

PEARSON

40周年中文  
纪念版

**THE**  
Mythical  
Man—Month  
Essays on Software Engineering, Anniversary Edition

# 人月 神话

(Frederick P. Brooks, Jr.)

[美]小弗雷德里克·布鲁克斯/著

UMLChina翻译组 汪颖/译

PEARSON

ALWAYS LEARNING ALWAYS LEARNING ALWAYS LEARNING

清华大学出版社

# 人 月 神 话

(40 周年中文纪念版)

[美] 小弗雷德里克·布鲁克斯 著  
UMLChina 翻译组 汪颖 译

清华大学出版社  
北 京

Authorized translation from the English language edition, entitled The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition, 9780201835953 by Frederick P. Brooks, JR., published by Pearson Education, Inc, publishing as Addison-Wesley, Copyright © 1995.

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

CHINESE SIMPLIFIED language edition published by PEARSON EDUCATION ASIA LTD., and TSINGHUA UNIVERSITY PRESS Copyright © 2015.

本书中文简体翻译版由培生教育出版集团授权给清华大学出版社出版发行。未经许可，不得以任何方式复制或抄袭本书的任何部分。

北京市版权局著作权合同登记号 图字：01-2002-5152

本书封面贴有 Pearson Education(培生教育出版集团)防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

#### 图书在版编目(CIP)数据

人月神话(40周年中文纪念版)/(美)布鲁克斯(Brooks, F. P.)著; UML China 翻译组, 汪颖译. —北京: 清华大学出版社, 2015(2015.6重印)

书名原文: The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition

ISBN 978-7-302-39264-4

I. ①人… II. ①布… ②U… ③… III. ①软件工程 IV. ①TP311.5

中国版本图书馆CIP数据核字(2015)第024400号



责任编辑: 陈莉

封面设计: 周晓亮

版式设计: 牛静敏

责任校对: 曹阳

责任印制: 杨艳

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦A座 邮 编: 100084

社总机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈: 010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印刷者: 清华大学印刷厂

装订者: 北京市密云县京文制本装订厂

经 销: 全国新华书店

开 本: 180mm×250mm 印 张: 24.5 字 数: 316千字

版 次: 2015年4月第1版 印 次: 2015年6月第3次印刷

印 数: 10001~13000

定 价: 68.00元



## 神品——代序\*

---

这是本书中唯一的一节废话。

我是一个书狂，积习甚深，费尽心机在软件工程、系统工程方面积累了一些书。书，在我看来当分为神品、精品和普通三等，其中神品、精品又分别有一、二和三品之分。我所收集的书中，软件工程书大都属于精品，神品只有两本，Frederick P. Brooks 的这本书就属于神品之列。

软件作为一个行业，逐步背起了“solving the wrong problem”（解决错误的问题）的名声。问题决定解决方案，这也就是说，我们一直在制造错误解决方案！这方面有大量的证据，其中最著名的是美国政府统计署（GAO）的数据：全球最大的软件消费商（美国军方）每年要花费数十亿美元购买软件，而在其所购软件中，可直接使用的只占 2%，另外 3% 需要做一些修改，其余 95% 都成了垃圾。一句话，不管这些软件是否符合需求规格，它们都显然没有满足客户的需求。面向对象技术并没有给我们带来“神奇的效应”，不管开发商如何吹嘘面向对象 OO(Object-Oriented) 工具多么万能，也不管那些 OO 狂热者多么毅然地前赴后继，这方面的数据从 20 世纪 80 年代以来并没有发生大的改观。

这实在令我们的软件工程专家和从业者们羞愧，因为它揭示了我

---

\* 编注：该序的作者是王计斌(Dave Wang)，清华大学博士，研究方向包括软件工程和集成产品开发(IPD)，长期从事 IPD/CMM 推行，创办软件工程研究与实践论坛(<http://www.seforum.net>)，现在华为技术有限公司工作。

们可能一开始就从根本上做错了什么！20世纪90年代中期，当软件工程一代宗师 Michael Jackson(非歌坛巨星 Michael Jackson)宣布他们的研究成果时，立刻在软件工程界激起了阵阵涟漪。Jackson指出，软件从业人员和方法学大师们只是简单地模仿和照搬其他学科的方法，却将最重要的方面(问题域)忽略了。他指出，面向对象方法和结构化方法对问题域的处理没有什么大的区别，却被人们过分地用美好的词汇美化了：

“...You can see the results clearly in many object-oriented modeling descriptions. Often they are accompanied by fine words about modeling the real world. But when you look closely you can see that they are really descriptions of programming objects, pure and simple. Any similarity to real-world objects, living or dead, is purely coincidental...”

(……从众多面向对象建模的描述中，你可以很清楚地看到这些恶果。而且它们还经常伴随着有关现实世界建模的非常美好的词汇。然而，仔细看看，你就会发现它们其实是彻头彻尾的编程对象！如果说有任何和现实世界对象相似的地方，不管是死是活，纯属巧合……)

回首软件工程近40年的发展，Jackson哀叹软件行业普遍缺乏专业性，充满了业余人员，“手中有一个锤子，看到什么都是钉子”，谁都可以开发性命攸关的软件。

这就是我们面临的严峻而复杂的现实，也许您会感到震惊！然而在大师 Frederick P. Brooks 眼里，是那么的平静。因为早在28年前，他就在《人月神话》(*The Mythical Man-Month*)这本不朽著作中对这些内容做了深入论述。

这本小册子行文优美，思想博大精深，字字真言，精读之有不尽的趣味，藏之又是极珍贵的文献，名眼高人，自能鉴之。

前些年，一位朋友从印度归来，说此书在印度极为普及，我也动起笔来，但惭愧终未成正果。汪颖兄素来勤恳，明知此翻译为“success without applause, diligence without reward”（没有掌声的成功，没有回报的勤勉），却兢兢业业，反复琢磨，历经单调、繁琐、艰辛的劳动，终于付梓。钦佩之余随即作序共勉。

Dave Wang  
SE Forum China  
2002年3月于深圳



# 人月神话 20周年纪念版序言

令我惊奇和高兴的是,《人月神话》在20年后仍然继续流行,印数超过了250 000册。人们经常问,我在1975年提出的观点和建议,哪些是我仍然坚持的,哪些是已经改变了的,又是怎样改变的?尽管我在一些讲座上也分析过这个问题,但我还是一直想把它写成文章。

Peter Gordon现在是Addison-Wesley的出版伙伴,他从1980年开始和我共事。他非常有耐心,对我帮助很大。他建议我们准备一个纪念版本。我们决定不对原版本做任何修订,只是原封不动地重印(除了一些无足轻重的修正),并用更新的思想来扩充它。

第16章重印了一篇在1986年IFIPS会议上的论文“没有银弹:软件工程的根本和次要问题”(No Silver Bullet: Essence and Accidents of Software Engineering)。这篇文章来自我在国防科学委员会主持军用软件方面研究时的经验。我当时的研究合作者,也是我的执行秘书,Robert L. Patrick,他帮助我回想和感受那些做过的软件大项目。1987年,IEEE的《计算机》(Computer)杂志重印了这篇论文,使它传播得更广了。

“没有银弹”被证明是富有煽动性的,它预言10年内没有任何编程技巧能够给软件的生产率带来数量级上的提高。10年只剩下一年了,我的预言看来是安全了。“没有银弹”激起了越来越多文字上的剧烈争论,比《人月神话》还要多。因此在第17章,我对一些公开的批评做了说明,并更新了在1986年提出的观点。

在准备《人月神话》的回顾和更新时，一直在进行的软件工程研究和经验已经批评、证实或否定了少数书中断言的观点，也影响了我。剥去辅助的争论和数据后，把那些观点粗略地分类，对我来说是很有帮助的。我在第 18 章列出了这些观点的概要，希望这些单调的陈述能够引来争论和证据，然后得到证实、否定、更新或精炼。

第 19 章是一篇更新的短文。读者应该注意的是，新观点并不像原来的书一样来自我的亲身经历。我在大学里工作，而不是在工业界，做的是小规模的项目，而不是大项目。自 1986 年以来，我就只是教授软件工程，不再做这方面的研究。我现在的研究领域是虚拟环境及其应用。

在这次回顾的准备过程中，我找了一些正在软件工程领域工作的朋友，征求他们现在的观点。他们很乐意与我分享他们的想法，并仔细地草稿提出了意见，这些都使我重新受到启发。感谢 Barry Boehm、Ken Brooks、Dick Case、James Coggins、Tom DeMarco、Jim McCarthy、David Parnas、Earl Wheeler 和 Edward Yourdon。感谢 Fay Ward 对新的章节进行了出色的技术加工。

感谢我在国防科学委员会军事软件工作组的同事 Gordon Bell、Bruce Buchanan、Rick Hayes-Roth，特别是 David Parnas，感谢他们的洞察力和生动的想法。感谢 Rebekah Bierly 对第 16 章的论文进行了技术加工。我把软件问题分成“根本的”和“次要的”，这是受 Nancy Greenwood Brooks 的启发，她在一篇“Suzuki 小提琴教育”的论文中应用了这样的分析方法。

在 1975 年版本的序言中，Addison-Wesley 出版社按规定不允许我在书中向该社的一些扮演了关键角色的员工致谢。可是，有两个人贡献必须特别指出：执行编辑 Norman Stanton 和美术指导 Herbert Boes。Boes 设计了优雅风格的版式和封面，他在评注时特别提到：“页边的空白要宽，字体和版面要有想象力。”更重要的是，他提出了至关重要的建议：



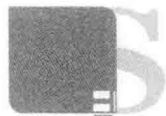
为每一章的开头配一幅图片(当时我只有“焦油坑”和“兰斯大教堂”的图片)。寻找这些图片使我多花了一年的时间，但我永远感激这个忠告。

*Soli Deo Gloria*——愿神独得荣耀。

Frederick P. Brooks, Jr.

Chapel Hill, N.C.

1995年3月





## 第 1 版序言

---

在很多方面，管理一个大型的计算机编程项目与管理其他行业的大型工程很相似——比大多数程序员所认为的还要相似；在另外一些方面，它又有差别——比大多数职业经理人所认为的差别还要大。

这个领域的知识在于累积。现在，AFIPS(美国信息处理学会联合会)已经有了一些讨论和会议，也出版了一些书籍和论文，但是还没有成形的办法对这一领域来进行系统地阐述。提供这样一本主要反映个人观点的小书看来是合适的。

虽然我原来从事计算机科学的编程方面的工作，但是在 1956—1963 年，自动控制程序和高级语言编译器开发出来的时候，我主要参加的是硬件构架方面的工作。1964 年，我成为操作系统 OS/360 的经理，我发现前些年的进展使编程世界改变了很多。

虽然是失败的，但管理 OS/360 的开发仍是一次很有帮助的经历。负责这次开发项目的团队，包括我的继任经理 F. M. Trapnell，有很多值得自豪的东西。该系统在设计 and 执行方面都很出色，并被成功地应用到很多领域，特别是设备独立的输入输出和外部库管理，在很多技术革新中被广泛复制。现在，这一系统是十分可靠的，相当有效且非常通用。

但是，并不是所有的努力都是成功的。所有 OS/360 的用户很快就能发现它应该能够做得更好。设计和执行上的缺陷在控制程序中特别普

遍，相比之下，语言编译器就好得多。大多数缺陷发生在1964—1965年的设计阶段，所以这肯定是我的责任。此外，这个产品发布推迟了，需要的内存比计划中的要多，成本也是估计的好几倍，而且第一次发布时并不能很好地运行，直到发布了几次以后，问题才得以解决。

按照当初接受OS/360任务时的协议，在1965年离开IBM后，我来到Chapel Hill。我开始分析OS/360的经验，看能不能从中学到什么管理和技术上的教训。特别要说明的是，System/360硬件开发和OS/360软件开发中的管理经验是大相径庭的。对Tom Watson关于为什么编程难以管理的探索性问题，这本书是一份迟来的答案。

在这次探索中，我和1964—1965年的经理助理R. P. Case，还有1965—1968年的经理F. M. Trapnell进行了长谈，从中受益很多。我还对比了其他大型编程项目经理的结论，这些项目经理包括M. I. T.的F. J. Corbato，贝尔电话实验室的V. Vyssotsky和John Harr，International Computers Limited的Charles Portman，苏联科学院西伯利亚分部计算实验室的A. P. Ershov和IBM的A. M. Pietrasanta。

我自己的结论体现在下面的文字中，送给专业程序员、职业经理，特别是程序员的职业经理。

虽然写出来的是各自独立的章节，但本书还是有一个中心的论点，特别包含在第2~7章。简言之，我相信由于人员的分工，大型编程项目碰到的管理问题和小项目碰到的管理问题区别很大；我相信关键需要的是维持产品自身的概念完整性。这几章探讨了其中的困难和解决的方法。而后续的章节则探讨了软件工程管理的其他方面。

这个领域的文献并不多，但散布很广。因此我尝试在书后给出了参考文献，说明某个特定知识点并指导感兴趣的读者去参阅其他有用的文献。很多朋友读过了本书的手稿，其中一些朋友还给出了很有帮助的意

见。这些意见很有价值，但为了不打乱文字的通顺，我把它们作为注解包含在本书中。

因为这本书是一部文集而不是一部教材，所有的参考文献和注解都被放到书的末尾，建议读者在读第一遍时略去不看。

深深地感谢 Sara Elizabeth Moore 小姐、David Wagner 先生和 Rebecca Burris 夫人，他们帮助我准备了手稿。感谢 Joseph C. Sloane 教授在图解方面的建议。

Frederick P. Brooks, Jr.  
Chapel Hill, N.C.  
1974 年 10 月



# 目 录

---

第 1 章	焦油坑 .....	1
	编程系统产品 .....	4
	职业的乐趣 .....	6
	职业的苦恼 .....	8
第 2 章	人月神话 .....	11
	乐观主义 .....	14
	人月 .....	16
	系统测试 .....	19
	空泛的估算 .....	21
	重复产生的进度灾难 .....	22
第 3 章	外科手术队伍 .....	27
	问题 .....	30
	Mills 的建议 .....	32
	如何运作 .....	35
	团队的扩建 .....	36
第 4 章	贵族专制、民主政治和系统设计 .....	39
	概念的完整性 .....	42

	获得概念的完整性.....	43
	贵族专制统治和民主政治.....	44
	在等待时，实现人员应该做什么.....	47
<b>第 5 章</b>	<b>画蛇添足.....</b>	<b>51</b>
	结构师的交互准则和机制.....	54
	自律——开发第二个系统所带来的后果.....	55
<b>第 6 章</b>	<b>贯彻执行.....</b>	<b>59</b>
	文档化的规格说明——手册.....	62
	形式化定义.....	63
	直接整合.....	66
	会议和大会.....	66
	多重实现.....	68
	电话日志.....	68
	产品测试.....	69
<b>第 7 章</b>	<b>为什么巴比伦塔会失败.....</b>	<b>71</b>
	巴比伦塔的管理教训.....	75
	大型编程项目中的交流.....	76
	项目工作手册.....	76
	大型编程项目的组织架构.....	80
<b>第 8 章</b>	<b>胸有成竹.....</b>	<b>85</b>
	Portman 的数据.....	89
	Aron 的数据.....	90
	Harr 的数据.....	90
	OS/360 的数据.....	92

Corbató的数据	93
<b>第 9 章 削足适履</b>	<b>95</b>
作为成本的程序空间	98
规模控制	99
空间技能	100
数据的表现形式是编程的根本	102
<b>第 10 章 提纲挈领</b>	<b>105</b>
计算机产品的文档	108
大学科系的文档	110
软件项目的文档	110
为什么要有正式的文档	111
<b>第 11 章 未雨绸缪</b>	<b>113</b>
试验性工厂和增大规模	116
唯一不变的就是变化本身	117
为变更设计系统	117
为变更计划组织架构	118
前进两步，后退一步	120
前进一步，后退一步	122
<b>第 12 章 干将莫邪</b>	<b>125</b>
目标机器	129
辅助机器和数据服务	131
高级语言和交互式编程	134
<b>第 13 章 整体部分</b>	<b>139</b>
剔除 bug 的设计	142

	构件单元调试 .....	144
	系统集成调试 .....	147
第 14 章	祸起萧墙 .....	153
	里程碑还是沉重的负担 .....	156
	“其他的部分反正会落后” .....	158
	地毯的下面 .....	159
第 15 章	另外一面 .....	165
	需要什么样的文档 .....	169
	流程图 .....	171
	自文档化的程序 .....	175
第 16 章	没有银弹 .....	181
	摘要 .....	184
	介绍 .....	184
	根本困难 .....	185
	以往解决次要困难的一些突破 .....	190
	银弹的希望 .....	192
	针对概念上根本问题的颇具前途的方法 .....	200
第 17 章	再论“没有银弹” .....	209
	人狼和其他恐怖传说 .....	212
	存在着银弹——就在这里 .....	212
	含糊的表达将会导致误解 .....	213
	Harel 的分析 .....	216
	Jones 的观点——质量带来生产率 .....	221
	那么，生产率的情形如何 .....	222



面向对象编程——这颗铜质子弹可以吗	223
重用的情况怎样	225
学习大量的词汇——对软件重用的一个可预见 但还没有被预言的问题	228
子弹的本质——形势没有发生改变	229
<b>第 18 章 《人月神话》的观点：是与非</b>	<b>231</b>
第 1 章 焦油坑	234
第 2 章 人月神话	235
第 3 章 外科手术队伍	236
第 4 章 贵族专制、民主政治和系统设计	237
第 5 章 画蛇添足	238
第 6 章 贯彻执行	239
第 7 章 为什么巴比伦塔会失败	240
第 8 章 胸有成竹	242
第 9 章 削足适履	243
第 10 章 提纲挈领	245
第 11 章 未雨绸缪	246
第 12 章 干将莫邪	249
第 13 章 整体部分	251
第 14 章 祸起萧墙	253
第 15 章 另外一面	255
第 1 版结束语	256
<b>第 19 章 20 年后的《人月神话》</b>	<b>257</b>
为什么要出版 20 周年纪念版本	260
核心观点——概念完整性和结构师	261