

<<软件估算>>

图书基本信息

书名：<<软件估算>>

13位ISBN编号：9787121052958

10位ISBN编号：7121052954

出版时间：2007-12

出版时间：电子工业出版社

作者：麦克康内尔

页数：324

字数：420000

译者：宋锐

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<软件估算>>

### 内容概要

在《软件估算——“黑匣子”揭秘》一书中，著名的软件开发书籍的作者Steve McConnell揭开了围绕在软件估算周围的层层迷雾。

作者在深入浅出地介绍了与软件估算有关的主要概念之后，深入、全面地介绍了与软件估算有关的多种估算方法。

本书的主要内容包括：估算与计划和项目控制，以及估算与目标和承诺之间的关系；不确定性锥与估算中的误差来源以及影响估算的各种因素；先计数、再计算，无法可想时才依靠判断的基本估算原则；用于估算软件项目的三个重要部分——规模、工作量和进度估算的基本方法；与规模、工作量和进度估算有关的特殊问题；估算的概率论观点以及如何采用适当的方式来表达估算结果中的不确定性；如何进行与估算有关的沟通，从而使技术人员和非技术人员达成共识。

本书主要面向软件开发项目中要进行估算的开发人员和技术管理人员。但本书所涉及的与软件估算有关的背景知识，以及有关估算谈判和表达方式的讨论，对于非技术人员出身的主管和项目的其他有关人员同样大有裨益。

## &lt;&lt;软件估算&gt;&gt;

## 作者简介

Steve McConnell是Construx Software公司的首席软件工程师，负责监督该公司的软件工程实践。Steve是软件工程知识体（SWEBOK，Software Engineering Body of Knowledge）项目的构造知识领域（Construction Knowledge Area）的负责人。

Steve在微软、波音以及西雅图地区的其他公司也从事过软件项目方面的工作。

他是Construx Estimate和SPC Estimate Professional项目开发的负责人，后一个项目获得过Software Development杂志的生产力大奖（Productivity Award）。

Steve是Rapid Development（1996）、Software Project Survival Guide（1998）、Professional Software Development（2004）和Code Complete, Second Edition（2004，《代码大全，第2版》）等书的作者。他的著作曾两次获得过Software Development杂志的年度卓越软件开发书籍震撼大奖（Jolt Product Excellence Award）。

Steve还是SPC Estimate Professional的开发负责人，该产品获得了软件开发生产力大奖（Software Development Productivity Award）。

1998年，Software Development杂志的读者们把Steve选为软件行业最有影响力的三个人之一，另外两人分别是Bill Gates（微软公司的创办人）和Linus Torvalds（Linux的作者）。

Steve在惠特曼学院获得了学士学位，在西雅图大学获得了软件工程硕士学位。

他现在居住在华盛顿州的贝尔维尤市。

如果想对本书提出任何评论或疑问，请通过[steve.mcconnell@construc.com](mailto:steve.mcconnell@construc.com)或通过[www.stevemcconnell.com](http://www.stevemcconnell.com)网站联系他。

## &lt;&lt;软件估算&gt;&gt;

## 书籍目录

第一部分 估算的关键概念 第1章 “估算”的含义 第2章 你的估算水平如何 第3章 准确估算的价值 第4章 估算误差的来源 第5章 影响估算的因素 第二部分 基本估算方法 第6章 估算方法概述 第7章 计数、计算和判断 第8章 估算校准和历史数据 第9章 专家的个人判断 第10章 分解和重组 第11章 类比估算 第12章 基于代理的估算 第13章 专家小组判断法 第14章 软件估算工具 第15章 使用多种估算方法 第16章 获得良好估算的软件项目中的估算流程 第17章 标准化估算规程 第三部分 特定的估算挑战 第18章 规模估算中的特殊问题 第19章 工作量估算中的特殊问题 第20章 进度估算中的特殊问题 第21章 计划参数的估算 第22章 估算结果的表达方式 第23章 政治、谈判和解决问题 附录A 估算合理性检查 附录B 第2章“你的估算水平如何？”测验的答案 附录C 软件估算提示 参考文献 索引

## &lt;&lt;软件估算&gt;&gt;

## 媒体关注与评论

中文版引言 对于软件项目管理，“坊间”流传着一个经典的“六拍”黑色幽默，如图1所示。

在此我略做演绎：在项目开始之前，你总是先“拍脑袋”得出进度和成本的承诺；在开工大会上领导“拍拍你肩膀”，是那样的语重心长、充满期待；而小酒刚下肚、春风正得意时，你不由得“拍胸脯”以表决心和能力；但在项目进展过程中遇到这样、那样的困难时，客户和业主不能不“拍桌子”了；这时充满悔意的你，只能“拍大腿”以示自责；而到了一切都腹水难收时，恐怕也只能“拍屁股”另谋高就了；不过却也总能够重新找个地方东山再起，再“拍脑袋”去了。

如上所示的六拍“怪圈子”已经将尚显稚嫩的软件行业折磨得心焦力竭，如何才能冲破这个行业的“紧箍咒”呢？

? 从业人员敬业精神不高？

非也！

虽然拍屁股走人后总能够再次上岗，一则因为软件项目成功率本来就不高，二则因为大家实际上“已尽人事”了。

? 项目经理们不思进取？

非也！

每当遇到项目超期、成本失控时，项目经理大凡都会真心地“拍大腿”，都不愿意自己的苦心最终收获苦果。

? 项目过程监控管理不足？

不全是！

没有过程数据、没有中间管理，哪能会有人半道“拍桌子”？

另则，大多失败的项目也并非源于大家不够努力。

? 项目经理经验技能不足？

不全是！

“没有金刚钻，不揽瓷器活”，敢“拍胸脯”的项目经理，绝不会都是纸上谈兵的赵括吧！

? 各种所需资源投入不足？

也不全是！

“拍拍你肩膀”的人恐怕也绝不会让你一个人独自承担所有的问题吧！

那么问题在哪里？

显然最为核心的问题就出在“拍脑袋”上，在软件项目管理中缺乏有效的估算方法与过程，一直以来是业界的心头之痛！

那又是什么原因导致这个大家都意识到的问题，长期以来却处于“无解”的状态呢？

也许“规划规划全是糊话，计划计划全是鬼话”这一说法从另一个侧面道出了从业人员的无奈，计划情况与实际情况的严重偏离导致整个行业对“软件估算”产生了不信任感，高级管理人员不敢采用，项目管理者也不愿意把时间花在软件估算这个“黑匣子”上。

幸运的是，Steve McConnell在本书中打开了这个潘多拉之盒背后的秘密。

如果你是一名管理人员 如果你是一名管理人员，相信对“财务预算表”不陌生吧，虽然财务预算也从来没有准确过，但它却总能够为管理提供一个范围。

对于业务管理、软件项目而言，其道理也是相同的，戴明博士在TQM理论中提出的管制图（如图2所示）就高度概括了这一思想： 管理层的合理预期是使软件估算结果在一个可以控制的范围之内，即管理下限和管理上限之间。

如果今后的项目执行情况都能够落在这个区域，那么就意味着是可控的；而如果有大量的数据点都落在管理上限之上，或管理下限之下，就意味着控制性很差。

我想有着丰富管理理论的你，当从项目经理那里听到整个项目需要35.2个人月的估算时，你的理解可能是类似于“需要30~40个人月”的概念吧！

鬼才相信一开始就能够给出如此精确的估算值哩。

## &lt;&lt;软件估算&gt;&gt;

你肯定知道，项目的估计准确度是随着项目进展而不断提高的，是一个“瞄准靶心”的过程，一开始要做的是确定出“1环”和“脱靶”之间的边界，然后才能确定“2环、3环、...、10环”，这种被Steve McConnell定义为“不确定性锥”的理念是使软件估算有意义的基础。

但项目经理似乎没有发现这种“观念”，这就让问题从一开始就深深埋下了，因此你该告诉他！本书就能够完成这个任务，因为它围绕着这一观念展开了有效、深入的讲解。

当然如果你有时间翻阅本书，就能够更深入地理解特定于软件估算所遇到的困难，更好地理解诸如软件开发人员产能不稳定、不同软件项目存在很大差异等特殊因素，以便为项目团队的估算提供更好的行政支持。

如果你是一名项目经理 如果你是一名项目经理，首先要理解在项目初期，估算的结果是一个靶子，而不是马上给出一个靶心，也就是说得到的结果应该是“需要30~40个人月，最可能的值是36个人月”，而不是精确的35.2个人月。

如果对这个概念还有些模糊，建议你回头看看上一小节，理解理解“管制图”的概念，不管怎么说你也是一个“经理”，算是个管理岗位吧！

除此之外，还需要建立以下几个关键的理念，而如果你开始有了点感觉，那么就不要再等待，马上翻开正文，让自己接受一次全新的思维洗礼吧。

1. 总估算值不能谈判，只对单个估算项进行修改 你向高级管理人员汇报，整个项目要5个月的时间，但却被要求只能3个月完成！

然后基于这个数字展开的讨价还价的场景，怎么听怎么像在菜市场。

这既不是一个严谨态度，也没有采用科学的方法！

问题出在哪里呢？

想想工程预算（诸如家装）时，你要调整整体造价时，不会直接修改总价吧！

该怎么做，谁都知道应该修改某个单项预算，因为总价是用公式自动累加出来的嘛。

因此要想避免前面说到的问题，就应该提供包含各种单项预算的表格，当高级管理人员需要减少时间时，要商量的就是去掉哪些单项。

放心吧，高级管理人员早在年度预算会上就适应这种模式了。

2. 估算不是玄学 《软件工程经济学》之类的里程碑性著作作为软件估算学奠定了坚实的基础，但同时也将许多道行尚浅的项目经理带到万丈高空，使估算活动彻底与开发实践脱节了。

相对于COCOMO、FP之类高深的估算学模型而言，无数一线的开发人员更需要的是学以致用、能够解决实际问题的方法，这些方法在本书中被称为“估算术”，而且多到几乎占据了全书所有的篇幅，没有给这些大名鼎鼎的模型留下一点篇幅。

在阅读本书时，其中列举的各种精致、有效、小巧的估算方法总让我联想到“瑞士军刀”，相信大家也会有相似的感觉。

3. 经验数据是提升估算准确率的关键 经历并不意味着经验，这是我常挂在嘴边的一句话，放在估算领域也是如此。

许多项目经理、开发人员虽然早已身经百战，但每当遇到新的开发任务时，却仍然由于没有经验数据而难以给出较好的估算，甚至连自己的产能都不太了解。

估算的结果需要进行校准，才能够更加准确。

而最无奈之举则是使用估算模型提供的行业平均数据进行校准，要想真正有效就必须收集、记录自己的经验数据。

本书不仅指出了应该收集、记录哪些数据，还指出了项目初期的数据能够用来修正项目中、后期的计划，这一理念必将为你的“经验数据收集工程”注入强有力的引擎。

4. 分解是复杂性的克星 对于任何复杂的问题，都可以借助“分解”这一伟大的工具来应对。

因此WBS（Work Breakdown Structure，工作分解结构）分解的质量对于估算而言意义重大，而仅从软件开发过程的阶段、活动来划分是不足以支撑估算的；因此必须从软件开发的产物（诸如用例、用户故事）的角度进行分解，这样才能够为估算奠定良好的基础。

因此将需求开发的输出作为WBS的基础，让负责具体实现的开发人员对每个单项进行估算，然后

## &lt;&lt;软件估算&gt;&gt;

在此基础上进行累加、考虑缓冲、重组，最后才能够得到真正合理的估算结果。

5. 估算的问题不仅困扰你一个 在多年的职业生涯中，老听到诸如“中国的客户水平差，连需求都说不清”、“这种东西放到中国来是行不通的”……之类的“国情托词”。其实不然，在规模估算、工作量估算、进度估算、计划参数估算、估算结果表达以及围绕着估算结果的谈判等活动中，全世界的开发团队都将公平地遇到相似的问题，Steve McConnell特意用了很大的篇幅讲解他在实践中总结出来的解决之道。

怎么样，是否感到有点兴趣盎然呀！

不要轻易放过“兴趣”这一最好的老师，现在就开始愉快的阅读之旅吧，相信全书简洁且充满哲理的文字会给你带去好的心情和收获。

如果你是开发团队的一员 嗨，开发团队中的普通一员！

别走开，这里同样欢迎你！

估算并非是头衔中带有“经理”字样的那些人的特权。

对于每个开发活动而言，都需要你对其进行估算，如果你的估算结果偏离太远，那么整个项目估算就是在浮沙之上筑高台。

积累反映自己产能的经验数据，以便正确地对完成任务所需时间进行估算，从而对团队做出现实的承诺，才能够使自己真正远离“无休止”加班的困扰，毕竟加班要解决的就是那些不应该设置的最后期限，是对不负责任的进度承诺的惩罚！

学会记录自己的时间，掌握基本的估算方法，这些是使自己成长为具有专业素养的开发人员所必备的过程。

而从本书中，你可以获得你想要的东西。

感言 在我看完整本书之后，脑海里便浮现出了金庸笔下的两门绝世武功。

用它们作类比，既显得十分贴近，却又有几分不像之处： ？它好像是“易筋经”，能够帮你打通奇经八脉，让你建立系统、清晰的估算知识体系；它也不像是“易筋经”，因为它没有那么神秘、晦涩，不必非得“有缘人”才可练就。

？它好像是“降龙十八掌”，每招每式都那么有气势，有威力；它也不像是“降龙十八掌”，因为使用它的人并不需要绝顶的内力。

哎呀！

不知不觉将你挡在这本“睿智小书”之前好长时间了，我还是赶快躲开吧，不影响你享受阅读之趣了。

徐 锋 2007年10月于厦门 译者序 软件估算是项目计划和控制的基础。

任何一个软件项目在开始实施之前和实施的过程中，都需要对软件的规模、成本、工作量和进度，等等方面进行估算。

但是由于软件开发是一个非常复杂的过程，故软件估算具有极高的复杂性和不确定性，以至于估算过程往往被看做是一种“黑匣子过程”。

在本书之前，还没有哪本书籍能够如此全面、深入地阐述如何才能对软件项目进行有效和准确的估算。

以往那些涉及软件估算的书更多的是对一些相当成熟的开发组织的估算方法进行理论分析。

这些开发组织采用的方法虽然很科学，但是对大多数开发组织而言可能并不具有很高的效费比。

本书的作者Steve McConnell是资深的软件开发人员和久负盛名的软件开发书籍作家，他领导开发的软件曾荣获Software Development杂志的生产力大奖，他的著作也曾两度荣获Software Development杂志的软件开发书籍震撼大奖。

在《软件估算——“黑匣子”揭秘》一书中，他为软件开发组织和开发人员获得基本的估算技能提供了有效的实践指南，并为他们指出了持续提高估算能力的基本途径。

本书虽然涉及一些数学计算（这在估算中是不可避免的），但它尽可能地避免了过于复杂的公式推导，并提供众多的提示，以帮助读者通过建立较少的工作来获得更准确的估算结果。

用作者的话说，本书关注的重点是实用的估算术，而不是复杂的估算学方法。

就在本书的翻译过程中，还传来了本书荣获Software Development杂志2007年生产力大奖的消息。

## <<软件估算>>

这再次肯定了本书提供的那些实际的、经过作者验证的亲身经验的价值。

本书主要由宋锐翻译，如果广大读者需要对本书的内容或与软件估算有关的问题进行讨论，可以发送电子邮件至coldmoon75@163.com。

Be Flying工作室负责人肖国尊负责本书译员的确定，翻译质量和进度的控制与管理。

此外，本书的出版离不开电子社编辑许艳所做的大量协调工作，没有她的积极推动，本书有可能延迟半年以上出版，在此予以特别感谢。

敬请广大读者提供反馈意见，读者可以将意见E-mail至be-flying@sohu.com，我们会仔细阅读读者发来的每一封邮件，以求进一步提高今后译著的质量。



## <<软件估算>>

### 编辑推荐

在《软件估算：“黑匣子”揭秘》一书中，著名的软件开发书籍的作者Steve McConnell揭开了围绕在软件估算周围的层层迷雾。作者在深入浅出地介绍了与软件估算有关的主要概念之后，深入、全面地介绍了与软件估算有关的各种估算方法。

<<软件估算>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>