

The Mythical Man-Month
Essays on Software Engineering
Anniversary Edition

Frederick P. Brooks, Jr.

University of North Carolina at Chapel Hill



ADDISON-WESLEY

Boston • San Francisco * New York « Toronto « Montreal
London « Munich * Paris ° Madrid
Capetown • Sydney • Tokyo • Singapore • Mexico City

Cover drawing: C. R. Knight, Mural of the La Brea Tar Pits. Courtesy of the George C. Page Museum of La Brea Discoveries, The Natural History Museum of Los Angeles County. Cover designer: Diana Coe.

The essay entitled, No Silver Bullet, is from Information Processing 1986, the Proceedings of the IFIP Tenth World Computing Conference, edited by H.-J. Kugler, 1986, pages 1069-1076. Reprinted with the kind permission of IFIP and Elsevier Science B.V., Amsterdam, The Netherlands.

Library of Congress Cataloging-in-Publication Data

Brooks, Frederick P., Jr. (Frederick Phillips)

The mythical man-month : essays on software engineering /

Frederick P. Brooks, Jr. — Anniversary ed.

p. cm.

Includes bibliographical references and index.

ISBN 0-201-83595-9

1. Software engineering. I. Title.

QA76.758.B75 1995

005.1'068—dc20

94-36653

CIP

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and Addison-Wesley was aware of a trademark claim, the designations have been printed in initial caps or all caps.

Copyright © 1995 Addison Wesley Longman, Inc.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of the publisher and author. Printed in the United States of America.

Text printed on recycled and acid-free paper.

ISBN 0201835959

17 1819202122 MA 05 04 03 02

17th Printing August 2002

Dedication of the 1975 edition

*To two who especially enriched my IBM years:
Thomas / . Watson, Jr.,
whose deep concern for people still permeates his company,
and
Bob O. Evans,
whose bold leadership turned work into **adventure**.*

Dedication of the 1995 edition

*To Nancy,
God's gift to me.*

Preface to the 20th Anniversary Edition

To my surprise and delight, *The Mythical Man-Month* continues to be popular after 20 years. Over 250,000 copies are in print. People often ask which of the opinions and recommendations set forth in 1975 I still hold, and which have changed, and how. Whereas I have from time to time addressed that question in lectures, I have long wanted to essay it in writing.

Peter Gordon, now a Publishing Partner at Addison-Wesley, has been working with me patiently and helpfully since 1980. He proposed that we prepare an Anniversary Edition. We decided not to revise the original, but to reprint it untouched (except for trivial corrections) and to augment it with more current thoughts.

Chapter 16 reprints "No Silver Bullet: Essence and Accidents of Software Engineering," a 1986 IFIPS paper that grew out of my experience chairing a Defense Science Board study on military software. My coauthors of that study, and our executive secretary, Robert L. Patrick, were invaluable in bringing me back into touch with real-world large software projects. The paper was reprinted in 1987 in the IEEE *Computer* magazine, which gave it wide circulation.

"No Silver Bullet" proved provocative. It predicted that a decade would not see any programming technique that would by itself bring an order-of-magnitude improvement in software productivity. The decade has a year to run; my prediction seems safe. "NSB" has stimulated more and more spirited discussion

in the literature than has *The Mythical Man-Month*. Chapter 17, therefore, comments on some of the published critique and updates the opinions set forth in 1986.

In preparing my retrospective and update of *The Mythical Man-Month*, I was struck by how few of the propositions asserted in it have been critiqued, proven, or disproven by ongoing software engineering research and experience. It proved useful to me now to catalog those propositions in raw form, stripped of supporting arguments and data. In hopes that these bald statements will invite arguments and facts to prove, disprove, update, or refine those propositions, I have included this outline as Chapter 18.

Chapter 19 is the updating essay itself. The reader should be warned that the new opinions are not nearly so well informed by experience in the trenches as the original book was. I have been at work in a university, not industry, and on small-scale projects, not large ones. Since 1986, I have only taught software engineering, not done research in it at all. My research has rather been on virtual environments and their applications.

In preparing this retrospective, I have sought the current views of friends who are indeed at work in software engineering. For a wonderful willingness to share views, to comment thoughtfully on drafts, and to re-educate me, I am indebted to Barry Boehm, Ken Brooks, Dick Case, James Coggins, Tom DeMarco, Jim McCarthy, David Parnas, Earl Wheeler, and Edward Yourdon. Fay Ward has superbly handled the technical production of the new chapters.

I thank Gordon Bell, Bruce Buchanan, Rick Hayes-Roth, my colleagues on the Defense Science Board Task Force on Military Software, and, most especially, David Parnas for their insights and stimulating ideas for, and Rebekah Bierly for technical production of, the paper printed here as Chapter 16. Analyzing the software problem into the categories of *essence* and *accident* was inspired by Nancy Greenwood Brooks, who used such analysis in a paper on Suzuki violin pedagogy.

Addison-Wesley's house custom did not permit me to acknowledge in the preface to the 1975 edition the key roles played by their staff. Two persons' contributions should be especially cited: Norman Stanton, then Executive Editor, and Herbert Boes, then Art Director. Boes developed the elegant style, which one reviewer especially cited: "wide margins, [and] imaginative use of typeface and layout." More important, he also made the crucial recommendation that every chapter have an opening picture. (I had only the Tar Pit and Reims Cathedral at the time.) Finding the pictures occasioned an extra year's work for me, but I am eternally grateful for the counsel.

Soli Deo gloria—To God alone be glory.

*Chapel Hill, N.C..
March 1995*

F. P. B., Jr.

Preface to the First Edition

In many ways, managing a large computer programming project is like managing any other large undertaking—in more ways than most programmers believe. But in many other ways it is different—in more ways than most professional managers expect.

The lore of the field is accumulating. There have been several conferences, sessions at AFIPS conferences, some books, and papers. But it is by no means yet in shape for any systematic textbook treatment. It seems appropriate, however, to offer this little book, reflecting essentially a personal view.

Although I originally grew up in the programming side of computer science, I was involved chiefly in hardware architecture during the years (1956-1963) that the autonomous control program and the high-level language compiler were developed. When in 1964 I became manager of Operating System/360, I found a programming world quite changed by the progress of the previous few years.

Managing OS/360 development was a very educational experience, albeit a very frustrating one. The team, including F. M. Trapnell who succeeded me as manager, has much to be proud of. The system contains many excellencies in design and execution, and it has been successful in achieving widespread use. Certain ideas, most noticeably device-independent input-output and external library management, were technical innovations

now widely copied. It is now quite reliable, reasonably efficient, and very versatile.

The effort cannot be called wholly successful, however. Any OS/360 user is quickly aware of how much better it should be. The flaws in design and execution pervade especially the control program, as distinguished from the language compilers. Most of these flaws date from the 1964-65 design period and hence must be laid to my charge. Furthermore, the product was late, it took more memory than planned, the costs were several times the estimate, and it did not perform very well until several releases after the first.

After leaving IBM in 1965 to come to Chapel Hill as originally agreed when I took over OS/360, I began to analyze the OS/360 experience to see what management and technical lessons were to be learned. In particular, I wanted to explain the quite different management experiences encountered in System/360 hardware development and OS/360 software development. This book is a belated answer to Tom Watson's probing questions as to why programming is hard to manage.

In this quest I have profited from long conversations with R. P. Case, assistant manager 1964-65, and F. M. Trapnell, manager 1965-68. I have compared conclusions with other managers of jumbo programming projects, including F. J. Corbato of M.I.T., John Harr and V. Vyssotsky of Bell Telephone Laboratories, Charles Portman of International Computers Limited, A. P. Ershov of the Computation Laboratory of the Siberian Division, U.S.S.R. Academy of Sciences, and A. M. Pietrasanta of IBM.

My own conclusions are embodied in the essays that follow, which are intended for professional programmers, professional managers, and especially professional managers of programmers.

Although written as separable essays, there is a central argument contained especially in Chapters 2-7. Briefly, I believe that large programming projects suffer management problems

different in kind from small ones, due to division of labor. I believe the critical need to be the preservation of the conceptual integrity of the product itself. These chapters explore both the difficulties of achieving this unity and methods for doing so. The later chapters explore other aspects of software engineering management.

The literature in this field is not abundant, but it is widely scattered. Hence I have tried to give references that will both illuminate particular points and guide the interested reader to other useful works. Many friends have read the manuscript, and some have prepared extensive helpful comments; where these seemed valuable but did not fit the flow of the text, I have included them in the notes.

Because this is a book of essays and not a text, all the references and notes have been banished to the end of the volume, and the reader is urged to ignore them on his first reading.

I am deeply indebted to Miss Sara Elizabeth Moore, Mr. David Wagner, and Mrs. Rebecca Burns for their help in preparing the manuscript, and to Professor Joseph C. Sloane for advice on illustration.

Chapel Hill, N.C.
October 1974

F. P. B., Jr

Contents

	Preface to the 20th Anniversary Edition	vii
	Preface to the First Edition	x
Chapter 1	The Tar Pit	3
Chapter 2	The Mythical Man-Month	13
Chapter 3	The Surgical Team	29
Chapter 4	Aristocracy, Democracy, and System Design	41
Chapter 5	The Second-System Effect	53
Chapter 6	Passing the Word	61
Chapter 7	Why Did the Tower of Babel Fail?	73
Chapter 8	Calling the Shot	87
Chapter 9	Ten Pounds in a Five-Pound Sack	97
Chapter 10	The Documentary Hypothesis	107
Chapter 11	Plan to Throw One Away	115
Chapter 12	Sharp Tools	127
Chapter 13	The Whole and the Parts	141
Chapter 14	Hatching a Catastrophe	153
Chapter 15	The Other Face	163
Chapter 16	No Silver Bullet—Essence and Accident	177
Chapter 17	"No Silver Bullet" Refired	205
Chapter 18	Propositions of <i>The Mythical Man-Month</i> : True or False?	227
Chapter 19	<i>The Mythical Man-Month</i> after 20 Years -	251
	Epilogue	291
	Notes and References	293
	Index	309