



SEI 软件工程丛书·影印版

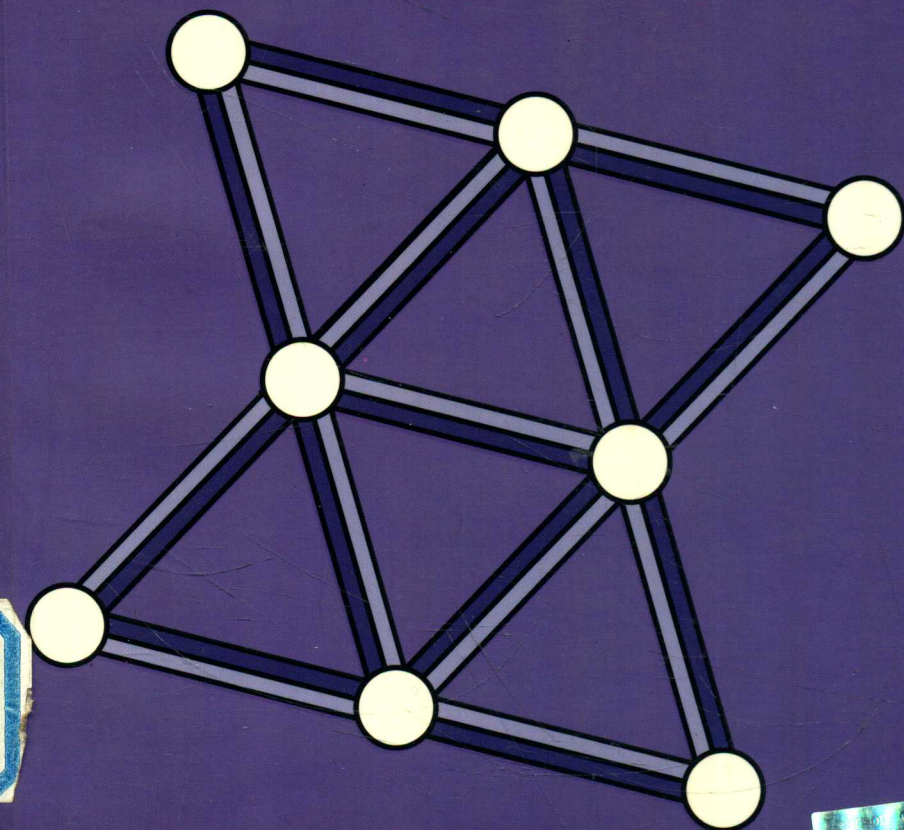
卡内基·梅隆大学软件工程研究所推荐·软件学院教材



基于构架的软件项目管理

Architecture-Centric Software Project Management:
A Practical Guide

[美] 丹尼尔·J·鲍里斯 [Daniel J. Paulish] 著



清华大学出版社



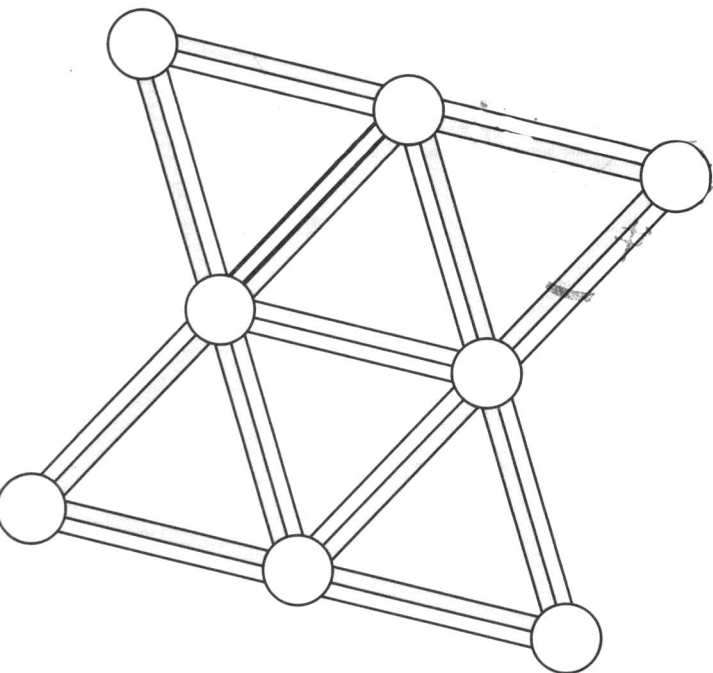
SEI 软件工程丛书·影印版
卡内基·梅隆大学软件工程研究所推荐·软件学院教材



基于构架的软件项目管理

Architecture-Centric Software Project Management:
A Practical Guide

[美] 丹尼尔·J·鲍里斯 [Daniel J. Paulish] 著



清华大学出版社
北京

内 容 简 介

基于构架的软件项目设计(ACSP)是一种重要的设计软件项目的软件开发方法,该方法可以准时、按预算、高效地完成任任务。本书说明了如何利用软件构架来设计进度表、生成评估表、决定作用域和管理开发团队。本书涉及高效软件管理的基础——计划、管理、实现和度量。作者给出了大量经过实践检验的宝贵例子,还以实际的案例来说明本书的策略、方法和技术。

本书可作为软件学院及大学计算机等专业相关课程的教材,也可以作为软件公司各级管理和开发人员参考。

EISBN: 0-201-73409-5

Architecture-Centric Software Project Management: A Practical Guide, 1e
Daniel J. Paulish

Copyright © 2002 by Pearson Education, Inc.

Original English language edition published by Pearson Education, Inc.
All right reserved.

本书影印版由 Pearson Education 授权清华大学出版社在中国境内,香港、澳门特别行政区和台湾地区除外,独家出版、发行。

未经出版者书面许可,不得以任何方式复制或抄袭本书的任何部分。

本书封面贴有 Pearson Education 的激光防伪标签,无标签者不得销售。

北京市版权局著作权合同登记号:图字 01-2002-0553 号

图书在版编目(CIP)数据

基于构架的软件项目管理 Architecture-Centric Software Project Management /

(美)鲍里斯著. — 影印版.—北京:清华大学出版社,2003

(SEI 软件工程丛书)

ISBN 7-302-06350-8

I. 基... II. 鲍... III. 软件开发—项目管理—英文 IV. TP311.52

中国版本图书馆 CIP 数据核字(2003)第 010095 号

出 版 者:清华大学出版社(北京清华大学学研大厦,邮编 100084)

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

<http://www.tup.tsinghua.edu.cn>

责任编辑:尤晓东

印 刷 者:北京广益印刷有限公司

发 行 者:新华书店总店北京发行所

开 本:787×960 1/16 印张:20.5 彩页:1

版 次:2003年3月第1版 2003年3月第1次印刷

书 号:ISBN 7-302-06350-8/TP·4792

印 数:0001~3000

定 价:39.00 元

出版说明

1984年,美国国防部出资在卡内基·梅隆大学设立软件工程研究所(Software Engineering Institute, 简称SEI)。SEI于1986年开始研究软件过程能力成熟度模型(Capability Maturity Model, CMM),1991年正式推出了CMM 1.0版,1993年推出CMM 1.1版。此后,SEI还完成了能力成熟度模型集成(Capability Maturity Model Integration, 简称CMMI)。目前,CMM 2.0版已经推出。

CMM自问世以来备受关注,在一些发达国家和地区得到了广泛应用,成为衡量软件公司软件开发管理水平的重要参考因素,并成为软件过程改进的事实标准。CMM目前代表着软件发展的一种思路,一种提高软件过程能力的途径。它为软件行业的发展提供了一个良好的框架,是软件过程能力提高的有用工具。

SEI十几年的研究过程和成果,都浓缩在由SEI参与研究工作的资深专家亲自撰写的SEI软件工程丛书(SEI Series In Software Engineering)中。

为增强我国软件企业的竞争力,提高国产软件的水平,经清华大学出版社和三联四方工作室共同策划,全面引进了这套丛书,分批影印和翻译出版,这套丛书采取开放式出版,不断改进,不断出版,旨在满足国内软件界人士学习原版软件工程高级教程的愿望。

清华大学出版社

2002年8月

“SEI 软件工程译丛”编委会

主任 周伯生

副主任 郑人杰

委员 (按姓名拼音顺序排列)

董士海 顾毓清 王 纬

吴超英 尤晓东

执行委员 尤晓东

秘 书 廖彬山

总 序

周伯生

美国卡内基·梅隆大学软件工程研究所 (CMU/SEI) 是美国联邦政府资助构建的研究单位, 由美国国防部主管。他们确认, 为了保证软件开发工作的成功, 由软件开发人员、软件采办人员和软件用户组成的集成化团队必须具有必要的软件工程知识和技能, 以保证能按时向用户交付正确的软件。所谓“正确的”就是指在功能、性能和成本几个方面都能满足用户要求且无缺陷; 所谓“无缺陷”就是指在编码后对软件系统进行了彻底的穷举测试修复了所有的缺陷, 或保证所编写的代码本身不存在缺陷。

CMU/SEI 为了达到这个目的, 提出了创造、应用和推广的战略。这里的“创造”是指与软件工程研究社团一起, 共同创造新的实践或改进原有的实践, 而不墨守成规。这里的“应用”是指与一线开发人员共同工作, 以应用、改进和确认这些新的或改进的实践, 强调理论联系实际。这里的“推广”是指与整个社团一起, 共同鼓励和支持这些经过验证和确认的、新的或改进的实践在世界范围内的应用, 通过实践进行进一步的检验和提高。如此循环, 往复无穷。

他们把所获得的成就归纳为两个主要领域。一个是倡导软件工程管理实践, 使软件组织在采办、构建和改进软件系统时, 具有预测的能力与控制质量、进度、成本、开发周期和生产效率的能力。另一个是改进软件工程技术的实践, 使软件工程师具有分析、预测和控制软件系统属性的能力, 其中包括在采办、构建和改进软件系统时, 能进行恰当的权衡, 作出正确的判断和决策。CMU/SEI 通过出版软件工程丛书, 总结他们的研究成果和实践经验, 是推广这两个领域经验的重大举措。

SEI 软件工程丛书由 CMU/SEI 和 Addison-Wesley 公司共同组织出版, 共分 4 个部分: 计算机和网络安全 (已出版了 2 本著作), 工程实践 (已出版了 8 本著作), 过程改进和过程管理 (已出版了 11 本著作), 团队软件过程和个体软件过程 (已出版了 3 本著作)。前两者属于软件

工程技术实践，后两者属于软件工程管理实践。目前这 4 个部分共出版了 24 本著作，以向软件工程实践人员和学生方便地提供最新的软件工程信息。这些著作凝聚了全世界软件工程界上百位开拓者和成千上万实践者的创造性劳动，蕴含了大量的宝贵经验和沉痛教训，很值得我们学习。

清华大学出版社邀请我和郑人杰教授共同组织 SEI 软件工程译丛编委会。清华社计划首先影印 6 本著作，翻译出版 15 本著作。据我所知，在 Addison-Wesley 公司出版的 SEI 软件工程丛书中，人民邮电出版社已经翻译出版了《个体软件过程》和《团队软件过程》，还拟影印出版《个体软件过程》和《软件工程规范》；电子工业出版社已经翻译出版了《净室软件工程的技术与过程》、《能力成熟度模型 CMM 1.1 指南》、《能力成熟度模型集成 CMMI》和《软件项目管理》；北京航空航天大学出版社已经翻译出版了《统计过程控制》。这些出版社共计影印 2 本著作，翻译出版 7 本著作。这样，可以预期我国在今年年底共可影印 8 本著作，翻译出版 22 本著作。各个出版社的有远见的辛勤劳动，为我们创造了“引进、消化、吸收、创新”的机遇。我们应该结合各自的实践，认真学习国外的先进经验，以大大提高我国软件工程的理论和实践水平。

在这套丛书中，特别值得一提的是，在过程工程领域被誉为软件过程之父的 Humphrey 先生所撰写的《软件过程管理》、《技术人员管理》、《软件工程规范》、《个体软件过程》、《团队软件过程》和《软件制胜之道》等 6 本著作，将于今年年内全部翻译出版，其中《软件过程管理》、《技术人员管理》、《软件工程规范》、《个体软件过程》和《软件制胜之道》等 5 本著作亦已经或将于今年年内影印出版。

《软件过程管理》是软件过程领域的开创性著作，是为软件公司经理和软件项目经理撰写的。用这本书提出的原理来指导软件开发，可以有效地按照预定进度得到高质量的软件，同时还可了解如何持续进行过程改进。美国 CMU/SEI 按照这本书提出的原理开发了能力成熟度模型，在国际上得到绝大多数国家的认可和广泛采用，是改进软件过程能力的有力武器。在信息技术迅速发展和企业激烈竞争的今天，能否持续改进过程往往决定企业的命运。

作为一个软件经理，在改进组织的能力之前，首先必须明确绝大多数软件问题是由管理不善所引起的。因此，要改进组织的性能，首先需

要改进自己的管理模式。同时还要认识到软件开发是一项智力劳动，需要拥有掌握高技能和忘我工作的技术人员。因此，有效的软件管理需要充分注意技术人员的管理。

《技术人员管理》这本著作就是为达到这个目的而撰写的。高质量的技术工作要求没有差错，这就要求人们高度专心和高度献身。因此要求人们对他所从事的工作不仅具有高度的责任感，而且具有浓厚的兴趣和高度的热忱。在当前知识经济群龙相争的今天，一个能激励人们进行创造性工作的领导群体，是众多竞争因素中最重要的因素。本书提供了大量的实用指南，可用来有效地改进工程人员、经理和组织的性能。

Humphrey 先生还认为这本书特别适合于在我国工作的软件经理。我国是一个人口大国，拥有大量能干的知识分子，而且信息领域的劳动力价格比国际市场的价格要低，因此吸引了许多国家到我国来投资。但若不提高人员的素质，不在产品质量和进度方面也狠下功夫，就不能在这方面持续保持优势。

《软件工程规范》是为编程人员撰写的。它精辟地阐述了个体软件过程（PSP）的基本原理，详尽地描述了人们如何来控制自己的工作，如何与管理方协商各项安排。在软件工程界，这本著作被誉为是软件工程由定性进入定量的标志。目前在世界范围内，有成千上万的软件工程技术人员正在接受有关 PSP 的培训，以便正确地遵循 PSP 的实践、开发和管理工作计划，在他们承诺的进度范围内，交付高质量的产品。

《软件制胜之道》这本著作描述了团队软件过程的基本原理，详尽地阐述了在软件组织中如何应用 PSP 和 TSP 的原理以及它所能带来的效益。此外，虽然 CMM 同样适用于小型组织，但在其他著作中都没有描述如何应用 CMM 于个体或小型团队，这本书填补了这个空白。应该指出，如果一个组织正在按照 CMM 改进过程，则 PSP 和 TSP 是和 CMM 完全相容的。如果一个组织还没有按照 CMM 改进过程，则有关 PSP 和 TSP 的训练，可以为未来的 CMM 实践奠定坚实的基础。

在软件工程技术实践方面目前共出版了 10 本著作，其中《用商业组件构建系统》、《软件构架实践》和《软件构架评估——方法和案例研究》等 3 本著作详尽地阐述了软件构架的构建、实践和评估。鉴于是否有一个稳定的软件构架，对软件的质量和成本影响很大，因此如何获得一个好的构架就成为当今软件界研究的重点。我相信这几本著作的出

版,将对我国软件构架领域的研究与实践有重要的参考价值。此外,众所周知,计算机与网络的安全问题对信息系统的可靠使用关系极大,《CERT 安全指南——系统与网络安全实践》的出版将会对我国在这一领域的研究和实践起积极的促进作用。《风险管理——软件系统开发方法》、《软件采办管理——开放系统和 COTS 产品》、《项目管理原理》、《软件产品线——实践和模式》和《系统工程:基于信息的设计方法》等 5 本著作,分别从风险管理、软件采办、项目管理、软件产品线以及信息系统设计方法等几个方面阐述了大型、复杂软件系统的开发问题,是有关发展软件产业的重要领域,很值得我国软件产业界借鉴。

目前我们所处的时代是信息化时代,是人类进入能够综合利用物质、能量和信息三种资源的时代。千百年来以传统的物质产品的生产、流通、消费为基本特征的物质型经济,将逐步进入以信息产品的生产、流通、利用和消费为基本特征的知识型经济。在这个历史任务中,建造和广泛应用各类计算机应用系统是其公共特征。计算机软件是计算机应用系统的灵魂,没有先进的软件产业,不可能有先进的信息产业,从而也不可能建成现代化的知识型经济。

我们应该看到,在软件领域中我国在总体上离世界先进水平还有相当大的差距。但是,我们不能跟随他国的脚印,走他人的老路。我们应该抓住机遇,直接针对未来的目标,在软件工程技术和软件工程管理两个方面,注意研究 SEI 软件工程从书中倡导的原理和方法,联系实际,认真实践,并充分利用我国丰富优秀的人力资源和尊重教育的优良传统,大力培养各个层次的高质量的软件工程人员,使其具有开发各类大型、复杂软件系统的能力。我衷心地预祝清华大学出版社影印和翻译出版这套丛书,在把我国建设成为一个真正现代化的软件产业大国的历史任务中起到推波助澜的作用,并请读者在阅读这些译著时,对这套丛书的选题、译文和编排等方面都提出批评和建议。

周伯生

于北京

2002 年 8 月 18 日

Architecture-Centric Software Project Management

A Practical Guide

Daniel J. Paulish

W Addison-Wesley

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

Architecture-Centric Software Project Management A Practical Guide

Daniel J. Paulish

◆ Addison-Wesley

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



**Carnegie Mellon
Software Engineering Institute**

The SEI Series in Software Engineering

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 with initial capital letters or in all capitals.

CMM, Capability Maturity Model, Capability Maturity Modeling, Carnegie Mellon, CERT, and CERT Coordination Center are registered in the U.S. Patent and Trademark Office.

ATAM; Architecture Tradeoff Analysis Method; CMMI; CMM Integration; CURE; IDEAL; Interim Profile; OCTAVE; Operationally Critical Threat, Asset, and Vulnerability Evaluation; Personal Software Process; PSP; SCAMPI; SCAMPI Lead Assessor; SCE; Team Software Process; and TSP are service marks of Carnegie Mellon University.

ANY MATERIAL FURNISHED BY CARNEGIE MELLON UNIVERSITY AND THE SOFTWARE ENGINEERING INSTITUTE IS FURNISHED ON AN "AS IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

The authors and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

The publisher offers discounts on this book when ordered in quantity for special sales. For more information, please contact:

Pearson Education Corporate Sales Division
One Lake Street
Upper Saddle River, NJ 07458
(800) 382-3419
corpsales@pearsontechgroup.com

Visit Addison-Wesley on the Web: www.aw.com/cseng/

Library of Congress Cataloging-in-Publication Data

Paulish, D. J. (Daniel J.)

Architecture-centric software project management : a practical guide / Daniel J. Paulish.

p. cm.—(SEI series in software engineering)

Includes bibliographical references and index.

ISBN 0-201-73409-5

1. Computer software—Development—Management. 2. Software architecture. I. Title.

II. Series.

QA76.76.D47 P378 2001

005.1'068—dc21

2001051260

Copyright © 2002 Pearson Education, 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 the prior consent of the publisher. Printed in the United States of America. Published simultaneously in Canada.

For information on obtaining permission for use of material from this work, please submit a written request to:

Pearson Education, Inc.
Rights and Contracts Department
75 Arlington Street, Suite 300
Boston, MA 02116
Fax: (617) 848-7047

ISBN 0-201-73409-5

Text printed on recycled paper

1 2 3 4 5 6 7 8 9 10—CRS—0504030201

First printing, December 2001

*To my family,
Ellen, Terry, and Nick,
for their love and support*

Contents

Foreword	iii
Preface	viii
PART ONE	Overview
Chapter 1	Introduction
1.1	What is Project Management?
1.2	What is Software Architecture?
1.3	Core Skills
1.4	Project Management Process
1.5	Architecture-Centered Project Management
1.6	Planning
1.7	Organizing
1.8	Implementing
1.9	Measuring
1.10	Summary
PART TWO	Planning
Chapter 2	Architecture-Centered Software Project Planning
2.1	Developing Realistic Schedules
2.2	Approach

Contents

Foreword	xiii
Preface	xvii
PART ONE Overview	1
Chapter 1 Introduction	3
1.1 What Is Project Management?	3
1.2 What Is Software Architecture?	5
1.3 Core Beliefs	6
1.4 Project Management Process	7
1.5 Architecture-Centric Project Management	7
1.6 Planning	10
1.7 Organizing	12
1.8 Implementing	13
1.9 Measuring	15
1.10 Summary	16
PART TWO Planning	17
Chapter 2 Architecture-Centered Software Project Planning	19
2.1 Developing Realistic Schedules	19
2.2 Approach	21

	2.3	Benefits	39
	2.4	Experience	40
	2.5	Rules of Thumb	41
	2.6	Summary	43
Chapter 3		Global Analysis	45
	3.1	What Is Global Analysis?	46
	3.2	Global Analysis Activities	48
	3.3	Using Global Analysis for Project Planning	56
	3.4	Using Global Analysis for Test Planning	64
	3.5	Benefits	65
Chapter 4		Managing Expectations	67
	4.1	When to Plan and When to Commit	68
	4.2	Managing Upward	70
	4.3	Managing Sideways	72
	4.4	Information Flow	74
	4.5	Using the Software Development Plan	75
	4.6	Summary	77
PART THREE		Organizing	79
Chapter 5		The Project Organization	81
	5.1	Using Software Architecture to Define the Project Organization	81
	5.2	Architecture Team Roles During Development	85
	5.3	Project Functions That Support Development	88
	5.4	Responsibilities, Roles, Authority, and Ownership	93
	5.5	Summary	94
Chapter 6		Global Development	97
	6.1	Why Global Development?	98
	6.2	Architectures for Supporting Global Development	100
	6.3	Development Processes for Global Development	101
	6.4	Multicultural Variables	111
	6.5	Recommendations for Global Development Teams	113
	6.6	Conclusions	116

Chapter 7	Building a Project Culture and Team	119
	7.1 Establishing Project Goals	120
	7.2 Characteristics of Good Teams	121
	7.3 Building a Project Culture	122
	7.4 Building Consensus	129
	7.5 Setting the Amount of Direction	130
	7.6 Summary	133
Chapter 8	The Role of the Software Project Manager	135
	8.1 Creating a Vision	136
	8.2 Coaching	136
	8.3 Making Decisions	137
	8.4 Coordinating	138
	8.5 Working with Your Project Team	139
	8.6 Software Project Management as a Career	142
	8.7 Summary	144
PART FOUR	Implementing	145
Chapter 9	Trade-Offs and Project Decisions	147
	9.1 Using the Project Goals to Make Decisions	148
	9.2 Managing Creeping Functionality and Architecture Drift	148
	9.3 Taking Responsibility	151
	9.4 When to Accept or Reject Changes	152
	9.5 Ethical Decisions of the Project Manager	155
	9.6 Summary	157
Chapter 10	Incremental Development	159
	10.1 Baselining the Software Development Plan	160
	10.2 Build Planning and Management	161
	10.3 Getting Everyone Involved	162
	10.4 Tracking Progress	163
	10.5 Incremental Testing	166

	10.6 Release Criteria Meeting	167
	10.7 Tooling	169
	10.8 Summary	169
Chapter 11	Creating Visibility and Avoiding Surprises	171
	11.1 Risk Management	172
	11.2 Communicating Status and Issues	174
	11.3 Building Credibility with Management	175
	11.4 Recognizing and Celebrating Success	176
	11.5 Summary	178
Chapter 12	Staying Calm in the Heat of Battle	179
	12.1 Cheerleading, Micro-Management, and Discipline ..	180
	12.2 Remaining Optimistic	183
	12.3 Playing the Quality Card	185
	12.4 Providing Support and Removing Obstacles	186
	12.5 Handling Problem Employees	187
	12.6 Emotions and Avoidance	187
	12.7 Quality of Work Life	189
	12.8 Summary	190
PART FIVE	Measuring	191
Chapter 13	Measures to Pay Attention To	193
	13.1 Global Metrics for Project Managers	194
	13.2 Phase Metrics for High-Level Design	199
	13.3 Cost-to-Completes	200
	13.4 Engineering Budgets	202
	13.5 Watching the Test Results	203
	13.6 Summary	204
Chapter 14	What Is a "Good Job"?	207
	14.1 Trading Off Among Schedule, Functionality, and Quality	208
	14.2 Defining Project Success	209
	14.3 Measuring Team Members' Contributions	210