235		
6.5 处理变化: COCOMO II与机构	236		
6.5.1 处理项目定义中的变更	236		

第 1 章

COCOMO II 介绍

1.1 COCOMO II 用户目标

COCOMO II 是帮助你推理必须做出的、软件决策中成本和进度关系的模型。近十八年来，在回复原COCOMO ([Boehm, 1981], 本书称之为COCOMO 81) 用户电话和电子邮件咨询的过程中，我们了解到了多种不同类型的COCOMO 81用户及其使用情况。最出人预料的应该是美国税务局审计员的咨询电话，在快速声明不是审计我们的纳税申报后，他们询问如何使用COCOMO 81以确认某些软件税的销账声明。

这里列出了我们确定的、用户将来可能要用COCOMO II进行决策的主要情况：

1) 涉及软件开发工作量的投资或其他财务决策。涉及软件开发的商业案例或投资回报分析需要软件开发成本的估算或者软件生命周期各阶段的支出概况。

2) 设定预算和进度作为项目计划和控制的基础。例如，在软件项目的早期、中期和晚期阶段应分配多少人？要花费多少工作量、成本和时间才能达到主要的项目里程碑？

1

3) 在软件成本、进度、功能、性能或质量因素之间进行决策或权衡。例如，我们在多大程度上可以说“质量不在考虑之列的”？难道构造与生命攸关的软件会比构造库存控制软件更低廉吗？

4) 做出软件成本和进度风险管理决策。影响项目成本和进度的许多因素难免会有不确定性。项目中要开发和复用的软件数就是不确定性的一个例子。如果成本模型考虑了这些不确定性的来源就有助于进行敏感性和风险分析。

5) 确定软件系统的哪些部分需要开发、复用、租借或购买。例如，好的成本模型有助于了解什么时候构造一个组件要比重新修改一个现有组件更划算。

6) 做出遗留软件的盘存决策：哪些部分要修改、逐步淘汰、外包等。例如，这样的模型有助于制定并定期更新一个多年的项目计划，以指导人们在遗留系统每次升级时应该优先开发哪些性能。

7) 通过复用、工具、过程成熟度、外包等相结合的投资策略改进机构的软件能力。对此，该模型有助于制定并定期地更新一个多年的技术投资计划。例如，COCOMO II既有能力估算生产可复用软件所需的额外投资，又能估算从其复用中节省的投资。

8) 确定如何实现过程改进策略，如SEI能力成熟度模型[Paulk et al.1995]。对第二级，如何把软件成本模型如COCOMO II，与计算规模、估算、计划和跟踪软件项目的成本、进度和工作量的其他方法相结合？对第三级，根据对应用、平台、语言和工具使用经验这些成本驱动因子的改进，如何评估培训的收益？对第四级，像COCOMO II这样的框架如何有助于设置有效的

定量软件管理过程？对第五级，如何改进这个框架以适应新的实践，如产品线（product lines）、快速应用程序开发和商业成品（commercial-off-the-shelf, COTS）软件集成？

至少，软件估算模型远非一种简单的输入成本驱动因子等级并得到预算和进度估算的工具。为了有助于实现这些额外的决策支持能力，我们确定了COCOMO II的一些目标和策略，详见下面的讨论。

2

1.2 COCOMO II模型目标

正如前言中所讨论的，我们将把COCOMO II模型的目标与多年以来用户对COCOMO 81中的优点和需要改进的能力反馈联系起来：

1) 为当前和未来可能的软件项目提供准确的成本和进度估算。COCOMO 81建立在20世纪70年代“瀑布模型”过程框架之上（依次经过需求、设计、编码和测试），而许多机构正朝并发、迭代、增量和循环过程相结合的方向发展。COCOMO II提供一个框架，可为特定的过程裁剪相应的模型版本，包括瀑布过程。

2) 使机构能方便地重新校准、裁剪或扩展COCOMO II，从而更好地适应它们的特殊情况。“重新校准”能使COCOMO II更好地与机构所收集的项目数据相适应，对项目终点、产品定义、劳动力定义和组织实践有自己特定的解释。“裁剪”和“扩展”使我们能改编COCOMO II以适应机构的产品、过程和人员的实际差异。COCOMO 81提供了“重新校准”功能，但“裁剪”只在瀑布模型的假设下才有效。

3) 提供细致的、容易理解的输入、输出和假设定义。没有这些定义，当利用模型估算时，不同的人就会有不同的解释，这很容易造成“百万美元的错误”。在这方面，COCOMO 81总的来说是令人满意的。COCOMO II为成本驱动因子的等级量表（rating scale）提供一些额外的辅助定义。

4) 提供一个构造性模型，其对成本和进度的估算可以帮助人们更好地理解所估算项目的本质，这是COCOMO 81的主要目标，也仍然是COCOMO II的主要目标。这个方向的主要改进是加入更多的成本驱动因子，以帮助人们理解复用和规模经济方面对项目决策的影响。

5) 提供一个标准化的模型，能为有效的软件开发或维护分配所需的资源；突出不切实际的成本和进度目标；以及识别能改进当前或未来项目的管理可控因素。按数据进行校准的软件成本模型一般在标准化上做得较好，因为管理不好的项目很少收集用于校准的数据。和前面讨论的目标一样，COCOMO II能识别出更多的在复用和规模经济方面的管理可控因素。

3

6) 提供一个发展的模型，能添加新的能力以满足新的需求（如COTS集成、快速应用程序开发），并保持与发展的软件实践相关联。设计COCOMO 81时没有考虑模型的发展问题。COCOMO II是个持续的项目，既持续细化主要的项目成本-进度估算模型，又提供不断的扩展（主要在第5章讨论）。