

PROJECT MANAGEMENT FOR ENGINEERING, BUSINESS, AND TECHNOLOGY

JOHN M. NICHOLAS AND
HERMAN STEYN



SEVENTH EDITION

“This book uses a systems view of project management that is very important at these challenging times as projects are being used to deal with systemic problems faced by our precious planet such as sustainable development and climate change that has received extra attention in its sixth edition. The book’s coverage goes beyond engineering, business and technology to the application of projects to a wide variety of applications in a ‘projectified society’. The latest edition clearly points out how the contents of the book align with a variety of knowledge areas and methodologies advocated by prominent peak bodies of project management making it also a valuable resource to project management professionals.”

—**Shankar Sankaran**, University of Technology Sydney, Australia,
and President International Society for the Systems Sciences

“This textbook is a standout resource in the field of project management. It masterfully blends theoretical frameworks with practical application, offering an accessible and insightful guide that is rich with real-world insights. It is an indispensable reference for students and professionals aiming to excel in engineering, business, or technology-driven projects.”

—**Edyta Kostanek**, Lecturer in Project Management and Deputy Programme
Director for BSc Information Management for Business, UCL, UK

“As a professor who has taught project engineering for the last 14 years, I have also performed large-scale project engineering throughout my first career (over 20 years) in aerospace, defense and information technology. When deciding on a textbook for my graduate project engineering class, I looked long and hard. I wasn’t finding what I was looking for and was going to write my own, until I found *Project Management for Engineering, Business, and Technology*. This is the textbook I would have written. It is robust, complete and easy to follow. The graphics, charts and figures are all very descriptive and real . . . I highly recommend this textbook for anyone teaching engineering, business or technology project management/engineering. I also recommend it as a ‘keeper’ for students who will be guiding projects in the future.”

—**Mark Calabrese**, University of Central Florida, USA

“As a postgraduate lecturer, I find *Project Management for Engineering, Business, and Technology* to be a comprehensive and indispensable resource. The numerous and detailed case studies offer relatable, real-world insights that are not only beneficial for students but also intriguing for experienced project managers looking to deepen their understanding. I highly recommend this book for anyone serious about mastering project management in engineering and technology.”

—**Taryn Bond-Barnard**, Stellenbosch University, South Africa

“*Project Management for Engineering, Business, and Technology* is one of the most comprehensive textbooks in the field. Nicholas and Steyn explain the matter in a readable and easy-to-understand way, illustrated with interesting examples. The authors combine the ‘hard matter’ of project management with relevant behavioural aspects. Overall, a useful work for anyone new to the field or as reference for the more advanced project manager.”

—**Martijn Leijten**, Delft University of Technology, The Netherlands

“Project management is essential for successfully achieving project objectives. Projects inherently bring about change, and effective project management is the best way to navigate that change. This book aims to inspire readers to engage with and participate in the evolution of project management and the management of projects.”

—**Benita Zulch**, University of The Free State South Africa

"Project Management for Engineering, Business, and Technology is a very comprehensive textbook that addresses project management across all industries. This textbook has the potential to be used as a tool to close the gap between academia and industry. This is done using real use case studies that reflect real scenarios from the industry. Thus, the textbook is suitable for advanced undergraduate and postgraduate students interested in project management as well as it can be used as a referencing point by practicing project managers across various industries."

—**Ramona Trestian**, Associate Professor, Design Engineering
and Mathematics Department, Middlesex

Project Management for Engineering, Business, and Technology

Project Management for Engineering, Business, and Technology is a highly regarded textbook that addresses project management across all industries. First covering the essential background, from origins and philosophy to methodology, the bulk of the book is dedicated to concepts and techniques for practical application.

The systems development cycle is used as a framework to discuss project management in a variety of situations, making this the go-to book for managing virtually any kind of project, program, or task force. It focuses on the ultimate purpose of project management—to unify and integrate the interests, resources, and work efforts of many stakeholders, as well as the planning, scheduling, and budgeting needed to accomplish overall project goals.

The seventh edition features:

- Updates to cover the latest developments in project management methodologies, including new material on applications of visual management, agile and hybrid methodologies, PM 2.0, and artificial intelligence to project management, and on the “dark side” of projects, projects in developing countries, and megaprojects.
- Sixty-two end-of-chapter case studies that apply concepts and practices from the book to real-life project situations.
- Updated support materials, including an instructor’s manual, PowerPoints, answers to chapter review questions, and a test bank of questions.

Taking a technical yet accessible approach, this book is an ideal resource and reference for all advanced undergraduate and graduate students in project management courses, as well as for practicing project managers across all industry sectors.

John M. Nicholas, PhD, is Emeritus Professor of Operations Management and Information Systems at Loyola University Chicago, Chicago, USA.

Herman Steyn, PhD, is Emeritus Professor of Project Management in the Graduate School of Technology Management, University of Pretoria, Pretoria, South Africa.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Project Management for Engineering, Business, and Technology

SEVENTH EDITION

John M. Nicholas
Loyola University Chicago

Herman Steyn
University of Pretoria

Designed cover image: alexsl/Getty Images®

Seventh edition published 2026

by Routledge

4 Park Square, Milton Park, Abingdon, Oxon, OX14 4RN

and by Routledge

605 Third Avenue, New York, NY 10158

Routledge is an imprint of the Taylor & Francis Group, an informa business

© 2026 John M. Nicholas and Herman Steyn

The right of John M. Nicholas and Herman Steyn to be identified as authors of this work has been asserted in accordance with sections 77 and 78 of the Copyright, Designs and Patents Act 1988.

All rights reserved. No part of this book may be reprinted or reproduced or utilised in any form or by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying and recording, or in any information storage or retrieval system, without permission in writing from the publishers.

Trademark notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation without intent to infringe.

Third edition published by Elsevier Inc. 2008

Fourth edition published by Routledge 2012

Sixth edition published by Routledge 2021

British Library Cataloguing-in-Publication Data

A catalogue record for this book is available from the British Library

ISBN: 978-1-032-66774-4 (hbk)

ISBN: 978-1-032-66588-7 (pbk)

ISBN: 978-1-032-66590-0 (ebk)

DOI: 10.4324/9781032665900

Typeset in Joanna MT

by Apex CoVantage, LLC

Access the Support Material: www.routledge.com/9781032665887

To Sharry, Julia, Joshua, and Abigail
J.M.N.

To Karen and Janine
H.S.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Brief Contents

About the Authors	xxi
Preface	xxiii
Acknowledgments	xxvii
Introduction	1
PART I: PHILOSOPHY AND CONCEPTS	21
1 What Is Project Management?	23
2 Systems Approach	43
PART II: PROJECT LIFE CYCLE	65
3 Project Life Cycle and Project Conception	67
4 Project Definition and System Definition	105
5 Project Execution and Closeout	153
PART III: SYSTEMS AND PROCEDURES FOR PLANNING AND CONTROL	173
6 Basic Project Planning Techniques	175
7 Project Schedule Planning and Networks	228
8 Advanced Project Network Analysis and Scheduling	272
9 Project Costs and Budgeting	317
10 Project Quality Management	351
11 Project Risk Management	384
12 Project Procurement Management and Contracting	429
13 Project Monitoring and Control	456
14 Agile, Lean, and Hybrid Project Management	500
PART IV: ORGANIZATIONAL BEHAVIOR	535
15 Project Organizational Structure and Integration	537
16 Project Roles and Stakeholders	571
17 Leadership, Teamwork, and Conflict	601
PART V: PROJECT MANAGEMENT IN THE CORPORATE CONTEXT	631
18 Project Meta-management and Program Management	633
19 Project Selection and Portfolio Management	671
20 Managing International Projects and Megaprojects	701
Appendix A: Request for Proposal from Midwest Parcel Distribution Company	739
Appendix B: Proposal for Logistical Online System Project (LOGON)	743
Appendix C: Project Execution Plan for Logistical Online System	750
Index	767



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Contents

About the Authors	xxi
Preface	xxiii
Acknowledgments	xxvii
Introduction	1
I.1 In the Beginning . . .	1
I.2 What Is a Project?	3
I.3 All Projects Are Not the Same	6
I.4 Project Management: The Need	8
I.5 Project Goal: Time, Cost, and Performance	9
I.6 Project Management: The Person, the Team, the Methodology	10
I.7 Project Management Standards of Knowledge and Competencies	10
I.8 About this Book	11
I.9 Study Project	12
Appendix: Relation Between Professional Standards and Chapters of this Book	13
Review Questions	18
Case I.1 Denver Airport	18
Questions About the Case	19
Notes	19
PART I: PHILOSOPHY AND CONCEPTS	21
1 What Is Project Management?	23
1.1 Functions of Management	23
1.2 Features of Project Management	24
1.3 Evolution of Project Management	26
1.4 Where Is Project Management Appropriate?	27
1.5 Management by Project: A Common Approach	28
1.6 Different Forms of Project-related Management	30
1.7 New Product and Systems Development Projects	32
1.8 Construction Projects	34
1.9 Service-sector Projects	34
1.10 Public-sector and Governmental Projects and Programs	35
1.11 Miscellaneous Projects	37
1.12 Summary	38
Review Questions	38
Questions About the Study Project	39
Case 1.1 Disaster Recovery at Marshall Field's	39
Case 1.2 Flexible Benefits System Implementation at Shah Alam Medical Center	41
Notes	42

2	Systems Approach	43
2.1	Systems and Systems Thinking	43
2.2	Systems Concepts and Principles	45
2.3	Systems Approach Framework and Systems Models	49
2.4	Systems Engineering	52
2.5	Project Management: A Systems Approach	56
2.6	Summary	57
	Review Questions	57
	Questions About the Study Project	58
	Case 2.1 Glades County Sanitary District	58
	Case 2.2 Jubilee Line Extension Project	60
	Case 2.3 Santa Clara County Traffic Operations System and Signal Coordination Project	61
	Notes	62
PART II:	PROJECT LIFE CYCLE	65
3	Project Life Cycle and Project Conception	67
3.1	Project Life Cycle	67
3.2	Systems Development Cycle	68
3.3	Phase A: Conception	74
3.4	Project Feasibility	76
3.5	The Project Proposal	89
3.6	Contractual Agreement and Negotiation	97
3.7	Contract Statement of Work and Work Requisition	97
3.8	Project Initiation: Variations on a Theme	99
3.9	Summary	99
	Review Questions	100
	Questions About the Study Project	101
	Case 3.1 West Coast University Medical Center	101
	Case 3.2 X-Philes Data Management Corporation: RFP Matters	102
	Case 3.3 Proposal Evaluation for Apollo Spacecraft	103
	Notes	103
4	Project Definition and System Definition	105
4.1	Phase B: Definition	105
4.2	Project Definition	108
4.3	Phased (Rolling Wave) Project Planning	110
4.4	System Definition	115
4.5	Alternate Approaches to Definition and Execution	122
4.6	Project Communication	125
4.7	Summary	127
	Appendix A: Stages of Systems Engineering	128
	Appendix B: Quality Function Deployment	143
	Review Questions	146
	Questions About the Study Project	147
	Case 4.1 Star-Board Construction and Santaro Associates: Requirements Snafu	147
	Case 4.2 Revcon Products and Welbar, Inc.: Client–Contractor Communication	148
	Case 4.3 Lavasoft.com: Interpreting Customer Requirements	149
	Case 4.4 Proposed Gold Mine in Canada: Phased Project Planning	150

	Refer to Case 2.3, Santa Clara County Traffic Operations System and Signal Coordination Project	152
	Notes	152
5	Project Execution and Closeout	153
	5.1 Phase C: Execution	153
	5.2 Detail Design Stage	154
	5.3 Production/Build Stage	157
	5.4 Implementation Stage	158
	5.5 Project Termination and Closeout	161
	5.6 Project Summary Evaluation	164
	5.7 After the Project—Phase D: Operation	166
	5.8 Ethics and the Dark Side of Projects	167
	5.9 Summary	169
	Review Questions	170
	Questions About the Study Project	171
	Case 5.1 SLU Information Central Building	171
	Notes	172
PART III:	SYSTEMS AND PROCEDURES FOR PLANNING AND CONTROL	173
6	Basic Project Planning Techniques	175
	6.1 Planning Process	175
	6.2 Project Execution Plan	176
	6.3 Scope and Statement of Work	179
	6.4 Work Definition	181
	6.5 Project Organization and Responsibilities	190
	6.6 Estimating and the Systems Development Cycle	190
	6.7 Estimating Process	194
	6.8 Estimating Methods	196
	6.9 Scheduling	203
	6.10 Planning and Scheduling Charts	205
	6.11 Summary	210
	Appendix: Information Management and Artificial Intelligence	210
	Review Questions	219
	Questions About the Study Project	221
	Case 6.1 Barrage Construction Company: Sean's WBS	222
	Case 6.2 Startrek Enterprises, Inc.: Deva's Project Plan	223
	Case 6.3 Walter's Project Plan	224
	Case 6.4 Planning the Boca Implementation at Kulczyński Products	225
	Notes	227
7	Project Schedule Planning and Networks	228
	7.1 Network Diagrams	228
	7.2 The Critical Path	233
	7.3 Converting to Gantt Calendar Schedules	241
	7.4 Schedule Reserve	242
	7.5 Alternative Relationships and PDM	242
	7.6 Scheduling with Resource Constraints	247
	7.7 Criticisms of Network Methods	255

7.8	Summary	256
	Appendix A: AOA Diagrams	257
	Appendix B: Alternate Scheduling Method: Project Starts at Day 1	259
	Review Questions and Problems	260
	Questions About the Study Project	265
	Case 7.1 Network Diagram for a Large Construction Project	266
	Case 7.2 Melbourne Construction Company, A	268
	Case 7.3 Melbourne Construction Company, B	269
	Case 7.4 Melbourne Construction Company, C	270
	Notes	271
8	Advanced Project Network Analysis and Scheduling	272
8.1	Reducing Project Duration with CPM	273
8.2	Variability of Activity Duration	280
8.3	PERT	282
8.4	Allocating Resources and Multiple Project Scheduling	292
8.5	Theory of Constraints and Critical Chain Method	294
8.6	TOC Method for Allocating Resources to Multiple Projects	302
8.7	Discussion and Summary	304
	Summary List of Variables	305
	Review Questions and Problems	306
	Questions About the Study Project	310
	Case 8.1 Bridgecon Construction	310
	Case 8.2 LOGON Project	313
	Case 8.3 Papua Petera Village Project	314
	Notes	315
9	Project Costs and Budgeting	317
9.1	Cost Estimates	317
9.2	Cost Escalation	318
9.3	Elements of Estimates and Budgets	323
9.4	Project Cost Accounting System	327
9.5	Budgeting Using Control (or Cost) Accounts	328
9.6	Cost Summaries	329
9.7	Cost Schedules and Forecasts	330
9.8	Life Cycle Costs	341
9.9	Summary	343
	Review Questions and Problems	344
	Questions About the Study Project	346
	Case 9.1 Life Cycle Costs for Fleet of Tourist Spaceships	347
	Case 9.2 Estimated Costs for the Channel Tunnel Project	348
	Case 9.3 Fiona's Estimate for the Gorgy Project	348
	Case 9.4 Melbourne Construction Company, D	349
	Notes	350
10	Project Quality Management	351
10.1	The Concept of Quality	351
10.2	Project Quality Management Processes	356
10.3	Techniques for Quality Assurance in System Development	360

10.4	Techniques for Quality Control	371
10.5	Summary	375
	Review Questions	375
	Questions About the Study Project	376
	Case 10.1 Ceiling Panel Collapse in the Big Dig Project	376
	Case 10.2 FIFA 2010 World Cup South Africa	378
	Case 10.3 Airbag Adversity	381
	Notes	382
11	Project Risk Management	384
11.1	Risk Concepts	384
11.2	Risk Identification	385
11.3	Risk Assessment	391
11.4	Risk Response Planning	402
11.5	Risk Monitoring and Response	407
11.6	Project Management <i>Is</i> Risk Management	408
11.7	Summary	412
	Appendix: Risk Analysis Methods	412
	Review Questions and Problems	418
	Questions About the Study Project	421
	Case 11.1 The Sydney Opera House	422
	Case 11.2 Infinity & Beyond, Inc.	423
	Case 11.3 The Nelson Mandela Bridge	424
	Case 11.4 Sunrise Beach Development	426
	Notes	427
12	Project Procurement Management and Contracting	429
12.1	Procurement and Procurement Management	429
12.2	Define and Plan the Procurement	431
12.3	Conduct the Procurement	432
12.4	Control and Closeout the Procurement	436
12.5	Contracting	438
12.6	Kinds of Contracts	441
12.7	Fixed-price Contracts	442
12.8	Cost-reimbursable Contracts	445
12.9	Contract-related Matters	448
12.10	Summary	450
	Review Questions and Problems	451
	Questions About the Study Project	452
	Case 12.1 Contract Mess-up at Polanski Developers	452
	Case 12.2 Consolidated Energy Company	453
	Case 12.3 Formal and Informal Agreements in Contractual Matters	454
	Notes	454
13	Project Monitoring and Control	456
13.1	Monitoring and Control Process	456
13.2	Project Monitoring	458
13.3	Control Emphasis	460
13.4	Work Packages and Control Accounts	467

13.5	Performance Analysis and Earned Value Management (EVM)	470
13.6	Visual Management: Kanban Board and Burn Chart	482
13.7	Issue Management	484
13.8	Change Control	485
13.9	Problems with Monitoring and Controlling Projects	488
13.10	Summary	490
	Summary of Variables	490
	Review Questions and Problems	491
	Questions About the Study Project	495
	Case 13.1 Cybersonic Project	495
	Case 13.2 SA Gold Mine: Earned Value after a Scope Change	497
	Case 13.3 Change Control Process at Dynacom Company	498
	Case 13.4 Status Report for the LOGON Project	499
	Notes	499
14	Agile, Lean, and Hybrid Project Management	500
14.1	Traditional Project Management	501
14.2	Agile Project Management	502
14.3	Scrum	505
14.4	APM Controversy	513
14.5	Lean Project Management	515
14.6	Kanban Project Management	522
14.7	Hybrid Methodologies	526
14.8	Summary	526
	Review Questions	527
	Questions About the Study Project	528
	Case 14.1 Grand Entry for Accent, Inc.	529
	Case 14.2 Technology to Track Stolen Vehicles	531
	Notes	532
PART IV:	ORGANIZATIONAL BEHAVIOR	535
15	Project Organizational Structure and Integration	537
15.1	Formal Organizational Structure	537
15.2	Organizational Design by Differentiation and Integration	538
15.3	Requirements of Project Organizations	540
15.4	Integration of Subunits in Projects	541
15.5	Liaison Roles, Task Forces, and Teams	541
15.6	Project Expeditors and Coordinators	543
15.7	Pure Project Organizations	544
15.8	Matrix Organizations	547
15.9	Selecting an Organizational Form for Projects	550
15.10	Project Office and PMO	552
15.11	Integration of Large-scale Projects	554
15.12	Integration in Systems Development Projects	558
15.13	Concurrent Engineering and DevOps	560
15.14	Summary	563
	Review Questions	564
	Questions About the Study Project	565

Case 15.1 Organization for the LOGON Project	566
Case 15.2 Pinhole Camera and Optics, Inc.: Why Do We Need a Project Manager?	567
Case 15.3 Implementing a Matrix Structure in an R&D Laboratory	567
Notes	569
16 Project Roles and Stakeholders	571
16.1 The Project Manager	572
16.2 Project Management Authority	576
16.3 Project Manager Qualifications	579
16.4 Filling the Project Management Role	584
16.5 Other Management Roles in the Project Team	585
16.6 Project Stakeholders	589
16.7 Stakeholder Engagement	589
16.8 Stakeholder Engagement Plan	594
16.9 Summary	594
Review Questions	595
Questions About the Study Project	596
Case 16.1 The LOGON Project	597
Case 16.2 Selecting a Project Manager at Nuwave Products Company	598
Case 16.3 Stakeholders in Boston's Big Dig	598
Notes	599
17 Leadership, Teamwork, and Conflict	601
17.1 Leadership in Project Management	602
17.2 Participative Management and Shared Leadership	604
17.3 Project Teams	605
17.4 Team Building	607
17.5 Improving Ongoing Work Teams	609
17.6 Building New Teams	610
17.7 Intergroup Problem Solving	612
17.8 Virtual Teams	613
17.9 Conflict	616
17.10 Conflict Management	619
17.11 Change Management	621
17.12 Managing Emotional Stress	622
17.13 Summary	624
Review Questions	625
Questions About the Study Project	626
Case 17.1 Wilma Keith	627
Case 17.2 Mars Climate Orbiter Spacecraft	628
Notes	629
PART V: PROJECT MANAGEMENT IN THE CORPORATE CONTEXT	631
18 Project Meta-management and Program Management	633
18.1 Project Management Maturity and Maturity Models	633
18.2 Project Management Methodology	636
18.3 Managing Project Knowledge	642
18.4 Project Management Office	647
18.5 Program Management	651

18.6	Program Life Cycle	653
18.7	Program Management Themes	654
18.8	Program Organization	655
18.9	Special Considerations	657
18.10	Summary	659
	Review Questions	660
	Questions About the Study Project	661
	Case 18.1 Maxim Corporation America (MCA)	661
	Case 18.2 Motorola's M-Gate Methodology and the RAZR Project	663
	Case 18.3 Tecknokrat Company	665
	Case 18.4 Mercury Exploration Program	666
	Notes	668
19	Project Selection and Portfolio Management	671
19.1	Project Portfolio Management	671
19.2	Framework for Project Selection and Portfolio Management	675
19.3	Methods for Assessing Individual Projects	677
19.4	Methods for Comparing and Selecting Projects	682
19.5	Integrating the Gating Process and Portfolio Management	692
19.6	Summary and Discussion	693
	Review Questions and Problems	693
	Question About the Study Project	695
	Case 19.1 Proposed Cement Factory for PCS Company	696
	Case 19.2 Selecting Projects at Galactic Mining, A	696
	Case 19.3 Selecting Projects at Galactic Mining, B	697
	Notes	699
20	Managing International Projects and Megaprojects	701
20.1	International Projects	701
20.2	Problems Managing International Projects	702
20.3	Culture and Local Customs and Institutions	702
20.4	Local Stakeholders	707
20.5	Financial and Geographic Issues	708
20.6	Project Manager	710
20.7	Local Representative	711
20.8	Steering Committee and PMO	712
20.9	Team and Relationship Building	713
20.10	Project Definition	713
20.11	Project Monitoring	717
20.12	Communication	717
20.13	Risks	718
20.14	Project Failure in Developing Nations	719
20.15	Megaprojects	720
20.16	Megaproject Conception, Planning, and Organization	722
20.17	Megaproject Stakeholders	723
20.18	Megaproject Leadership	724
20.19	Shortening the Megaproject Life Cycle	728
20.20	Summary	729
	Review Questions	730

Questions About the Study Project	732
Case 20.1 Mozal Project: International Investment in an Undeveloped Country	732
Case 20.2 Spirit Electronics' Puerto Rico Office	736
Notes	737
Appendix A: Request for Proposal from Midwest Parcel Distribution Company	739
Appendix B: Proposal for Logistical Online System Project (LOGON)	743
Appendix C: Project Execution Plan for Logistical Online System	750
Index	767



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

About the Authors

JOHN M. NICHOLAS is Emeritus Professor of Operations Management and Project Management in the Quinlan School of Business at Loyola University Chicago. He is a teacher, writer, and researcher in project management and operations management, and has conducted executive seminars and consultation on project management and process improvement. John is the author of numerous academic and technical publications, and five books including *Lean Production for Competitive Advantage* (2018). Prior to academia, he held the positions of team lead and engineer on aircraft development projects at Lockheed-Martin Corporation, project lead on operations projects at Bank America, and researcher on energy-environmental research projects at Argonne National Laboratory. He has a BS degree in aerospace engineering and an MBA in operations research from the University of Illinois, Urbana-Champaign, and a PhD in industrial engineering and applied behavioral science from Northwestern University.

HERMAN STEYN is Emeritus Professor of Project Management in the Graduate School of Technology Management, University of Pretoria, South Africa. He has been involved in projects in industry since 1975, has managed a variety of large and small engineering projects (system, product, and process development) in the minerals, defense, and nuclear industries, and has also managed programs and project portfolios. In 1996 he was appointed as professor at the University of Pretoria where he initiated a masters' program in project management. Besides supervising project management research and teaching graduate project management courses, Herman has conducted more than 200 seminars and workshops on project management. He has a bachelor's degree and a graduate diploma in metallurgical engineering, an MBA, and a PhD in engineering management.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Preface

When people see or use something impressive—a bridge arching high over a canyon, a space probe touching down on a distant planet, an animated game so realistic you think you’re there, or a nifty phone/camera/computer the size of your hand—they maybe wonder, “How did they do that?” By *they*, of course, they are referring to the creators, designers, and builders, the people who thought up and made those things. Seldom do they wonder about the *leaders* and *managers*, the people who organized and led the efforts that brought those wondrous things from concept to reality and without whom most neat ideas would never have been achieved. This book is about them—the managers of projects, the mostly unsung heroes of engineering, business, and technology who stand outside the public eye but are responsible for practically everything that requires collective human effort.

The project manager is but one of many people involved in the creation of society’s products, systems, and artifacts, yet it is he or she who gets the others involved and organizes and directs their efforts so everything comes out right. Occasionally, the manager and the creator happen to be the same: Burt Rutan, Clint Eastwood, and Gutzon Borglum are examples; their life work—in aerospace, motion pictures, and monumental sculptures, respectively—represent not only creative or technological genius, but leadership and managerial talent as well.

In the last several decades businesses have expanded from domestic, nationalistic enterprises and markets into multinational, global enterprises and markets. As a result, from a business perspective there is more of everything to contend with—more ideas, competitors, resources, constraints, and, certainly, more people doing and wanting things. Technology is advancing and products and processes evolving at a more rapid pace; as a result, the life cycles of most things in society are getting shorter. This “more of everything” has had a direct impact on the conduct of projects—including projects to develop products, systems, or processes that compete in local, domestic, and international markets; projects to create and implement new ways of meeting demand for energy, recreation, housing, communication, transportation, and food; and projects to answer basic questions in science and resolve grave problems such as disease, climate change, and the aftermath of natural disasters. All of this project activity has spurred a growing interest in improved ways to plan, organize, and guide projects to better meet the needs of customers, markets, and society within the bounds of limited time and resources.

Associated with this interest is the growing need to educate and train project managers. In the past—and still today—project managers were chosen for some demonstrated exceptional capability, although not necessarily managerial. If you were a good engineer, systems analyst, researcher, architect, or accountant, eventually you would become a project manager. Somewhere along the way, presumably, you would pick up the necessary skills to manage projects. The flaw in this reasoning is that project management encompasses a broad range of skills—managerial, leadership, interpersonal—that are much different and independent of skills associated with technical competency. And there is no reason to presume that the project environment alone will provide the opportunity for someone to “pick up” those necessary skills.

As a text and handbook, this book is about the “right” way to manage projects. It is intended for advanced undergraduate and graduate university students and practicing managers in engineering, business, and technology. It is a book about principles *and* practice, meaning that the topics in it are practical and meant to be applied. It covers the big picture of project management—origins, applications, and philosophy, as well as the nitty-gritty, how-to steps.

Why a book on project management in engineering *and* business *and* technology? In our experience, engineers, programmers, architects, scientists, and other technology specialists—those involved in “engineering/technology” projects often have little management or leadership training. This book, which includes many engineering and technology examples, provides somewhat broad exposure to business concepts and management specifics to help these specialists get started as managers and leaders.

What about those people involved in product-development, marketing, process-improvement, and related projects commonly thought of as “business” projects? Just as technology specialists seldom receive formal management training, students and practitioners of business rarely get formal exposure to practices common in technology projects. For them, this book describes not only how “business” projects are conducted, but also the necessary steps in the conception and execution of engineering, system development, construction, and other “technology” projects. Of course, every technology project is *also* a business project: it is conducted in a business context and involves business issues such as customer satisfaction, resource utilization, deadlines, costs, and profits.

Virtually all projects—engineering, technology, and business—originate and are conducted in a similar way, in this book conceptualized using a methodology called the systems development cycle (SDC). The SDC serves as a general framework for discussing the principles and practices of project management and illustrating commonalities and differences among a wide range of projects.

This book is an outgrowth of the authors’ combined several decades of experience teaching project management at Loyola University Chicago and University of Pretoria to business and engineering students, preceded by several years’ experience in business and technology projects, including for aircraft design and flight test, large-scale process facility construction, and software applications development and process improvement. This practical experience gave us an appreciation not only for the business-management side of project management, but also for the human-interpersonal side as well. We have seen the benefits of good communication, trust, and teamwork, as well as the costs of poor leadership, emotional stress, and group conflict. In our experience, the most successful projects are those where leadership, trust, communication, and teamwork flourished, regardless of the formal planning and control methods and systems in place. This book largely reflects these personal experiences. Of course, comprehensive coverage of project management required that we look much beyond our own experience and draw upon the published works of many others and the wisdom and suggestions of colleagues and reviewers.

In this seventh edition we have revised and added material to incorporate new topics of interest, current examples, and the growing body of literature in project management. Among significant changes are new material on project management applications of visual management and agile and hybrid methodologies (Chapters 13 and 14), and PM 2.0 and artificial intelligence (Chapter 6). Coverage has been expanded of the role of project management in “green projects” (Chapter 3), stakeholder engagement (Chapter 16) and change management (Chapter 17). Also new is material on ethics, the “dark side” of projects, projects in developing countries (Chapter 5), and managing megaprojects (Chapter 20). The Introduction includes updated tables that relate sections of the book to the project management knowledge areas and methodologies of PMI-PMBOK, IPMA, APM, and PRINCE2. Books tend to grow in size with each new edition; to combat that all chapters have been rewritten to make everything

more readable and concise. Despite the inclusion of new material, we've held the page count to roughly the same as it was in the previous edition.

Our goal in writing this book is to provide students and practicing managers with the most practical, current, and interesting text possible. We appreciate hearing your comments and suggestions. Please send them to us.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Acknowledgments

Like most projects, writing a book reflects the contributions of many people. We want to acknowledge and give special thanks to those who contributed the most to the book's seven editions. First, thanks to our research assistants. Research assistants in general do a lot of work—academic as well as gofer, and without their toiling efforts most professors would accomplish far less. We were fortunate to have had the assistance of several such bright and capable people, particularly Thomas Slomiany, Dominick Fuentes, Elisa Denney, Hollyce James, Diane Petrozzo, Miguel Velasco, Gaurav Monga, Cary Morgan, Louis Schwartzman, and Brian Whelan.

Special thanks to current and former colleagues at Loyola University Chicago and the University of Pretoria. In Chicago, thanks to Drs. Gezinus Hidding, Enrique Venta, Harold Dyck, Samuel Ramenofsky, and Donald Meyer, and Elaine Strnad, Paul Flugel, John Edison, Sharon Tylus, and Debbie Gillespie for their suggestions and support for earlier editions. In Pretoria, thanks to Dr. Tinus Pretorius and Dr. Elma van der Lingen for encouraging education and research in project management at the Graduate School of Technology Management and for supporting the work on this book. I (Herman) also want to express appreciation to Drs. Taryn Bond-Barnard, Corro van Waveren, Suzaan Pretorius, Giel Bekker, Clive Enoch, Pieter Pretorius, Lavagnon Ika, Leon Pretorius, Marie-Louise Barry, Philip Viljoen, Krige Visser, and Michael Carruthers for their direct and indirect contributions to this book and for all I have learned from them.

I (John) want to acknowledge the influence of three of my professors, Dr. Charles Thompson and Dr. Gustave Rath at Northwestern University, and Dr. Dick Evans at the University of Illinois, whose philosophies and teachings helped shaped this book. I also want to thank Chris Phares and Bob Zimmerman, project managers extraordinaire, for sharing their wisdom on the meaning and significance of project leadership.

Special thanks also to our wives Sharry and Karen. Sharry provided numerous suggestions to the first edition that helped reduce the amount of “techno-jargon” in the book, and managed the home front so that I (John) could pursue and complete this project. Karen provided wifely support and encouragement; as in the case of so many other projects I (Herman) have been involved in, my contribution to this project would never have materialized had it not been for her support.

Thanks also to Ruth Bourne for the fine copyediting work, and the folks at Routledge and Taylor and Francis for their ongoing support throughout preparation of this seventh edition.

Other colleagues, students, and friends, some mentioned in the endnotes throughout the book, provided support, encouragement, and reference materials; to them also we say thank you. Despite the assistance of so many people and our own best efforts, there are still likely to be omissions or errors. We had final say and accept responsibility for them.

John M. Nicholas
Herman Steyn



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Introduction

I.1 In the Beginning . . .

Sometime during the third millennium BC, workers on the Great Pyramid of Khufu set the last stone in place. They must have felt jubilant, for this event represented a milestone of sorts in one of humanity's grandest undertakings. Although much of the ancient Egyptians' technology is still a mystery, the enormity and quality of the finished product remains a marvel. Despite the lack of sophisticated machinery, Egyptian workers were able to raise and fit some 2,300,000 stone blocks, weighing 2 to 70 tons apiece, into a structure the height of a modern 40-story building. Each facing stone was set against the next with an accuracy of 0.04 inch (1 mm), and the base, which covers 13 acres (52,600 m²), deviates less than 1 inch (25 mm) from level (Figure I.1).¹

Equally as staggering was the number of workers involved. To quarry and transport stones down the Nile, 20,000–30,000 laborers were levied. In addition, skilled masons and attendants were employed in preparing and laying the blocks and erecting and dismantling the construction ramps. Public works were essential to keep the working population employed and fed, and it is estimated that at least 150,000 women and children also had to be housed and fed.² But just as mind-boggling was the managerial ability exercised by the Egyptians throughout the estimated 20-year duration of the pyramid construction. Francis Barber, a nineteenth-century pyramid scholar, concluded that:

*It must have taken the organizational capacity of a genius to plan all the work, to lay it out, to provide for emergencies and accidents, to see that the men in the quarries, on the boats and sleds, and in the mason's and smithies shops were all continuously and usefully employed, that the means of transportation was ample . . . that the water supply was ample . . . and that the sick reliefs were on hand.*³

Some have suggested the pyramid was built by slaves, but research indicates that such a massive undertaking could only have been accomplished by a highly skilled, motivated, and well-fed workforce, eager to participate in an historic and honorable endeavor.

Building the Great Pyramid is what we today would call a large-scale project. It stands among numerous projects from early recorded history that required massive human works,



Figure I.1

The Great Pyramid of Khufu, center back, an early (circa 2500 BC) large-scale project.

Source: Sculpies/iStock/Getty Images ®.

complex and thorough planning, and managerial competency. For the Great Pyramid's construction, we know that the pharaoh Khufu chose his vizier, Hemunu, to be the project manager.

Also worthy of note are the managerial and leadership accomplishments of Moses. The Biblical account of the exodus of the Hebrews from the bondage of the Egyptians gives some perspective on the preparation, organization, and execution of this tremendous undertaking. Supposedly Moses did a magnificent job of personnel selection, training, organization, and delegation of authority.⁴ The famed ruler Solomon was also the “manager” of great projects. He transformed the battered ruins of many ancient cities and crude shantytowns into powerful fortifications. With his wealth and the help of Phoenician artisans, he built the Temple in Jerusalem. Seven years went into the construction of the Temple, after which Solomon took 13 years more to build a palace for himself. He employed a workforce of 30,000 Israelites to fell trees and import timber from the forests of Lebanon.⁵ That was almost 3,000 years ago.

With later civilizations, notably the Greeks and Romans, projects requiring extensive planning and organizing escalated. To facilitate their military campaigns and commercial interests, the Romans constructed networks of highways and roads throughout Europe, Asia Minor, Palestine, and northern Africa so that all roads would “lead to Rome.” The civilizations of Renaissance Europe and the Middle and Far East undertook river engineering, construction of aqueducts, canals, dams, locks, and port and harbor facilities. With the spread of modern religions, construction of temples, monasteries, mosques, and massive urban cathedrals was added to the list.

With the advent of industrialization and electricity, projects for the construction of railroads, electrical and hydro-electrical power facilities and infrastructures, subways, and factories became commonplace. Recently, development of large systems for communications, defense, transportation, research, and information technology have spurred more complex kinds of projects.

As long as humans are around, there will be projects. Many projects of the future will be like those in the past. Others will be different either in terms of increased scale of effort or more advanced technology. Representative of the latter are the English Channel Tunnel and the International Space Station (ISS). The Channel Tunnel required tremendous resources and took a decade to complete. The ISS (Figure 1.2) required development of new technologies and the efforts of the US, Russian, European, Canadian, and Japanese space agencies.

1.2 What Is a Project?

From these examples it is clear that humankind has been involved in project activities for a long time. But why are these considered “projects” while other activities like planting and harvesting a crop, stocking a warehouse, issuing payroll checks, or manufacturing a product, are not?

What is a project? This is a question we will cover in much detail later. As an introduction though, below are listed some characteristics that warrant classifying an activity as a project.⁶

1. A project is **goal-oriented**—it has a purpose with well-defined deliverables, products, or outcomes (called **end-items**) meant to provide specific benefits. Fundamentally, it provides something of **value**.
2. It is **unique**; it requires doing something differently than was done previously. It is a one-time activity, never repeated.
3. It is a **temporary** organization formed to accomplish the goal in a limited time frame.
4. It is **cross-functional**—it utilizes people and resources from different organizations and functions.
5. Given that each project is unique, it carries some **unfamiliarity** and **risk**.

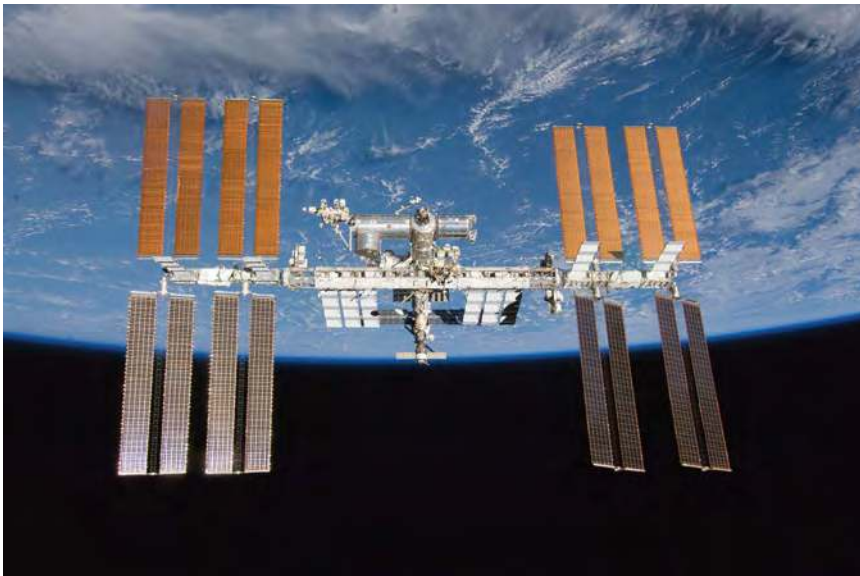


Figure 1.2
The International Space Station, a modern large-scale project.
Source: Photo courtesy of NASA.

The examples described are for familiar kinds of projects such as construction (pyramids) and technology development (space station). In general, the list of activities that qualify as projects is long and includes many that are commonplace. Weddings, remodeling a home, and moving to another house are projects; so are company audits, major litigations, corporate relocations, and motion picture productions; and so are efforts to develop new products and implement new systems. Military campaigns also qualify as projects; they are temporary, unique, risky efforts directed toward a goal. The Normandy invasion in World War II on June 6, 1944, is an example:

The technical ingenuity and organizational skill that made the landings possible was staggering. The invasion armada included nearly 5,000 ships of all descriptions protected by another 900 warships. The plan called for landing 150,000 troops and 1,500 tanks on the Normandy coast *in the first 48 hours*.⁷

Most artistic endeavors are projects, too. Composing a song or symphony, writing a novel, or making a sculpture are one-person projects. Some artistic projects also require the skills of engineers and builders, for example Mount Rushmore, the Statue of Liberty, and the Eiffel Tower.

Many efforts at saving human life and recovering from man-made or natural disasters become projects. Examples are the massive cleanup following the Soviet nuclear accident at Chernobyl, and rescue and recovery operations following disastrous earthquakes throughout the world, the Indian Ocean tsunami of 2004, and the Ebola outbreak in western Africa in 2014. The COVID-19 pandemic spurred projects around the world to discover and distribute a vaccine. Ongoing efforts to stem climate change and mitigate its global effects will of necessity spur innumerable projects.

Figure I.3 shows diverse project endeavors and examples of well-known projects, and where the projects fall with respect to complexity and uncertainty. Complexity is measured by the magnitude of the effort—the number of groups and organizations involved and the diversity of expertise needed to accomplish the work. Time and resource commitments tend to increase with complexity.

Uncertainty is measured roughly by the difficulty in predicting the final outcome in terms of the dimensions of time, cost, and technical performance. In most projects there is some uncertainty in one or two dimensions (e.g. weddings); in complex projects there is uncertainty in all three (e.g. the space station).

Generally, the more often something is done, the less uncertainty there is in doing it. This is simply because people learn by doing and so improve their efforts—the “learning curve” concept. Projects that are very similar to previous ones and about which there is abundant knowledge have lower uncertainty. These are found in the lower portion of Figure I.3 (e.g. weddings, highways, dams, system implementation). Projects with high uncertainty are in the upper portion of the figure.

When the uncertainty of a project drops to nearly zero, and when the project effort is repeated many times, then the work is usually no longer considered a project. For example, building a skyscraper is a project, but mass construction of prefabricated homes more closely resembles a scheduled, repetitive operation than a project. The first flight to the South Pole by Admiral Byrd was a project, but modern daily supply flights to bases there are not. Early human missions to Mars will be projects, but future chartered tourist trips to hotels and excursions on Mars will not be. They will just be run-of-the-mill scheduled operations.

The cost curve in Figure I.3 indicates that a project’s expense tends to increase in proportion to its complexity and uncertainty. Cost, represented in terms of time or economic value,

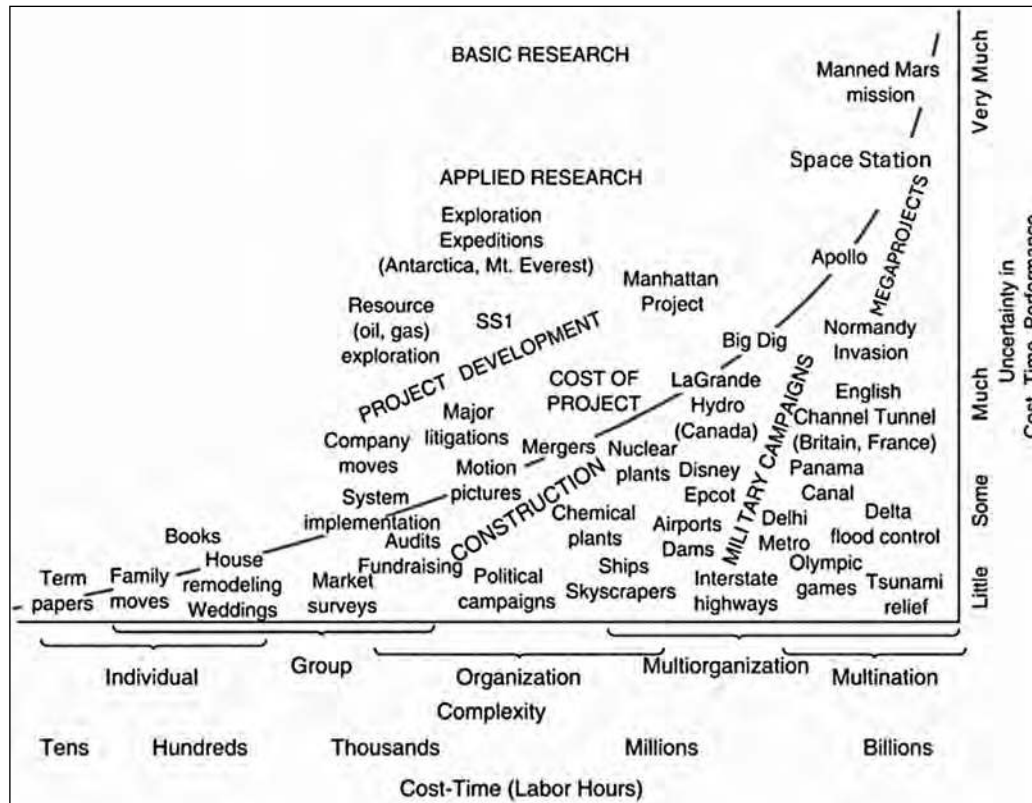


Figure I.3
A typology of projects.

is at the level of tens or hundreds of labor hours for projects with low complexity and uncertainty, but increases to millions and billions of hours for projects with the greatest complexity and uncertainty.

In all cases, projects are conducted by organizations that after the project is completed go on to do something else (construction companies) or are disbanded (Admiral Byrd's crew, a Mars exploration team). In contrast, repetitive, high-certainty activities (prefabricated housing, supply flights, tourist trips to Antarctica or Mars) are performed by permanent organizations that do the same thing repeatedly, with little changes in operations other than scheduling. Because projects are not repetitive, they must be managed differently.

I.3 All Projects Are Not the Same⁸

Another way, besides Figure I.3, to illustrate the diversity in projects is with the so-called NTCP model, which classifies projects and their end-items or products into four dimensions, each with three or four possible levels. The dimensions and levels are:

- **Novelty:** How new the project end-item or product is to customers and users and how well defined its initial product requirements are. This dimension has three levels:
 - **Derivative**—the project end-item or product is an extension or improvement of an existing product or system, e.g. new features added to an existing car model.
 - **Platform**—the end-item or product is a new generation of an existing product line in a well-established market, e.g. a new car model.
 - **Breakthrough** (*new to market (NTM) or new to the world (NTW)*)—the product or end-item is brand new, e.g. first-generation electric vehicles (NTM) or the Space Shuttle (NTW).
- **Technology:** The project's technological uncertainty and whether it is new or mature; this addresses the question of how much new technology is required to create, build, manufacture, and enable the use of the product and how much technical competency is needed by the project manager and the team. It has four levels:
 - **Low-tech**—involves only well-established technologies.
 - **Medium-tech**—uses mainly existing technologies, but also limited use of some new technology or new features, e.g. designing automobiles and appliances.
 - **High-tech**—uses technologies that are mostly new to the firm but already exist and are available at project initiation; typical of many defense and computer projects; is synonymous with "high-risk."
 - **Super-high-tech**—relies on new technologies that do not exist at project initiation. The project goal is well defined, but the solution is not; is synonymous with "very high-risk," e.g. landing humans on the moon.
- **Complexity:** The complexity of the product and the project organization; it has three levels:
 - **Assembly**—the project involves combining a collection of elements, components, and modules into a single unit or entity that performs a single function, e.g. creating a new coffee maker or software to manage a single function (such as payroll).
 - **System**—involves a complex collection of interactive elements and subsystems that jointly perform multiple functions to meet specific operational needs, e.g. developing a new car, new computer, or entirely new business.

- **Array**—the project involves a large variety of dispersed systems (a system of systems, or “super system”) that function together to achieve a common purpose, e.g. creating a national communications network, mass transit infrastructure, regional power generation and distribution network, or an entire corporation.
- **Pace**: The time available for the project—the urgency or criticality of meeting the project’s completion targets; it has four levels:
 - **Regular**—no urgency; time is not critical to immediate success.
 - **Fast/competitive**—complete project in adequate time to address market opportunities, create a strategic positioning, or form a new business unit, e.g. launching a new drug or introducing a new product line.
 - **Time-critical**—complete project by a specific deadline; missing the deadline means project failure, e.g. construction of facilities for the Olympic Games; launching a space probe to a comet.
 - **Blitz**—a crisis project; the criterion for success is solving a problem as fast as possible, e.g. rescuing the survivors of a tsunami.

All projects can be characterized according to the four dimensions. In Figure I.4, each of the dimensions is represented by a quadrant on the graph. The diamond-shaped profiles show the four dimensions for two examples, the Apollo lunar program and the space shuttle program.

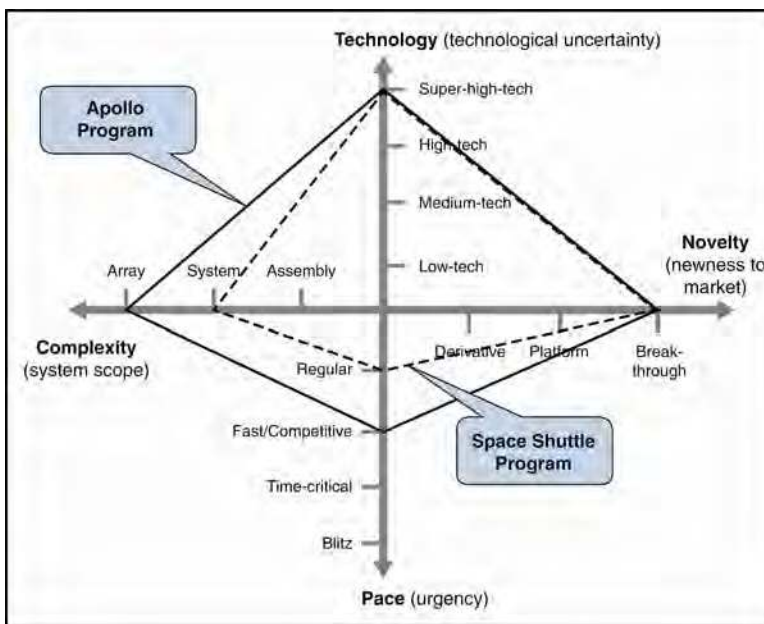


Figure I.4

Shenhar & Dvir’s NTCP Diamond model contrasting the Apollo and space shuttle programs.

Source: Shenhar A. and Dvir D. *Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation*. Cambridge, MA: Harvard Business School Press, 2007.

1.4 Project Management: The Need

Although mankind has been involved in projects since the beginning of recorded history, obviously the nature of projects and the environment have changed. Many modern projects involve technical complexity and challenges in terms of assembling and directing large temporary organizations while subject to constrained resources, limited time schedules, and environmental uncertainty. An example is the NASA Pathfinder Mission to land and operate a rover vehicle on the surface of Mars. Such a project was unparalleled not only in terms of technical difficulty and organizational complexity, but also for the requirements imposed on it. In ancient times, requirements were more flexible. When Renaissance builders ran out of funds during construction of a cathedral, they stopped the work until more funds could be raised (one reason why cathedrals took decades or centuries to complete). When a king ran out of money while building a fortress or palace, he could levy higher taxes. In cases where additional money or workers could not be found or the project not delayed, then the scale of effort or quality of workmanship was reduced.

More commonly, however, project requirements are not flexible. Khufu's Great Pyramid had to be completed before the pharaoh died to serve as his tomb and portal to the afterlife, and tens of thousands of skilled artisans and laborers were recruited so as to meet that "deadline" (pun intended). The Mars Pathfinder project was challenged with developing and landing a vehicle on Mars in less than 3 years' time and on a \$150 million budget—less than half the time and 1/20th the cost of the last probe NASA had landed on Mars. The project involved advanced research and explored new areas of science and engineering. Technical performance requirements could not be compromised.

Beyond large-scale engineering efforts, constraints and uncertainty are common in everyday business and technology projects where organizations strive to develop and implement new products, processes, and systems, and to adapt to changing requirements in a changing world. Consider Dalian Company's development of "Product J," a project the company undertook as part of its efforts to remain competitive. Product J was a promising but radically new idea. To move that idea from a concept to a real product required the involvement of engineers and technicians from several Dalian divisions and suppliers. The project required meeting tough technical challenges, launching the product ahead of competitors, and doing it for an affordable cost.

Another example is Shah Alam Hospital's installation of a new employee benefits plan. The project involved developing new policies, training staff workers, familiarizing 10,000 employees with the plan, and installing new software and a database, and required participation from personnel in human resources, financial services, and information systems, plus experts from two consulting firms. It typifies "change" projects everywhere—projects initiated in response to changing needs and with the goal of transforming the organization's way of doing things.

Finally, consider that virtually every company has a website. Behind each site are multiple projects to develop or enhance the website and to integrate electronic business technology into the company's mainstream marketing and supply-chain operations. Such projects exemplify all organizations' need to change, in this case to keep pace with advances in information technology and business processes.

Activities such as these defy traditional management approaches for planning, organization, and control. They are representative of activities that require the use of project management to meet technological or market-related performance goals in spite of limited time and resources.

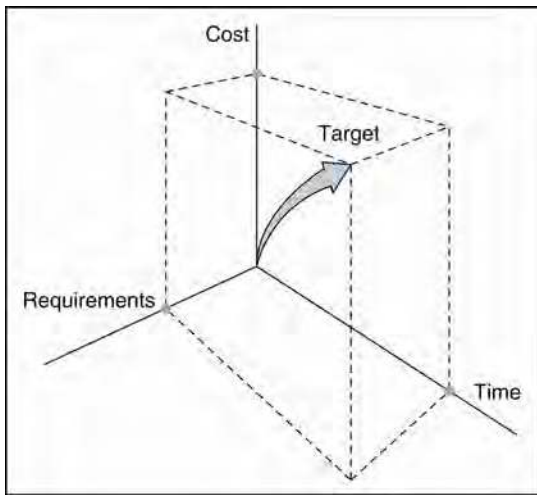


Figure 1.5

Three-dimensional project goal.

Source: Adapted from Rosenau M. *Successful Project Management*. Belmont, CA: Lifetime Learning Publications; 1981, p. 16.

1.5 Project Goal: Time, Cost, and Performance

The goal of every project can be conceptualized in terms of hitting a target that floats in three-dimensional space—the dimensions of **cost**, **time**, and **performance** (Figure 1.5). Cost is the specified or budgeted cost for the project. Time is the scheduled period over which the work is to be done. Performance is what the project end-item, deliverables, or result must be or do; it includes whatever the project customer, user, and other stakeholders consider necessary or important. The target represents a goal to deliver what’s required by the client within the budget and by a specific deadline. The purpose of project management is to hit the project target goal.⁹

But technological complexity, changing markets, and an uncontrollable environment make hitting the target difficult. Time, cost, and technical performance are interrelated, and exclusive emphasis on just one will likely undermine the others. In trying to meet schedules and performance requirements, costs increase; conversely, in trying to contain costs, work performance erodes and schedules slip. In earlier times, one or two aspects of the goal could be allowed to slide so that the “most fixed” could be met. Many projects such as the Pathfinder, Dalian Company, and Shah Alam Hospital examples do not have this luxury.

Project management is a way to maintain focus on all three dimensions and to control the tradeoffs among them, but it is not always successful. Failure of a project to meet time and cost targets is referred to as **project management failure**—the project exceeded targets according to the project schedule and budget. Oftentimes, however, even when that happens, the project end-item or solution is a success because most stakeholders—customers, users, the market, or the community—got what they expected, or more. To them, the project was a success. The obverse is when the end-item performs poorly or is not what stakeholders wanted or expected; this is called **project failure**. For everyone, the best scenario is both project management success (time and cost) and project success (outcome performance); the worst is project failure—when stakeholders are stuck with something that does not meet their expectations—even if it was delivered on time and on budget.

I.6 Project Management: The Person, the Team, the Methodology

Three features distinguish project management from traditional forms of management: the person, the team, and the methodology.

The most prominent feature of project management is the role of the **project manager**—the individual who has overall responsibility to *plan, direct, and integrate* the efforts of everyone associated with the project to achieve the project goal. In the role of project manager, one person is held accountable for the project and is dedicated to achieving its goals. The project manager coordinates the efforts of every functional area and organization associated with the project and oversees the planning and control of costs, schedules, and work tasks. Although numerous other parties are also involved in the project (**stakeholders**), the role of project manager is a key distinguishing feature between project- and non-project management.

Doing a project is a **team effort**, and project management means bringing together individuals and groups and directing them toward the common goal. The team will often consist of people and groups from different functional areas and organizations. Depending on the project, the size and composition of the team may fluctuate; usually the team disbands after the project is completed.

The project manager and project team typically perform work in phases according to a **project management methodology**. This methodology provides for *integrative planning and control* of projects, which, says Archibald, refers to:

the pulling together of all important elements of information related to (1) the products or results of the project, (2) the time, and (3) the cost, in funds, manpower, or other key resources . . . for all (or as many as practical) phases of the project. [It] requires continual revision of future plans, comparison of actual results with plans, and projection of total time and cost at *completion* through interrelated evaluation of all elements of information.¹⁰

As a project proceeds from one phase to the next, the project manager relies on the methodology to (1) identify the project tasks, (2) identify the required resources and the costs, (3) establish priorities, (4) plan and update schedules, (5) monitor and control end-item quality and performance, and (6) measure project performance.¹¹

I.7 Project Management Standards of Knowledge and Competencies

Project management has become a recognized vocation supported by several professional organizations around the world. These organizations have advanced project management by establishing standards, guidelines, and certifications. Among the globally more well-known of these organizations are PMI (Project Management Institute), IPMA (International Project Management Association), and APM Group (Association for Project Management). PMI is based in the US and is the largest of these organizations; IPMA, based in the Netherlands, is an international group of national project management associations in Europe, Africa, Asia and North and South America; and APM is based in the UK.

These organizations have published “best practices” of project management as “bodies of knowledge” (standards) and competencies for the profession.¹² Although none of the standards covers everything about project management, they have become recognized norms

about what minimally a project management professional should know. The organizations also offer levels of qualification and certification that include, for example, PMI's PMP (Project Management Professional) certification; APM's APMP (APM Professional), and IPMA's CPMA (Certified Project Management Associate). PMI's and APM's certifications are "BOK-based"; IPMA's certifications are "competency-based." Another certification popular in Europe is based upon PRINCE2 (Projects IN Controlled Environments, Version 2), a project management methodology originated by the UK Office of Government Commerce.¹³

For readers interested in professional certification, Tables I.1 through Table I.4 in the Appendix to the chapter show the correspondence between the knowledge areas, competencies expected, and methods from PMI, IPMA, APM, and PRINCE, and chapters in this book most relevant to them.

I.8 About this Book

Philosophy and objectives

As a philosophy and an approach, project management is broader and more sophisticated than traditional management of repetitive activities. It has roots in many disciplines, including management science, systems theory, accounting, operations management, organizational design, law, and applied behavioral science. What has evolved, and will continue to evolve, are a philosophy, approach, and set of practices, the sum total of which comprise project management. Some managers fail to understand this, believing that application of techniques alone, such as "Gantt charts," "critical path," or "matrix management" (all explained later) make for successful project management. Project management is much more than these.

C.P. Snow wrote an essay entitled "Two Cultures" about the cultural gap that separates scientists from the rest of society.¹⁴ Managers and management scholars also tend to separate the world into either of two perspectives: (1) the "quantitativists" tend to view projects in terms of costs, dates, and economic variables; (2) the "behaviorists" view projects in terms of peoples' behavior, skills, attitudes, and systems of organization.

The intent of this book is to give a balanced view that emphasizes both the behaviorist and quantitativist views of project management. The philosophy of this book is that for managers to "do" project management, they need familiarity with four topical areas: system methodology; systems development process; management methods, procedures, and systems; and organization and human behavior; correspondingly, the objectives of this book are to cover in depth:

1. The principles and philosophy that guide project management practice.
2. The logical sequence of stages in the life of a project.
3. The methods, procedures, and systems for defining, planning, scheduling, controlling, and organizing project activities.
4. The organizational, managerial, and human behavioral aspects of project management.

In recent years the scope of project management has grown to encompass more than the management of individual projects, recognizing that project success involves more than managerial skills and talent; hence, a final objective of the book is to cover:

5. Responsibilities of the organization and other stakeholders for effective project management and project success.

Organization of this book

Beyond this introductory chapter, the book is divided into five main parts. The first part is devoted to the basic concepts of project management. It describes project management principles, systems methodologies, and the systems approach—the philosophy that underlies project management. It also covers the origins and concepts of project management, situations where it is needed, and examples of applications. The second part describes the logical process in the creation and life of a system. Called the *systems development cycle*, it is the sequence of phases through which all human-made systems move from birth to death. The cycle is described in terms of its relation to projects, project management, and another cycle called the *project life cycle*. The third part is devoted to methods and procedures for aspects of project management such as planning, scheduling, budgeting, risk management, termination, and others. The fourth part is devoted to project organizations, teams, and the people in projects. It covers forms of project organization, roles and responsibilities of project managers and team members, leadership styles, stakeholders, and methods for managing teamwork and conflict. The last part covers topics that lie beyond the project manager but are crucial for project success and, more broadly, the success of the organizations and communities that sponsor and undertake projects. It also covers two topics that span most other topics in this book but require special attention: managing international projects and megaprojects.

The five stated objectives of this book are roughly divided among the book's five parts:

1. Basic concepts and systems philosophy: Chapters 1 and 2.
2. Systems development and project life cycle: Chapters 3 through 5.
3. Methods, procedures, and systems for planning and control: Chapters 6 through 14.
4. Organizations, management, and human behavior: Chapters 15 through 17.
5. The corporate context and international and megaproject management: Chapters 18 through 20.

Three appendices at the end of the book provide in-depth examples of topics covered throughout the book: request for proposal (Appendix A), project proposal (Appendix B), and project execution plan (Appendix C).

I.9 Study Project

The best way to learn about project management is to participate in it or, failing that, to observe it. At the end of every chapter in this book are two kinds of questions: the first kind are the usual chapter review questions, the second are called “Questions About the Study Project.” The latter are intended to be applied to a particular project of the reader's choosing. This will be called the “study project.” These questions and the study project help the reader relate concepts from each chapter to real-life situations.

The study project questions can be used in two ways:

1. For readers who currently work in projects as managers or project team members, the questions can be related to their current work. They serve to increase the reader's awareness of key issues surrounding the project and to guide managers in the conduct of project management.
2. For readers who currently are full- or part-time students, the questions can be applied to “real-life” projects they are permitted to observe and research. Many business firms

and government agencies are happy to allow student groups to interview managers and collect information about their projects. Though secondhand, this is nonetheless an excellent way to learn about project management practice (and mismanagement).

Assignment

Select a project to investigate. It should be a “real” project; that is, a project that has a real purpose and is not contrived just so you can investigate it. It can be a current project or one already completed; whichever, it must be a project for which you can readily get information.

If you are not currently involved in a project as a team member, then you must find one for which you have permission to collect data and interview people as an “outsider.” The project should include a project team (minimum of five people) with a project leader and be at least 2 or 3 months in duration. It should also have a specific goal in terms of a target completion date, a budget limit, and a specified end-item result or product. In general, larger projects afford greater opportunities to observe the concepts of project management than smaller ones.

If you are studying a project as an outsider, it is also a good idea to do it in a team with three to six people and an appointed team leader. This, in essence, becomes *your project team*—a team organized for the purpose of studying a project. You can then readily apply many of the planning, organizing, team building, and other procedures discussed throughout the book as practice and to see how they work. This “hands-on” experience with your own team combined with what you learn from the project you are studying, will give you a fairly accurate picture about problems encountered and management techniques used in real-life project management.

APPENDIX: RELATION BETWEEN PROFESSIONAL STANDARDS AND CHAPTERS OF THIS BOOK

Table I.1 (a) *PMI Standard for PM and PMBOK Guide, 7th Edition, 2021.*

PM Standard for Project Management and PMBOK Guide, 7th Edition	Book Chapters Addressing These Areas	
	Most Relevant	Related
PM Standard for Project Management		
1. Introduction	0, 1	
2. System for value delivery	2	
2.1 Creating Value	4, 10, 19	13
2.2 Organizational Governance Systems	12, 15, 18, 19	16
2.3 Functions Associated with Projects	12, 15, 19	16
2.4 The Project Environment	2, 18	
2.5 Product Management Considerations	4, 5, 18	2, 10
3. Project management principles		
3.1 Be a Diligent, Respectful, and Caring Steward	16	17

Table I.1 (a) *(Continued)*

PM Standard for Project Management and PMBOK Guide, 7th Edition	Book Chapters Addressing These Areas	
	Most Relevant	Related
3.2 Create a Collaborative Project Team Environment	17	14, 15, 16
3.3 Effectively Engage with Stakeholders	4, 14, 16	5, 10, 18
3.4 Focus on Value	4, 10, 14, 19	9, 13
3.5 Recognize, Evaluate, and Respond to Systems Interactions	2	13
3.6 Demonstrate Leadership Behaviors	17	15, 16
3.7 Tailor Based Context	4, 14, 19	13
3.8 Build Quality into Processes and Deliverables	10	13, 14
3.9 Navigate Complexity	11	
3.10 Optimize Risk Responses	11	
3.11 Embrace Adaptability and Resiliency	14, 16	4, 17
3.12 Enable Change to Achieve the Envisioned Future State	14, 16	2
PMBOK Guide, 7th Edition		
1. Introduction	0	
2. Project Performance Domains		
2.1 Stakeholder Performance Domain	16	4, 13, 18, 20
2.2 Team Performance Domain	17	15, 16, 18, 20
2.3 Development Approach and Life Cycle Performance Domain	2, 3, 4	5, 15, 16, 17
2.4 Planning Performance Domain	4, 6, 7, 9, 14	5, 8, 10, 11, 12, 18, 20
2.5 Project Work Performance Domain	4, 12, 13, 17, 18	6, 7, 16
2.6 Delivery Performance Domain	4, 6, 10, 13, 16	5, 7, 8, 12, 14, 18
2.7 Measurement Performance Domain	13	4, 5, 6, 9, 14, 16, 19
2.8 Uncertainty Performance Domain	11	13
3. Tailoring	4, 14	6, 11, 13, 16, 17, 18
4. Models, Methods, and Artifacts		
4.1 Overview	--	
4.2 Commonly Used Models	17	4, 13, 20
4.3 Models Applied Across Performance Domains	--	
4.4 Commonly Used Methods	3, 4, 6, 10, 12, 13, 19	8, 11
4.5 Methods Applied Across Performance Domains	--	
4.6 Commonly Used Artifacts	4, 6, 9, 11, 12, 14, 15, 16, 19	10, 11
4.7 Artifacts Applied Across Performance Domains	--	

Since the principle-based PMBOK7 does not entirely replace the more process-based PMBOK6, the table below is included to show the book's relation to the 6th Edition of the PMBOK.

Table I.1(b) PMI Bodies of Knowledge and Process Groups, *PMBOK Guide*, 6th Edition, 2017.

PMBOK Guide, 6th Edition	Chapters Addressing These Areas	
	Most Relevant	Related
Part 1 Bodies of Knowledge		
1. Introduction	0, 1	16, 17
2. The environment in which projects operate	15, 18	5, 13
3. The role of the project manager	16, 17	12
4. Project integration management*	5, 6, 7, 13	13, 18
5. Project scope management*	4, 5, 6	2, 14, 20
6. Project schedule management*	6, 7, 8, 13	14, 20
7. Project cost management*	9, 13	20
8. Project quality management*	10	2, 13, 14
9. Project resource management*	7, 12, 17	8, 13, 15, 16, 20
10. Project communications management*	4	12, 14, 20
11. Project risk management*	11	8, 13, 19, 20
12. Project procurement management*	12	13, 20
13. Project stakeholder engagement*	16	1, 2, 3, 12, 17, 20
*PMBOK Knowledge area		
Part 2 Process Groups		
Introduction	1, 3, 16, 18, 19	
Initiating Process Group	3, 4	
Planning Process Group	6, 7, 8, 9	10, 11, 14, 20
Executing Process Group	5	14, 20
Monitoring and Controlling Process Group	13	11, 14, 20
Closing Process Group	5	

Table I.2 IPMA Individual Competence Baseline, 4th Version, 2015.

IPMA ICB4	Chapters Addressing These Competencies	
	Most Relevant	Related
1. <i>Introduction (Note: An introduction to the ICB standard)</i>	-	
2. <i>Purposes and intended users</i>	-	
3. <i>The individual competence baseline</i>	-	
4. <i>Individuals working in project management</i>		
4.1 Managing projects	1, 3–6, 13, 14	2, 7–12, 15–20
4.2 Competences overview	-	
4.3 Perspective		
4.3.1 Strategy	3, 19	4
4.3.2 Governance, structures, and processes		15, 18, 19

Table I.2 (Continued)

IPMA ICB4	Chapters Addressing These Competencies	
	Most Relevant	Related
4.3.3 Compliance, standards, and regulation		18, 20
4.3.4 Power and interest	16	17
4.3.5 Culture and values	17	20
4.4 People		
4.4.1 Self-reflection and self-management		17
4.4.2 Personal integrity and reliability	16	17
4.4.3 Personal communication	13	20
4.4.4 Relations and engagement	16, 17	
4.4.5 Leadership	17	18
4.4.6 Teamwork	17	15
4.4.7 Conflict and crisis	17	
4.4.8 Resourcefulness		2, 6
4.4.9 Negotiation	12	17
4.4.10 Results orientation	13	4–12
4.5 Practice		
4.5.1 Project design	2–4	17–20
4.5.2 Requirements and objectives	2–4, 10, 16	19
4.5.3 Scope	6	2, 4, 10
4.5.4 Time	3, 6, 7, 8	14
4.5.5 Organizations and information	13, 15	6, 18
4.5.6 Quality	10	2, 4
4.5.7 Finance	9, 13	
4.5.8 Resources	6–8	12, 18
4.5.9 Procurement	12	
4.5.10 Plan and control	2–4, 6–9, 13	10–12, 14
4.5.11 Risk and opportunity	11	19
4.5.12 Stakeholders	16	13
4.5.13 Change and transformation		17
5. Individuals working in programme management	18	
6. Individuals working in portfolio management	19	

Table I.3 APM Body of Knowledge, 7th Edition, 2019.

APM Body of Knowledge	Chapters Addressing These Areas	
	Most Relevant	Related
<i>Chapter 1 Setting up for success</i>		
1.1 Implementing strategy	19	15, 16
1.2 Life cycle options and choices	3	2, 4, 5
1.3 Establishing governance and oversight	3	10, 15, 19
<i>Chapter 2 Preparing for change</i>		
2.1 Shaping the early life cycle	3, 12	18, 19
2.2 Assurance, learning, and maturity	3, 18	10
2.3 Transition into use	3, 5	19

Table I.3 (Continued)

<i>APM Body of Knowledge</i>	<i>Chapters Addressing These Areas</i>	
	<i>Most Relevant</i>	<i>Related</i>
<i>Chapter 3 People and behaviours</i>		
3.1 Engaging stakeholders	16, 17	6, 13
3.2 Leading teams	17	16
3.3 Working professionally	16	13, 17
<i>Chapter 4 Planning and managing deployment</i>		
4.1 Defining outputs	2, 4, 6, 10, 11	
4.2 Integrated planning	6, 7, 8, 9, 11, 12	
4.3 Controlling deployment	13, 10, 11, 12	

Table I.4 Managing Successful Projects with PRINCE2, 6th Edition, 2017.

<i>Managing Successful Projects with Prince2</i>	<i>Chapters Addressing Principles, Themes, Processes</i>	
	<i>Most Relevant</i>	<i>Related</i>
1. Introduction		
2. Project management with PRINCE2	1, 16	13, 14, 17, 18
3. Principles	3, 16, 19	18
4. Tailoring and adopting PRINCE2	3, 14	18
5. Introduction to PRINCE2 themes	3, 6, 10, 11, 13, 15,	
6. Business case	3	19
7. Organization	15, 16	3, 13, 14, 17
8. Quality	10	2, 13, 14
9. Plans	6, 7, 8, 9	10, 11, 14, 20
10. Risk	11	8, 13, 19, 20
11. Change	4	10
12. Progress	13	14, 20
13. Introduction to processes	3	
14. Starting a project	3, 4	19
15. Directing a project	3, 5, 13	
16. Initiating a project	3, 4, 6	11, 19
17. Controlling a stage	13	3, 11, 14, 20
18. Managing product delivery	5, 13	14, 20
19. Managing a stage boundary	3	11
20. Closing a project	5	
21. Considerations for organizational adoption	18	



Review Questions

1. Look at websites, newspapers, magazines, or television for examples of projects. Surprisingly, a great number of newsworthy topics relate to current and future projects, or to the outcome of past projects. Prepare a list of these topics.
2. Prepare a list of activities that are not projects. What distinguishes them from project activities? Which activities are difficult to classify as projects or non-projects?
3. Because this is an introductory chapter, not very much has been said about why projects must be managed differently from ordinary “operations,” and what constitutes project management—the subject of this book. Now is a good time to speculate about these: Why do you think projects and non-projects need to be managed differently? What do you think are some additional or special considerations necessary for managing projects?

CASE I.1 DENVER AIRPORT¹⁵

At the initiation of the project to build Denver Airport in 1989, the planned 4-year timeframe seemed adequate, yet despite abundant political backing and adequate funding it was completed 16 months late and \$1.5 billion over budget. The NTCP model can be used in retrospect to explain the root cause of much of the project's unsatisfactory performance. With 20–20 hindsight one may argue that a simple NTCP analysis of the project and its sub-projects at the start (and adjustment of management style accordingly) might have improved project performance.

To enable aircraft turnaround in less than 30 minutes, United Airlines, one of the airport's largest tenants, requested an automated baggage sorting and handling system over the traditional manual handling system. In December 1991 BAE Automatic Systems was contracted to design and implement the system in an estimated 2.5-year timeframe. By August 1994 the system, already 11 months late, was still not functioning properly and was severely hampering airport operations. Management decided to build an alternative, more traditional baggage system for an additional \$50 million, and only United would use the BAE system at its own terminal concourse. In January 1995 a full-scale practice run of the BAE system was executed successfully, and in February 1995 the airport was opened—16 months late.

Building the airport was mostly a typical large construction project; in terms of NTCP it would be classified as follows: Novelty—Platform; Technology—Low-tech; Complexity—Array; Pace—Fast/Competitive. The snag in the project was the BAE automatic baggage-handling system: it was new technology and, thus, riskier than the rest of the project, a risk that was not considered. It was the first of its kind (the technology had been used before but only on a much smaller scale) and required several design cycles and intensive testing. In terms of technology, it should have been considered High-tech. As discussed later in the book, high-risk projects need to be managed differently from low-risk projects. The NTCP profiles of the total project and the baggage-handling system are illustrated in Figure I.6.

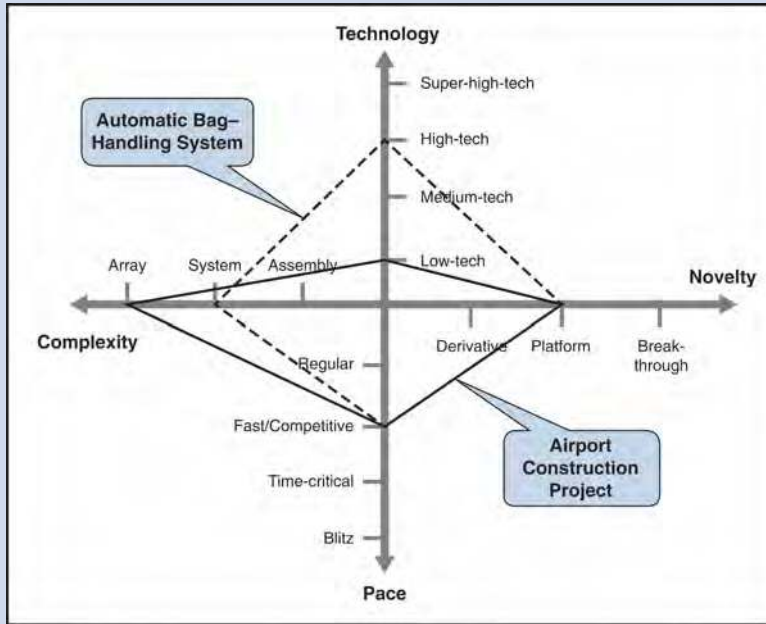


Figure I.6

Diamond model profiles for the Denver Airport project and for its bag-handling system.



Questions About the Case

1. In what ways should High-tech projects be managed differently from Low-tech ones?
2. BAE Automatic Systems is a reputable high-technology corporation and was familiar with building automated baggage-handling systems. What might have convinced the company to accept a schedule of only 2.5 years for designing and construction of the baggage-handling system?
3. If an NTCP analysis had been done and the profile of the baggage-handling system identified, what should the project manager have done to help ensure project success?
4. Explain how the NTCP model makes provision for 144 different types of projects.

Notes

1. Tompkins P. *Secrets of the Great Pyramids*. New York, NY: Harper & Row; 1976, pp. 233–234; Poirier R. *The Fifteen Wonders of the World*. New York, NY: Random House; 1961, pp. 54–67; Smith C. *How the Great Pyramid Was Built*. Washington DC: Smithsonian, 2004. The last reference is noteworthy for its engineering and project management perspective.

2. Tompkins P. *Secrets of the Great Pyramids*, pp. 227–228.
3. Barber F. *The Mechanical Triumphs of the Ancient Egyptians*. London, UK: Tribner; 1900.
4. George C.S. *The History of Management Thought*. Upper Saddle River, NJ: Prentice Hall; 1968, p. 11.
5. Potok C. *Wanderings*. New York, NY: Fawcett Crest; 1978, pp. 154–162.
6. See Archibald R.D. *Managing High-Technology Projects*. New York, NY: Wiley; 1976, p. 19; Meredith J. and Mantel S. *Project Management: A Managerial Approach*, 3rd edn. New York, NY: Wiley; 1995, pp. 8–9; Roman D. *Managing Projects: A Systems Approach*. New York, NY: Elsevier; 1986.
7. See Terraine J. *The Mighty Continent*. London, UK: BBC; 1974, pp. 241–242.
8. This section is adapted from: Shenhar A. and Dvir D. *Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation*. Cambridge, MA: Harvard Business School Press; 2007. Since publication of the book, the NTCP model has been revised: “Breakthrough” has been split into New-to-Market, and New-to-World; to “Complexity” the level of Component has been added below Assembly.
9. See Rosenau M.D. *Successful Project Management*. Belmont, CA: Lifetime Learning; 1981, pp. 15–19.
10. Archibald, *Managing High-Technology Projects*, pp. 6–7.
11. Kerzner H. *Project Management: A Systems Approach to Planning, Organizing, and Controlling*, 10th edn. Hoboken, NJ: John Wiley & Sons; 2009, p. 16.
12. *APM Body of Knowledge*, 7th edn. Association for Project Management, 2019; *IPMA Competence Baseline*, International Project Management Association. Available for download at https://products.ipma.world/wp-content/uploads/2016/03/IPMA_ICB_4_0_WEB.pdf [accessed June 17, 2024]; *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*, 7th edn. Project Management Institute, 2021.
13. *Managing Successful Projects with PRINCE2*, 2017 edn. Office of Government Commerce. Available for download at <https://www.prince2.com/usa/downloads> [accessed June 17, 2024].
14. Snow C.P. *The Two Cultures and a Second Look*. Cambridge, UK: Cambridge University Press; 1969.
15. Shenhar and Dvir. *Reinventing Project Management*.

Part I

Philosophy and Concepts

1	What Is Project Management?	23
2	Systems Approach	43

The two chapters in this section describe the philosophy and concepts that distinguish project management from non-project management. The first chapter introduces features associated with project management and variations of project management. Project management is an application of what has been called the systems approach to management, and the second chapter describes the principles, terminology, and methodology of that approach. The two chapters set the stage for detailed coverage in later sections.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Chapter 1

What Is Project Management?

The projects mentioned in the Introduction—the Great Pyramid of Egypt, the International Space Station, the Channel Tunnel, and the development of Product J all have something in common with each other and with most every other undertaking of human organizations: they all require, in a word, *management*. Although the resources, work tasks, and goals of these projects vary, none of them could have happened without management. This chapter contrasts project management with non-project management and looks at the ways in which project management is used.

1.1 Functions of Management¹

The role of management is to plan, organize, and integrate resources and tasks to achieve goals. Although the specific responsibilities of managers vary greatly, all managers—whether corporate presidents, agency directors, line managers, school administrators, movie producers, or project managers—have this same role.

The activities of managers, including project managers, can be classified into five functions, virtually all of which are directed at a purpose or goal, as shown in Figure 1.1. First is the function of deciding what must be done and how to do it. This is the **planning** function, which is to establish the means to achieve the goal that is consistent with organizational-level goals, resources, and constraints.

Second, and related to planning, is forming or configuring the organization to do the work; this is the **organizing** function. It involves (1) hiring, training, and gathering people into teams with responsibility and accountability relationships; (2) acquiring and allocating materials, capital, and other resources; and (3) creating an organization structure with policies, procedures, and communication channels.

Third is doing the work; this is the **execute** function. The team or organization executes the work according to the plan.

Fourth is monitoring work progress to achieve the goal and taking necessary action whenever work deviates from the goal; this is the **controlling** function.

The **close** function brings the project to an end when the purpose or goal has been reached.

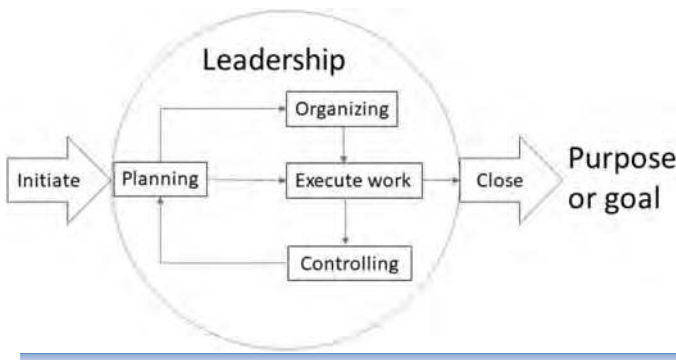


Figure 1.1
The functions of management in the context of a project.

Finally is the overarching function of directing and motivating people to attain the goal, which is the **leadership** function. As implied in the figure, managers exercise leadership in doing everything—planning, organizing, executing, and controlling.

Different managers' jobs carry different responsibilities depending on the functional area and managerial level of the job. Some devote most of their time to planning and organizing, others to controlling, and others to directing and motivating. At some time or another, project managers perform all these functions. Figure 1.1 represents management within the context of a project, beginning with the project's initiation and concluding with the project's closeout.

1.2 Features of Project Management

Project management is a **systems approach** to management. A system is a collection of inter-related components or elements that in combination serve a purpose and work toward a goal. A project can be thought of as a system: it is a collection of elements—work tasks, resources, and **stakeholders** (individuals, teams, organizations)—aimed at achieving a goal. The emphasis in the systems approach is to optimize the overall system to achieve the goal. The approach starts by defining the goal, identifying elements of the system that contribute to or detract from the goal, and then managing the elements to best achieve the goal. It involves all the functions of management—planning, organizing, and so on.

As mentioned in the Introduction, projects differ from non-projects. Non-project activities such as mass production of products or delivery of repetitive services are routine and seldom change. They tend to involve the *same people* doing the *same procedures*, *day-in, day-out*, without much change or risk. In contrast, every project is *unique* and unfamiliar in some sense, and requires people or teams from different functions or organizations to do different things. This creates uncertainty and risk that make it harder to achieve the goal. So, the question is: How do you manage such a thing as a project?

The answer is, of course, project management, the key features of which are:²

1. A **single person**, the project manager, heads the project team or organization.
2. Because each project is unique, it requires a unique variety of skills and resources; and therefore, the work is typically performed by **people** from **different functional areas** or outside **contractors**.
3. The project team or organization reflects the project: it is **goal-oriented** and **temporary**.

4. The project manager is responsible for **integrating the work** of people from the different functional areas or outside contractors.
5. A project might have **two chains of command**, one functional and one project, so often people working in a project have both a project manager and a functional manager.
6. Decision making, accountability, outcomes, and rewards are **shared** between the project team and supporting functional areas and outside contractors.
7. Although the project organization is temporary, the functional or subcontracting units from which it is formed are usually permanent. When a project ends, the project organization is disbanded, and people return to their functional or subcontracting units.

Because projects require the coordinated efforts of individuals and groups from within and outside the organization, managers and workers in different units must work directly with each other. Formal lines of communication and authority are frequently bypassed, and a **horizontal hierarchy** is created to enable people in the project from different functional areas and outside organizations to work directly with each other.

In non-project organizations managers tend to be specialized and responsible for a single functional unit or department. A project that involves multiple departments, however, needs someone from beyond these departments to be responsible for meeting the project's goals; this is the project manager. Responsibility for meeting project goals versus functional or departmental goals is a key feature that distinguishes project managers from functional managers.

Project managers often direct people who are not “under” them but who are “assigned” to them from different departments or contractors, as needed, which makes being a project manager more complicated (and difficult) than being a departmental manager.

Outcomes, benefits, and value

In its broadest sense, project management is the business of **generating value**. Projects create **outcomes** (also called **deliverables**, **products**, or **end-items**), and those outcomes provide **benefits** to project stakeholders.³ For example, consider a project whose product outcome is a business software application. The product benefits its users by providing more efficient processing of information, and users value that because it saves them time and effort.

One interpretation of the term value is the “relative worth” of something, in other words, is it worth the time and cost? As mentioned in the Introduction section, projects often provide stakeholders with what they wanted or expected (project success) even when the time and cost were more than they expected (project management failure). If a software product saves the user time and money, the project (to the user) is a success, even if it costs too much, takes too long, or causes investors and contractors to lose money.

In the business world, projects are but one element in a larger scheme called the **value delivery system**, which besides projects includes programs and portfolios—topics described in Part V.

Change and stakeholder engagement

Every project involves “change”; it creates a product or end-item that did not exist before, or it replaces or modifies something that did. You can think of a project as being the start of

someone's intention to change something, and the words "change" and "project" as being interchangeable.

But not all changes are welcomed. Yes, each project provides value and benefits to somebody, but almost always there are others who do not recognize those benefits, will not receive them, or see negative outcomes and oppose or openly resist the project. Consequently, an important aspect of project management is to understand the attitudes, perspectives, wants, and needs of different stakeholders, and to gain their support or reduce their resistance to the change. Ways to do this include altering the project to better meet different stakeholders' needs and wants or altering stakeholders' attitudes and perspectives through communication and training. This happens through a planned process of **stakeholder engagement** and **change management**, topics discussed in Chapters 16 and 17.



See
Chapters 16
and 17

1.3 Evolution of Project Management

No single individual or industry can be credited with the idea of project management. It is often associated with the early missile and space programs of the 1960s, but clearly its origins go back much earlier. Techniques of project management were first used in the major construction works of antiquity, such as the Egyptian Pyramids and Great Wall of China, later modified for use on other projects such as water management and shipbuilding. Starting in the early twentieth century, managers developed techniques for use in other kinds of projects, such as designing and testing new products, and building and installing specialized machinery.

By the 1950s, the size and complexity of many projects had increased so that existing management techniques proved inadequate. Repeatedly, large-scale projects for developing aircraft, missiles, communication systems, and naval vessels suffered enormous cost and schedule overruns. To grapple with the problem, two new methods for planning and control were developed, one called PERT, the other called CPM (described in Chapter 8). A decade later, so-called network-based methods were developed to integrate project cost accounting with project scheduling, and these became widespread when the US government mandated their usage in projects for the Department of Defense, NASA, and other large-scale efforts such as nuclear power plants. In the 1970s, the *earned value* method of project tracking was developed (see Chapter 13), leading to performance measurement systems that simultaneously track work expenditures and work progress.



See Chapter 8
and 13

The last 50 years have witnessed the increased computerization of project management. Early project planning and tracking systems cost \$10,000 to \$100,000, but today low-cost software and freeware make possible the use of a variety of planning, scheduling, costing, and controlling tools for virtually any project. Applications of web-enabled software and artificial intelligence to project management are expanding.

Associated with the evolution of project management was the emergence of project forms of organization and the role of project manager. Not until World War II was "the project" recognized as a distinct organizational form. In the urgency to develop sophisticated weaponry and organize massive task forces of troops and material, the "pure project" form of organization evolved, and by the 1960s companies began to use the term "project manager" as a formal title and role (described in Chapters 15 and 16).



See
Chapters 15
and 16

In recent years, project management has proliferated throughout all industries around the world. The most popular applications are discussed in the following sections.

1.4 Where Is Project Management Appropriate?⁴

Project management is applied everywhere, and there are few industries or situations where it is not. It can be applied to any “ad hoc” undertaking, such as shown in Figure I.3 in the Introduction chapter and includes activities that range from writing a term paper or remodeling a kitchen, to fundraising and constructing theme parks. Major corporations and governments frequently mandate the use of formal project management methods because they believe it offers better cost, schedule, and quality control, and they want a single point of contact—the project manager—with whom to deal.

Criteria

Cleland and King list five criteria for determining when to use project management:⁵

1. Unfamiliarity

By definition, a project involves doing different things or doing the same things but differently. The same tried-and-true methods no longer apply, and new methods have to be adopted; this might require adopting new approaches and learning new practices.

2. Magnitude of the effort

When a job demands substantially more resources (people, capital, equipment, etc.) than are normally needed, project management might be necessary. Examples include relocating a facility, merging two corporations, or developing a new product and placing it on the market. Even when the job lies primarily within the realm of one functional area, the task of coordinating the work with other functional areas might be large.

3. Dynamic environment

Industries such as aerospace, biotechnology, computers, electronics, pharmaceuticals, and communications face continual change, driven by high innovation, intense competition, and shifting markets and consumer demand. Project management provides the flexibility to deal with threats and opportunities in such industries.

4. Multifunctional effort

Functional areas tend to be self-serving and work independently. Whenever they must work together, project management builds the necessary relationships between the areas, expedites work, and reconciles conflicts. The project manager coordinates the efforts of all areas, whether internal functional departments or outside contractors.

5. Reputation of the organization

Whenever failure to satisfactorily complete a project would result in financial ruin, injuries, loss of market share, damaged reputation, or loss of future contracts, that is a strong case for use of project management. Although project management cannot guarantee success, it improves the odds.

Example 1.1: Renovating the Statue of Liberty⁶

Ninety-five years after the Statue of Liberty was presented to the American people, its surface and interior structure had become so badly corroded that it was judged structurally unsound. To oversee the badly needed restoration of the statue and other buildings on nearby Ellis Island, the US Department of Interior created a special project.

Very little of the restoration work qualified as “standard.” It involved highly specialized skills such as erecting intricate scaffolding, constructing a new torch, building windows for the crown, and replacing the interior framework—expertise for which tends to be found in smaller firms. In fact, the project consisted of a legion of over 50 small businesses, many of whose workers were immigrants or descendants of immigrants whom the statue had welcomed to America.

There were myriad notable features about the job. The scaffolding surrounding the statue never touched it at any point. Constructed of hundreds of thousands of pieces of aluminum, it qualified for the *Guinness Book of World Records* as the largest free-standing scaffolding ever built. To renovate the statue’s interior, 1,699 5-foot (1.5 m) bars were painstakingly fashioned from 35,000 pounds (15,900 kg) of stainless steel, and then individually installed. Around the crown, 25 windows were replaced. Each was handcrafted and had to be treated as a project unto itself. To fashion an entirely new torch, French artisans practiced an ancient copper shaping technique. The project was truly a marriage of art and engineering.

The 30-month, \$31 million project involved thousands of tasks performed by hundreds of people. Most of the tasks were non-routine and interrelated, and all had to be completed within a tight budget and schedule—a situation that clearly needed project management. (Chapter 17 discusses the company and managers responsible for the renovation.)

Where is project management not appropriate?

The obverse of all of this is that the more familiar and routine the undertaking, the more stable the environment, the less unique and more standardized the product or outcome, and the lower the stakes in the result, the less project management is needed. Production of standardized industrial and agricultural outputs, for example, is generally more efficiently managed by tried-and-true operations planning and control procedures than by project management. This is because for many kinds of repetitive operations, the process is well known, and standard, routine procedures for work planning, scheduling, and control are well-suited. You do not need a project to do these things.

1.5 Management by Project: A Common Approach

Beyond large-scale, infrequent undertakings, project management applies to all kinds of smaller, more frequent activities as well. Whenever an undertaking involves activities that are unique or unfamiliar and requires cooperation from several parties, project management applies.

For example, consultants in most every industry perform work on a project-by-project basis. Whenever their work calls for coordinated participation of several individuals or groups, project management applies. The more people or groups involved, the greater the applicability.

Similarly, groups that develop or implement new products, systems, or services also work on a project-by-project basis. The larger, riskier, more complex, costly, innovative, or different the thing being developed or implemented, the greater the applicability of project management.

Further, any group that performs unique work on a *client-by-client* basis (so-called made-to-order or made-to-engineer) is performing project work.

Managing any kind of work as a discrete project is referred to as **managing by project**, or **MBP**.⁷ With MBP, an undertaking is planned and managed *as if it were* a project. MBP implies that the undertaking will have a well-defined goal and objectives, requirements for the end-items, a plan of work, a completion date, and a budget for the required resources. A team is formed to do the work, and a project manager or team leader is assigned to guide and coordinate the work.

At some time, all organizations do projects. Even in repetitive industries, projects involving a few individuals are always in progress: new equipment is installed; the office is remodeled; a new product is conceived; in all cases, a formalized project group is formed, and a project manager appointed.

Example 1.2: Relocation of Goman Publishing Company

Many companies at some point face the decision to relocate. Relocation requires planning and coordination of numerous tasks involving many individuals, departments, and outside contractors. It is an important event that if done properly can be an exciting and profitable experience, but if done poorly can lead to financial loss. It is also representative of a situation wherein a company must do something it does not ordinarily do.

Goman Publishing was experiencing rapid growth and expected to outgrow its current facility. The initial task in relocating the company was to decide between two options: buying land and constructing a new building or leasing or buying an existing structure. After deciding to build, the next task was to select a site. The main selection criteria were purchase price, distance from current location, prestige and size of the new location, and access to major highways. The next task was relocation planning, which had two major phases: design and construction of the new facility, and the physical move; each involved numerous considerations. For example, Goman wanted to keep its current employees, and so to maximize the new facility's appeal, it built an indoor employee parking area and a large, well-appointed cafeteria. Among the many move-related considerations were furniture procurement, special handling of computers, hiring movers, informing employees and clients about the move, and maintaining corporate security. Further, the relocation would have to be scheduled to minimize downtime and interruption of operations.

To oversee the project and ensure that construction and the physical move would go as planned, Goman appointed a project manager. The project manager worked with architects and building contractors during the design and construction phases, and with representatives from functional departments and moving contractors during the relocation phase. Despite the scope and unfamiliarity of the project, Goman was able to complete the construction and physical move on time and on budget.

1.6 Different Forms of Project-related Management

Project management takes different forms with different names, but common to all of them are two features: (1) a project team or project organization, created uniquely for the purpose of achieving the goal, and (2) a single person—a project manager—who is responsible for accomplishing the goal. Beyond these, the features of the different forms of project management vary. The first section below covers “basic” project management, the most understood concept of project management. The others cover forms that are similar or related to project management.

Basic project management

Commonly the project manager and functional managers in a company are on the same organizational level and report to the same senior-level persons. The project manager has formal authority to plan, direct, organize, and control the project from start to finish. She may work directly with any organizational level and functional area to accomplish project goals. She reports to the general manager or owner and keeps him apprised of project status. Sometimes the project manager has authority to hire personnel and procure facilities, although more often she must negotiate with functional managers to “borrow” them.

Basic project management is implemented in two widely used forms—matrix and pure project. In matrix management, the project team is created from resources borrowed from the functional units. The project must share its resources with other projects and with the functional areas from which they are borrowed. In pure project management, a complete, self-contained organization is created; the needed resources belong to the project and do not have to be borrowed. These project management forms are described



See Chapter 15 in Chapter 15.

Often found in construction and technology industries, basic project management also applies to small, nontechnical activities. Adams, Barndt, and Martin cite examples:⁸

- Health, Education, and Welfare (HEW) performs social work largely based on grants allocated through state and local agencies. Associated with each grant are time, cost, and performance requirements. In essence, each grant results in a project to which the concepts of project management apply.
- An advertising firm conducting promotional campaigns utilizes the services of marketing research, accounting, graphics, sales, and other departments. The campaigns are like the projects in other industries: they have time and cost constraints and require planning and coordination of multiple departments as provided by project management.

Program management

The term **program management** is sometimes used interchangeably with project management due to the similarities of programs and projects: both are defined in terms of goals about what must be accomplished, and both require plans, budgets, and schedules to accomplish goals.

But programs and projects are different. The main distinctions are that a program extends over a longer time horizon (sometimes indefinitely) than a project, and it consists of several

parallel or sequential projects working to meet a higher-level, program goal. The projects within a program share common goals and resources and, often, are interdependent. As examples, an urban development program would include several projects such as housing rehab, job and skill training, and small business consulting assistance; a Mars exploration program might include several projects for unmanned probes to Mars and its two moons, followed by manned missions to Mars.

Another distinction is that a project produces and delivers an end result or product, but afterward the project organization is dissolved and responsibility is handed off to someone else to operate the product. In a program, however, the **program manager** is responsible for not only delivering the product but also for integrating it with other systems and, sometimes, for operating it for as long as needed. For example, program management oversees not only development of a satellite and its booster rocket, but also launching the satellite and then operating and monitoring it.

Many concepts for managing projects also apply to managing programs, though with modification to handle the larger scope and magnitude of programs and enable the program manager to oversee and coordinate the projects within the program. The Project Management Institute (PMI) has published a Standard for Program Management that aligns with its PMBOK; the UK Office of Government Commerce has produced one that aligns with PRINCE2.⁹ Program management is discussed more in Chapter 18.



See Chapter 18

Product management

Product management refers to a special type of program management whereby a single person is responsible for overseeing all aspects of a product's production scheduling, inventory, distribution, and sales. The **product manager** coordinates and expedites the product's launch, manufacture, distribution, and support. Like the project manager, the product manager communicates directly with functions inside and outside the organization, and coordinates efforts directed at product goals. The product manager is active in resolving problems that would degrade manufacturing capability, forestall distribution, alter prices, harm sales, or affect financing, production, and marketing of the product. For long-life-cycle products, the role is filled on a rotating basis.

Project portfolio management

Projects and programs can be grouped into a **project portfolio**, which is similar to an investment portfolio and where its purpose is to maximize the portfolio's value in terms of profit, rate of return, or meeting company strategic goals. The **portfolio manager** helps make decisions about adding, removing, or changing projects or programs in the portfolio based on their financial performance, resource utilization, risks, and other factors affecting business value.

Whereas the purpose of project and program management is to manage particular projects and programs, the purpose of portfolio management is to make sure the right projects and programs are selected, i.e., to assure that they align with the organization's strategic and financial goals and fit within the available limited resources. Portfolio management is not about managing projects, per se, but selecting and retaining the right projects. The PMI has created the Standard for Portfolio Management, and so has the UK OGC.¹⁰ Portfolio management is covered in Chapter 19.



See Chapter 19

1.7 New Product and Systems Development Projects

The development of every new product and system is a project. Examples include development of home appliances, pharmaceuticals, information systems, medical equipment, industrial machinery, computers, and systems for defense, aerospace, energy, and telecommunication.

When the development of new products and systems comprises new technologies, the early phases of the projects typically involve testing and experimentation. Although each project focuses on a newly designed product or system, the actual project deliverable could be either the physical product or system, or merely the design documentation or instructions for how to produce it. The following are two examples of systems and product development projects.

SpaceShipOne and the X-Prize competition¹¹

In 2025, Virgin Galactic (VG) was set to begin commercial spaceflight operations, a journey that began more than 20 years earlier when SpaceShipOne (SS1) was rolled out to the public and VG announced that SS1 was entering the \$10 million X-Prize competition against 23 other teams from seven countries. The goal: to be the first manned vehicle to make two trips into space in less than 2 weeks (Figure 1.2). Space is internationally recognized as beginning at 100 km or 62 miles (commercial jets fly at a height of about 8 km). SS1 was the brainchild of celebrated aerospace engineer Burt Rutan and the culmination of almost 8 years of design and development work. Rutan's major challenge was not just winning the prize but designing and building a complete space launch system—spacecraft, aerial launch vehicle, rocket motor, and all support subsystems—without relying on government support (i.e., many hundreds of engineers and many millions of dollars) to do it. Rutan would try to do it with his own little



Figure 1.2

SpaceShipOne beneath its mother ship, White Knight.

Source: Flight 16P taxi pre-launch, photo by D Ramey Logan, <http://creativecommons.org/licenses/by-sa/4.0/>.

company of 130 people, a handful of subcontractors, and \$25 million from billionaire Paul Allen, cofounder of Microsoft.

Besides Rutan and Allen, the principal stakeholders in the program included the Ansari Foundation, Sir Richard Branson, and the FAA (Federal Aviation Administration). The Ansari Foundation sponsored the X-Prize competition, and its requirements for “a non-government-funded program to put three people safely into space twice within 2 weeks with a reusable spacecraft.” Sir Richard Branson, founder of the Virgin Group, was the customer; his plan was to buy spaceships and the associated technology for his fledgling space airline, Virgin Galactic. Branson estimated VG would be able to turn a profit if it can carry 3,000 customers into sub-orbit over a 5-year period at about \$190,000 a ticket—to include medical checks, 3 days of training, custom-molded seats, and 5 minutes of floating weightless in space. (By comparison, a civilian trip aboard the International Space Station costs about \$50 million.) Paying passengers was another stakeholder group. Although none would be aboard SS1, the vehicle was designed with them in mind. For instance, SS1’s cabin was designed to provide a “shirtsleeve” environment so passengers would not have to wear spacesuits. The FAA was also a stakeholder; it imposes the requirements necessary for the spaceship to be “certified safe” and commercially viable.

As in many technical projects, the project manager shared oversight with a project engineer who was responsible for identifying technical requirements and overseeing system design, integration, and testing.

Product development at Dalian Company¹²

The future of Dalian Company depends on its ability to continuously develop and market new products. Dalian specializes in food and drink additives, but it is representative of firms in industries such as pharmaceuticals, food products, biotechnology, computer and entertainment electronics, and communications that must always generate new products and remain competitive.

The product development process at Dalian is facilitated by the New Product Development Department, which is responsible for managing and coordinating all internal and externally contracted development projects; its purpose is to ensure that good ideas can be developed and quickly brought to market. The department has three directors who are the project managers.

For each new product concept, a team is created with representatives from various departments. A director works with the team to direct and assess the project’s progress. Functional managers decide what is to be done and how, but the director has final say over project direction. The director always knows the status of the project and reports any problems or delays to upper-level managers who manage project portfolios (i.e. they are portfolio managers). Projects facing big problems or near failure are cancelled so resources can be allocated to more promising projects.

New product development projects involve participation from several departments: R&D (research & development) develops a product prototype and prepares specifications; engineering defines where and in what ways the product would be used; marketing determines the commercial market and how to position the product; manufacturing develops the process for making the product; finance determines the initial product costing; and legal obtains regulatory approval and performs patent research.

The director works with R&D scientists and marketing experts to establish the project’s feasibility, and to gain upper management’s approval. She works with department managers to prepare project plans and schedules. When additional labor, equipment, instruments, or raw materials are needed, she writes the funding requests. During the project she schedules and chairs project review meetings and issues monthly and quarterly progress reports.

1.8 Construction Projects


See Chapters
9, 10, 11,
and 20

Similar to developing new products and systems, construction projects often have a front-end piece devoted to architectural/engineering design followed by a back-end piece for fabrication and construction. This is typical for most construction projects, whether for new buildings, transport infrastructure (roads, bridges, rail systems, harbors, airports), factories, oil and gas installations, dams, mines, and plants for renewable energy and utilities. When undertaken for profit, such projects are classified as “capital expenditure projects.” Throughout this book are many examples of construction projects; including Case 9.2, the Channel Tunnel Project; Case 10.1, Big Dig; Case 11.1, Sydney Opera House; Case 11.3, Nelson Mandela Bridge; and Case 20.1, the Mozal Aluminum Smelter. These are all large construction projects, but look around and you see numerous smaller construction works—homes and small businesses being built or renovated; city streets being repaved; bridges and sewers being rebuilt; and solar panels being installed on rooftops. These are all projects.

1.9 Service-sector Projects

Project management is employed in a range of services like banking, consulting, insurance, and accounting. The following examples show how it is used in corporate auditing and non-profit fundraising.

Auditing at CPAone¹³

Large audits conducted by the auditing division at CPAone require the involvement of many people. In the audit of a national corporation, for example, numerous auditors with diverse specialties are needed to investigate all aspects of operations in various geographic areas. Given the number of people and the variety of expertise involved, a project manager is needed to oversee the audit. Every audit begins by assigning the client to a CPAone partner who is familiar with the client’s business. The partner becomes the audit’s “project director” and is responsible for the project’s initiation, staffing, scheduling, and budgeting.

In determining the general approach for the audit, the director considers the company’s size and number of departments, and assigns auditors on a department-by-department basis. The audit team is created anew for every audit and composed of people who have the skills needed for the audit. Each team has one or two staff accountants and one or two senior accountants. During the audit the director monitors the work for adherence to the Book of Auditing Standards and scheduled completion. The client and project director meet weekly to review progress. When problems cannot be solved, the director may call in people from CPAone’s tax or consulting divisions.

Nonprofit fundraising campaign project: Archdiocese of Boston¹⁴

The Archdiocese of Boston contracted the American Services Company, a fundraising consulting firm, to manage a 3-year campaign to raise \$30 million for education, social and health care services, building renovations, and a clergy retirement fund. American Services appointed a project manager to prepare the campaign strategy and organize and direct the campaign staff. The project manager worked with three stakeholder groups: donors, the

Archdiocese Board of Directors, and campaign volunteers. Potential target donors had to be identified and provided with evidence to show how their financial commitments would benefit the community and the Archdiocese; the board had to be involved in and kept apprised of campaign planning and progress; and volunteers had to be identified, organized, and motivated.

One of the project manager's first tasks was to conduct a feasibility study to determine whether there was sufficient leadership capability, volunteer willingness, and "donor depth" within the community to achieve the \$30 million goal. The study indicated the goal was achievable, and pastors were invited to a kickoff meeting where the campaign was introduced and the process of identifying potential donors and volunteers was explained.

The project manager provided guidance for forming a campaign leadership team and project office, enlisting volunteers, forming campaign committees, and recruiting and training volunteers. He also convened several "reality sessions" with chairpersons to remind them of the importance of the campaign and renew their commitment to the campaign goal, and organized frequent meetings with the volunteers to instill a sense of pride and involvement in the campaign.

1.10 Public-sector and Governmental Projects and Programs

The following two examples illustrate how project and program management is performed in large public sector and joint government/commercial undertakings.

Disaster recovery

The aid, assistance, cleanup, rebuilding, and return-to-normalcy efforts following a disaster (floods, fires, earthquakes, hurricanes, even wars) involve numerous organizations. A large disaster can impact many countries and require the support and coordinated efforts of host governments, non-governmental agencies (NGOs), local business, religious and community organizations, and international aid, charitable, and funding groups.

Almost by definition, post-disaster recovery is a program—or several programs—a host of project efforts devoted to the goals of rescuing and providing immediate relief to victims and, ultimately, returning the lives of people in the areas affected back to normal. The programs involve many projects to address the multiple aspects of a recovery, including:

- Immediate rescue of victims
- Food and medical care
- Temporary shelter and housing
- Clothing, blankets, and other immediate physical needs
- Social, moral, and spiritual assistance.

Ideally, disaster recovery is treated as an organized, coordinated effort that enables quick assessment of the scope of the situation, identification and organization of needed and available resources, and deployment of those resources. For all of that to happen requires leadership, usually in the person of someone with exceptionally strong organization and leadership abilities—in effect, a *program leader*. In the chaos and frenzy immediately following a disaster, however, it is often not clear who is in charge. Indeed, the poor immediate response and confused rescue and recovery efforts in New Orleans and the surrounding US Gulf coastal region

following Hurricane Katrina in 2005 has been blamed on a lack of leadership and coordinated management at all levels of government—federal, state, and local.

In the period following a disaster, the focus turns to (1) obtaining and allocating aid funding; (2) reconstruction, redevelopment, and rebuilding; (3) permanently situating victims; (4) dealing with waste and debris; and (5) providing jobs and ongoing support. Each requires many projects.

For example, the December 2004 tsunami caused severe damage to coastal areas all around the Indian Ocean; in India alone, it affected an estimated 2.7 million already-poor people. The Indian government launched the Emergency Tsunami Reconstruction Project, estimated to cost US \$682.8 million, to help repair or reconstruct damaged houses in two coastal regions and assist with the reconstruction of public buildings and the revival of livelihoods in fisheries and agriculture.¹⁵ It was a program that consisted of hundreds of projects and spanned many years.

NASA project and program management¹⁶

NASA has had a successful history of working in partnership with researchers in universities, industry, and the military. They work closely together on resolving technical problems, but technical initiatives and decisions are made by NASA field installations.

NASA organization includes (1) top management, (2) functional support for top management, (3) program offices for developing and controlling major programs, and (4) field installations that conduct the programs and their projects either on-site or at universities or contractors.

NASA has four mission directorates or offices: Exploration Systems, Space Operations, Science, and Aeronautics Research, shown in Figure 1.3. Each is responsible for development, justification, and management of programs that support broad NASA goals. Each office performs permanent activities for itself, and also does projects or tasks overseen by other directorates.

In a typical non-NASA government project, the government agency prepares specifications for a program, awards a contract, and then relies on the contractor for results. NASA uses a different approach since usually no one company has all the capability to execute a large project. Traditionally, NASA has relied upon industry to build, integrate, and test-fly hardware, and its own in-house management and technical competence to monitor and work with contractors. However, in recent years it has begun to transfer much or almost all this responsibility to contractors such as SpaceX and Blue Origin with the goal of developing a space industry outside of NASA.

Because space projects call for a diversity of technical and managerial competency, NASA project managers practice the philosophy of “participative responsibility”—an integration of technical and managerial competency across industry, academia, and NASA laboratories. Regardless of location, NASA brings in experts from its own field installations, universities, and other government laboratories to assist contractors in tackling difficult problems. This participative team approach avoids the usual delays caused by working across boundaries that separate government and commercial organizations.

At NASA a *program* is defined as a series of undertakings that over several years will accomplish broad scientific or technical goals. A *project* is defined as an undertaking within a *program* with a scheduled beginning and end, and normally involves design, construction, and/or operation and support of specific hardware items. This project–program duality is reflected in a management duality. Whereas the project manager has final responsibility for a project’s success, a program manager in Washington DC is responsible for developing and administering NASA guidelines and controls regarding the project. A project’s success can depend on

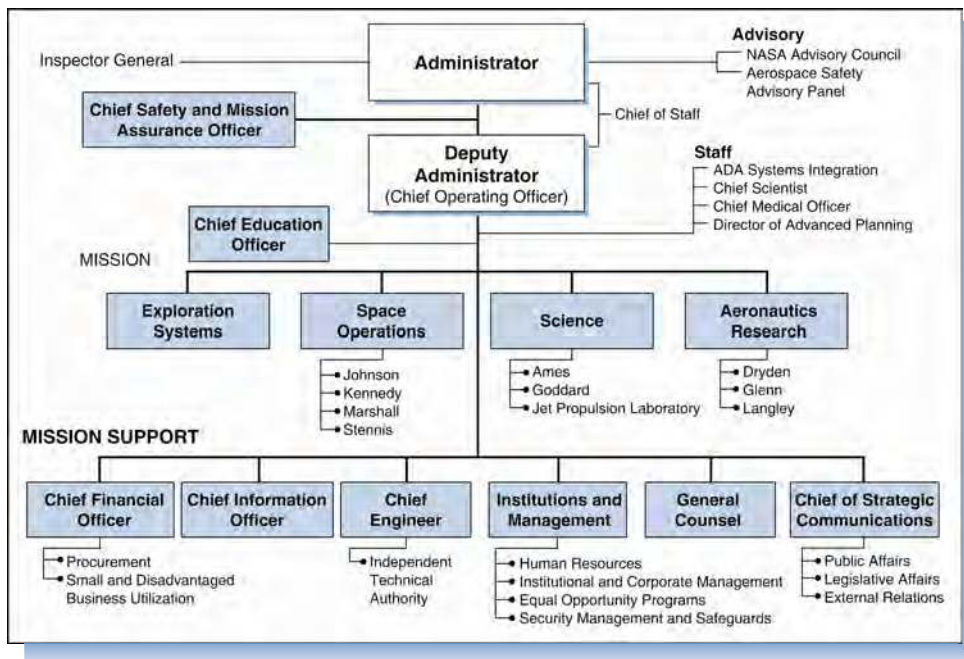


Figure 1.3
NASA program and organization chart.

Source: NASA.

how well the project and program managers work together. An example of NASA program management is Case 18.4, the Mercury Exploration Program.



See Chapter 18

1.11 Miscellaneous Projects

If you still wonder about the myriad situations where project management applies, read on.

Maintenance

All major facilities and machines require maintenance programs that include small repairs and preventative maintenance jobs. Some maintenance projects are relatively large, for example shutdowns of facilities like chemical plants and power-generating plants. Also, airplanes are removed from service after a certain number of flight hours, stripped down, inspected, and parts are replaced, repaired, or rebuilt. Jobs like that put facilities and equipment out of operation and require project management to do quality work, fast.

Events

Activities such as fundraising, political campaigns, and entertainment and sports events, if small, can be managed without project management (called **event management**), but as they



See Chapter 10

grow in size and complexity, they need project management. For major events like national fundraising and political campaigns, league championships, and the Olympic Games, project management is a necessity. Case 10.2, the FIFA 2010 World Cup, is an example.

1.12 Summary

The most identifiable aspect of project management is the project manager, the person who functions to unify project-related planning, communications, control, and direction to achieve project goals. The project manager is the integrator who ties together the efforts of functional areas, suppliers, and contractors, and keeps top management and the customer apprised of project progress. Project management includes many things, but it is the organization, systems, and methods that enable the project manager to plan, organize, direct, and integrate everything necessary to achieve project goals.

Project management can be applied to any temporary, goal-oriented activity, but it becomes more essential as the magnitude, unfamiliarity, and stake of the undertaking increase. Organizations in rapidly changing business and technology environments especially need project management.

Project management takes on various forms such as pure project, matrix, product, and program management forms. Project management is applied in similar ways in commercial, nonprofit, service sector, and public sector projects, with variations to account for differences in the environments.

Project management is a “systems approach” to management. The next chapter expands on that concept and discusses systems philosophy and methodologies that underlie project management.



Review Questions

1. Filming a motion picture and landing a spacecraft on the moon are both expensive projects conducted by teams and subject to budgetary and schedule constraints. The technical expertise for a moon landing is similar to *creating the illusion* of a spacecraft landing in a motion picture. Use the NTCP model described in the Introduction to show how the two project types differ.
2. Describe five functions of management. Are any of these ever not performed by managers? How do you think each of these functions comes into play during a project?
3. List the main characteristics of “projects.” How do these features distinguish projects from other, non-project activities?
4. What are the characteristics of “project management”? Contrast these to functional and other types of non-project management.
5. What makes project management more suitable to project environments than traditional management and organization?
6. Where did project management methods and organization originate? What happened during the twentieth century that made project management necessary?
7. What five criteria do Cleland and King suggest for determining when to use project management? From these, describe briefly how a manager should know when project management is appropriate for the task.

8. When is project management clearly not appropriate? List some “project-type” activities where you think project management should not be used. Describe organizations or kinds of work where both project and non-project types of management are appropriate.
9. Briefly compare and contrast the following forms of project management: program, product, program and portfolio. Give at least one illustration of an organization where each one is used.
10. In the industry, service sector, and government examples in this chapter, what common characteristics of the environment, the project goals, and the project tasks make project management appropriate or necessary? Also, what seem to be the common characteristics of the roles and responsibilities of the project managers in these examples? What are the differences?
11. Now that you know a little about projects and project management, list some government and private organizations where you think project management might be useful. You might want to check to see if, in fact, they are using project management.
12. The Introduction section of the book listed features of projects; each is: goal-oriented, unique, temporary, cross-functional, and unfamiliar. Describe these features for each of the projects depicted in the chapter in Sections 1.7–1.10 (SS1, Dalian, CPAone, etc.).



Questions About the Study Project

1. In the project you are studying, what characteristics of the company, project goals, tasks, or necessary expertise make the use of project management appropriate or inappropriate? Consider the project size, complexity, risk, and other criteria in answering this question.
2. How does the project you are studying fit the definition of a project?
3. What kind of project management is used—program, product, or other? Explain. Is it called “project management” or something else?
4. What functions does the project manager serve? What is his or her title?
5. In which way(s) does the industry of the study project differ from other industries described in the chapter? Do the differences affect how projects are managed?

CASE 1.1 DISASTER RECOVERY AT MARSHALL FIELD'S¹⁷

Early one morning, basements in Chicago's downtown central business district began to flood. A hole the size of a car had developed between the river and an adjacent abandoned tunnel that was built in the early 1900s for transporting coal. When the tunnel flooded, so did the basements of 272 buildings linked to it, including that of major retailer Marshall Field's.

The problem was first noted at 5:30 am when a member of Marshall Field's trouble desk saw water pouring into the basement. He notified his manager who immediately contacted the Chicago Fire and Water Departments, and Marshall Field's parent company, Dayton Hudson in Minneapolis. Electricity—and with it all elevator, computer, communication, and security services for the 15-story building—would soon be lost. The building was evacuated, and elevators were moved above basement levels.

A command post was set up and a team formed from various departments such as facilities, security, human resources, public relations, and financial, legal, insurance, and support services. Later that day, members of Dayton Hudson's risk management group arrived from Minneapolis to coordinate the team's efforts. The team's goal was to ensure employee and customer safety, minimize flood damage, and resume normal operations as soon as possible.

An attempt was made to pump the water out; however, as long as the tunnel hole remained unrepaired, the water continued to pour back into the basements. Thus, basements remained flooded until the tunnel was sealed, and the Army Corps of Engineers gave approval to start pumping. Everything in the second-level basement was a loss, including equipment for security, heating, ventilation, air-conditioning, fire sprinkling, and mechanical services. Most merchandise in the first-level basement stockrooms was also lost.

Electricians worked around the clock to install emergency generators and restore lighting and elevator service. Additional security officers were hired. An emergency pumping system and new piping to the water-sprinkling tank were installed so the sprinkler system could be reactivated. Measures were taken to monitor ventilation and air quality, and dehumidifiers and fans were installed to improve air quality. Within the week, inspectors from the city and OSHA (Occupational Safety and Health Administration) allowed the store to reopen.

After water was drained from the basement, damaged merchandise was removed and sold to a salvager. The basement had to be gutted to remove contaminants, and salvageable machinery had to be disassembled and sanitized.

The extent of the damage was assessed, and insurance claims filed. A construction company was hired to manage restoration of the damaged areas. The public relations department dealt with the media, being candid yet showing confidence in the recovery effort. Customers had to be assured that the store was safe. The team overseeing the recovery met twice a week to evaluate progress and make decisions, then slowly dissolved as the store recovered.

This case illustrates crisis management, an element of which is a team that can move fast to minimize losses and quickly recover damages, and someone to coordinate everything. At the beginning of a disaster there is little time to plan, so companies and public agencies often have crisis guidelines for responding to emergency situations. When an emergency occurs, they then develop more specific, detailed plans to guide short- and long-term recovery efforts.

QUESTIONS

1. In what ways is the Marshall Field's flood disaster recovery effort a project? Why are large-scale disaster response and recovery efforts projects?
2. In what ways do the characteristics of crisis management as described in this case correspond to those of project management?
3. Who was (were) the project manager(s) and what was his or her (their) responsibility? Who was assigned to the project team and why were they on the team?
4. Comment on the appropriateness of using project management for managing disaster recovery efforts such as this.
5. What form of project management (basic, program, and so on) does this case most closely resemble?

CASE 1.2 FLEXIBLE BENEFITS SYSTEM IMPLEMENTATION AT SHAH ALAM MEDICAL CENTER¹⁸

Senior management of Shah Alam Medical Center decided to procure and implement a new system that would reduce the cost and improve the service of its employee benefits coverage. The new system would have to meet four goals: improved responsiveness to employee needs, added benefits flexibility, better cost management, and greater coordination of human resource objectives with business strategies. A cross-functional team of 13 members was formed with representatives from Human Resources (HR), Financial Systems (FS), and Information Services (IS), and technical experts from Hun and Bar Software (HBS) consultants.

Early in the project a workshop was held with participants from Shah Alam and HBS to finalize project objectives and develop a project plan. Project completion was set to be 10 months. In that time HBS had to develop and supply all hardware and software for the new system; the system had to be brought online, tested, and approved; HR workers had to be trained to operate the system and load employee data; and all Shah Alam employees had to be educated about and enrolled in the new benefits process.

The director of FS was chosen to oversee the project. She had the necessary IT background and had previously worked with the IS group on another project. She selected two team leaders to assist her, one each from HR and IS. The HR leader's task was to ensure that the new system met HR requirements and the needs of Shah Alam employees. The IS leader's task was to ensure that the new software interfaced with other Shah Alam systems.

The Shah Alam team members worked half their time on the project, devoting the other half to their normal daily duties. The project manager and team leaders also worked part-time on it, although each gave the project priority. Shah Alam's senior management made it clear that meeting project requirements and time deadlines was imperative and so gave the project manager authority over functional managers and project team members for all project-related decisions.

QUESTIONS

1. What form of project management (basic, program, etc.) does this case most resemble?
2. The project manager is also the director of FS, one of the departments to be affected by the new benefits system. Does this seem like a good idea? What are the pros and cons of her being selected?
3. Comment on the team members' part-time assignment to the project and the expectation that they give the project top priority.
4. Much of this project's success depends on the performance of team members not employed by Shah Alam, namely the HBS consultants. They must develop the entire hardware/software benefits system. Why was an outside firm chosen for such an important part of the project? What difficulties might this pose to the project manager in meeting project goals?

Notes

1. Adapted from Szilagyi A. *Management and Performance*, 2nd edn. Glenview, IL: Scott, Foresman; 1984; pp. 7–10, 16–20, 29–32.
2. Portions of this section are adapted from Cleland D. and King W. *Systems Analysis and Project Management*, 3rd edn. New York, NY: McGraw-Hill; 1983, pp. 191–192.
3. As defined in the *PMBOK Guide*, 7th edn., 2021. Newtown Square, PA: Project Management Institute.
4. Portions of this section are adapted from Johnson R., Kast F., and Rosenzweig J. *The Theory and Management of Systems*, 3rd edn. New York, NY: McGraw-Hill; 1973, pp. 395–397.
5. Cleland and King, *Systems Analysis and Project Management*, p. 259.
6. Based upon Hofer W. Lady Liberty's business army. *Nation's Business* July; 1983: 18–28.
7. Sharad D. Management by projects, an ideological breakthrough. *Project Management Journal* March; 1986: 61–63.
8. Adams J., Barndt S. and Martin M. *Managing by Project Management*. Dayton, OH: Universal Technology; 1979, pp. 12–13.
9. Project Management Institute. *The Standard for Program Management*, 5th edn. Newton Square, PA: PMI; 2024; Office of Government Commerce. *Managing Successful Programmes (MSP)*, 5th edn. UK: The Stationery Office; 2020.
10. Project Management Institute. *The Standard for Portfolio Management*, 4th edn. Newton Square, PA: PMI; 2017; Office of Government Commerce. *Management of Portfolios*. UK: The Stationery Office; 2011.
11. This example and examples in later chapters of SpaceShipOne illustrate concepts. Much factual information about the project and associated systems is available from published sources, but design and development information is confidential. SpaceShipOne, the X-Prize, and the stakeholders described are all true-life, but for lack of information portions of this and subsequent examples are hypothetical.
12. Based upon information compiled by Jenny Harrison from interviews with managers in Dalian Company (factual company, fictitious name).
13. Based upon information compiled by Darlene Capodice from interviews with managers in the accounting firm (factual company, fictitious name).
14. Information about this project was contributed by Daniel Molson, Mike Billish, May Cumba, Jesper Larson, Anne Lanagan, Madeleine Pember, and Diane Petrozzo. Case is factual.
15. India: Emergency Tsunami Reconstruction Project. The World Bank Group, May 3, 2005, Press Release No: 453/SAR/2005, *ReliefWeb*, <http://www.reliefweb.int/rw/RWB.NSF/db900SID/VBOL-6C3CF8?OpenDocument&rc=3&cc=ind>, (accessed June 18, 2024).
16. Portions of this section are adapted from Chapman R. *Project Management in NASA: The System and The Men*. Washington, D.C.: NASA SP-324, NTIS No. N75-15692; 1973.
17. Information about this case was contributed by Jennifer Koziol, Sussan Arias, Linda Clausen, Gilbert Rogers, and Nidia Sakac. Case is factual.
18. Information about this case was contributed by Debbie Tomczak, Bill Baginski, Terry Bradley, Brad Carlson, and Tom Delaney. Organizational names are fictitious, but the case is factual.

Chapter 2

Systems Approach

A project can be conceptualized as a collection of people, equipment, materials, and facilities organized and managed to achieve a goal. In other words, it is a system, and much of what it takes to manage a project comes from a perspective called the **systems approach**. At the same time, many projects are devoted to *creating* systems, and such projects commonly employ methodologies such as “systems analysis” and “systems engineering.” This chapter introduces systems concepts that form the basis for project management, and systems methodologies used in technical projects.

2.1 Systems and Systems Thinking

By definition, a system is “an assemblage of things or parts interacting in a coordinated way.” The parts could be physical entities such as players on a football team or components in a machine, or abstract or conceptual things, such as words in a language or steps in a procedure. Beyond “assemblage of parts,” however, a system has other features:¹

1. The parts of the system affect the system and are affected by it.
2. The assemblage of parts serves a purpose or goal.
3. The assemblage is of particular interest.

The first feature means that, in a system, the whole is greater than the sum of the parts. For example, the human body is comprised of separate components—the liver, brain, heart, nerve fibers, and so on. The parts of the body cannot live outside the body, and without the parts, the body cannot live either. The parts affect the whole, and the whole affects the parts.

The second feature of a system is that the parts work together to do something. This can usually be observed in the outputs of the system or the way the system converts inputs to outputs. In a human-made system the parts of the system interact to achieve a purpose or goal.

The third feature is that systems are conceived by people, which means that what you choose to call a system depends on what you want to look at; it depends on your purpose.² For example, in diagnosing a sick patient, a doctor may consider the *human body* and all its organs as “the system.” If the doctor suspects the illness is an intestinal infection and sends the patient

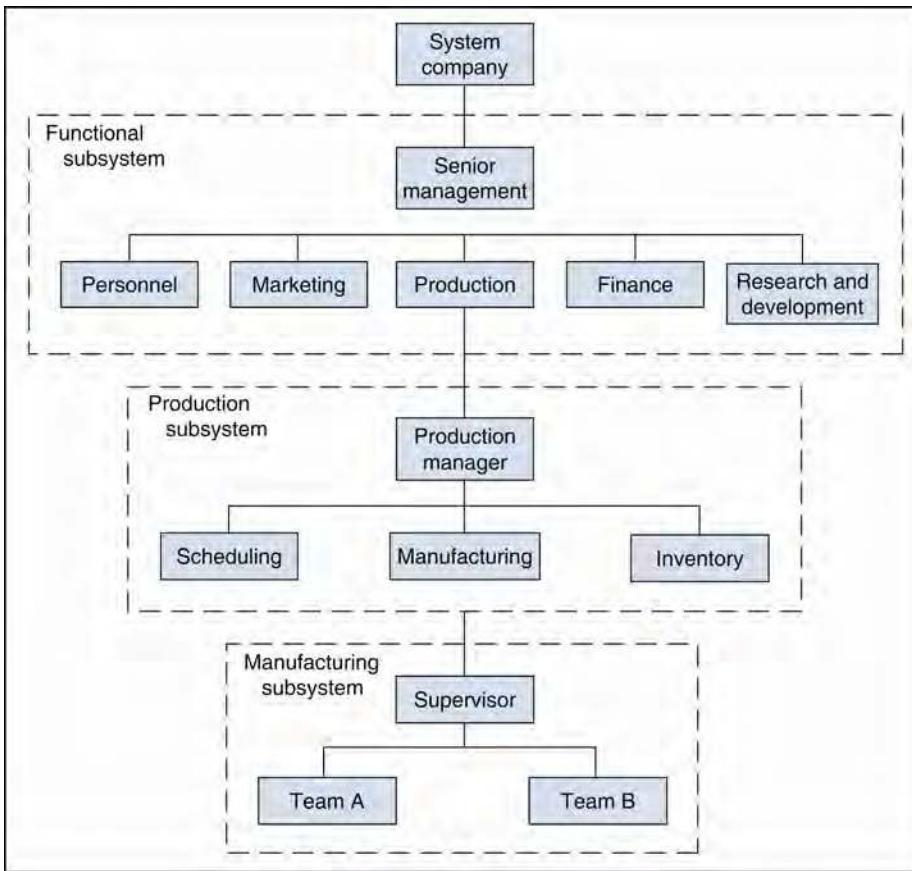


Figure 2.1

A company portrayed in terms of systems, subsystems, and elements.

to a gastroenterologist (GI), the GI will consider only the *digestive tract* of the patient as “the system.” Suppose the diagnosis is food poisoning and the patient thinks it came from a restaurant. In filing a lawsuit, the patient’s attorney will expand “the system” to include the *restaurant* where the patient ate and supposedly got sick.

Systems thinking is a particular way of viewing the world, but its key feature is a focus on the big picture—the whole system or organism—rather than just the parts of the system. Systems thinkers do look at the parts too, but they try to understand the relationships among the parts and how they contribute to the whole system.³ Systems thinking means being able to take a seemingly confused, chaotic situation and perceive a degree of order or harmony in it. As such it is a useful way of dealing with complex human-created systems and endeavors such as projects.

Project managers must be familiar with the individual parts of the project, but most responsibility for those parts lies with the managers and subject matter experts who specialize in them. Project managers are primarily concerned with the big picture, the whole project, and with all its elements working to achieve the project goal; as such, project managers are systems thinkers.

2.2 Systems Concepts and Principles

The following concepts and principles apply to all systems.

Goals and objectives

Human-made systems are designed to *do something*; they have goals and objectives that are conceived by people. For the intentions of this book, a **goal** (sometimes also called a *mission*) is defined as a broad, all-encompassing statement of the purpose of a system, and an **objective** as a more-detailed, usually quantifiable statement of purpose pertaining to some aspect of the system. The system goal is met by achieving a group of system objectives.

The goal of the project may be defined as, for example, “build a space station for \$15 billion in 10 years.” Starting with the goal, the project can then be defined in terms of many objectives such as “select overall design for the station,” “train crew,” “launch components into orbit,” “assemble components,” and so on. The objectives can be broken down into more detailed, specific objectives called **requirements**. Requirements are the specific criteria to which the system and its parts must conform for the system to meet its goals and objectives.

Elements and subsystems

Any system can be broken down into smaller parts or **elements**. These parts in combination form “the assemblage of parts” that constitutes the system. Some systems can also be broken down into parts that are also systems, called **subsystems**. A subsystem is a system that functions as an element of a larger system. When it is not necessary to understand its inner workings, a subsystem can simply be thought of as an element. Figure 2.1, a common organization chart, illustrates this: the production subsystem may be viewed as an element in the company; if we choose to delve into it, however, production itself becomes a system with elements of scheduling, manufacturing, and inventory. Each of these elements could in turn be viewed as a subsystem containing elements. In a project, an element could be a unit of work, a person or group doing the work, or a component of the final product or end-item.

Attributes

Systems, subsystems, and elements all have distinguishing characteristics called **attributes**; these describe the condition of systems, subsystems, and elements in qualitative or quantitative terms. In human-made systems, attributes are designed so the system will perform as required. Often, the attributes of a system and its components are monitored to track the system’s behavior and performance. Time and cost are universal attributes of most of elements in a project, and they are tracked to assess the project’s performance.

Environment and boundary

The term *environment* refers to anything *outside* the system that influences the system’s behavior or outcomes. In human-made systems it usually refers to things over which system designers and managers have no control. The environment can include, for example, community or

society, the atmosphere, or people associated with the project—although it is not necessarily any of these. A system is separated from its environment by a *boundary*. In many systems the boundary is obscure, and it is difficult to separate the system from its environment. To determine whether something is in the environment or in the system, ask two questions: “Can I do anything about it?” and “Is it relevant to the system and its objectives?” If the answer is “no” to the first question, then “it” is part of the environment. If it is yes to both, “it” is in the system. The table below shows how to distinguish a system from its environment:

Can system designers or managers control it?	Is it relevant to the system?	
	Yes	No
	Yes System	Irrelevant Environment
No	Environment	

“Irrelevant” includes things that do not influence the system and do not matter. To a project manager the planet Jupiter is irrelevant—unless the project is to send a space probe there, in which case Jupiter is relevant, even though the manager can’t control it, and it is part of the project environment. From here on, mention of the environment will always refer to the relevant environment—factors that matter to and affect the system in some way but must be lived with.

Although customarily anything outside the system is considered the environment, what is *inside* the system might be considered the environment as well. For example, what you see and experience *inside* a house—furniture, rooms, pictures on walls, etc., these are considered the **internal environment** of the house. Now look outside the house and what you see—foliage, trees, rain, etc., these constitute the **external environment**. A project can be viewed the same way—what is happening inside the project—people working and creating things, versus what is happening outside the project—weather, geography, society, politics, the economy, and so on.

System structure

Elements and subsystems are linked together by relationships. The form taken by the relationships is called the **system’s structure**. The functioning and effectiveness of a system is largely determined by the “appropriateness” of the structure to the system’s objective or purpose. Many complex systems have hierarchical structures consisting of organized levels.

Figure 2.2 is an example of a hierarchical structure. Element X represents the entire project; elements A, B, and C are areas of work or management divisions in the project; elements a through g are specific work tasks. The structure implies that tasks a, b, and c are all subsumed under management division A, tasks d and e are under division B, and so on. This particular structure is called a *work breakdown structure*, which is explained in Chapter 6.



See Chapter 6

Inputs, process, outputs, interfaces

Systems achieve goals and objectives by converting inputs into outputs through a defined process. This is illustrated in Figure 2.3. **Outputs** are the end-result of a system and, generally, the purpose for which the system exists. All systems have multiple outputs, including desirable

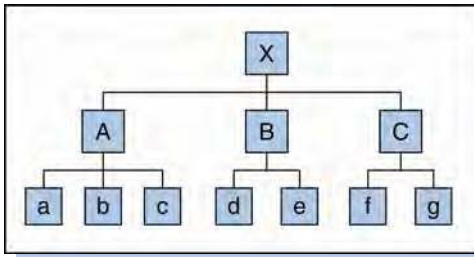


Figure 2.2
One way to conceptualize project structure.

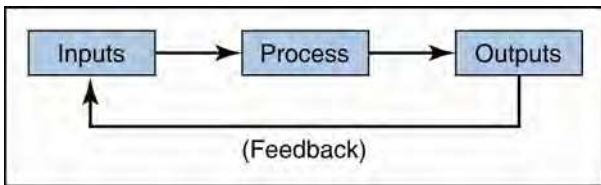


Figure 2.3
Input–process–output relationship.

ones that contribute to system objectives, neutral ones, and undesirable or wasteful ones that detract from system objectives and/or negatively impact the environment. Subsystems and most elements have inputs and outputs too.

Inputs are the raw materials, resources, or procedures necessary for the system to function and produce outputs. They include controllable factors such as labor, materials, information, capital, energy, and facilities, as well as uncontrollable factors such as weather and natural phenomena (i.e., the environment). Inputs that originate from the system itself are called *feedback*. For example, all systems produce information; usage of that information for guiding system behavior is called feedback input.

Process (also termed *function*) is how the system physically transforms inputs into outputs. An important aspect of system design is to create a process that effectively produces the desired outputs, meets system objectives, yet minimizes wasteful outputs and consumption of inputs.

In a hierarchical structure where systems are divided into subsystems, the subsystems each have their own inputs, process, and outputs that are interconnected in some way. In Figure 2.2, each of the project elements produces outputs, some of which become inputs for other elements. Two elements that fit together or function together (e.g., where the output of one becomes the input of the other) are said to **interface**.

Constraints and conflicts

All systems have **constraints** or limitations that inhibit their ability to reach goals and objectives. Often the constraints are imposed by the environment. Time and money are two universal constraints in projects. Weather and economy are two constraints imposed by the environment.

In human-made systems, and especially projects, the objectives of the subsystems sometimes conflict, which makes it difficult to achieve the overall system's goal. Removing conflict from among the system's elements to enable the system to meet its goal is called *integration*.

System integration

For a system to achieve its goal, all its elements, the “assemblage of parts,” must work in unison. Designing, implementing, and operating a system that achieves its objectives and requirements through the coordinated (so-called “seamless”) functioning of its elements and subsystems is called **system integration**. Project management seeks to integrate tasks and resources to achieve the project goal; in technological projects, it also addresses the integration of the physical components and modules that compose the project end-item. The subject of systems integration is covered in later chapters.

Closed systems and open systems

Systems can be classified as closed or open. A **closed system** is one that is viewed as self-contained with no interaction with its environment. “Closed systems thinking” means to focus on the internal operation, structure, and processes of a system without regard to the environment. For some kinds of systems, closed system thinking applies: to understand how a machine functions, you need only study the machine, its components, and not anything else. This does not mean that the environment does not affect the system, but only that the person looking at the system has chosen to ignore the environment. For analyzing or improving the design of many kinds of mechanical systems, closed system thinking works fairly well.

But what about systems that *interact* with and must be adaptive to the environment? These are **open systems**. To understand their behavior and functioning, you cannot ignore the environment. Since mechanical systems rely upon resources from and inject byproducts (e.g. pollutants) into the environment, in many cases they too should be treated as open systems. In fact, any system that must be adaptable to the environment, including projects, must be treated as an open system.

Organizations and environment⁴

As open systems, human organizations interact with stakeholders in the environment (customers, suppliers, unions, stockholders, governments, etc.) and rely upon the environment for inputs (energy, information, and raw materials). In turn, they export to the environment outputs of goods, services, and waste (represented in Figure 2.4).

As an open system, an organization must choose goals and conduct its operations to respect opportunities presented and limitations imposed by the environment. Cleland and King call this the “environmental problem,” meaning that a manager must:⁵

1. appreciate the need to assess forces in the environment
2. understand the forces that significantly affect the organization, and
3. somehow, integrate those forces into the organization’s goals, objectives, and operations.

To the extent that every project is influenced by outside forces, the project manager must try to understand these forces and, having done that, try to guide the project to its goal. A project that is primarily influenced by forces in the environment will be difficult to control and subject to failure.

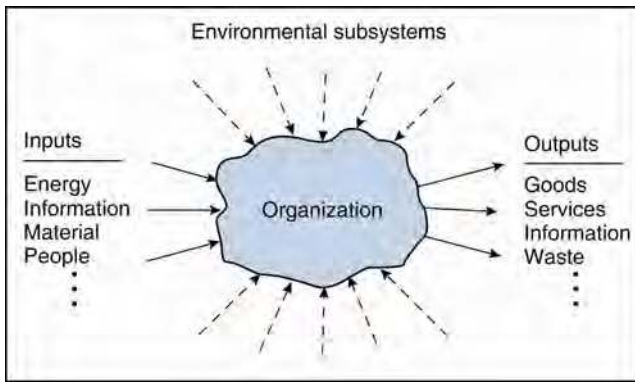


Figure 2.4
Organization as an
input–output system.

Natural systems and human-made systems

Systems can also be classified as natural or human made. **Natural systems** originate by natural processes (e.g. ecology of organisms; planetary systems). **Human-made systems** are designed and operated by people (e.g. communication systems; human organizations). Projects are human-made systems formed to create other human-made systems.

Natural systems can be altered by or become intertwined with human-made systems. An example is altering a river system and formation of a lake by building a dam; another is altering the composition of the atmosphere through CO₂ produced by internal combustion engines.

Human-made systems are embedded in and utilize inputs from natural systems, and both systems interact in significant ways. In recent years, large-scale human-made systems have had a significant, mostly undesirable, impact on the natural world. Examples include climate change, ocean acidification, air and water contamination, and species extinction. Such consequences, called **side effects**, arise largely because designers and users of human systems fail to consider (or deny or choose to ignore) the impacts of their systems on natural systems.

2.3 Systems Approach Framework and Systems Models

The **systems approach** is a way to visualize and analyze physical things and conceptual systems, but more than that it is an approach for abstracting and solving problems and making decisions.

Systems approach framework

The systems approach framework utilizes systems concepts such as goals, objectives, subsystems, elements, relationships, integration, and environment. It formally acknowledges that the behavior of any one element may affect other elements, and that no single element can perform effectively without help from the others. This recognition of **interfaces**, **interdependency**, and **cause–effect** relationships among elements of a system is what most distinguishes the systems approach.⁶

Managers who adopt the systems approach recognize the multitude of “elements” in the systems they manage, the relationships among the elements, and reciprocal influences between human-made systems and the environment. As a result, they are better able to grasp the magnitude of a problem and anticipate the consequences of their actions. This reduces the chances that important elements in a situation or consequences of actions will be overlooked.

The systems approach keeps attention on the big picture and the goal; it allows focus on the parts of the system, but only on their contributions to the whole. For instance, a university system can be viewed as separate elements of students, faculty, administrators, and alumni, and it is possible to act on any one of them while ignoring impacts on the others and the environment. But actions that focus exclusively on parts of the system are likely not optimal for the total system because they disregard negative repercussions on the other parts of the system. For example, although curtailing the hiring of faculty reduces costs, it can also lead to larger class sizes and classroom overcrowding, less faculty time for research, fewer research grants, lower prestige to the university, and ultimately, lower enrollments and less revenue.

Every problem is inextricably linked to the environment and attempts to solve it may cause other problems. Churchman calls this the “environmental fallacy.”⁷ Examples abound of situations where solutions directed at part of the system have led to worse problems for the whole. These include trying to reduce traffic congestion by building more highways, trying to eliminate drug abuse by outlawing drug sale and consumption, and trying to increase the appeal of wilderness areas by building resorts near national parks. The negative consequences of these problem-solving attempts are well known. The systems approach tries to avoid the environmental fallacy.

Orderly way of appraisal⁸

By its holistic nature, the systems approach avoids tackling problems narrowly, head-on. It says, “Let’s stand back and look at this situation from all angles.” The problem solver does this by keeping in mind the system concepts discussed, namely:

1. The goals and objectives of the system.
2. The environment of the system.
3. The resources and constraints of the system.
4. The elements of the system, their functions, attributes, and performance measures.
5. The interface and interaction among the elements.
6. The management of the system.

The systems approach starts with hardheaded thinking about the goals and objectives of the system and real ways to measure them. Project managers use this kind of thinking: they begin with the mission or goals of the system and, thereafter, organize and direct the work to achieve those objectives. The stated goals must be precise and measurable in terms of specific performance criteria (the system requirements). Criteria are the basis for ranking alternative solutions or courses of action to a problem. In a project, criteria for the end-item are referred to as user requirements and specifications, explained in later chapters.

The environment of the system (other systems, organization, and natural systems that affect or are affected by the system) must also be identified—no easy matter because external forces are sometimes hidden and work in insidious ways. Looking to the future, questions must be raised about likely changes in the environment and how they will affect the system. The project manager needs to ask, what can happen on the “outside” that will affect the project and its outcomes?

The resources to be used to accomplish system goals must be identified. These are assets or the means that the system utilizes and influences to its advantage; they include capital, labor, materials, facilities, and equipment. Most system resources are exhaustible. The system is free to utilize them only for as long as they are available or affordable. Depleted resources become constraints. In the systems approach, the project manager considers the resources needed and available to the project.

The systems approach identifies the key elements of the system—the pieces that comprise it. In a project there are actually two systems, the one being produced by the project (the project end result or end-item) and the one producing it (this is the project itself). Defining these involves defining, on the one hand, the subsystems, components, and parts of the hardware or software end-item system being produced, and, on the other hand, the work tasks, resources, organization, and procedures of the project. This topic is elaborated in Chapter 5.



See Chapter 5

Finally, the systems approach pays attention to the management of the system, that is, to its planning and control, considering its goals, objectives, environment, constraints, resources, and so on. This is precisely the role of project management.

The concepts listed are not necessarily addressed in this sequence, and each concept might need to be dealt with several times before it is completely described and clearly defined. More importantly, each concept serves to suggest numerous open-ended questions that aid in investigating the system:⁹ What are the goals and criteria? What are the elements and the relationships among them? What functions should each perform? What are the resources?

Systems models

The output of a system depends not only on the system's elements, but on the way the elements interact. Thus, designing a system or resolving problems in a human-made or natural system requires understanding the way the elements interact. Designers use models of the system to help understand how the elements interact and how altering the elements and their relationships impact system behavior and outputs. A model is a simplified representation of a system that abstracts the essential features of the system under study. It may be a physical model, mathematical formulation, computer simulation, or simple checklist.

An example of a **physical model** is a model airplane. It is a scaled-down abstraction of the real system. It includes some aspects of the system (configuration and shape of main components—wings, fuselage, tail) and excludes others (interior components and crew members). Another kind of model is a **conceptual model**; it depicts the elements, structure, and interactions in a system. The conceptual model in Figure 2.5, for example, helps demographers to understand relationships among the elements contributing to population size and to make population predictions.¹⁰

Models are used to conduct experimentation and tests. Many human-made systems are too expensive or risky to do “real-life” experiments on. The model permits assessment of various alternatives and their consequences before committing to a decision. For instance, engineers use model airplanes in wind tunnels to test design alternatives and measure the effects of different design parameters on airplane performance. A good model allows analysts to ask “what if” questions and explore the effects of altering the inputs; it accounts for the requirements, elements, resources, and constraints, and allows the consequences of different alternatives to be compared in terms of costs and benefits. Models employed for quality assurance are discussed in Chapter 10.



See Chapter 10

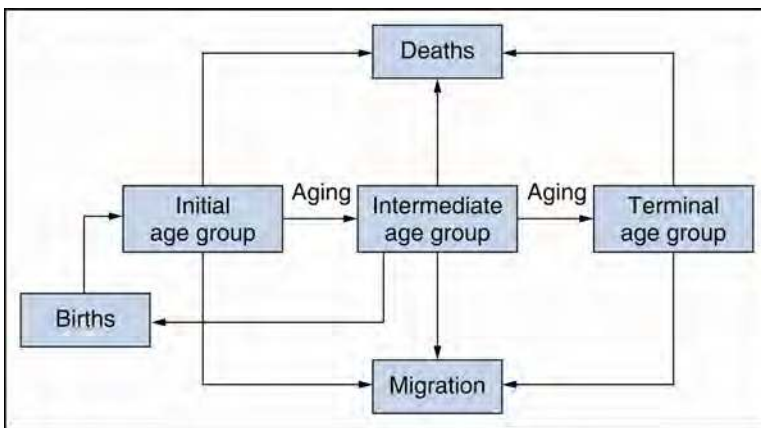


Figure 2.5
A generalized
population
sector model.

Systems life cycles

Natural and human-made systems change over time in a way that tends to be systematic and evolutionary, and similar kinds of systems follow similar evolutionary life cycles. One cycle, that of all organisms, is the pattern of conception, birth, growth, maturity, senescence, and death. Each of these can be thought of as a life “stage.” Historically, even civilizations and societies have followed this pattern. Nonliving, electro-mechanical systems also follow a cycle with the stages of design, fabrication, installation, burn-in, normal operation, and deterioration or obsolescence. Similarly, all products follow a cycle—the **product life cycle**, which consists of the stages of development, introduction, growth, maturity, and decline. Products such as cell phones may have life cycles of only a year or two; others like Kool-Aid and jeans are decades-long—their technology and popularity change little and are long-lasting.¹¹ As mentioned in Chapter 1, virtually all human-made systems start out as projects, and most projects follow a cycle called the **project life cycle**.¹² This is discussed in Chapter 3.



See Chapters 3 a cycle called the **project life cycle**.¹² This is discussed in Chapter 3.

2.4 Systems Engineering

Systems engineering (SE) is an application of the systems approach. It is defined as “the science of designing complex systems in their totality to ensure that the component subsystems making up the system are designed, fitted together, checked and operated in the most efficient way.”¹³ SE refers to the conception, design, and development of complex systems wherein the components of the systems themselves must be designed, developed, and integrated together to meet system objectives. It is a way to bring an *entire* system into being and to account for its entire life cycle—including operation and phase-out—during its early conception and design.

SE overview¹⁴

SE can be applied to any system that must be developed (perhaps from scratch), implemented, and operated to fulfill an immediate or future goal. Examples can readily be found in the design and implementation of local, national, and global systems for communication, transportation, water supply, power generation and transmission, and defense.

SE can be described in terms of three dimensions as illustrated in Figure 2.6. First, it is a **multidisciplinary** effort. Systems engineers (parties responsible for oversight of designing

Example 2.1: All Systems Go

An example of SE is the design and operation of a space vehicle. The expression “all systems go,” popularized during the early US space program, means that the overall system comprised of the millions of components that make up the space vehicle, launch rocket, and supporting systems, and the hundreds of people in its technical and management teams are all ready to “go” to achieve the mission objectives.

To get to the point of “all systems go” planners must first have defined the overall system and its objectives. Designers must have analyzed the requirements of the system and broken them down into more detailed requirements, and have designed the components and subsystems so as to meet those requirements. They must then have built and combined the components into subsystems, and the subsystem into the total system comprised of space vehicle, rocket boosters, launch facilities, ground support, crew selection and training, and technical and management capability. In the end, every component and person must be assigned a role and be *integrated* into a subsystem that has been integrated into the overall system.

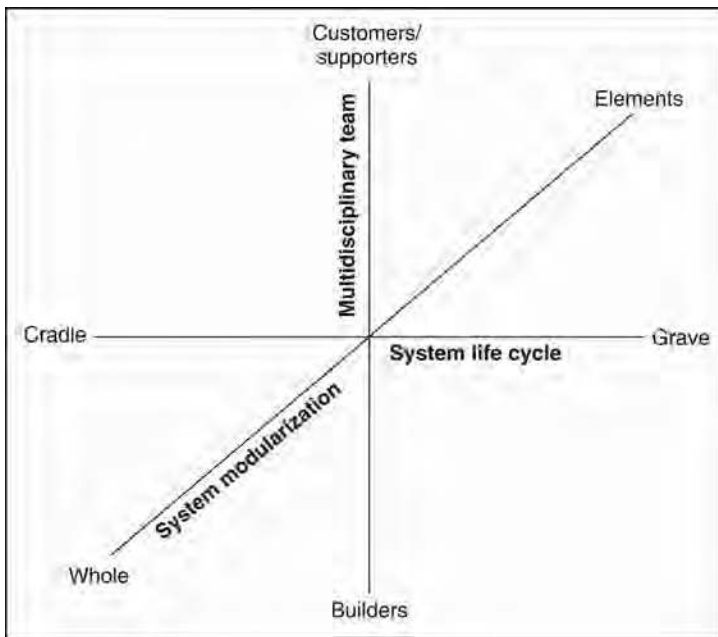


Figure 2.6
Dimensions of systems engineering.

and building the system) work with the system’s **stakeholders** to determine their needs and what the system must do to fulfill the needs. A stakeholder is any individual or group that affects or is affected by the system; primary stakeholders are customers, builders, and end users. Customers finance and own the system; builders design and create it; users operate and maintain it. Stakeholders’ objectives and needs are the basis for determining the system



See Chapters 4
and 15

requirements regarding what the system will do. The practice of involving key stakeholders in the early phases of a system's design and development, called **concurrent engineering**, is discussed in Chapters 4 and 15.

Second, SE addresses every aspect of the system, starting with the whole system and ending with its individual elements. System elements, modules, and subsystems are designed to perform the functions necessary to satisfy the objectives and requirements of the whole system. None of the elements and subsystems function independently; they all rely on the outputs of other elements and, in turn, provide inputs to still others; in a word, they **interface**. SE addresses the necessary interactions between elements, and how the elements should interface.

Third, SE also considers how the system will be produced, operated, maintained, and ultimately disposed of—the system's life cycle, cradle to grave. This helps ensure that the system will be economical to develop, build, operate, and maintain, and friendly to users and the environment. A multidisciplinary team-approach involving all the system's key stakeholders promotes this life cycle kind of thinking.

Once systems engineers have learned what stakeholders want and defined the system's objectives and requirements, they then investigate ways to meet the requirements. This involves research, analysis, and studies of alternative system designs, and the estimated costs, schedules, risks, and benefits associated with each design. Says Brooks, "The hardest part of building a [system] is deciding precisely what to build. No other part of the conceptual work is so difficult as establishing the detailed technical requirements [and] no other part of the work is so crippling to the resulting system if done wrong. No other part is more difficult to rectify later."¹⁵

Example 2.2: Advanced Automation System¹⁶

The centerpiece of the Federal Aviation Administration's (FAA) program to modernize the air traffic control system was the Advanced Automation System (AAS), which would provide controllers with new displays and computer equipment for processing radar and flight data. The FAA awarded the contract for AAS to IBM following a 4-year design competition. Requirements from the FAA initially filled a thick book, but as the program progressed, they kept increasing and eventually grew to a stack 20 feet high. As the number of requirements grew, so did program delays, costs, and tensions between the FAA and IBM. Congress balked, and after 10 years and an estimated \$1.5 billion it cancelled the program.

Eliciting the expectations and needs of operators and users and then translating them into measurable requirements can be difficult for engineers, which is why multidisciplinary SE teams sometimes include behaviorists and psychologists. Developing the flight deck for a commercial aircraft, for example, should include the suggestions of pilots, the airlines, pilot associations, and human factors experts. A common way to elicit responses to or suggestions about a proposed design is for users to try out a mockup or simulator of the system.

Modularization: iterative analysis–synthesis–evaluation cycle¹⁷

Systems are designed and assembled from subsystems that are themselves designed and assembled from subsystems, and so on, so the process of defining what will go into the system is a series of steps to define the subsystems and elements that will comprise the system. The

practice of forming systems as a collection of subsystems, called **modularization**, is what makes the design, assembly, and operation of complex systems feasible and practical. Herbert Simon gives the example of a watchmaker who assembles a watch of 100 parts. The process requires concentration and is time-consuming and expensive. If the watch should need repair, finding and fixing the problem might be difficult. If, instead, the watch were made of ten modules, each with ten parts, assembly will be simple. If the watch develops a problem, the repair will be simple: just identify the module with the malfunction and replace it.¹⁸

The design of systems occurs by designing subsystems and modules that each performs a necessary function of the system. Functions are how a system meets its objectives and requirements. In everyday systems it is easy to identify the modules and the functions they perform. A desktop computer, for example, is almost completely modularized: it has a processor, controllers, and drives, and peripheral devices that each performs a specialized function such as data processing, data storage, and input/output processing.

The process of defining the functions a system and its subsystems and modules must perform is illustrated by Forsberg and Mooz's two-stroke **V-model** in Figure 2.7.¹⁹ It involves iterative cycles of (1) top-down analysis of details (i.e. decompose the system into smaller pieces), (2) bottom-up synthesis (build up and integrate the pieces into successively larger parts), and (3) evaluation (check that the resulting pieces meet requirements).

1. The **downstroke** of the V represents subdividing the functions of the system into sub-functions and requirements, level by level. At each level, designers address the question: "What must the functions at this level do to meet the requirements of the function at the next higher-level?" In this way, requirements are defined for functions at all levels.

The way in which system functions are arranged into modules is called the **system architecture**. The architecture of an airplane is an example: an airplane must perform several major functions including propulsion, lift, and payload stowage, which are served by the visibly familiar modules of engines, wings, and fuselage, respectively. But each function is itself a composite of several subfunctions, hence each module is

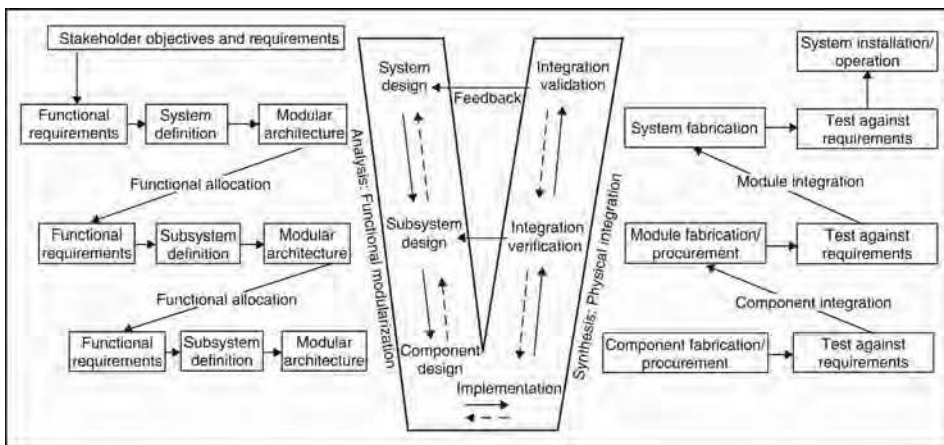


Figure 2.7
Forsberg and Mooz's V-model.

Source: Adapted from Forsberg K. and Mooz H. *Software Requirements Engineering*, 2nd edn, Taylor R., Dorfman M., and Davis A. (eds). Los Alamitos, CA: IEEE Computer Society Press; 1997, pp. 44–77.

comprised of submodules. A wing, for example, is comprised of ailerons, flaps, spoilers, etc., each of which performs a specific aerodynamic function.

2. The **upstroke** of the V represents implementing design decisions, converting designs into physical parts (by either procuring or building them), integrating the parts, and verifying that the integrated parts meet the requirements.
3. Components are checked individually and then assembled into modules; the modules are tested and then combined with others and tested again. If tests reveal that parts or modules do not meet requirements, then the process returns to the downstroke to determine why, and the analysis–synthesis–evaluation cycle repeats. As illustrated by the many dashed arrows in Figure 2.7, the process moves back and forth within each downstroke/upstroke; and at times during the upstroke, it loops back and over to the downstroke.

One rule of the systems approach is “Don’t rush to solutions! Look for alternatives.” Multidisciplinary teams are good at this: they combine knowledge from experts in disparate areas and generate alternatives that transcend any one person’s or field’s area of expertise.



See Chapters 3
and 4

The design and development of complex technical systems can be vexing, but SE offers a way to do it. In practice, SE follows a process similar to the project life cycle, described in Chapter 3, and it employs practices for defining systems, described in Chapter 4. But whereas the project life cycle applies to generic projects, SE applies more specifically to complex, usually technical projects. The stages of SE and the tools used to define functions and requirements are described in more depth in Appendix A to Chapter 4.

2.5 Project Management: A Systems Approach²⁰

Project management is a systems approach to management: its focus is on achieving the overall goal and objectives of the project; it emphasizes decisions that optimize the overall project rather than the elements of the project; and it recognizes interaction and synergy among working elements of the project. This contrasts with more traditional management, which tends to focus narrowly on individual functions and tasks, and on decisions that optimize the performance of individual departments, even if at the expense of the larger organization.

In *Winning at Project Management*, author Robert Gilbreath²¹ describes the “right” way to visualize a project. From an outsider’s perspective, he says, a project may look like something with no discernable parts, like a barrel containing thousands of earthworms. Obviously, if you must manage a project, such a perspective is not very useful. A better perspective is one that subdivides the mess into a collection of elements and defines the characteristics of each.²² Good project managers, says Gilbreath, conceptually subdivide the project into pieces and make sure each piece is well managed. Additionally, they know how the pieces impact each other and the overall project.

Gilbreath mentions another feature of project managers: the ability to “change focus,” to zoom-in on the performance of discrete elements, then zoom-out and check the direction and performance of the overall project. The zoom-out view enables the project manager to direct the project toward its goals and not get hung-up with the pieces.²³ The ability to zoom-in and zoom-out, to see and know what is important to the big picture—that is the essence of the systems approach. It does not matter whether you call it the “systems approach” or not, the point is, when managing a project, it helps to think of it as a system.

2.6 Summary

A system is an assembly of parts where (1) the parts are affected by being in the system, (2) the assembly does something, and (3) the assembly is of particular interest. What is called “the system” depends upon one’s point of view and purpose. Projects are systems created for the purpose of making systems.

Systems thinking is a way to deal with complex phenomena. It imparts the ability to impose order and structure on a seemingly confused or chaotic situation. Systems thinking includes the “systems approach,” which is a way of conceptualizing physical entities and addressing problems. The principal components of the systems approach are (1) the objectives and performance criteria of the system, (2) system environment and constraints, (3) resources of the system, (4) elements of the system, their functions, attributes, and performance measures, (5) interaction among the elements, and (6) management of the system. For development and operation of large technical systems, the systems approach is implemented through the systems engineering methodology.

Part I of this book gives you an overview of project management. Projects are temporary—they have a beginning and an ending. What happens in between—the stages of tasks and activities—tends to be remarkably similar, regardless of the kind of project. These stages are analogous to stages in the system life cycle and were alluded to in the examples in Chapter 1. Part II discusses these stages and describes a framework for conducting projects: the systems development cycle.



Review Questions

1. What distinguishes systems thinking from analytical thinking? Is systems thinking something new or is it just another perspective? Explain.
2. Define “system.” What notable features enable you to see something as a system? Describe briefly the American legal or education system in terms of these features.
3. How can several people looking at the same thing see the “system” in it differently?
4. Define the following concepts and explain how they fit into systems thinking: objectives, elements, subsystems, attributes, environment, boundary, structure, inputs, outputs, process, and constraints.
5. Describe the difference between open and closed systems, and between human-made and natural systems. Are all natural systems open systems?
6. Is a space vehicle an open system? Is an organization an open system? Explain.
7. Describe the systems approach. Where does the systems approach apply? Explain in a sentence what a manager does in the systems approach that she might not do otherwise.
8. What is the “environmental fallacy”?
9. What things does the problem solver keep in mind when applying the systems approach?
10. Describe how the following elements of the systems approach apply to projects and project management: objectives, environment, resources, subsystems, and management.
11. Give some examples of physical models; of graphical models; of mathematical models.
12. What is the systems life cycle? What is the systems development cycle?
13. Discuss the dimension of systems engineering in Figure 2.6.
14. What is modularization? What are its benefits in system design and operation?
15. In systems engineering the first stage is identification. Identification of what?
16. Who are the stakeholders in systems engineering?

17. What are the requirements? What aspects of the system or stakeholder needs should the requirements incorporate?
18. Distinguish stakeholder requirements and system requirements.
19. Why is project management a systems approach?
20. What is the relevance of the systems approach to project management?



Questions About the Study Project

1. Conceptualize the project organization (the project team and the parent organization of the team) you are studying as a system. What are the elements, attributes, environment, and so on? What are its internal subsystems—functional breakdown and management-hierarchy subsystems? What is the relevant environment? Who are the decision makers?
2. Describe the role of the project manager with respect to these subsystems, both internal and external. What is the nature of his or her responsibilities in these subsystems? How aware is the project manager of the project “environment” and what does he or she do that reflects this awareness?
3. Now, conceptualize the output or end-item of the project as a system. Again, focus on the elements, relationships, attributes, subsystems, environment, and so on. All projects, whether directed at making a physical product (e.g. computer, space station, skyscraper, research report) or a service (e.g. giving consultation and advice), are devoted to producing systems. This exercise will help you better understand what the project is doing. It is also good preparation for topics in the next chapter.
4. If the study project involves engineering or integration of many components, was the systems engineering (SE) process used? Is there a section, department, or task in the project called SE? If so, elaborate. Are there aspects of the project that seem to resemble the SE process?
5. As described in this chapter, besides the main end-item or operating system (i.e., the output objective of the project), SE also addresses the support system—that system which supports installation, operation, maintenance, evaluation, and enhancement of the operating system. Describe the support system in the study project and its development.
6. Were the stakeholder requirements clearly defined at the start of the project? Were system requirements clearly defined? What are the requirements? In your opinion, were stakeholders identified and involved early in the project? Were their needs identified and addressed? Did the project deliver a system that met their needs?

CASE 2.1 GLADES COUNTY SANITARY DISTRICT

Glades County is a region on the Gulf Coast with a population of 600,000. About 90 percent of the population resides in and near the city of Sitkus. The main attractions of the area are its clean, sandy beaches and nearby fishing. Resorts, restaurants, hotels, retailers, and the Sitkus/Glades County economy in general rely on these attractions for tourist dollars.

In the last decade, Glades County has experienced a near doubling of population and industry. One result has been the noticeable increase in the level of water pollution along the coast due to the increased raw sewage dumped by Glades County into the Gulf. Ordinarily, the Glades County sewer system directs effluent waste through filtration plants before pumping it into the Gulf. Although the Glades County Sanitary District (GCSD) usually can handle the county's sewage, during heavy rains the runoff from paved surfaces exceeds sewer capacity and must be diverted past filtration plants and directly into the Gulf. Following heavy rains, the beaches are cluttered with dead fish and debris. The Gulf fishing trade is also affected since pollution drives away desirable fish. Recently, the water pollution level has become high enough to damage both the tourist and fishing trade. Besides coastal pollution, there is also concern that as the population continues to increase, the county's primary fresh water source, Glades River, will also become polluted.

The GCSD has been mandated to prepare a comprehensive water waste management program that will reverse the trend in pollution along the Gulf Coast as well as handle the expected increase in effluent wastes over the next 20 years. Although not yet specified, it is known that the program will include new sewers, filtration plants, and stricter anti-pollution laws. As a first step, GCSD must establish the overall direction and mission of the program.

QUESTIONS

Answer the following questions [given the limited information, it is okay to advance some logical guesses; if you are not able to answer a question for lack of information, indicate how and where, as a systems engineer, you would get it]:

1. What is the system? What are its key elements and subsystems? What are the boundaries and how are they determined? What is the environment?
2. Who are the decision makers?
3. What is the problem? Carefully formulate it.
4. Define the overall objective of the water waste management program. Because the program is wide-ranging in scope, you should break this down into several sub-objectives.
5. Define the criteria measures of performance to determine whether the objectives of the program are being met. Specify several criteria for each sub-objective. As much as possible, the criteria should be quantitative, although some qualitative measures should also be included. How will you know if the criteria that you define are the appropriate ones to use?
6. What are the resources and constraints?
7. Elaborate on the kinds of alternatives and range of solutions to solving the problem.
8. Discuss some techniques that could be used to help evaluate which alternatives are best.

CASE 2.2 JUBILEE LINE EXTENSION PROJECT²⁴

The Jubilee Line Extension Project (JLEP) was an expansion of the London Underground (LU) system. It expanded LU through six London boroughs, linking Westminster to Docklands and Stratford. The project actually comprised 30 projects (i.e., it was a “program”) that included 22 km of tunnels, five underwater crossings, 11 new stations, and complex installations of machinery and equipment. Everywhere care had to be taken to ensure the safety of over 30 buildings in Central London. JLEP in many ways mirrors another large underground project—Boston’s Big Dig (see Cases 10.1 and 16.3).

Started in 1993 for an estimated £2.1 billion cost, JLEP was completed in December 1999, 20 months behind schedule and over £1.4 billion over budget (at the time, the most expensive project in the world). Four major events contributed to the overruns:²⁵

- Work stoppage to secure private sector funding.
- Collapse of express tunnels at Heathrow Airport, which utilized the same tunneling method in JLEP and necessitated a complete safety review of the method.
- Failure of the new signaling system.
- Decision to site the Millennium Dome at Greenwich, for which JLE was to be a major source for access.

Other contributors were the differences in contracts and resulting ambiguities over roles and responsibilities of the involved parties. Two kinds of contracts were used; one was based upon payment schedules and milestones, the other upon design and performance specifications. These differences later proved incompatible.

JLEP required significant design changes throughout the project; many of them were poorly controlled and managed or were approved post facto. Differences between early proposed designs and working design drawings were poorly communicated, and many designs were “frozen” by engineering and architectural groups even though elements of the design were still in the conceptual stage. Construction contractors were minimally involved in the design. The project team, which faced political pressure to complete JLE in time to serve as a main transport link to the Millennium Dome, which was then in construction, set an overly ambitious project deadline of 53 months.

The project was managed by a project director and project team. Contractors were chosen independently and interfaces between them were not defined. This led to confusion and left the project team with the substantial task of managing all contractor interfaces and coordinating their work. Sizable changes in design and lack of clear targets and milestones led to difficulties in monitoring progress and applying the milestone payment system.

London Underground management treated JLE as a “bolt on” to the existing railway line, i.e. it treated JLE as almost independent of existing transportation lines and communication systems to which it would be linked. The project team actually took the view that *existing LU lines had nothing to do with them*. Even though JLE would substantially increase the size of the LU system—and impact the system and be impacted by it—LU management viewed JLEP as simply a construction project whose ultimate operation would be independent of the overall LU system. Only a relatively small amount was budgeted to other parts of the LU to handle increased passenger traffic resulting from JLE. Early planning of JLE did not fully address operational issues, and it was *more than a year after* the project started that a plan for the operation of JLE was first addressed.

The project was originally scheduled to go “online” all at once; only much later after setbacks was it decided that JLE would open in a phased manner.

JLEP was completed with no fatalities and has been successful in relieving congestion; several of its stations have received awards for architectural design; and JLE is cited as a contributor to the success of the 2012 London Olympic Games. But it was completed for £3.5 billion instead of the budgeted £2.1 billion and in 73 months instead of the planned 53 months, this despite a substantial reduction in its scope (replacing an intended new technology signaling system with a traditional system).

QUESTIONS

1. The case illustrates a situation where the systems approach to design and management is necessary. Why is it necessary?
2. Is there evidence in the case to demonstrate that JLE planners and management used the systems approach or systems engineering? In your discussion, consider the following: JLE as a “system,” stakeholders’ identification and needs identification, requirements definition, interface management, and system operation.

CASE 2.3 SANTA CLARA COUNTY TRAFFIC OPERATIONS SYSTEM AND SIGNAL COORDINATION PROJECT²⁶

The road infrastructure of Santa Clara County consists of (1) freeways managed by the state of California, (2) city streets and highways managed by individual municipalities, and (3) limited-access highways and signalized intersections managed by the county. The county conducted a study of the feasibility of integrating all of these traffic operations and signaling systems into one Intelligent Transportation System (ITS). The ITS would upgrade the county’s Traffic Operations Center, traffic signal system, communications, and intersection surveillance, and provide communication links with municipal control centers. The study identified interfaces among the disparate systems and described the ITS architecture. The project began in 1998 and within the legislated budget and 7-year timeframe the ITS was fully operational.

Among challenges experienced during the project were:

- Rapid changes in video-camera and video-transmission technology for traffic surveillance.
- Availability of new technologies to allow traffic signaling and ITS communication systems to transition from analog to digital Internet protocol.
- The “dot.com” boom, which affected the supply of fiber optic cabling and led to an 18-month delivery schedule and a potential doubling of costs.

A post-hoc analysis of the project conducted by the INCOSE Transportation Working Group concluded that the project’s management had taken the following major steps:

- Created a clear statement of the operational concept.
- Developed system requirements.

- Controlled the revision of the requirements during the design and construction phases to accommodate changes in technology.
- Clearly defined during the design stage the verification tests necessary for acceptance of sub-systems.
- Defined early in the project the performance measures to be used in system validation.

The INCOSE group also concluded that the project's management had effectively used risk management planning, especially regarding potential delivery delays due to shortages of fiber optic cable. Soon after the communications requirements were declared fixed, the client initiated procurement of the fiber cable and processes to incorporate the cable into construction contracts.

During system design and implementation, senior staff (both client and consultant) reviewed the user requirements and revised the design concept, which removed technological biases in requirements and made it possible to accommodate later revisions in technology.

Twelve years after the start of the project, communications protocols had changed. The modularity of the system design, however, enabled the system to be upgraded in stages without changing the equipment or underground communication infrastructure.

QUESTION

Although the project team did not intentionally set out to follow the systems approach, much of what it did, in fact, conformed to systems engineering practices. Compare the limited information provided about the project with the systems engineering concepts and V-model described in the chapter.

Notes

1. Naughton J. and Peters G. *Systems Performance: Human Factors and Systems Failures*. Milton Keynes, UK: The Open University; 1976, pp. 8–12.
2. Innumerable systems can be perceived from any one entity. K. Boulding, in *The World as a Total System* (Beverly Hills, CA: Sage, 1985) describes the world as physical, biological, social, economic, political, communication, and evaluative systems.
3. Schoderbek P., Kefalas A. and Schoderbek C. *Management Systems: Conceptual Considerations*. Dallas: Business Publications; 1975, pp. 7–8.
4. Kast F. and Rosenzweig J. The Modern View: A Systems Approach. In Beishon J. and Peters G. (eds), *Systems Behavior*, 2nd edn. London, UK: Harper & Row; 1976, pp. 19–25.
5. Cleland D. and King W. *Management: A Systems Approach*. New York, NY: McGraw-Hill; 1972, p. 89.
6. Churchman C.W. *The Systems Approach and Its Enemies*. New York, NY: Basic Books, 1979.
7. Churchman, *The Systems Approach and Its Enemies*, pp. 4–5.
8. Much of the discussion in this section is based on Churchman C.W. *The Systems Approach*. New York: Dell; 1968, pp. 30–39.
9. Thome P. and Willard R. The Systems Approach: A Unified Concept of Planning. In Optner S. (ed.), *Systems Analysis*. Harmondsworth, UK: Penguin Books; 1973, p. 212.
10. Hamilton H. *Systems Simulation for Regional Analysis*. Cambridge, MA: The MIT Press; 1972.
11. The life cycle of technological products is eloquently described by Foster R. in *Innovation: The Attacker's Advantage*. New York, NY: Summit Books; 1986.

12. As common parlance, the term *project life cycle* is recognition that all projects tend to follow a similar sequence of activities, start to finish. Since every project, however, has a start and finish, when referring to a particular project the more precise term is *project life span*.
13. Jenkins G. The Systems Approach. In Beishan J. and Peters G. (eds), *Systems Behavior*, 2nd edn. London, UK: Harper and Row; 1976, p. 82.
14. Auyang S. *Engineering—An Endless Frontier*. Cambridge, MA: Harvard University Press; 2004, pp. 175–189.
15. Brooks F. *The Mythical Man Month*. Reading, MA: Addison Wesley; 1995, p. 199.
16. Auyang, *Engineering—An Endless Frontier*, p. 183.
17. Auyang, *Engineering—An Endless Frontier*, pp. 192–197.
18. Herbert Simon, quoted in Auyang, *Engineering—An Endless Frontier*, p. 194.
19. Forsberg K. and Mooz H. In Taylor R., Dorfman M. and Davis A. (eds), *Software Requirements Engineering*, 2nd edn. Los Alamitos, CA: IEEE Computer Society Press; 1997, pp. 44–77; V-model adapted from reprint in Auyang, *Engineering—An Endless Frontier*, p. 197.
20. Cleland and King, *Management: A Systems Approach*, pp. 171–173; and Johnson R., Kast K. and Rosenzweig J. *The Theory and Management of Systems*, 3rd edn. New York, NY: McGraw-Hill; 1973, pp. 135–136.
21. Gilbreath R. *Winning at Project Management*. New York, NY: John Wiley and Sons; 1986.
22. Gilbreath R. *Winning at Project Management*, pp. 95–96.
23. Gilbreath R. *Winning at Project Management*, pp. 98–102.
24. Information for this case is derived from two sources, last accessed June 18, 2024: (1) *Project Profile, Jubilee Line Extension*. Omega Centre, University College London (no date), https://www.omegacentre.bartlett.ucl.ac.uk/wp-content/uploads/2014/12/UK_JLE_PROFILE.pdf; (2) Systems Engineering Case Study # 8 Jubilee Line Extension, *Systems Engineering in Transportation Projects: A Library of Case Studies*, INCOSE Transportation Working Group, Dec. 6, 2014, https://www.incose.org/docs/default-source/TWG-Documents/incose-twg-case-study-library-7_0.pdf
25. Mitchell, B. *Jubilee Line Extension: From Conception to Completion*. London, UK: Thomas Telford Publishing; 2003.
26. Adapted from Systems Engineering Case Study # 7 Santa Clara County Traffic Operations System and Signal Coordination Project, *Systems Engineering in Transportation Projects: A Library of Case Studies*, INCOSE Transportation Working Group, Dec. 6, 2014, https://www.incose.org/docs/default-source/TWG-Documents/incose-twg-case-study-library-7_0.pdf; accessed June 18, 2024.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Part II

Project Life Cycle

3	Project Life Cycle and Project Conception	67
4	Project Definition and System Definition	105
5	Project Execution and Closeout	153

Most systems move through a series of developmental stages. In human-made systems, the developmental stages follow an intentional, logical sequence of prescribed activities called the systems development cycle. Project management occurs within this cycle and is the function responsible for planning the work activities and organizing and guiding their execution. The three chapters in this section introduce the systems development cycle and describe its first three phases: conception, definition, and execution/closeout.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Chapter 3

Project Life Cycle and Project Conception

One feature of the systems approach is the concept of “life cycle”—the basic pattern of change that occurs throughout the life of a system. Two ways the systems approach accounts for this are to (1) recognize the *natural process* that occurs in all dynamic systems—that of birth, growth, maturity, and death, and (2) incorporate that process into the planning and management of systems. The practice of project management does both.

The process of developing, implementing, and operating any human-made system follows a logical sequence of phases called the systems development cycle. Projects also follow a sequence of phases from beginning to end called the project life cycle. This chapter describes the system development and project life cycles and the first phase of both, conception.

3.1 Project Life Cycle

Systems are dynamic—they change over time. Often, the change follows a distinct pattern repeated. Mentioned in Chapter 2 was the life cycle of organisms—birth, growth, maturity, senescence, and death, and its similarity to cycles in human-made products and systems.

Similarly, projects follow a cycle called the **project life cycle**. Each project has a starting point and progresses toward a predetermined conclusion; during this time, work effort in the project grows, reaches a peak, and then declines—the pattern shown in the lower curve in Figure 3.1. The upper curve shows the cumulative work effort. Effort can be measured in various ways such as the amount of money being spent on the project and materials being used, number of people working on it, and so on.

Besides the level of work effort changing, the nature and emphasis of the effort change too, and so do the people involved. For example, customers and planners are active early in the project; then designers and builders take charge, then at the end, users and operators take over.

Managing the project life cycle requires special treatment. Unlike non-project operations where everything tends to be somewhat familiar and stable, many things in projects—resources, schedules, work tasks, etc.—are either unfamiliar or in a constant state of change. Much of what is done in a project can be considered non-repetitive or non-routine. Work schedules, budgets, and tasks must be uniquely tailored to fit the work. Unforeseen obstacles can cause missed deadlines, cost overruns, and poor project performance. Managers must try

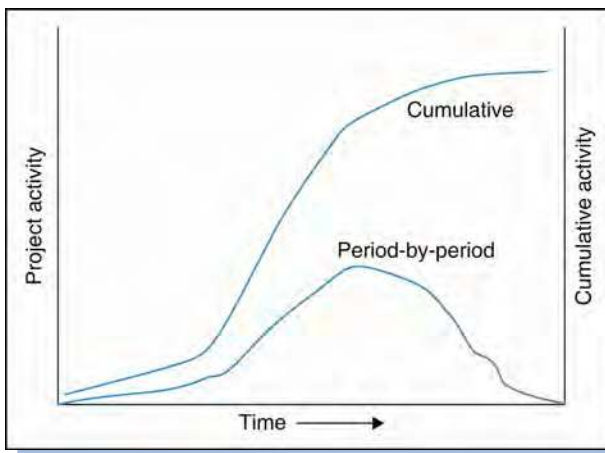


Figure 3.1
Level of effort during the project life cycle.

to anticipate the problems, plan for them, and adjust activities and shift resources to mitigate or overcome them.

3.2 Systems Development Cycle

The project life cycle is part of a larger life cycle called the **systems development cycle, SDC**, that virtually all human-made systems follow. The SDC has four phases (Figure 3.2):

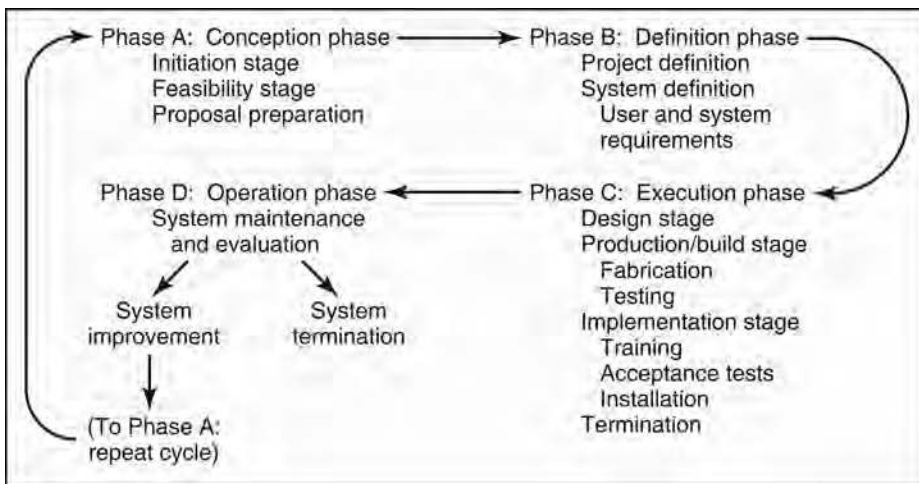
1. Phase A: Conception phase
2. Phase B: Definition phase
3. Phase C: Execution phase
4. Phase D: Operation phase

An important part of the SDC (and, from the perspective of this book, the most important part) is the previously mentioned project life cycle, which spans Phases A, B, and C. Virtually all human-made systems and products start out as projects; when Phase C ends, so does the project, after which the project end-item or result becomes an operational entity or product, which is Phase D of the SDC.¹

Phase A: Conception

Every project is an attempt to solve a problem or fulfill a need, so a first step in system development is **conception**—recognizing that the problem or need exists. After that, those facing the problem—the customers, users, stakeholders—seek out someone who can help. The steps they take—soliciting contractors who can do the work, reviewing the contractors’ proposals, and selecting a contractor—are part of the procurement management process.

If the customer is an organization that has someone internally capable of doing the work, it turns to them. If not, it turns to outside contractors, possibly by sending them a formal

**Figure 3.2**

Four-phase model and stages of the systems development cycle. The project life cycle is Phases A, B, and C.

request for help called a **request for proposal**, or **RFP**. Each contractor examines the customer's problem and requirements as stated in the RFP and determines the technical and economic feasibility of undertaking the project. If the contractor decides to respond to the request, it sends the customer its proposed solution (or "system concept") in a formal **proposal** or **letter of interest**. (Because of this, Phase A is sometimes also called the **proposal phase**.) The customer then examines the proposals from all the contractors and makes a choice. The result is a formal agreement or contract between the customer and the chosen contractor. Most ideas or proposals never go beyond Phase A because the problems addressed are judged as insignificant or the proposals as impractical, infeasible, or lacking benefits to justify funding and resources. The few that are approved and reach a contractual agreement move on to Phase B. All of this is discussed in depth later in the chapter.

Phase B: Definition

Having reached agreement with the customer, the contractor begins a detailed analysis and **definition** of the system concept, during which it defines requirements that the system must fulfill to meet the customer's needs, and the necessary functions and elements of the system to meet those requirements. The result is a preliminary design for the system. As the process continues, the major subsystems, components, and support systems of the proposed system are determined, as are the resources, costs, and schedules necessary to build the system. Meanwhile, as part of the definition phase, project management assembles a comprehensive project plan that defines the work activities, schedules, budgets, and resources to design, build, and implement the system.

In some industries the tasks in Phases A and B are referred to as "front-end loading" (FEL) or "front-end planning," which refers to everything that happens in a project prior to the execution of work in Phase C. FEL is discussed in Chapter 4.



See Chapter 4

Phase C: Execution

Execution is when the work as specified in the project plan is carried out; the phase is sometimes also referred to as the “acquisition” phase because most system resources are acquired then, and following the phase the user acquires the end-item system.

The execution phase often includes several stages and substages, including “design,” “production/build,” “test,” and “implementation”; this is the progression through which a system moves from being an idea to a finished, operational end-item. All systems are comprised of elements arranged in some configuration or structure, and in the **design stage** these elements and their configurations are defined. In the **production stage** the system is built, either as a single-item or a mass-produced item. Sometimes systems move through *iterations* of **design–build–test** during which the end-item is created during a structured, formalized trial and error testing procedure. Finally, in the **implementation stage** (also referred to as a *handover* or *closeout* stage), the system is installed and becomes a part of the user’s operational environment.

Phase D: Operation

In the **operation** phase the customer or user takes over the system and operates and maintains it. For systems such as products and equipment that people use or rely upon daily, Phase D may last for years or decades, and the phase includes not only operation and maintenance of the system, but improvements and enhancements to keep the system viable and useful. Every system eventually outlives its purposes or simply wears out. When that happens, the choices are to either scrap the system or upgrade it so it remains useful. In the latter case the “upgrade” is essentially a new system concept, which initiates a new SDC and a new project.

For some systems Phase D is a short event or nonexistent: examples are political campaigns and concerts—the project ends on Election Day or upon completion of the concert performance.²

Virtually all projects progress through the phases and stages as shown in Figure 3.2, although the way they are managed to do this varies. Projects managed to follow the phases and stages linearly (one after another) conform to the so-called **traditional** or **waterfall approach** to project management. This approach works well for projects where the outcomes are predictable (known with certainty) and the requirements are unlikely to change. But when the project requirements and outcomes are less certain, unknown, or likely to change, the project phases and stages must proceed more tentatively. Especially in Phases B and C, the project moves through **iterations** wherein the project stages are repeated; this is called the **agile approach** to project management, discussed in Chapter 14.

Regardless of variations, many or most projects follow the A, B, and C phases and the respective stages within each, even if informally. For instance, although not every project involves proposal preparation, every project does start with a proposal from *someone*. Similarly, while many projects do not involve “production,” every project involves the creation and development of something— even if only information. The point is that a great many projects follow a life-cycle pattern similar to Figure 3.2. Examples 3.1 and 3.2 illustrate.

Most project-oriented companies undertake projects in ways that best suit them, and they prescribe the ways to manage and perform tasks in those projects, i.e., they create their own **project methodology**. The following two examples illustrate such project methodologies; the first illustrates traditional or waterfall methodology; the second illustrates traditional plus agile methodology—that is, a **hybrid** methodology. This book repeatedly refers to the traditional



See Chapter 14

Example 3.1: Product Development Cycle at Jamal³

Jamal Industries is a medium-sized manufacturing firm that produces products for major retailers under their own labels such as Costco Kirkland and True Value. It develops and produces the products in the phases of *initiation*, *feasibility*, *analysis*, *design*, and *manufacturing*—the traditional approach to project management. Although Jamal initiates and implements most projects internally, sometimes it contracts out development and manufacturing work. This example is such a case.

A competitor introduced a digital timer that would greatly impact Jamal's market share. To examine the feasibility of launching a new product, Jamal engineers analyzed samples of the competitor's device to see whether they could quickly develop their own version. The analysis focused on whether a device as good or better could be made and sold under the retailers' private labels for 20 percent less than the competitor's price. As an alternative, Jamal could seek other distribution channels to sell the product under its own label. The study indicated that only the second alternative was feasible.

An in-depth analysis was done to determine how Jamal could contract out work to avoid a capital investment that it could not afford. The research director, who served as the project manager, and his engineering staff analyzed contracting alternatives and decided to hire a general contractor to design and manufacture the product. They identified a foreign contractor that could design and make a superior timer that Jamal could market at a price twelve percent lower than the competition. The bulk of the planning, scheduling, and budgeting associated with the project was delegated to the contractor. Within a year the product was designed, manufactured, distributed, and in stores.

The contractor will continue to produce the device if Jamal markets it. Jamal's design team was transferred to other projects, although the research director continues to monitor the contractor to ensure quality standards are maintained.

Example 3.2: Software Development Cycle at Microsoft⁴

New software development at Microsoft commonly follows the phases of *planning*, *development*, and *stabilization*.

The planning phase produces a vision statement, specification document, and plans for marketing, integrating components from other products, testing, and documentation. The phase runs from 3 to 12 months depending on whether the product is an upgrade or is new. The vision statement guides the project; it is a short statement about the product goals, focus, and priorities. The specification document is a preliminary statement of the product's features and packaging. It starts out small (sometimes a single sentence) but expands as the project progresses. This document is combined with project plans and time estimates to create a schedule. The phase concludes when management approves the plans and schedule.

The development phase is subdivided into four sub-phases with three internal product-release milestones. Each sub-phase is scheduled to run for 2 to 3 months, which includes time buffers to accommodate unanticipated problems and enable sub-phases to be

completed by the milestone date. Three sub-phases are devoted to development and coding, testing for bugs and functionality, and documentation of product features; they happen iteratively such that the product evolves as the development team gains experience with it. The goal of each iteration is to meet the requirements for a set of product features that would be fully ready to “ship” to customers, even though shipping isn’t yet possible because all the desired features have yet to be integrated into the product. If a competitor threatens to release a similar product, the third, or even second, sub-phase can be bypassed to cut 4 to 6 weeks from the development process. The product would have fewer features but would beat the competition to launch. During the fourth sub-phase, product features are further tested and debugged, and a “freeze” imposed, which means no major changes can be introduced thereafter. This enables the education group to write documentation that will correspond to the product when released.

In the last phase, *stabilization*, all the features developed in the previous phase are combined and tested. “Zero bug release” occurs when all bugs are fixed or features with bugs are removed from the product (to be fixed and included in later product releases). This phase concludes with the release of a “golden master” file from which marketing will distribute the software product. The project concludes with a project team meeting to review the project and what was learned from it.

Although the first and last phases of these projects conform more to the *traditional* approach to project management, the development phase follows a more *agile* approach. The agile approach to project management and system development is described in Chapters 4 and 14.



See Chapters 4
and 14

or waterfall methodology in Figure 3.2 and its Phases A, B, and C. We use this methodology not because it is the best for every project but because it conveys a common pattern that is very similar to methodologies we have seen in many companies. Another methodology that is similar to Figure 3.2 is the **systems engineering process**, which is discussed in Chapter 4. Other methodologies are discussed in Chapters 14 and 18.



See Chapters 4,
14 and 18

In product-oriented industries where the system of interest is a product, the SDC is called the **product life cycle**, which is the cycle of market phases through which a product moves in its lifetime. It has five phases: **Product Development** (product creation), **Introduction** (launch to market), **Growth** (rise in demand and sales), **Maturity** (rise in production output and efficiency, reduced costs), and **Decline** (drop in demand and sales, product discontinued). Some current (circa 2025) examples are: a drug to cure Alzheimer’s disease (Development), generative AI (Introduction), electric vehicles (Growth), cell phones (Maturity), and floppy disks (Decline, or, more accurately, dead!). Although similar to the SDC, the product life cycle places more emphasis on the product’s marketing, financing, and distribution than its design and development, i.e., the “project” portion of the life cycle.

Phased project planning, fast-tracking, and iterations



See Chapter 4

Sometimes the phases and stages in a project life cycle are undertaken in a stepwise fashion called **phased project planning** or project **stage-gating**. At the end of each stage or phase, top management reviews the project objectives, costs, and outcomes so far and decides whether to continue, suspend, or even cancel the project. This is elaborated in Chapter 4.

The project phases are not always performed in discrete sequence but can be overlapped in a practice called **fast-tracking**. Before Phase B is completed, aspects of Phase C are started; before Phase C is completed, Phase D is started. Fast-tracking compresses the project duration, although it poses the risk of overlooking or miss-defining tasks and having to repeat or undo them.

And, as mentioned, sometimes stages within phases are repeated. The stages of the execution phase and, sometimes, the definition phase are repeated in cycles, each cycle intended to refine, enhance, or build upon the results of the previous iteration.

Stakeholders

The SDC has many stakeholders. The main stakeholder groups are:

1. System **customers** (also called **buyers, clients, or owners**), including:
 - a. Customer management
 - b. Users and operators
2. System **contractor** (also called the **systems development organization—SDO, developer, promoter, consultant, or seller**), including:
 - a. Contractor top management (corporate and functional managers)
 - b. Project management (project manager and staff)
 - c. “Doers”—professional, trade, assembly, and other workers
3. Other parties, stakeholders, discussed later in the book.

Customers are people or groups for whom the project is being done and who will acquire and/or operate the system when it is completed. Customers pay for and make decisions about the project. **Users** and **operators** utilize, maintain, or in other ways are the recipients of the project end-item. It is important to identify the actual users since, ultimately, it is for them the end-item system is created. Although the terms customer and user are used somewhat interchangeably, it is important to keep in mind the distinction between them:

- The customer (owner, buyer, sponsor) *pays* for the system.
- The users *use* and operate it.

The **contractor** is the party that studies, designs, develops, builds, and/or installs the system. The contractor is usually external to the customer organization, although it might well reside within the same organization, as is the case of internal consulting/support groups. Since the contractor is usually an organization, it is also called the **systems development organization (SDO)**.

Because in most cases the customer pays the contractor to perform the project, think of the customer as the *buyer* and the contractor as the *seller*. These terms make sense when you think of a project in the context of being a contract between two parties, wherein one, the contractor (seller) agrees to provide services in return for payment from another, the customer (buyer). This is discussed in detail in Chapter 12.

In addition to these, the life cycle involves other parties—individuals, groups, and organizations with vested interests and/or influence on the conduct of the project. Anyone who is affected by the project, perceives being affected by it, or potentially can alter its outcome is a **stakeholder**. Project stakeholders are discussed throughout the book and in depth in Chapters 16 and 18.



See Chapter 12



See Chapters 16 and 18

Value creation and management

Simple question: Why do projects? Simple answer: To create value. All projects generate outcomes—products, end-items, and deliverables, which provide benefits to certain stakeholders that are important, useful, or worthwhile.

Perhaps obviously, the total value of benefits from both the project and the resulting product should exceed the total costs, and to be assured of that stakeholders must know the estimated costs and benefits *before* they approve the project. To that end, throughout the life cycles of both the project and the product, the costs and benefits are estimated, monitored, managed, and reported in a process called **benefits realization management (BRM)**. As discussed later, the process begins early in the project with **feasibility studies** and the **business case**.

For example, consider a project to create a new product, say, a digital assistant device. The device's product life cycle starts when the idea for the device is initially conceived and ends (presumably years later) when the device is discontinued and is no longer marketed (end of Phase D). The *project* forms the first part of the product life cycle, the early development part, starting with feasibility studies to justify the device, and ending with giving the device design to a manufacturer to produce and market it (Phases A–C). The BRM process is active throughout all this, starting with estimating the expected costs and benefits, and thereafter assessing the actual costs and benefits throughout the product's life cycle. Attempting to consider all costs and benefits even before the project is approved is the purpose of **life-cycle costing (LCC)**, which is discussed in Chapter 9. In the case of the digital assistant device, the costs and benefits to be included for LCC are twofold: (1) for the project itself—its feasibility, planning, and execution, and (2) for the device itself—operating and maintaining it over its lifetime and, eventually, discontinuing it.

The remainder of this chapter will focus on the first phase of the project life cycle and how projects are conceived and initiated.



See Chapter 9

3.3 Phase A: Conception

The conception phase nominally comprises two stages. The first, project initiation, establishes that a “need” or problem exists and that it is worthwhile to investigate. The second, project feasibility, is a detailed investigation of the need or problem, a formulation of possible alternative solutions, and the selection of one. The phase ends with an agreement that a chosen contractor will provide a specified solution to the customer.

Project initiation

Conception begins when the customer or user perceives a *need*, i.e., it recognizes a problem or opportunity and, possibly, ways to deal with it.⁵ Sometimes the need is expressed as a vision.

Beyond perceiving the need, project initiation requires proving that the need is significant and can be fulfilled at practical cost. It is easy to identify problems and muse about solutions, but most ideas are ephemeral and not worth much. If a customer decides to take an idea beyond speculation, he might take the “quick and dirty” route and simply accept the first solution that comes along; or, he might undertake a more protracted, albeit systematic and thorough approach and consider only ideas with a reasonably high degree of success or return on investment. To prescreen for good project ideas, the customer organization undertakes a brief, initial investigation.

Example 3.3: Vision Statement at Microsoft⁶

As mentioned in Example 3.2, each new product development project at Microsoft starts with a short statement about a product and its goals called a vision statement. For one version of Excel, it was five pages long.

The purpose of the vision statement is to communicate the concept and requirements of the product to the development team, management, and other product groups. The statement includes an executive summary with a one-sentence objective, a list specifying what the product will do and not do, and definitions of the typical customer and competition. It might describe product features and priorities in enough detail to begin preparing schedules for development, testing, user education, and preparation of English and non-English product versions. It might also list requirements for the operating system, memory, disk space, processor speed, graphics, and dependencies on printer drivers and components. The statement gives everyone a common overview about what they need to do.

Initial investigation

Many users know a problem exists but not what it is or even how to explain it. Before committing resources to a full-fledged study, the user undertakes a short internal investigation to clarify the problem and evaluate possible solutions. The investigation starts with fact-finding—interviewing managers and users, gathering data, and reviewing existing documentation. A clear statement of the problem is formulated, objectives are defined, and a list of alternative solutions is compiled. The investigation focuses on the elements of the problem, including:

- The environment
- The needs, symptoms, problem definition, and objectives
- Preliminary solutions and estimated costs, benefits, strengths, and weaknesses of each
- Affected individuals and organizations.

Based on the investigation, the customer decides whether to proceed. Most ideas never get further than this, and it is obvious why: there are endless ideas about needs and potential solutions, but resources are scarce and organizations can commit only to those comparatively few that provide the most benefits and have the best chances of success. To approve the concept for further study, the customer must be convinced that the idea:

- Fits a need that is real, and funding is available to support it.
- Has sufficient priority in relation to other ideas.
- Has value in terms of, for example, applying new technology, enhancing reputation, increasing market share, or raising profits.
- Is consistent with the organization's goals and resources.

Pertaining to the last bullet, some organizations prescreen proposed projects and consider for further analysis only those that align with organizational goals and available resources. Prescreening is an aspect of **project portfolio management**, discussed in Chapter 19.



See Chapter 19

The initial investigation is usually conducted by the customer and requires but a few days or weeks at most. Also called the *idea stage* or *pre-feasibility stage*, its purpose is to determine if the idea deserves further study; if it does, it then becomes a “potential project” and moves to the next stage, feasibility.

3.4 Project Feasibility

All projects generate outcomes—products, end-items, and deliverables, which to certain stakeholders provide benefits that are important, useful, or worthwhile, i.e., they create **value**. Perhaps obviously, the total value of benefits from both the project and the resulting product should exceed the total costs, and to be assured of that stakeholders must know the estimated costs and benefits before they approve the project. To that end, throughout the life cycles of both the project and the product, the costs and benefits are estimated, monitored, managed, and reported through the benefits realization management (BRM) process. The process begins early in the project with a look at the project’s **feasibility**, which involves studying the need, problem, or solution in sufficient depth to determine if the idea is economically viable and worth pursuing. The initial investigation is a form of feasibility study (*pre-feasibility*), but is usually rather cursory and, hence, insufficient to commit to a project. A **feasibility study** is a more protracted, rigorous study that considers alternative solutions (or “system concepts”) and the benefits and costs of each. The customer might perform the feasibility study, but will hire an outsider (contractor) to do it if the study requires special expertise. Deciding to build a new airport, power plant, factory, highway, or tunnel are examples where feasibility studies are undertaken by contractors and the studies are themselves big, expensive projects.

When several alternatives exist for the project, project feasibility may depend on the **life cycle cost (LCC)** of the end-item system; this is the system’s total cost over its entire useful life, including installation, operation, and disposal. The topic of LCC is covered in Chapter 9.

Some projects involve multiple feasibility studies. Product development and research projects often have both a *technical feasibility* (to assess the risk that the technology might not work) and a *commercial feasibility* (assess the risk that the product might fail in the market). Another matter regarding feasibility is the question of how the project will be financed and its expenses covered throughout the project life cycle. Project financing is a subject unto itself and beyond the scope of this book, however it is of major importance and, quite often, the deciding factor in project approval. In other words, a project’s feasibility might have less to do with the technical or commercial merit of the project than with funds available to support the project. For large projects, the execution plan (see Chapter 6) might include a section on project financing that addresses funding arrangements and means for controlling cash flow and managing money.

If the feasibility study indicates that the concept is viable, one of two things happens (Figure 3.3).

Theme A: if the concept is something the customer can handle itself, the project is passed along to an internal group for development and execution.

Theme B: if the concept cannot be executed internally, the project is given to outside contractors (SDOs).

Companies like Boeing, Microsoft, and Toyota routinely do feasibility studies for new products and then hand the approved concepts to their own teams for the design, development, and production of the products. But companies like Ritz-Carlton and Swissotel, after deciding to build a hotel at a specific location, hire contractors to execute the project. They solicit proposals from multiple contractors and select the best.



See Chapter 9



See Chapter 6

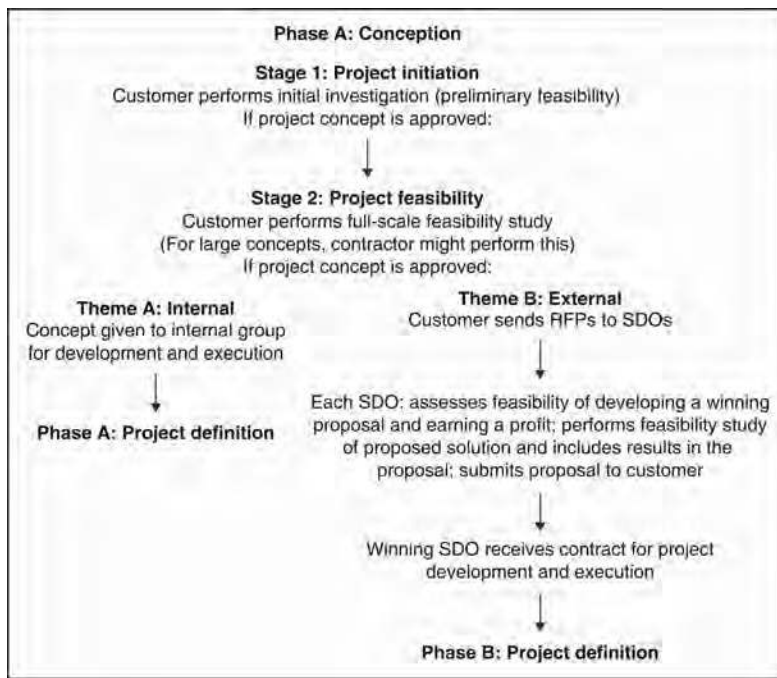


Figure 3.3
Feasibility study as part of the conception phase.

Each contractor competing for the project must also perform a feasibility study to assess the project's merits and whether it should participate. If a contractor decides to go forward, it will investigate solutions (system concepts) to the customer's problem, choose one, and describe it in the proposal. This is called the "proposal preparation process." Upon receipt of the proposals, the customer reviews them and selects the one it thinks is best, i.e., is the "most feasible."

In summary, project feasibility involves multiple studies and decisions—the customer assessing the "feasibility" of funding the project; the contractor determining the "feasibility" of winning the contract; the contractor conceiving and proposing the "most feasible" solution to the customer; and the customer assessing proposed solutions and choosing the "most feasible" to buy. When the customer reaches an agreement with a contractor, the project moves forward to Phase B.

Sometimes contractors decide not to prepare proposals because they don't have solutions that will make a profit or fit the customer's request. Sometimes customers conclude that none of the proposals meet the requirements. Either way, the concept is judged as "not feasible," and the process ends there.

Request for proposal⁷

The **RFP—request for proposal**—is a document the customer sends to potential contractors explaining the customer's problems, objectives, requirements and desire to hire someone; it might also state what the customer wants to see in the proposal (proposal requirements) and

Statement of work • Description of problem, need, or general type of solutions to be investigated. • Scope of work to be performed by contractor, work to be included, work excluded, and work restrictions; criteria of acceptance for results or end-items. • Requirements for the solution, results, or end-item, including specifications and standards; description of how work will be measured; expected relationship between user and contractor; expected completion date; constraints on cost of work to be performed. Proposal requirements Conditions placed on the proposal such as proposal contents and format, data requirements, sample forms to include, and submission location and deadline. Contractual provisions Type of contract to be awarded, sample contract, and nondisclosure provisions. Technical information or data Any additional data, or name of a contact person for requesting additional data, necessary to develop a solution and prepare the proposal or price quote. Proposal evaluation criteria Explanation of criteria and procedures for assessing the proposal and selecting a contractor.

Figure 3.4
Contents of a request for proposal.



See Chapters
12 and 19

how the winning proposal will be selected (proposal evaluation criteria) (see Figure 3.4). Additional forms of customer requests and the process of selecting the best proposal are described later in this chapter and Chapters 12 and 19.

A purpose of the RFP is to outline the user's need, problem, or idea, another is to solicit suggestions (proposals) for solutions—usually with the intent of awarding a contract. The customer can send RFPs to contractors on its own **bidders list**—a list of pre-qualified contractors, or it can also distribute RFPs to the broader market via the Internet using commercial “online sourcing tools.” Thus, contractors learn about upcoming jobs either by directly receiving RFPs from customers or by requesting RFPs through online newsletters and bulletins. For example, the web publication *Commerce Business Daily* gives a synopsis of all US federal jobs over \$10,000. Businesses scan the jobs and request RFPs for those they find of interest.

Often the customer will precede the RFP with a **request for qualifications**, which is a request for contractors to describe their qualifications. The customer sends RFPs only to contractors it deems qualified for the work.

Project foul-ups can often be traced back to a poor RFP. The RFP must be clear, concise, and complete: when it is, the customer can expect contractors to respond with proposals that are clear, concise, and complete; when it is not, the customer can expect proposals in kind. Ultimately, the ability of contractors to develop solutions that uniquely fit the customer's needs will depend in part on their understanding of the requirements as specified in the RFP. Similarly, the ability of the customer to select a contractor that is qualified and has the best proposal will depend on information provided in the RFP. Appendix A at the end of the book is an example RFP.



See Appendix A

Each competing contractor must consider its capability to prepare a winning proposal and, should it win, to perform the proposed work. Among the considerations are:

- Have competitors already got a head start?
- Does the contractor have sufficient money, facilities, and resources to invest in the project?
- Could performance on the project enhance (or damage) the contractor's reputation?
- Other considerations similar to the criteria employed by the customer in the initial investigation.

Sometimes a contractor will submit a proposal knowing full well it cannot win the project, doing so to maintain its relationship with the customer, be added to or remain on the customer's bidders list, or keep the field competitive. Sometimes a customer sends out RFPs with no intention of ever signing with a contractor; it simply wants to gather ideas—a situation of which contractors must be wary.

Contractors can also submit proposals to potential customers without an RFP. Whenever a developer believes it has a system or solution that satisfies a need or solves a problem, the project manager works with his marketing department to identify prospective customers, to which they might send **unsolicited proposals** describing the merits of the new system. Unsolicited proposals are also sent to current customers for potential follow-up work on current projects.

Feasibility study

As mentioned, a feasibility study can be performed at multiple times and with different parties in a project: minimally the customer performs a study to determine if the project is financially viable or otherwise worth pursuing; and the contractor also performs one to determine if the job is worth doing. In this section we consider the latter, although the same steps would apply to most anybody doing a feasibility study.

The statement of the problem (or **statement of work**) as defined in the RFP is frequently incomplete, vague, or even incorrect. If the contractor has received an RFP, it will possibly contain such a statement. Thus, one of the first steps before responding to an RFP is to develop a definition of the problem that is more concise, accurate and complete than the one in the RFP.

The prime source of information about the problem is interviews and documented information provided by the customer and users. It is thus important that the contractor identifies who the users really are—parties who are familiar with both the problem and the workings of the organization. Surprisingly, this is not always obvious. The real users, those who will operate, maintain, or be the main beneficiary of the system, are often confused with people who only represent the users. If the customer is an organization, the contractor must determine the individuals whose needs are to be met, then try to work with them during the feasibility study. Sometimes, however, the RFP specifies that to make the competition “fair” the customer will maintain an “arms-length” relationship with competing contractors. Even then, however, contractors are usually permitted to make inquiries to or seek additional information from a customer contact person.

Business case



See Chapter 19 costs. This is discussed in Chapter 19.

Sometimes the feasibility study results in a business case, a document that assesses the value and risks (feasibility) of a project at an early stage and attempts to convince the customer/sponsor to authorize and undertake the project. It is sometimes used to obtain financing for the project from commercial banks (a **bankable business case**). In choosing between projects, some companies use the business case to compare projects in terms of benefits versus risks and

The content of the business case depends on the findings of the feasibility study. Whereas the feasibility study compares alternative solutions and determines if the project is viable, the business case tries to justify the chosen alternative.

A business case typically includes:

- Cost–benefit analysis: estimated project costs compared to the benefits.
- Estimated project duration (when a financial return depends on the timescale).
- Financial aspects such as the funding approach.
- Risks, issues, and a preliminary risk management plan.
- Assumptions.

The business case contains estimates for costs and benefits, although often those estimates are updated as the project moves through its early phases. For example, the PRINCE2 methodology specifies developing a *verified outline business case* at the pre-project stage and a *verified business case* in the subsequent initiation stage, and thereafter reviewing and updating the estimates after each later project phase. In some industrial megaprojects, the early phases of the project are devoted to a three-step estimating and planning process called FEL, or **front-end loading**: FEL-1 (called Concept phase), delivers a *preliminary business case*; FEL-2 (Pre-Feasibility phase) delivers a *business case*, and FEL-3 (feasibility phase) delivers a conceptual design and project execution plan based upon the business case.

Sometimes the business case is included as part of a more comprehensive feasibility report that, in addition to arguing the business case, addresses technical, environmental, financial, and other aspects of the project in greater detail than would a typical business case.

Needs definition

A *problem* is an unsatisfied need, and a *solution* is a way to satisfy the need. Since both problems and solutions originate from needs, it is important that the solution adopted for the project addresses the right needs. Conducting a feasibility study and preparing a proposal should begin with defining the user or business needs. J. Davidson Frame suggests the following steps:⁸

1. Ask the user to state the needs as clearly as possible.
2. Ask the user a complete set of questions to further elicit the needs. For example:
 - a. Are these real needs, or are there other, more fundamental ones?
 - b. Are they important enough to pursue?
 - c. Are we capable of fulfilling these needs, or is someone else better suited?
 - d. If the needs are fulfilled, will they give rise to other needs?
 - e. Will satisfying these needs also satisfy other needs too?
 - f. What effect do the unmet needs have on the organization and the user?
 - g. What other parties are affected by these needs and how will they react to our efforts?

3. Conduct research to better understand the needs. “Research” means probing to gather whatever information is necessary to better understand the needs, define the problem, and propose solutions. Information sources include interviews, reports, memos, observation, models, and analysis of technical data and empirical test results.
4. Based on information from Steps 2 and 3, restate and document the needs.
5. Give the restated needs to the user.

The steps are repeated as often as necessary, concluding with a needs statement that the user agrees best represents his interests (rather than the interests of the contractor or other parties).

Since every project is an effort to fulfill needs, a clear, well-stated, and correct needs statement is necessary to avoid a project that is meandering or irrelevant. But attaining such a needs statement is not easy. Frame describes the following troublesome aspects.⁹

- *Some needs are ever-changing.* They are a moving target; thus, for each need the question must be asked “Is this likely to change?” When the answer is yes, the solutions and project plans that address the need must be flexible and easy to change.
- *Solutions are confused with needs.* Rather than stating a need, the user or contractor states a solution. For example, the statement “We need a new building” is a solution, not the need. True, maybe a new building will be required, but a building is only one of perhaps many ways of satisfying the need to, for example, overcome a space shortage.
- *The needs identified are for the wrong user.* Who is the user? Is it the party that feels the need and is most affected by it, or is it the party who pays to resolve it? Usually, these are different parties. The needs statement should reflect the opinion of the party to which the solution will be directed—the user. Do not let one party tell you another party’s need. Talk to the other party.
- *There is more than one user, and their needs differ.* The user embodies several parties, all with valid needs. The question is “Can all of their needs be addressed?” Given multiple users, an attempt must be made to organize, classify, and prioritize their needs.
- *User’s needs are distorted by the “experts.”* Inadvertently or intentionally, the contractor leads the user to a distorted definition of needs. The customer should be wary that the contractor might:
 - Extend the list of needs to be much broader than the user thought. This increases the problem’s size and, no surprise, the contractor’s billable work as well.
 - Reframe the needs in terms of what he, the contractor, is best suited to do. The contractor readily fulfills the stated needs, but the user’s needs remain unaddressed.
 - Not ask for but rather *state* the user’s needs (because, after all, the contractor is the expert).

Sometimes it is the users who are resistant to clarifying needs; they expect the contractor to do it for them. The contractor should involve the user and ensure the two parties work together until they reach an agreed-upon statement of user needs. The process helps both parties to better understand the needs and problems, and to ensure that the adopted solution is the right one.

User requirements definition

Conversation between a user and contractor:

USER: “You installed my computer. Why didn’t you install the network router, too?”

CONTRACTOR: “You said you wanted the computer installed.”

USER: “But the computer won’t be of much use to me without a router.”

CONTRACTOR: “You said you wanted the computer installed. I did just what you requested.”

Another exchange:

CONTRACTOR: “The lighting for the office addition is finished. As we agreed, I wired 20 ceiling lights.”

USER: “But the room seems kind of dark.”

CONTRACTOR: “You said you wanted 20 lights.”

USER: “Yes, but the room isn’t bright enough for what I need.”

Both cases illustrate user–contractor disagreements about end results. Misunderstandings like this delay project completion, drive up costs, and sometimes become legal disputes that end up in court. The problem is the lack of clear **user requirements**. User requirements should describe in clear, concise, and measurable, verifiable terms what the user wants in the solution. Derived from user needs, the requirements are the measures by which the user determines whether the end result or deliverable is acceptable. They are the *quality measures* for the project. In the above examples, they would address the functions that the computer system and overhead lighting must serve.

Ideally, user requirements address the needs not only of users but also builders, suppliers, and other stakeholders who will benefit from, manage, maintain, or otherwise be impacted by the system. Perhaps obviously, user requirements are stated in the *language* of the users and other stakeholders. The project should not begin until the requirements have been combined into a user requirements **list**, and the customer and contractor agree that the list is complete.

Often users do not understand the necessity for and importance of good requirements; thus, it is the project manager’s responsibility to make sure the requirements are complete, clear, and accurate. When the project is completed and the contractor says, “Here’s what you ordered,” the user should be able to say “Yes, it satisfies all my requirements.”

Objectives and life cycle requirements

There are many kinds of user requirements. Some account for the system’s objectives, life cycle, and operational modes, others for constraints and interfaces with other systems.

Every project and the end-item system start with a statement of objectives that elaborate on the needs; these objectives provide the basis for defining requirements. Consider the SpaceShipOne example from Chapter 1. The need—“a reusable three-person vehicle that can be launched into space twice within a 2-week period” can be defined in terms of the following set of objectives:

Develop a spaceship that can:

1. Attain a minimal altitude of 100 km (the altitude where “space” begins).
2. Be reused (launched) every 2 weeks.
3. Carry three people.

Each objective can then be elaborated in terms of a set of requirements. The requirements must account for whatever the users and other stakeholders think will be significant throughout the expected *life cycle of the system*, which means they should incorporate issues regarding how the system will be developed, built, used, marketed, financed, maintained, and disposed.



See Chapter 1

Operational modes requirements

During the product life cycle, what are the different ways and kinds of environments in which the system will be used or operated? These are referred to as **operational modes**. For example, the modes for the previously mentioned reusable spacecraft include:

- Flight mode
 - launch and boost into space
 - in-space
 - return from space
 - landing
- Turn-around-between-flights mode
- Crew training mode
- Ground transport mode
- Maintenance and testing mode

The system will be expected to perform different functions and satisfy different conditions in each of the modes, and these functions and conditions must be specified in the requirements.

Constraints and interface requirements

Every system is subject to limitations imposed by the environment and other systems with which it must interface, as well as mandated policies, procedures, and standards, and limits on resources, time, funding, technology, and knowledge. In addition, it faces environmental constraints including technological requirements, laws, and even social norms and customs. For instance, among numerous constraints and interfacing systems, the spaceship must conform to FAA regulations, technical standards of the aerospace industry, and local noise and pollution laws, and it must be able to interface with existing systems for air traffic control and communication.

The current system

Conceptually, a need arises because of inadequacies of the **current system**; there is a gap between the current system's capability and a desired capability. (Of course, sometimes needs arise for lack of a current system; that would be the case for SpaceShipOne.) One purpose of the feasibility study is to *understand and document* the current system, including its inputs, outputs, functions, subsystems, components, relationships, attributes, resources, and constraints. The schematic diagram in Figure 3.5, for example, shows the elements and supply flow for a hospital system; it was developed for a project to reduce the supply chain costs in the operating room.

It is common to refer to alterations or expansions of an existing system as a **brownfield project**, as opposed to a **greenfield project**, which is creating a system where no current one exists.

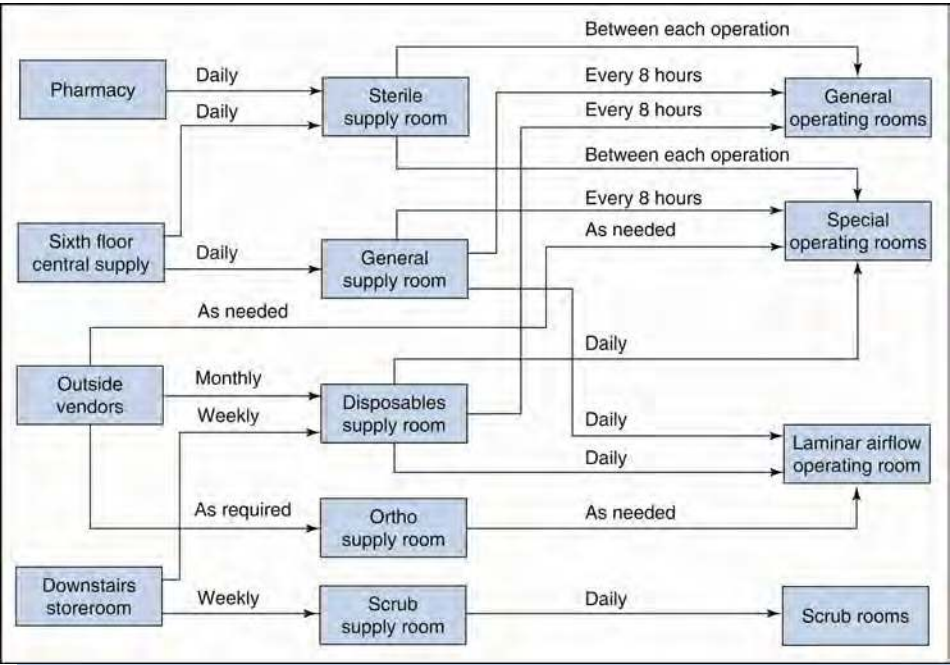


Figure 3.5
System schematic: flow of supplies to the operating rooms.

Analysis of alternative solutions

Through defining and documenting needs, requirements, and the current system, the contractor develops a good understanding of the problem and can delimit the scope of the alternatives to solve it. The contractor begins to develop alternative high-level (system-level) solutions to the problem from studies and models that account for what the system must do (user requirements), how it could be done (technical considerations), and what it would cost (economic considerations). The solutions may include new systems developed from scratch, or modifications of **off-the-shelf systems** and existing technology. Good project managers encourage creativity and free flow of ideas in the search for solutions.

Alternatives are analyzed for ability to satisfy objectives and user requirements within the available resources and imposed constraints. The best solution is chosen and proposed to the customer. Example 3.4 illustrates.

Environmental impact

Part of a project’s feasibility is determining the impact of the project or its end-item system on the natural environment. In 1969 the US enacted legislation mandating that all projects receiving federal funding or licensing must assess and report on the project’s environmental impacts in an **Environmental Impact Statement (EIS)**. Since then, Canada, Australia, New Zealand, Japan, countries of the European Union, and others have ratified laws requiring Environmental Impact Assessments (EIAs).

Example 3.4: User Requirements and Feasible Solution for the X-Prize Project

The X-Prize competition described in Chapter 1 required developing a complete system that would meet numerous requirements relating to everything necessary to design, build, and operate a spaceship. Among numerous user requirements for the spaceship are:

- Climb to an altitude of at least 100 km.
- Carry three people.
- Provide safe and comfortable flights.
- Be relatively inexpensive to design, build, and launch.
- Have a maximum “turn-around” time for reuse of at most 2 weeks.

Associated with each requirement are many issues, problems, and alternative solutions. One issue that impacts all the requirements is the basic question of how, exactly, do you get people into space and then back home safely?

The alternatives are:

1. Getting into space
 - a. Launch spaceship from atop a booster rocket.
 - b. Launch spaceship from a high-flying airplane.
2. Being in space
 - a. Enter Earth orbit (orbital).
 - b. Do not enter Earth orbit (sub-orbital).
3. Getting back to Earth.
 - a. Follow a wide parabolic arc.
 - b. Follow a narrow arc going almost straight up and straight down.
4. Landing
 - a. Land in a “zone” using a parachute.
 - b. Land at an airport like an airplane.

Designer Burt Rutan chose the combination of alternatives 1-b, 2-b, 3-b, 4-b: launch the spaceship from a high-flying airplane, do not enter orbit, follow a narrow parabolic trajectory up and down, and land airplane-like. These choices were the result of much study about the overall best way to meet the requirements in terms of technology, costs, and risks.



See Chapter 1

The contents of the EIS vary by state, country, and region but typically include:

1. A summary of proposed development and/or management plans.
2. Alternative sites and technologies for the proposed project.
3. A description of the project’s existing site and surrounding area.
4. Potential project impacts on:
 - a. Greenhouse gas emissions
 - b. Quality of air, soil, watersheds, wetlands, flood plains
 - c. Fisheries; sensitive plants; sensitive, endangered, or threatened species
 - d. Scenic resources; societal and aesthetic experiences

- e. Heritage resources (sites, structures, buildings, districts, objects)
- f. Historical resources (logging, ranching, grazing, mining, recreation)
- 5. Adverse impacts that cannot be avoided.
- 6. Long-term impacts on resources.
- 7. Ways to prevent, minimize or offset impacts; ways to monitor actual impacts.

The EIS is followed by a series of public reviews and hearings to discuss the findings and determine follow-up actions, especially concerning the last bullet above. Since the results of the EIS often affect the project plan and the system's design, the project's managers and supporters should try to develop a positive working relationship with the environmental assessment team.

Sustainability and green projects

In recent decades, increased energy consumption and usage of non-renewable resources has led to harmful environmental effects such as habitat destruction, biodiversity loss, desertification, and climate change. Projects themselves and the end-items they produce contribute significantly to such effects. One way to mitigate the harm is to design and build end-item systems—and manage the projects that produce them—to minimize these effects: the concept of sustainability in project management.

Related to sustainability is the term **green projects**, which refers to a classification of projects according to their purpose or impacts on the natural and social environment. For example, the “greenness” of a project can be categorized using the following criteria:¹⁰

1. The project's **purpose** (green by purpose): The purpose of these projects is associated with stemming or mitigating damage to the natural environment and efforts to “save the planet,” for example building wind farms, efforts to save a species, or converting to an all-electric vehicle fleet.
2. The **impact of the project** as it is underway (green by project impact): The conduct of these projects is potentially disruptive or damaging to the community and natural environment; they become “greener” when steps are taken to minimize those negative impacts. Example of such projects include London's Underground projects, China's high speed rail network program, Boston's “Big Dig,” and India's Navi Mumbai airport.
3. The **impact of the product or project outcome** (green by product impact): The products and services that result from these projects will potentially impact the future environment. An example is a project to create a new air conditioner: the project itself might have a small carbon footprint, but the air conditioner it creates might create tons of carbon every year and by every customer that uses it.

Projects can fall into multiple categories. Constructing a dam requires moving earth and tearing up the landscape (#2), but also results in flooded and nonrecoverable areas (#3); a project to save one species (#1) might put other species in jeopardy (#2 and #3).

While project management might focus on #2, the project, the big benefits for sustainability come from focusing on its lasting outcomes—#1 and #3. Focusing only on #2 is the tip of the iceberg—the ton of carbon produced in developing the air conditioner versus the many tons required to operate the air conditioner thereafter. In other words, sustainability must address the project life cycle, the life cycle of the resulting product, and the larger system of which the project and product are a part.

GPM Global, an organization devoted to environmental sustainability, has developed a project methodology and standards for sustainability in project management and offers a professional certification in green project management.¹¹ The methodology, called **PRISM** for Projects Integrating

Sustainability Measures, integrates sustainability standards into all phases of a project. The standards address the so-called **triple bottom line**, or **3Ps** (people—society, planet—environment, profit—economics) and align with the sustainable development goals of the United Nations.

Sustainability in projects and their outcomes is being addressed on many fronts. For example, the construction industry has created guidelines (sometimes by government mandate) for designing and constructing buildings (so-called “design for environment” and “green construction,” respectively) to reduce air and water pollution, landfill waste, and carbon emissions. As examples, building design guidelines in the UK focus on the building and mandate the use of:

- Passive ventilation systems
- Whole building heat recovery systems
- Renewable/recycled materials
- Materials with no damaging effects on the environment and energy-efficiency in terms of manufacture, use, and disposal.

Construction guidelines in the US that focus on the project include:

- Reduce landfill waste: crush/reuse aggregates (stone, etc.); use suppliers who accept returns/exchanges, reuse packaging, and use reclaimed/recycled building products.
- Minimize dust from concrete/mortar; avoid air/water pollution.
- Use timber and wood products with the Forest Stewardship Council’s trademark.
- Use low energy forms of construction to minimize CO₂ from site activities.
- Reduce trips to/from the site and use local suppliers to reduce transport CO₂.

In the US, the LEED certification program (Leadership in Environmental and Energy Design) has created standards of sustainable building design and development. The standards include many of the guidelines listed above, as well as:¹²

- Installation of windows that provide ample fresh air and natural light.
- Site selection: do not build on prime farmland or too close to a threatened animal habitat.
- Build nearby transportation alternatives and within walking distance to ten basic services.

Example 3.5: Addressing Climate Change in Major Projects.¹³

The European Regional Development Fund and the Cohesion Fund provide financial backing for major projects, where a “major project” is defined as one that has a total eligible cost exceeding €50 million (about \$54 million), and €75 million (\$80 million) for transport projects.

To receive consideration for funding, measures for climate change adaptation and mitigation must be integrated in the preparation and approval of each major project. **Adaptation** measures seek to ensure adequate resilience of a project to adverse impacts of climate change such as flooding and drought. Consideration of such measures requires identifying which climate hazards the project is vulnerable to, assessing the project’s level of risk, and adoption of alternatives to reduce the risk to an acceptable level.

Mitigation measures seek to reduce the emission of greenhouse gases, for example by selecting low-carbon options in project operations. This is addressed through the quantification of the expected greenhouse gas (GHG) emissions, or “carbon footprint,” of the project and its end-item system, and adopting measures to reduce those GHG emissions.

Consideration of climate change requirements for adaptation and mitigation into a project and its products must be initiated as early as possible. Only by doing so can climate adaptation measures and mitigation options be optimally integrated in the system development cycle. This is illustrated in Table 3.1, which gives an overview of the system development cycle stages and where and how in the cycle considerations regarding climate change can be included.

Table 3.1 Integrating climate change requirements in the development of major projects.

Systems Development Cycle	Adaptation Measures: Enhance resilience to the adverse impacts of climate change	Mitigation Measures: Reduce emission of greenhouse gas
Strategy	Strategic climate vulnerability screening—use principal steps as for vulnerability and risk assessment	- Link to climate policy and GHG emission targets - Consider less carbon intensive solutions in planning
Feasibility Design	- Assess vulnerability and risk - Do option analysis: climate risk and adaptation - Take measures to ensure resilience to current/future climate - Review technical aspects, e.g. location and design - Review environment and climate change factors - Do economic analysis - Do risk assessment/sensitivity analysis	- Measure CO₂ footprint shadow prices - Assess contribution to climate targets in EU2020 Strategy - Consider less carbon intensive options - Review environmental and other factors - Do economic analysis
Build/Procure Operate Decommission	- Implement adaptation measures in construction and operation - Monitor critical climate hazards - Regularly review climate hazards (which may change over time), update risk assessment, review structural and non-structural adaptation measures, and report to the project owner and others as required	- Reduce GHG emissions in construction and operation - Verify actual GHG emissions

Source: Adapted from European Commission Climate Action, "Climate Change and Major Projects," p. 4. EU Publications Office, 2016. https://ec.europa.eu/clima/sites/clima/files/docs/major_projects_en.pdf.

Project managers tend to focus on what will happen and is happening in the project, although sustainability also requires taking a longer view as to what happens after the project is completed. The reality, however, is that the project manager often has limited ability to influence the project's sustainability goals and outcomes because such decisions are made by other stakeholders—the customer, the market, the community, or senior management (although a well-respected project manager might be able to sway these decisions to make the project greener). Even with limited influence on project goals, the project manager often can impact how those goals are achieved—for example by selecting contractors and suppliers with a focus on minimizing the project's environmental impact.¹⁴ At minimum, the project manager should ensure that the project and its outcomes comply with local and national environmental laws and regulations; where laws and regulations are inadequate, the project manager must take the lead.

3.5 The Project Proposal

Proposal preparation¹⁵

The feasibility study results in a preferred solution to the stated problem and user requirements. The feasibility study, when combined with a rudimentary project plan, bid price, and contractor qualifications, form the **project proposal**. In the proposal, the contractor tells the customer what it intends to do, and the customer uses the proposals to select the best contractor.

The effort to prepare the proposal is itself a project and, thus, should be managed like one. Since preparing a proposal sometimes involves significant time and money, it usually requires top management authorization. Upon authorization, management identifies a technically competent person to oversee the proposal preparation, and sometimes this person becomes the project manager if the contract is won. She might be responsible for managing the entire proposal preparation effort or, alternatively, work with another manager who specializes in conducting proposal-related activities. The project manager selects the project team, or part of it, to help prepare the proposal, however the bulk of the team will not be chosen until after the contract is won.

The project manager reviews the requirements in the RFP and prepares a detailed summary of the to-be-proposed project. This summary guides the effort and prevents the focus from shifting to irrelevant technical or managerial considerations.

The proposal team outlines the work to be done for the solution identified in the feasibility study and prepares a *statement of work* or SOW. The SOW will include the system and project objectives, technical solution, high-level requirements, and major areas of work required to deliver the solution. If a SOW appeared in the RFP (e.g. Figure 3.4) then the SOW in the proposal might repeat it but should also include new information collected during the feasibility study and particulars about the chosen solution. In cases where the contractor believes the SOW in the RFP is inaccurate or incorrect, that should be stated in the proposal.

During preparation, the proposal team should seek input representing many perspectives; for example, from people in marketing, legal, accounting, estimating, engineering, and procurement.

Companies often have checklists/templates for preparing proposals, and standard clauses that address most of the issues that arise in projects. Matters the team must address include:¹⁶

- Scope of work
- Completion date

- Contract type
- Payment terms
- Customer obligations and items to be supplied
- Warranties and guarantees
- Limitations of liabilities
- Insurance and tax matters
- Confidentiality, patents, and proprietary information
- Ways for handling project termination, changes, delays, and cost escalation.

During proposal preparation, the proposal team must think through the entire project and prepare a rudimentary project plan that will address project time, cost, and performance issues. It uses a **work breakdown structure (WBS)** to determine the tasks necessary to achieve the requirements and to prepare a schedule and cost estimate (topics discussed in later chapters). The proposal sometimes includes the WBS, schedule, and a cost breakdown showing how the project price was derived. When multiple solutions are proposed, a rough plan of each is included.

The proposal is a sales device and, if accepted, also a contract: a good proposal gives not only the price, schedule, and other details, but convinces the customer that the contractor is competent and capable of doing the work.

All functional departments in the contractor organization able to provide relevant information are called upon to assist with the proposal. This increases the accuracy of proposal estimates and builds commitment from groups that will later work on the project.

During proposal preparation, the contractor should try to establish a dialogue with the customers to determine which solutions they prefer and which requirements are dominant among time, cost, and performance. Even with a clear RFP, this will help ensure that the proposal will satisfy the user's requirements. Proposal preparation can be iterative: acceptance of one proposal leads to preparation of another, more detailed proposal, as illustrated next.

Example 3.6: Proposal Development for Real Estate Projects at Wutzrite Company

The real estate department at Wutzrite Company helps clients choose from real estate investment alternatives. A meeting is set up with the client to define the client's investment "problem" and goals, and the client and several Wutzrite employees brainstorm to get a clear, accurate definition of the problem. Afterward, a project director prepares a proposal for the client that includes the problem statement, a proposed solution, and the price. Proposals that involve site development or designing and constructing a building include a feasibility study; for proposals that involve only evaluating, improving, or determining the value of a site, no feasibility study is needed. If the client likes the proposal, the director prepares a second, more detailed proposal that includes a WBS and updated schedule. If the client approves it, the second proposal becomes the high-level project plan. It specifies tasks to be done and target dates, and is the basis for assigning personnel to the project.

Approval of the second proposal usually calls for a feasibility study, demographic study, and analysis of financing, tax, and accounting ramifications of the recommended solution. The results are combined and submitted to the client in a third proposal that suggests particular courses of action regarding the solution.

The feasibility study and proposal preparation may take weeks or months to complete. Although enough time must be spent to produce a good proposal, not so much time should be spent that it becomes overly time-consuming or expensive. A rule of thumb is: Do not do the entire project while preparing the proposal! In some technical projects this may be unavoidable since the proposal must include a full-scale demonstration of the proposed solution. Developing the system for demonstration is itself tantamount to a full-sized project.

To assure nothing is overlooked in the proposal preparation, project managers employ checklists that, over the years, grow to accumulate all important matters to be addressed, including, for example, considerations for design, assembly, test, shipment, documentation, facilities, subcontractors, travel, labor, training, and payment. Before submitting the proposal to the customer, contractor top management must be briefed about the project's scope, resources needed, price, etc., and approve it.

Proposals range in length from a few pages to many hundreds. The content varies depending on, for example, format favored by the customer, relationship between customer and contractor, technical complexity of the work, and whether the proposal was solicited or unsolicited. Figure 3.6 shows the main ingredients of a typical proposal.¹⁷ If the proposal is in response to an RFP, its content and format should conform exactly to the proposal requirements or guidelines stated in the RFP. Appendix B at the end of the book is a proposal in response to the RFP in Appendix A.

The cost to a contractor in preparing proposals and the proportion of contracts it wins significantly affect the contractor's overhead since expenses for proposal preparation are charged to overhead. Only in rare cases such as major US defense contracts are the winning contractors reimbursed for proposal expenses.

Selecting the winning proposal

Upon receiving proposals from multiple contractors, the customer compares and evaluates them. Selecting the best proposal, reaching an agreement with the contractor, and committing funds are all part of the **project selection** process. Most companies follow a prescribed procedure for evaluating and comparing proposals. When the selection involves assessing each proposed project for its contribution to a portfolio of projects, the procedure also includes appraising the project's contribution to company strategic goals, the resources it will entail, and its comparative financial benefits. The topics of project selection and project portfolios are expansive and covered more fully in Chapter 12 and Chapter 19. Here we give a brief overview of the project selection process.



See Chapters 12
and 19

In general, project selection is based upon consideration of many criteria (sometimes provided to contractors in the RFP), for example:

- Project price
- Ability of contractor to satisfy stated needs (solution or technical approach)
- Return on investment
- Project plan and contractor management
- Contractor qualifications and reputation
- Likelihood of success or failure (risks)
- Fit to contractor resources and technological capability.

The customer may assume that a competent contractor with a good plan will do a good job and, thus, select the contractor with the best qualifications or best plan rather than best solution or technical approach. Thus, the proposal should highlight contractor credentials

Executive summary

Perhaps the most important part of the proposal, this section must convince the customer that the remainder of the proposal is worth reading. It should be more personal than the proposal, briefly state the qualifications, experience, and interests of the contractor and draw attention to the unique or outstanding features of the proposal, the price, and the contractor's ability to do the project. In case the customer has questions, the contractor "contact" person is identified here. From reading this section the customer decides whether or not to examine the rest of the proposal.

Technical section (SOW)

- (a) Indicates the scope of the work—the planned approach. It must be specific enough to avoid misunderstandings and demonstrate the method and appropriateness of the approach, yet not so specific as to "give away" the solution. It should also discuss any problems or limitations inherent to the approach.
- (b) Describes realistic benefits in sufficient detail to demonstrate that user needs will be fulfilled, but not so specific or enthusiastic as to promise benefits that might be difficult to deliver.
- (c) Contains a schedule of when end-items will be delivered. It should be based upon a work breakdown structure and include the major project phases and key tasks, milestones, and reviews. In developmental projects, portions of this section might have to be negotiated.

Cost and payment section

Breaks down projected hours for direct, indirect, and special activities and associated labor charges, materials expenses, and price of project. The preferred or required contractual arrangement and method of payment are also included.

Legal section

Contains anticipated, possible, or likely problems, and provisions for contingencies; e.g. appropriate procedures for handling changes to the scope of the project and for terminating the project.

Management/qualifications section

Describes the background of the contractor, related experience and achievements, and financial responsibility. Also includes organization of management, and resumes of project manager and key project personnel.

Figure 3.6
Contents of a proposal.



See Chapters 6
to 11

and include a basic but concise project plan. Methods for preparing the plan are discussed in Chapters 6 through 11.

Selecting the best proposal often begins with prescreening the proposals and rejecting the ones that fail to meet certain cut-off requirements such as too-high price tag, too-low rate of return, or insufficient experience of the contractor.

Proposals that survive prescreening are subjected to closer scrutiny: a common evaluation method, called **simple rating**, rates proposals according to multiple evaluation criteria on a checklist. Each proposal is given a score for each criterion *j*. The overall score for the proposal is the sum of the scores for all criteria,

$$S = \sum_j s_j, \text{ where } j = 1, 2, \dots, n$$

The proposal receiving the highest overall score wins.

A limitation of the method is that all evaluation criteria are treated as equally important. When some criteria are clearly more important than others, a method called **weighted rating** is used instead wherein the relative importance of each criterion j is indicated with an assigned weight w_j . After a given criterion has been scored, the score is multiplied by the weight of the criterion, $s_j \cdot w_j$. The overall score for the proposal is the sum of the $s_j \cdot w_j$. For all criteria,

$$S = \sum_j S_j w_j, \text{ where } j = 1, 2, \dots, n$$

$$\sum w_j = 1, \text{ and } 0 \leq w_j \leq 1.0$$

The procedures for the two methods are illustrated in Example 3.7.

Example 3.7: Evaluating the Proposals at MPD Company

In response to its RFP for the LOGON project (Appendix A, end of book), MPD Company received proposals from three contractors: Iron Butterfly Contractors, Inc.; Lowball Company; and Modicum Associates. Each proposal was reviewed and rated by a group of operations managers at MPD on five criteria using a four-point scale, where 4 is most favorable:

The results of the assessments for the three proposals were as follows:

Criteria	Scores		
	Iron Butterfly	Lowball	Modicum
Technical solution approach	3	1	4
Price of contract	4	4	1
Project organization/management	4	2	3
Likelihood of meeting cost/schedule targets	3	2	4
Reputation of contractor	3	3	4
Sum	17	12	16

Based on the sum of simple ratings, Iron Butterfly was rated the best.

Weighted rating

Using the simple rating, Lowball was clearly the worst, but Iron Butterfly and Modicum were considered too close to differentiate. The rating group then decided to look at the criteria more closely and to assign weights to the criteria based on their relative importance (column 2, below, where $\sum w_j = 1$, and $0 \leq w_j \leq 1.0$).

Taking the weights into account, the proposals scored as follows (columns 4 and 6):

Criterion	Weight(w)	Iron Butterfly		Modicum	
		Score (s)	(s)x(w)	Score (s)	(s)x(w)
Technical solution approach	0.25	3	0.75	4	1.0
Price of contract	0.25	4	1.0	1	0.25
Project organization/management	0.20	4	0.8	3	0.6
Risks of solution	0.15	3	0.45	4	0.6
Reputation of contractor	0.15	3	0.45	4	0.6
	Sum		3.45		3.05

Using the sum of the weighted ratings, Iron Butterfly is clearly the superior proposal. More examples of weighted ratings for project selection are provided in Chapter 19.


See Chapter 19


See Chapter 11

Assessment of proposals commonly also includes evaluation of project risk, especially when the proposed solutions and associated levels of risk differ significantly between proposals. Methods for identifying and assessing risks are discussed in Chapter 11.

Sometimes the contract award depends more on the contractor’s qualifications than the proposed solution. Among factors the customer might consider are:¹⁸

- Is the contractor big enough or adequately financed to do the project?
- Does it have a good track record with this kind of project?
- Does it have a good reputation in the industry?
- Has it been involved in litigations and arbitrations?
- Will its management be accessible?
- Does it have ISO 9000, ISO 14000, or other certification?
- Will the relationship with the contractor likely be amicable or touchy?

Proposal finalists are notified and might be requested to provide more data or give live demonstrations of their proposed solutions or systems. If none of the proposals are acceptable or the feasibility studies conclude the project would be too costly, risky, or time-consuming or not provide adequate benefits, the process ends with nobody getting a contract.

For contractors who did not get the job, good practice is to conduct a “proposal postmortem” to determine why not, lessons learned, and what to do differently next time.

The RFP/proposal process outlined in the last few sections addresses the question of who will do the work. Whether it is a customer choosing a contractor, or a contractor choosing subcontractors, each party follows a similar process to define its needs, solicit ideas, and choose potential contractors. The effort required of all parties to define their requirements, prepare proposals, and review and hire qualified contractors can be quite time consuming, especially in international projects. Thus, extra time must be allotted for it in the project schedule; if not, the process can substantially delay the start of the project. Initiating projects and preparing proposals often involves convolutions that are hard to anticipate. The following story illustrates.

Example 3.8: Proposal for the Apollo Spacecraft¹⁹

The US space program to land human beings on the moon involved thousands of contracts awarded by NASA in separate competitions. The biggest contracts were for the biggest components, namely, the Apollo spacecraft, the lunar lander, and the first, second, and third stages of the rocket that would propel the spacecraft and lander to the moon. Harrison Storms was vice president of North American Aviation's Space Division (NA) in Los Angeles when NASA opened bidding for Apollo. His division had already been working feverishly to solve difficult technical problems for a proposal to build the rocket's second

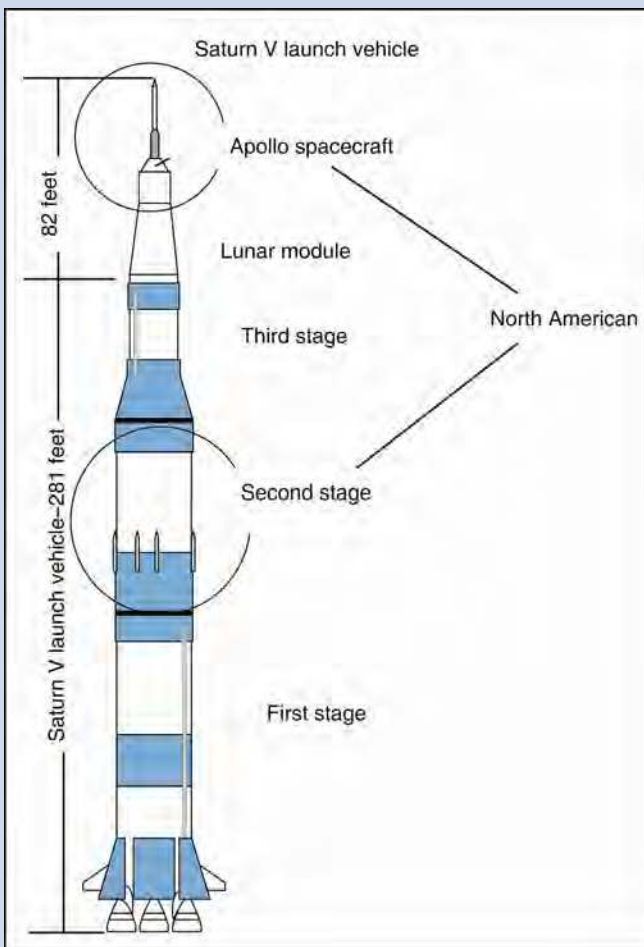


Figure 3.7

Apollo/Saturn moon rocket and North American components.

Source: Image courtesy of NASA.

stage. The technical requirements were so demanding that only a handful of contractors stayed in the competition. Most managers engaged in such a big effort would have considered themselves already overextended, but not Storms: he wanted to go after the *big prize* contract—the Apollo spacecraft contract. The Apollo spacecraft would contain systems for life-support, guidance, and navigation (ultimately comprising over 2 million parts) and would take three astronauts to the moon and back. The problem was, NA had never built a spacecraft before, and it would be expensive to learn how. Storms gathered up his best people and put together a presentation for the company chairman and founder, James “Dutch” Kindleburger, arguing that NA should prepare a proposal for Apollo. Dutch was skeptical but he pledged \$1 million support. Storms knew that wouldn’t be nearly enough but accepted it anyhow. Now NA would bid on *both* the second stage *and* Apollo (Figure 3.7).

The allotted \$1 million had long since been exceeded—maybe by three times, but no one knew. Back then cost statements ran months behind billings, and Storms gambled that NASA would receive the proposal before anyone saw the final bill. With less than 6 weeks to go he picked John Paup to be Apollo program manager, a “witty, engaging person” who understood the technology. For the next month Paup worked 18 hours a day, slept in a cot, and ate from vending machines. Every morning, he gathered his team for a standup meeting; anyone not there by 7:45 was locked out. No coffee or seats, he wanted to hear about the problems and how each would be fixed within 24 hours.

The proposal was encyclopedic in size, and NASA wanted dozens of copies submitted no later than 5:00 pm two days before the presentation. The whole bundle, weighing 100 pounds, was hand-delivered just under the wire. Next day Paup and his team, looking like zombies from lack of sleep, boarded the company plane for the presentation to NASA in Virginia. Each company had 60 minutes to present its proposal to an evaluation team of 75 top engineers. Undaunted, Paup hit all the presentation high points and finished 10 minutes early.

Meantime NA headquarters had determined that the proposal cost five times the allotted \$1 million, and it was fuming. But to say the overrun was worth it would be an un-



Figure 3.8
Apollo spacecraft.

Source: Photo courtesy of NASA.

derstatement. North American won the contract: in return for a target cost of \$884 million and a fee of \$50 million, NA was to deliver several mockups, test versions, and flight-ready Apollo spacecraft (Figure 3.8). Since the risks of sending humans to the moon were overwhelming, the contract was cost-plus. By the time the lunar program ended 10 years later in 1972 with the return of the seventh crew from the moon, NA as prime contractor had earned \$4.4 billion—over \$29 billion in 2025 dollars.

3.6 Contractual Agreement and Negotiation

The usual result of the RFP/proposal process is the selection of a preferred contractor and the signing of a contractual agreement. If several contractors receive close marks during the selection process, then the parties might enter a process of negotiation to settle upon a final agreement.

The purpose of contract negotiation is to clarify technical or other terms in the contract and reach agreement on project price, schedule, and performance obligations. The process is not necessary for standardized projects wherein the project terms are simple and costs are fairly well known. In fact, in such cases and where *price is the only selection criterion*, the customer will issue an **invitation for bid** (IFB) rather than an RFP; price is all that matters, and negotiation is prohibited. Nonetheless, for complex systems projects that involve high risk or difficult developmental work, reaching the contractual agreement almost always involves negotiation.

Different kinds of contractual agreements offer different advantages to the customer and contractor, depending on the nature of the project. The most common forms of contractual agreements, described in much more depth in Chapter 12, are:



See Chapter 12

- **Fixed price contract:** the price paid by the customer for the project is fixed regardless of the costs incurred by the contractor. The customer knows what the project will cost.
- **Cost-reimbursement contract:** the price paid is based on the costs incurred in the project plus the contractor's fee. The contractor is assured his costs will be covered.
- **Incentive contract:** Variation on fixed-price and cost reimbursement contracts: the amount paid depends on the contractor's performance as compared to the target price, schedule, or technical specification. The contractor receives a bonus for exceeding the target or pays a penalty for not meeting it.

In general, work that is contracted or procured from contractors and suppliers is referred to as **procured goods, work, and services**. Like everything else in projects, such work must be planned, scheduled, budgeted, and controlled—it must be managed; this includes keeping the agreement up to date with respect to changes to the project, customer needs, and the contractor capabilities, and checking that all work conforms to the agreement. This, called **contract administration**, and the broader topic of **procurement management**, are the subjects of Chapter 12.



See Chapter 12

3.7 Contract Statement of Work and Work Requisition

Each contract contains a statement of work (SOW) that is like the SOW in the original RFP or the proposal, or is a restatement of it to reflect the negotiated agreement. This so-called

contract statement of work (CSOW) defines the expected performance of the project in terms of scope of work, requirements, end results, schedules, costs, price, payment schedules, and ways to handle changes or variations, plus each party's responsibilities and liabilities.

When the customer and the contractor both agree on the CSOW, the project is considered "approved" and ready to go. Before work can begin, however, it must be divided among the involved departments of the SDO and contractors, and requirements specified in the CSOW must be translated into terminology that people in these groups understand. The translations, aimed at the groups that will perform the work, must be identical interpretations of the requirements and work scope specified in the CSOW. The document containing the SOW for each work group is called a *work requisition* or *work order*. Its purpose is to describe the work expected of each party and to authorize the work to begin. This topic is discussed further in



See Chapter 12 Chapter 12.

Authorization of the work to begin marks completion of Phase A and authorization for the project to begin and to proceed to Phase B. The steps in Phase A are summarized in Figure 3.9.

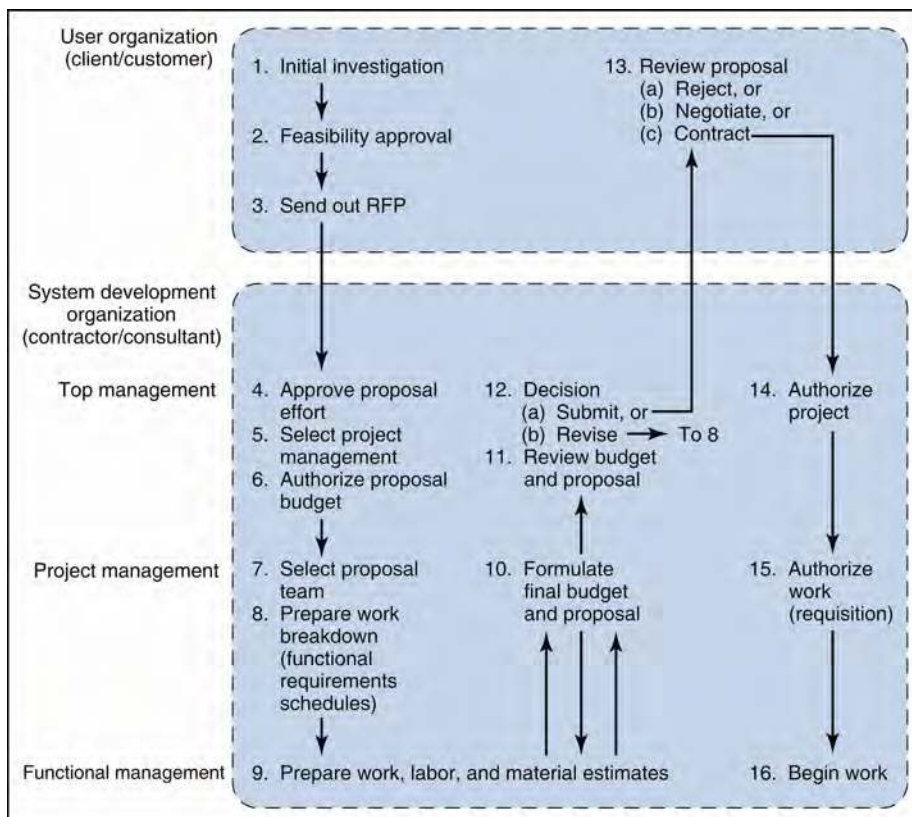


Figure 3.9

Project initiation, proposal preparation, and authorization process.

3.8 Project Initiation: Variations on a Theme

Projects are initiated in response to a need, but they do not always involve an RFP or even a proposal. The RFP/proposal process as described largely applies to projects where the work is **contracted out**, i.e., where the customer and the contractor are not in the same organization. For internal projects—projects where the organization has the capability to do the work on its own, a project might initiate with a business case study, as described earlier. Common examples of this are projects in product development (PD) and IT—two areas where companies often exhibit significant internal prowess. In PD the “need” is manifest as the desire or mandate to fill a perceived market niche or respond to a competitive threat. The business case study addresses the proposed product, market, competition, risk, cost, and benefits, and argues in favor of launching a new PD effort. If the case is approved, the project is turned over to the PD department to begin work. The business case study thus serves a dual purpose: it is both the feasibility study and the proposal. IT projects are similarly initiated by business case studies.

The department that would do the project, if approved, prepares the business case study and argues for the proffered end-item or solution. Approval or denial of the project involves rating the case against competing cases in terms of the resources required, benefits compared to goals, and priority of needs—the selection process described in Chapter 19. If the project is approved, a project charter is created, as described in Chapter 4.

The RFP/proposal process as described represents projects with relatively few stakeholders or a single, clearly identified customer and its potential contractors. In large technical projects that touch many stakeholders the process is more protracted. Examples include projects for infrastructure and transportation systems (Boston Big Dig, Delhi Metro, telecommunication systems, Channel Tunnel), technical systems wherein subsystems and components must be developed from scratch (commercial aircraft, SpaceShipOne, medical devices) and large-scale property developments (resorts, airports, planned communities). In such cases, it is more difficult to identify the stakeholders and define their multiple and sometimes conflicting needs, so the RFP/proposal process must include a “front-end” component to identify the important stakeholders and combine their needs into a list of stakeholder requirements. The contractor that gathers the stakeholder requirements might be hired solely for that purpose and not be the same as contractors hired to perform the project work; after the stakeholders and their needs have been identified, other contractors are solicited to review the requirements and suggest solutions through the RFP/proposal process. The process of identifying stakeholders and their requirements is a systems engineering effort that considers the end-item system’s far-reaching effects and the many stakeholders it will touch throughout its life cycle.



See Chapters 4
and 19

3.9 Summary

The systems development cycle can be divided into four phases: conception, definition, execution, and operation. The first three phases constitute the project life cycle.

Phase one, conception, includes formulating the problem, defining needs and user requirements, evaluating alternative solutions, and preparing a proposal to conduct the project. At the start of this phase most activities are in the hands of the customer; by the end of the phase, most activities have been taken over by the contractor or system developer. The relationship between the customer and the contractor is initiated and cemented through the RFP/proposal process, negotiation, and contractual agreement.

Phase A is the “foundation” part of the systems development cycle; it establishes the needs, objectives, requirements, constraints, agreements, and patterns of communication

upon which the remaining phases are built. It is the phase where the project is conceptualized, its feasibility examined, and contractors are identified, proposals prepared and evaluated, and contract agreements reached. It is a crucial phase and the place where, often, the seeds of project success or failure are planted.



Review Questions

1. How are projects initiated? Describe the process.
2. What factors determine whether an idea should be further investigated?
3. Who is the user in the systems development process? Who is the contractor?
4. Besides the user and the contractor, what other parties are involved in the systems development cycle? Give examples for particular projects.
5. What does the term “fast-tracking” imply?
6. How does the contractor (SDO) become involved in a project?
7. What is the purpose of an RFP? Describe the contents of an RFP.
8. What is a feasibility study? Describe its contents and purpose.
9. What are user needs? Describe the process of defining user needs and the problems encountered.
10. What are user requirements? How do they differ from user needs?
11. Who prepares the proposal? Describe the proposal preparation process.
12. What is the statement of work (SOW)? In what documents does the SOW appear?
13. Describe the contents of the proposal.
14. How is the best proposal selected? Describe the process and the criteria used.
15. Three proposals (W, X, and Y) have been rated on six criteria as follows: 1 = poor, 2 = average, 3 = good. Choose between the three proposals using (a) the simple rating method and (b) the weighted rating method.

Criteria	Weight	W	X	Y
Attention to quality	0.25	2	1	3
Cost	0.20	3	3	1
Project plan	0.20	2	2	1
Project organization	0.15	3	2	3
Likelihood of success	0.10	2	3	3
Contractor's credentials	0.10	2	2	3

16. What contractor qualifications might the customer look for in a proposal? What else about the contractor might the customer look for?
17. What parties are considered subcontractors in a project?
18. Discuss the purpose of a business case study for internal projects. What does the study include and who prepares it?
19. How is the RFP/proposal process adapted to large projects that potentially have numerous stakeholders but initially only a few have been identified?
20. In contracting out work, does the customer relinquish all control over the project to the contractor? Explain.
21. How can a contractor be both the sender and receiver of RFPs, i.e., how can it both prepare and submit proposals, and receive and review proposals?
22. When a contractor hires a subcontractor, to whom is the subcontractor obligated—the end user customer or the contractor?

23. What must the project manager know to be able to effectively negotiate a contract? Consider aspects of the customer, competition, and technical content of the proposal.
24. Discuss the difference between the SOW, CSOW, and work requisition or work order.



Questions About the Study Project

As appropriate, answer Questions 1–14 regarding your project. Also answer the following questions: How are contracts negotiated and who is involved in the negotiation? What kinds of contracts are used in the project?

CASE 3.1 WEST COAST UNIVERSITY MEDICAL CENTER

West Coast University Medical Center (WCMC) is a large teaching and research hospital with a national reputation for excellence in health care practice, education, and research. Seeking to sustain that reputation, the senior executive board decided to install a comprehensive medical diagnostic system incorporating AI. The system would be linked to WCMC's servers and available to physicians from their homes and offices via the Internet. By clicking icons to access a medical specialty area, then keying answers to queries about a patient's medical symptoms and history, a physician could receive a list of diagnostics with associated statistics.

The senior board sent a questionnaire to every department asking managers about the needs of their areas and how they felt the system might improve doctors' performance. Most managers replied that the system would save doctors' time and improve performance. The hospital information technology (IT) group was assigned to assess the cost and feasibility of implementing the system. They interviewed managers at WCMC and several vendors of diagnostic software. The study showed high enthusiasm among the managers and a long list of potential benefits. Based on the feasibility study, the board approved the system.

The IT manager invited three well-known consulting firms that specialized in medical diagnostic systems to give presentations, and then hired one to assist his group in selecting and integrating several software packages into a single, complete diagnostic system.

One year and millions of dollars later the project was completed, but 6 months later it was clear the system was a failure. Although it did everything the consultants and software vendors had promised, few doctors used it; of those that did, many complained that the "benefits" were irrelevant and that features of the system they would have liked were lacking.

QUESTIONS

1. Why was the system a failure?
2. What was the likely cause of its lack of use?
3. What steps or procedures were poorly handled in the project conception phase?

CASE 3.2 X-PHILES DATA MANAGEMENT CORPORATION: RFP MATTERS

X-Philes Data Management (XDM) Corporation (motto: “The truth is out there”) is preparing to contract out work for two large projects: Scully and Mulder. The projects are comparable in terms of size, technical requirements, and estimated completion time, but are independent and will be performed by separate project teams.

Two managers at XDM, one each assigned to Scully and Mulder, prepare RFPs for the projects and send them to several contractors. The RFP for Scully includes the following: a SOW that specifies system performance and quality requirements, maximum price, completion deadline, and contract conditions; an incentives clause stating the contractor will receive a bonus for exceeding minimal quality requirements and finishing the project early, or will be penalized for poor quality and late completion; and a requirement that the contractor submit detailed monthly status reports showing progress on key quality measures. The RFP for Mulder includes a brief SOW, a maximum budget, and the desired completion date.

Based on proposals received in response to the RFPs, the managers responsible for Scully and Mulder each select a contractor. Unknown to either manager is that they selected the same contractor, Yrisket Systems. The Scully manager selects Yrisket because its bid price is somewhat below the budget limit and its reputation in the business is good. The Mulder manager selects Yrisket for similar reasons—good price and reputation. In preparing the Scully proposal, Yrisket managers had to work hard to meet the maximum price specified in the RFP, but they felt that by doing quality work they could make a tidy profit from the incentive offered.

A few months after the projects are underway some of Yrisket’s employees quit. To meet their commitments to both projects, Yrisket workers must work long hours and weekends. It is apparent, however, that these extra efforts might not be enough, especially because Yrisket has a contract with another customer and must begin work soon.

QUESTIONS

1. What do you think will happen?
2. How do you think the crisis facing Yrisket will affect the Mulder and Scully projects? The two projects are very similar, yet do you expect Yrisket to treat them the same?

CASE 3.3 PROPOSAL EVALUATION FOR APOLLO SPACECRAFT²⁰

Five proposals were submitted to NASA to design and build the Apollo spacecraft. An evaluation board of more than 100 specialists reviewed the proposals and ranked them as follows (maximum = 10).

	Technical Approach (30%)	Technical Qualification (30%)	Business Strength (40%)	Weighted Total
Martin Company	5.58	6.63	8.09	6.90
General Dynamics Astronautics	5.27	5.35	8.52	6.59
North American Aviation	5.09	6.66	7.59	6.56
General Electric Company	5.16	5.60	7.99	6.42
McDonnell Aircraft Corporation	5.53	5.67	7.62	6.41

The board unequivocally recommended to NASA senior management that Martin be awarded the contract but suggested North American as the next-best alternative based upon NA's experience in developing high-performance military and research aircraft. This experience (technical qualification) sufficiently impressed the board that it put NA ahead of General Dynamics, despite NA's lower ratings on technical approach (design of the space capsule) and business strength (organization and management). The board mentioned that any shortcomings in NA's technical approach could be corrected through additional design effort. Seeing the board's recommendations—and aware of NA's long, close association with NACA (NASA's predecessor agency), NASA senior management immediately selected North American.

QUESTIONS

1. How were the points in the "Weighted Total" column determined? Show the computations.
2. North American rated third out of five contractors in the Weighted Total column yet was awarded the contract. How did that happen? What are the lessons from this example?

Notes

1. Phases of the project life cycle can be categorized in many ways. PMBOK, 6th edn categorizes them as (1) Starting the project, (2) Organizing and preparing, (3) Carrying out the work, and (4) Ending the project; PMBOK, 7th edn describes project life cycles as predictive, incremental, and adaptive. PRINCE2 specifies the final phase as Post-project Assessment. The 3-phase life cycle in this book addresses virtually everything in these other categorizations, regardless of terminology.

2. It could be argued that Phase D in an election-campaign project will be extended *if* the candidate is elected, whereupon the “operation” phase represents the elected official’s full term in office—but that would be stretching the analogy.
3. Based upon information collected and documented by Cary Morgen from interviews with managers from Jamal Industries (factual case, fictitious name).
4. Cusumano M. and Selby R. *Microsoft Secrets*. New York, NY: Free Press; 1995, p. 210.
5. A need is a value judgment that a problem exists. Different parties in an identical situation might perceive the situation differently; therefore, a need is always identified with respect to a particular party—e.g. the user. See McKillip J. *Need Analysis: Tools for the Human Services and Education*. Newbury Park, CA: Sage Publications; 1987.
6. Cusumano and Selby, *Microsoft Secrets*. p. 210.
7. In the US a request for quotation or invitation for bid (IFB) commonly suggests that selection of a contractor will be based primarily on price; in an RFP, the nature of the solution and competency of the contractor are as or more important than price. Elsewhere in the world, the terms *proposal* and *bid* often are used interchangeably, a bid being the equivalent of a full-fledged proposal.
8. Adapted from Frame J.D. *Managing Projects in Organizations*. San Francisco, CA: Jossey-Bass; 1987, pp. 109–110.
9. Frame, *Managing Projects in Organizations*, pp. 111–126.
10. Green projects <https://onlinepmcourses.com/green-project-management-ready-to-think-sustainably/> Accessed January 23, 2024.
11. <https://greenprojectmanagement.org/about/what-is-sustainable-project-management> Accessed January 23, 2024.
12. US Green Building Council, LEED v4 for Homes and Midrise—ballot version. <https://www.usgbc.org/resources/leed-v4-homes-and-midrise-ballot-version> Accessed May 30, 2024.
13. European Commission Climate Action, “Climate Change and Major Projects,” European Union Publications Office, 2016. <https://op.europa.eu/en/publication-detail/-/publication/5535c968-7a41-11e6-b076-01aa75ed71a1/language-en>. Accessed June 19, 2024.
14. Hamilton G., Byatt G. and Hodgkinson J. “How project managers can help their companies ‘go green’: Program and project managers can contribute to sustainability.” *CIO*, November 2, 2010. http://www.cio.com.au/article/366509/how_project_managers_can_help_their_companies_go_green_/, accessed November 14, 2014.
15. Hajek V.G. *Management of Engineering Projects*, 3rd edn. New York, NY: McGraw-Hill; 1984, pp. 39–57; Rosenau M.D. *Successful Project Management*. Belmont, CA: Lifetime Learning; 1981, pp. 21–32.
16. Kerzner H. *Project Management*, 8th edn. Hoboken, NJ: John Wiley; 2003, p. 829.
17. Roman D. *Managing Projects: A Systems Approach*. New York, NY: Elsevier; 1986, pp. 67–72; Stewart R. and Stewart A. *Proposal Preparation*. New York, NY: John Wiley and Sons; 1984.
18. Murphy O. *International Project Management*. Mason, OH: Thompson; 2005, pp. 159–161.
19. Primary source for this example is Gray M. *Angle of Attack: Harrison Storms and the Race to the Moon*. New York, NY: W.W. Norton; 1992, pp. 87–116; the other source is Brooks C., Grimwood J. and Swenson, Jr., L. *Chariots for Apollo: A History of Manned Lunar Spacecraft*. Washington, DC: NASA Scientific and Technical Information Office, SP-4205; 1979, sections 2.5 and 4.2.
20. Brooks, Grimwood, and Swenson, *Chariots for Apollo*, Chapters 2–5.

Chapter 4

Project Definition and System Definition

The result of Phase A is a formalized systems concept that includes a (1) clear problem formulation and list of user requirements, (2) rudimentary but well-conceptualized solution to the need or problem, (3) an elemental project plan in the proposal, and (4) agreement between the customer and the contractor about all of these. The project is now ready to move on to the “middle” and “later” phases of the systems development cycle and bring the systems concept to fruition.

Much of this chapter and the previous one concern conceptualizing and defining the “system”—the end-item of the project. This is also the thrust of the systems engineering methodology, which is often applied to projects that involve more stakeholders, greater technical complexity, greater risk, and with farther-reaching consequences than other projects. Interested readers are encouraged to see Appendix A at the end of this chapter for additional systems engineering methods and tools.

4.1 Phase B: Definition

As Figure 4.1 shows, with approval of the project in Phase A, the thrust of the effort now moves to **Phase B, Definition**. Most of the effort in Phase A was devoted to investigating the *problem*—what is it, is it significant, should it be resolved, and can it be resolved in an acceptable fashion. Now in Phase B, the *solution* is scrutinized: it is analyzed and defined sufficiently so that designers and builders will be able to produce a system that meets the customer’s needs.

The underlying principle behind Phase B, Definition, is, simply, prepare as best you can for what you intend to do *before* you start doing it. Definition means, “Think through what you want to happen and how best to make it happen; do not just jump in and begin!” Definition is important because its outcomes dictate what will happen in the future. In the Definition phase, before things are defined and plans are set, planners still have broad latitude in decisions and the ability to influence project outcomes. Things are still easy to change because the “things” are just plans. Later in the project, after plans are set and work is underway, things are hard to change because the “things” include work already done or fully committed to. At some point the project will be stuck conforming to decisions already made, even bad ones. For instance, it is easy to decide in Definition whether a building will have 5, 10, or 15 floors. But once you

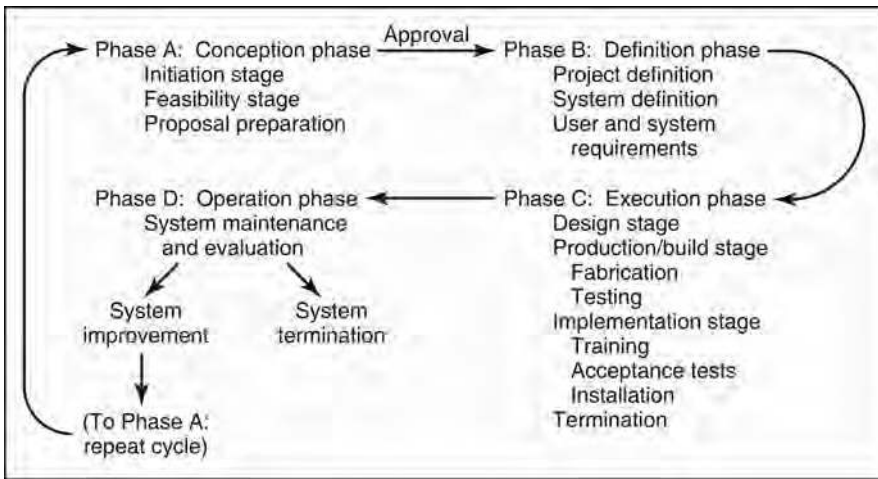


Figure 4.1

Four-phase model of system development cycle.

decide on 10 floors, after 6 floors have been built you cannot change the decision to 5 floors (without tearing down a floor) or to 15 floors (since the foundation was built for only 10).

This is illustrated by three curves in Figure 4.2. Early in the project when it is easy to make decisions that will affect project outcomes, the cost of changing those decisions is little. Early on, little will have been spent (cumulative cost), so it is also easy to cancel the project then. As the project progresses, however, and especially after it enters Execution, the cumulative cost rises dramatically. It is not so easy to cancel the project then because of the high sunk cost. It is also not so easy to change decisions (go from 10 floors to 5 or 15 floors) because so much has already been done and it is costly to redo, undo, or alter it. Definition is that phase where ideas and plans are flushed out before final commitments are made and work begins. The thrust of the phase is twofold: project definition and system definition.

Project definition vs. system definition

There are two ways to look at a project: one is to see the end-item, product, or result of the project, the other is to see the effort being directed at achieving that result. Looking at both is necessary: if you focus too much on the end-item and too little on the effort, the project will run into problems for lack of preparation and coordination of resources, costs, and schedules; if you focus mostly on the effort and less on the end-item, the project will again run into problems—this time for not meeting user requirements. System definition and project definition are equally important. **System definition** aims at achieving a good understanding of what the project end-item or product must do to satisfy user requirements; **project definition** aims at specifying what the project team must do to produce the end-item. While it is not surprising that much of the literature on project management is preoccupied with project definition, it is surprising how little attention it gives to system definition.

System definition begins with defining user needs and requirements; project definition begins with addressing those requirements in the project proposal. Hence, some of the definition work necessary for the project is initiated in Phase A. Phase B continues this definition

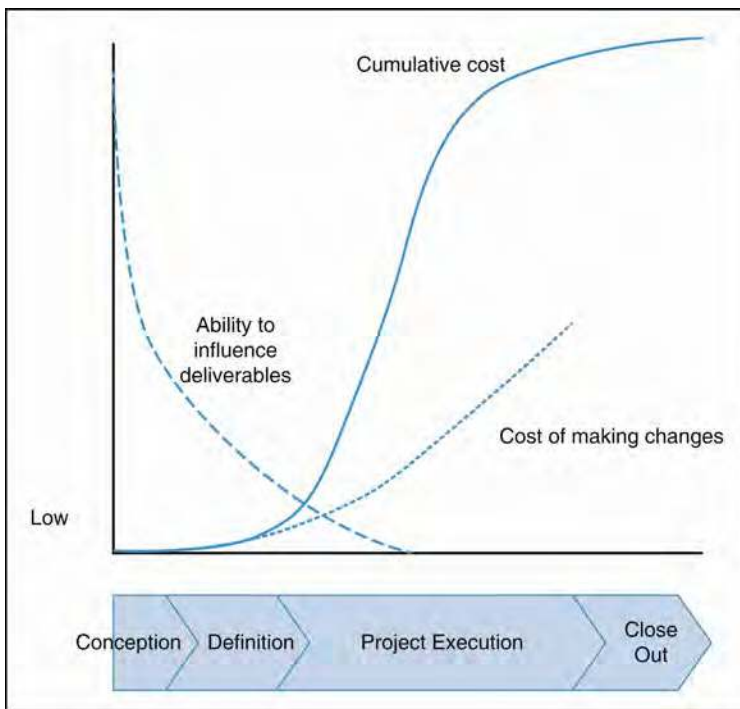


Figure 4.2

Project costs and ability to influence outcomes vs. project phase.

work and concludes with a set of system specifications and a project plan—a full suite of everything necessary to execute the project in Phase C.

Project kickoff

The project formally begins with a **kickoff meeting**—the first formal meeting of the project team and key stakeholders. The meeting’s purpose is to announce that the project is about to commence, communicate what it is about, develop common expectations, and generate enthusiasm and commitment to project goals and deliverables. The project manager plans and runs the meeting. Attendees include the project team (or, if too large, only managers, team leads, and project staff), supporters, and others who should know that the project is about to begin. For a multi-location project, multiple kickoffs at each location or a video or phone conference might be necessary. The kickoff runs 1.5 to 2 hours and is mostly a formal presentation followed by questions and answers.

Invited attendees should be formally notified in advance and provided with information about the meeting agenda, a list of invited participants and their project roles, and a rudimentary project plan. The meeting introduces the following: the project manager; the project SOW, goals, and deliverables; the proposed plan—budget, schedule, main work packages; constraints and risks; the customer, other key stakeholders, and their needs and requirements; the project organization and key team members; and immediate next steps and who is to do

what. Much of this information will have been worked out for the project proposal; if not, the project manager and project team must prepare it for the meeting.

Every project should start with a kickoff meeting. A large project might also have a kickoff for the proposal preparation effort, and then one again for each project phase.



See Chapter 17 shortly after the kickoff. Team building is discussed in Chapter 17.

The purpose of the kickoff is to provide information, not to reach consensus of opinion, develop working relationships, or establish guidelines so team members can work together. The latter is the purpose of team building, for which subsequent meetings should be held

Project name

The project name is often the first thing that people hear about the project—often with no accompanying explanation.¹ The name will appear repeatedly in virtually all communication and persist for as long as the project—and perhaps longer. A carelessly chosen name can cause misunderstanding or a blank stare about the project; it can cause people to confuse the project with other projects; and it can influence the way they react to the project. Unless the intention is to obfuscate the project's purpose (“Manhattan Engineering District”—the atomic bomb project; “Have Blue”—the F-117 stealth fighter project), the name should clearly suggest what the project is about.

Clever or cute names should be avoided; they tend to be ambiguous and, sometimes, annoying. All projects are apt to acquire nicknames, which tend to indicate how people *feel* about the project (“Project from Hell”) but not much else. If, however, the nickname gains widespread usage, then sometimes the sensible thing is to formally adopt it. (Boston's Central Artery/Tunnel became the “Big Dig”—not to be confused with Canada's “Big Dig,” the Wascana Lake Urban Revitalization Project in Saskatchewan. The 1960s geological research project to drill through the earth's crust to the Mohorovicic discontinuity was named Project Mohole, but as political and technical problems mounted, it became known as “Project No Hole.”) A project is often named for a place, person, or the end-item it creates (Petronas Towers; Bandra-Worli Sea Link Bridge), and for long-named end-items it is okay to adopt an acronym (BWSL)—though it's a good idea to first check the acronym before adopting it for the project name; a serious project should not make people chuckle whenever they see its acronym (Automated Network for Uniform Security—ANUS).

4.2 Project Definition

Project definition addresses the question: What must the project do to deliver the system concept and satisfy the user and system requirements? Project definition and system definition happen concurrently and interactively. The work to be done as laid out in the project plan must meet the system requirements, but the system requirements must conform to the work methods, budgets, and schedules specified in the project plan.

Detailed project planning

Prior to Phase B a portion of the project definition will already have been done. At minimum, some definition was necessary in Phase A to prepare the project proposal. But that definition effort will have resulted at best in an outline of what is to come. During Phase B that outline must be expanded and elaborated in detail. The renewed definition effort will involve

identifying the necessary work tasks and resources, project team and leaders, subcontractors, and support staff, and creating schedules, budgets, and cost control systems.

The project team begins to evolve from the skeletal group that prepared the proposal, sometimes in a cascading manner: the project manager selects team leaders who, in turn, fill in team positions under them. The project manager negotiates with functional managers to get specific individuals with the requisite expertise assigned to the project. Sometimes she seeks the customer's approval to add members to the project team, which is advisable whenever the customer must work closely with the team or might have an objection. Good customer–project team rapport is crucial to maintaining a healthy customer–contractor relationship.

Project execution plan

As key members of the project team are assembled, they begin preparing the detailed project plan—the **execution plan**. The audience of the execution plan is whoever will be doing the project, so the plan should address whatever they will need to know, including, for example:

- Scope statement or SOW that includes high-level user requirements and system requirements.
- Work breakdown structure and work packages or tasks.
- Project organization.
- Responsibility assignments of key personnel to work packages.
- Project schedules showing events, milestones, or points of critical action.
- Budget and allocation to work packages.
- Quality plan for monitoring and accepting project deliverables, including testing plan.
- Risk plan and contingency or mitigation measures.
- Procurement plan.
- Communication plan.
- Work review plan.
- Change control plan.
- Implementation plan to guide conversion to, or adoption of, deliverables.
- Change management plan.
- Stakeholder engagement plan.
- Health, safety, and environmental (HSE) policies and plans.

The execution plan is described more in Chapter 6, and its elements are described throughout the book. Concerning the last element above, HSE: perhaps obvious is that project management is responsible for protecting the project team, stakeholders, and society from injury arising from immediate and long-term health hazards associated with the project and its outcomes. Minimally, the project plan includes measures to guard against accidents and health hazards, comply with industry standards and municipal, state, and federal laws and regulations, and to meet the unique circumstances of the project.² The plan also includes measures to mitigate negative environmental impacts of the project, although such measures might also be addressed in sections of the project plan such as SOW, work breakdown, risk, etc. Company policies regarding HSE should be referenced in the plan. The project manager is responsible for ensuring the policies are implemented, that specific HSE roles and responsibilities are defined, and staff receive appropriate health and safety training.³ Significant hazards that cannot be eliminated should be included in the risk management plan. Preparation of the HSE plan includes attention to environmental and sustainability matters as discussed in Chapter 3.



See Chapter 6



See Chapter 3



See Appendix C

The elements of the execution plan must be integrated; each must be tied to, compatible with, and supportive of the others. Details of these elements are discussed in Part III of the book, and a sample project execution plan is in Appendix C at the end of the book.

In large projects the planning is divided into sub-plans created by members of the project team, including subcontractors. The project manager coordinates their efforts to ensure that the sub-plans are thorough and tied together. Contractor top management and the customer review the final plan for approval. Contractor top management makes sure that the plan fits into existing and upcoming projects and capabilities, and the customer makes sure it conforms to user requirements and the conditions stated in the contract.

Anxious to get the project underway, many contractors skip reviewing the project plan with the customer. This is shortsighted since the plan could contain elements to which the customer might object. Often the project is conducted and implemented within the customer's organization, so everything in the plan must fit: the project schedule must fit the customer's schedule; project cash flow requirements must fit the customer's payment schedule; the contractor's personnel and procedures must complement those of the customer; and the materials and work methods must be acceptable to the customer.



See Chapters 5 and 13

Once management and the customer approve the project plan and system specifications, the project team turns its attention to the detailed design and building of the system, which is what happens in Phase C, as covered in Chapters 5 and 13. As explained next, however, project planning never stops; it continues throughout the project life cycle.

4.3 Phased (Rolling Wave) Project Planning

A major thrust of Phase B is to develop the project plan, but seldom does it result in a comprehensive plan for the entire project. The fact is, despite all the effort devoted to planning in Phase B, often the plan is developed in phases, not all at once. At the start of a project there are too many unknowns, and it is impossible to specify exactly what will or should happen for the whole project. Only as the project progresses and the unknowns decrease can the details of the plan be filled in. The situation is analogous to planning an off-the-road route to some destination, but without the benefit of knowing the obstacles. Since you can only see the landscape directly ahead, you can only plan the first part of the route in detail; beyond that, the route is vague. This is represented by Phase I in Figure 4.3a. As you move through Phase I, you see more of the obstacles ahead, which enables you to plan the next part of the route, Phase II (b). The process continues, filling in details of the route, phase by phase, until you reach the destination (c and d).

At the beginning of a project the customer wants to know the project cost and completion date, which can be estimated by preparing an initial rough plan. Although much of the initial plan is somewhat vague (analogous to the shaded blob in Figure 4.3), the plan is usually sufficient to enable managers to estimate project resources, time, and cost. As the project gets underway, more-detailed plans are created but only for the most immediate phase of the project (dotted lines, Figure 4.3). Whereas the initial plan was based upon information from similar projects, estimates, and forecasts, the detailed portions of the plan are based upon facts about upcoming work, facts identified as the approaching work gets closer.

For highly unique projects, the initial rough plan should be seen as just that—a rough indication of project deliverables, cost, and delivery date—but not necessarily a commitment. That plan was first prepared during the feasibility study or business case study, but as the project progresses, it is replaced with more detailed plans and more-specific work tasks and schedules. Only for the most immediate phase where the “terrain” is clearly visible is it possible to create

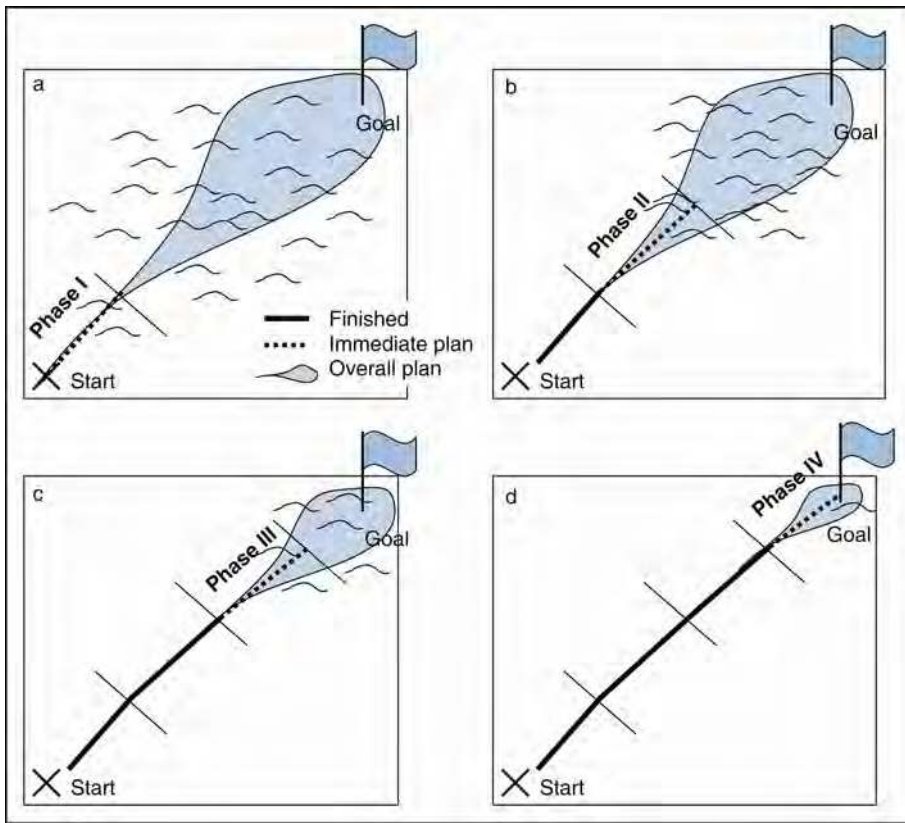


Figure 4.3
Phased project planning.

a detailed plan and make commitments to work, dates, and costs. Application of this **phased** or **rolling-wave planning** is an aspect of **agile projects**, which are described in Chapter 14.

In some projects, each formally identified project phase concludes with a **phase gate**, at which time the customer or executive managers review the project's deliverables and performance; if satisfied, they approve the deliverables and pay for work done thus far. At the same time, they review the plan for the next phase and assess the costs, risks, etc. of the updated high-level plan for the rest of the project. Note, this requires the plan for each phase to be largely prepared in the prior phase, as illustrated in Figure 4.4. If satisfied with the plan, they authorize the project to proceed to the next phase. If a project is to be terminated, that happens only at the end of a phase, unless termination is forced as the result of unforeseen external circumstances.

In some organizations this **phase-gate review** process is a mere formality, a “rubber stamp” to proceed to the next phase. More often, however, it serves an important purpose, in some cases having strategic implications. For example, each phase-gate review can result in a:

1. **Green light:** all relevant stakeholders are satisfied with the work performed so far. They accept the plans for the rest of the project and believe that the risks and business impact of the project justify continuing with the project. The project is authorized to proceed to the next phase.



See Chapter 14

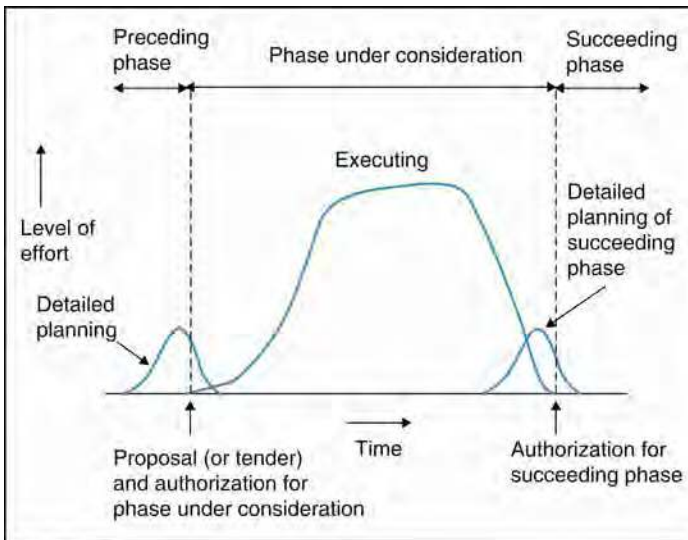


Figure 4.4

Effort and detail of planning vs. project phase.

Adapted from Steyn H. (ed.) *Project Management—A Multi-Disciplinary Approach*. Pretoria: FPM Publishing; 2003. p: 27.

2. Yellow light: the stakeholders feel the business impact of the project justifies continuing the project but not that the objectives of the phase were met or plans for the rest of the project are adequate. The project team must re-do part or all of the preceding phase and/or part of the plan.
3. Red light: due to changes in the business environment or risks, or disappointing results so far, stakeholders might consider the business impact of the project as insufficient and decide to terminate the project. If the possibility exists that conditions might improve later, the project can be “put on ice” for later reconsideration.



See Chapter 19 Companies doing multiple simultaneous projects sometimes use phase-gate reviews to compare the projects—their benefits, resource requirements, and relative performance, and to determine which to give the green, yellow, or red light; this is discussed in Chapter 19. The phased approach to project planning and approval is illustrated in Example 4.1.

Example 4.1: Mary and Peter’s New House

Mary and Peter buy land to build a new house upon. They approach NewHome Construction and describe to Paul, its owner, what they have in mind. Among other things, they want to know what it would cost. Having been in the business for many years, Paul has an idea of the cost but is wary of quoting a fixed price since he does not know Mary and Peter very well and whether their tastes are cheap or expensive. Also, he is wary of possible hidden costs arising from, for example, poor soil conditions of the site. He therefore gives a range of possible prices based upon the estimated square footage of the house and its estimated

completion date. Nobody has yet made any commitments. On the question, “Where do we go from here?” Paul answers that the first phase is to create a concept design and sketches of the house. He also outlines the other phases of the project he foresees, and the deliverables and approximate schedule and cost for each.

Mary and Peter sign a contract for Paul to provide the preliminary concept design and sketches. The contract specifies when they will see the design and sketches, what the sketches will include and exclude, and what they will cost. Within a month they receive and approve the design and sketches.

Paul now presents them with a second contract, this time for detailed drawings that the construction team will use to build the house. Just like the first phase, the contract specifies the deliverables (drawings), delivery date, and price. A few months later Mary and Peter approve the drawings and construction begins.

Paul notes that the construction work will also be done in phases, although now, he says, there is sufficient information about the project and its cost so that only one contract is needed. He shows them the contract, which lists the remaining phases of the project and includes a guarantee period (following completion of construction) during which NewHome will fix any defects free of charge. The contract indicates milestones and deliverables for each of the phases and specifies that a payment will be due upon reaching each milestone. Before each payment, Mary and Peter will have the chance to inspect the work and verify that it has been completed and meets workmanship standards specified in the contract.

This example illustrates the benefits of phased project planning: at the start, NewHome does not have to commit to the cost of building an as-yet undefined structure, and during the project Mary and Peter do not have to commit to work beyond any one phase (in fact, at the conclusion of any contracted phase they can walk away from the project). The milestone payments improve NewHome’s cash flow and reduce interest payments on money for construction borrowed from the bank. They also provide NewHome with some protection against bad debt: if Mary and Peter miss a milestone payment, NewHome simply stops work.

Project charter

The **project charter** is a proclamation that management has approved a project. For some projects, it is created once, following a feasibility study or acceptance of a proposal; for larger projects, it is created and expanded at multiple points during the conception and definition phases. For internal projects, the charter serves the purpose of announcing and formally authorizing the project’s start. For external projects, that same purpose is served by a contract, so, generally, there is no charter.

The charter is issued by the project sponsor or originator. It describes the project to stakeholders in the organization and establishes the project manager’s authority to organize and use resources; thus, it should be signed by at least one executive manager. It includes whatever information is necessary to give the reader a good overview of the project. Often the charter contains sections like the project plan, and sometimes it is the project plan, although commonly it is somewhat brief and provides only an overview of the execution plan, described earlier.

For any reasonably-sized project, the project charter is developed after some prior planning and a feasibility study. In large projects conducted in phases (e.g. FEL, described later), a charter is created for and used to authorize each phase. The PRINCE2 methodology prescribes

three charters: one (called a “mandate”) authorizes the first, pre-project, stage of the project; another (a “brief”) authorizes the second, initiation, stage; and a final (set of “project initiation documents”) authorizes subsequent stages.

For a small project or initial phase of a project, the charter can be rather short. For most projects, however, it is more comprehensive and can include the following:

- Project vision, purpose, benefits; problem it will solve or opportunity it will exploit.
- Project justification (business case and environmental impact analysis findings).
- Approach to be followed.
- Project scope statement.
- Main deliverables, criteria for acceptance, individuals responsible for acceptance.
- Clients and key stakeholders.
- Identification of the project manager and his authority and responsibilities.
- Identification of other decision makers, their authority, responsibilities, and reporting relationships.
- Listing of resources, including project team staff, required training, subcontractors, etc.
- Project organization and work breakdown structure.
- Project budget summary and cash flow plan.
- Master schedule, project phases, key milestones, planned due dates.
- Perceived risks and issues.
- Objectives for sustainability.
- Procurement policies.
- Plans for safety, health, environmental protection; communication.
- Control procedures.

Despite similarities, the project charter differs from the execution plan in one important way. Whereas the purpose of the charter is to *describe*, *justify*, and *authorize* the project; the purpose of the execution plan is to *give direction* to parties working on the project (team, contractors, etc.). This leads to major differences in the content of each and, usually, an execution plan that is substantially longer, more detailed, and more comprehensive than the charter.

Front-end loading⁴

A variation of phased project planning and approval used by some industries (chemical, mineral, oil, and gas) in major industrial infrastructure projects or **megaprojects** (typically costing over \$1 billion) is the **front-end loading (FEL)** approach. FEL overlaps the Conception and Definition phases in Figure 4.1 and includes all the data gathering, analysis, and documentation necessary to justify and launch a project. It is divided into three phases, FEL-1, FEL-2, and FEL-3.

FEL-1, called “opportunity identification,” is the idea-generation and evaluation phase; it corresponds somewhat with the “pre-project stage” in the PRINCE2 methodology. The proposed project is in competition with other projects to receive funding, thus the objective of FEL-1 is to confirm that the project is compatible with organizational strategy and is a “preferred project” (discussed in Chapter 19). The output of FEL-1 is a preliminary business case that confirms the feasibility of the proposed capital investment.

FEL-2 goes by different names depending on industry, for example “business planning,” “concept study,” and “appraise.” In this phase the project is “shaped” in terms of scope, technology selection, and execution strategy. The output of FEL-2 is a detailed business case and a scope statement that enables reliable cost and schedule forecasts. Typically, only 1 percent of the total project cost is incurred during FEL-1 and FEL-2. Although that sounds like a small



See Chapter 19

amount, on megaprojects of \$1 billion or more, it is substantial and often leads to pressure to reduce the budgets for FEL-1 and FEL-2. But just like cutting costs on the foundation of a building, it is a poor idea to skimp on the early FEL phases and steam ahead.

FEL-3, called “project definition,” includes preparation of a detailed project execution plan, advanced conceptual design, and some detailed system design (which in the Figure 4.1 methodology is placed as the first stage of Execution). FEL-3 is often divided into sub-phases that go by such terms as “facilities planning/execution planning,” “feasibility,” and “select/FEED (front-end engineering design).” The output of FEL-3 is a project execution plan, conceptual (ready for detail) design, basic engineering plan, and detailed project charter. By the end of FEL-3, typically 3 to 5 percent of the total project cost is spent, a relatively small amount not to be stinted on, to ensure that project risks are acceptable before committing full funding to the project.

Each FEL phase is followed by a gate: FEL-1 to assess the robustness of the business case, FEL-2 to assess the completeness of the scope definition, and FEL-3 to determine if the project is ready to execute. Since FEL-3 is the most expensive part of FEL, it is not undertaken unless the project has already been approved. The project approval decision happens at the FEL-2 gate, which means that the FEL-2 phase must be very thorough and address all factors important in the decision. Project cancellation at the FEL-3 gate happens rarely and only with changes in the project environment (sharp drop in market, business downturn, dropout of a major business partner).

Besides project definition, FEL also addresses system definition, described next.

4.4 System Definition

Systems are defined from their requirements; thus, requirements are the starting point for all systems development projects and the foundation for project planning. Each requirement impacts end-item scope and complexity, which in turn impacts project work effort, time, cost, and risk. Unless the requirements are clearly defined and agreed upon, it will be difficult to fully conceptualize the end-item and create a viable project plan. With the contract signed and the project about to get underway, the user requirements defined in Phase A should be reviewed, the details filled in, and any gaps and ambiguities eliminated.

User requirements revisited

For products and systems in competitive markets, user requirements are initially framed in general terms; for example, outperform the F-35, taste better than Joe’s beef jerky, obtain at least a 20 percent rate of return, or upgrade to software release 10.2. General requirements such as these must be expanded before serious development work and project planning can be started. As Example 4.2 shows, poor requirements definition can lead to project failure.

Example 4.2: User Requirements for Product Development

The marketing group for a kitchen appliance manufacturer wrote the requirements for a new food processor. The requirements specified the general size, weight, usage, price, and sales volume of the proposed product, but nothing about product performance, which the engineering design group had to set by studying competitors’ products. The food processor

as developed met all the requirements set by marketing and engineering, yet it was obsolete before even being launched because competitors had released products better suited to customer needs. In defining the product, both the marketing and engineering groups had ignored the **user** requirements for the food processor—i.e., the requirements as specified by actual user customers.

Defining complete, accurate requirements is not easy. Among the problems are:

- Requirements must incorporate information from not only the user but also functional areas such as marketing, engineering, manufacturing, and outside stakeholders.
- The information needed to define requirements is not always available when definition occurs, so it is easy to overlook necessary requirements or include unnecessary ones.
- The requirements include vague terms that cannot be accurately measured (e.g. “modern,” “enough,” “comfortable,” “sufficient,” or “low cost”).
- The user or contractor cannot adequately describe the requirements because the end-item is complex, abstract, or artistic.
- The customer or contractor intentionally defines requirements in ambiguous terms to allow latitude in results later in the project.

Problems like these result in confused project planning and, later, customer–contractor disputes over whether the end-item met the requirements. The following steps can reduce such disputes.⁵

- Convince both the user and contractor groups of the importance of a clear, comprehensive definition of requirements. Users and contractors often are reluctant to devote the time necessary to define clear and complete requirements.
- Check for ambiguities and redefine the requirements so none remain.
- Augment written requirements with nonverbal aids such as pictures, schematics, graphics, and visual or functional models.
- Avoid rigid specification of requirements that are likely to change due to uncertainty or changing environment.
- Treat each requirement as a commitment to which the users and the contractor both agree and sign off.
- After the project begins, monitor the requirements and resist attempts to change them.
- Use a change control system to assess the necessity and impacts of changes before deciding whether to approve them.

Detailed user requirements come from one source: users. The project manager, however, should not accept just any requirements provided by users but offer help defining them. Just as users sometimes require help in understanding the problem or need, they also might need help in specifying their requirements. They may not understand the ramifications of requirements in terms of cost or schedule, or what will be necessary to fulfill them.

For most projects the list of high-level user requirements (summary or bullet points) should fit on one page for easy reference. Early in the project the contractor will refer to the

list when preparing the project’s scope statement; at the end of the project the customer will refer to it to determine the acceptability of project results and end-items.

Preliminary definition of user requirements happens during the feasibility study and proposal preparation, and a summary of user requirements is included in the contract. In simple systems, user requirements rarely exceed a few lines or a page. In big systems, however, they might fill volumes. An example of the former is user requirements for a contract to perform a 1-day management seminar; an example of the latter is user requirements for the 9-year, multibillion dollar Delta Project to prevent the North Sea from flooding the Netherlands.

System requirements

A major thrust of Phase B is translating user requirements into **system requirements**. System requirements are oriented toward the solution; they specify the contractor’s approach and objectives for satisfying the needs as spelled out in the user requirements. But beyond fulfilling user requirements, a project must also fulfill contractor needs, such as being profitable and keeping skilled workers employed and costly production facilities operating.

System requirements define the system or solution approach—including the principal functions, system architecture, and resulting end-item (system, solution, or product)—and provide a mutual understanding among project team members as to what must be done in the project. Whereas user requirements represent the user’s perspective, system requirements derive from the contractor’s perspective. They state what the end-item system must do to satisfy the user requirements. Following are examples contrasting user requirements and system requirements:

Table 4.1 Example user requirements and corresponding system requirements.

User Requirements	System Requirements Will Address
1. Vehicle must accelerate from 0 to 60 mph in eight seconds and accommodate six people.	Vehicle size and weight, engine horsepower, kind of transmission.
2. House must accommodate a family of four.	Number and size of rooms.
3. House must be luxurious.	Quality and expense of materials and decorative features.
4. Space station must generate electricity for life support, manufacturing, and experimental equipment.	Type and kilowatt capacity of power-generating equipment; technology for primary operation and backup operation.
5. Aircraft must be “stealthy.”	Design of configuration and external surfaces; types of materials; usage of existing or newly developed components.

System requirements specify targets that the project’s designers and builders must achieve in designing and building the end-item. The following illustrates this for the X-Prize/SpaceShipOne project introduced in Chapter 1.

Example 4.3: High-level System Requirements for Spaceship

Below are five user requirements for the spaceship, each followed by one or more system requirements. The former specifies the user requirements, the latter what the spaceship and its subsystems and components must do to satisfy those requirements.

1. Attain altitude of at least 100 km:
 - 1.1 Motor must provide enough thrust (i.e., be powerful enough)
 - 1.2 Motor must burn long enough
 - 1.3 Vehicle must be lightweight
2. Capacity for three people:
 - 2.1 Cabin must be large enough
3. Comfortable flight:
 - 3.1 Cabin temperature must remain at comfortable level
 - 3.2 Cabin pressure must remain at comfortable level
 - 3.3 Vehicle must not exceed certain acceleration force
 - 3.4 Cabin must have sufficient elbowroom
4. Relatively inexpensive to design, build, and launch:
 - 4.1 Fuel and fuel handling procedure must be economical
 - 4.2 Structural materials of vehicle must be economical
 - 4.3 Whenever possible, uses existing, off-the-shelf technology and systems
 - 4.4 Requires few people to maintain vehicle
5. Capable of being “turned around” in at most 2 weeks:
 - 5.1 Minimum repair/replacement of parts/modules between flights
 - 5.2 Minimum refueling time
 - 5.3 Minimum cabin cleaning time.

Notice, the system requirements specify “what” the system must do, not “how” it will do it. They say, e.g. “the motor must generate enough thrust to propel the spaceship to 100 km. before it runs out of fuel,” but not how. Addressing the “how” comes later.

Defining requirements in sufficient detail so designers will know what they are striving for is called **requirements analysis**. The result of the requirements analysis is a comprehensive list of functional requirements.

Functional requirements

Functional requirements specify the functions the new system must perform to meet the user requirements. For example, the functions of the spaceship include propulsion, handling and maneuverability, human habitability, safety, and support and maintenance. The common tool for identifying the functional requirements of a complex system is the **functional flow block diagram, FFBD**, described in Appendix A of this chapter. All significant functions for the system, its subsystems, components, and interfaces, including for support and

maintenance must be identified. Most systems perform several basic functions, each of which has numerous sub-functions.

Associated with each functional requirement are targets or **performance requirements**. These specify in technical terms—e.g. physical dimensions, miles per hour, turning radius, sound decibels, acceleration, percent efficiency, operating temperature, operating cost—the target requirements that the function must satisfy as well as the tests, procedures, and measures to be used to prove that the targets have been met. The project team refers to these performance requirements in the design or purchase of components for the system.

In addition, other requirements might be imposed on the overall system or specific subsystems and components. These requirements are called, perversely, **non-functional requirements** (!) because they are not tied to any specific functions and are desirable for the entire system and all its components.

The following are typical:⁶

1. **Compatibility.** Ability of subsystems to be integrated into the whole system or environment and to contribute to objectives of the entire system.
2. **Commonality.** Ability of a component to be used interchangeably with an existing but different type of component. A “high commonality” system contains many available off-the shelf (OTS) components; a “low commonality” one has many that must be newly developed.
3. **Cost-effectiveness.** Whether the total cost of the system, including its design, implementations, and operation, is commensurate with its benefits.
4. **Reliability.** Ability of the system or component to function at a given level or time before failing.
5. **Maintainability.** Ability and ease of the system to be repaired within a reasonable period.
6. **Testability.** Degree to which the system can be systematically tested and measured for its performance capabilities.
7. **Availability.** Degree to which the system can be expected to operate when it is needed.
8. **Usability.** Amount of physical effort, technical skill, training, or ability required to operate and maintain the system.
9. **Robustness.** Ability of the system to survive in a harsh environment.
10. **Expandability.** Ability of system to be easily expanded to include new functions or adapted to new conditions.

Requirements priority and margin

Two properties of each requirement are its **priority** and **margin** (the latter also referred to as **tolerance**). The *priority* of a requirement is, simply, the relative importance of the requirement. When multiple requirements conflict so that not all of them can be met, priority determines which will be met and which not. Suppose a product is specified to, e.g., perform in a certain way and be a particular maximum height, but performance has priority. This will be useful to the design team if later they determine that to achieve the specified performance the height requirement must be exceeded.

Related to priority is the *margin* on a requirement—the amount by which the requirement can vary. For example, the requirement “maximum height of four feet; margin of two inches” tells designers that in case they must exceed the height requirement, they have at most two more inches.

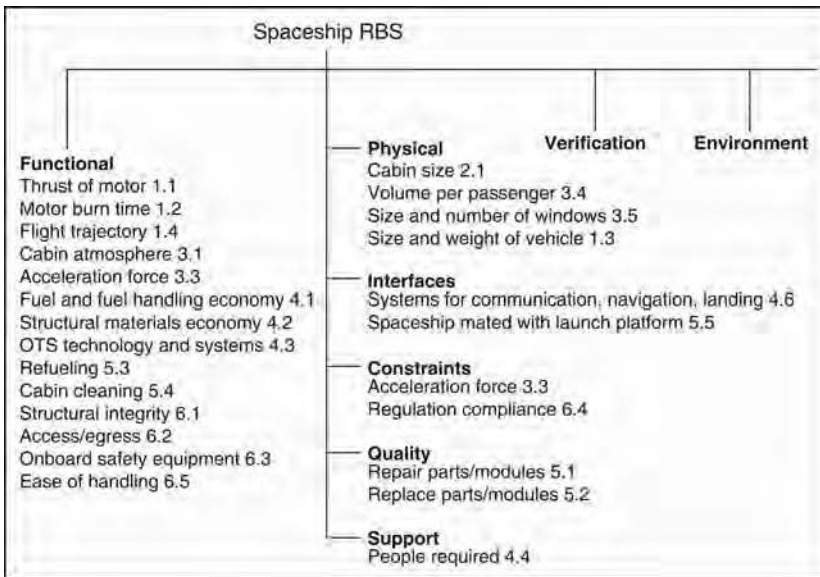


Figure 4.5
Requirements breakdown structure.

Requirements breakdown structure

During requirements analysis, system functions are sorted and clustered into logical groups. The **requirements breakdown structure** (RBS) in Figure 4.5 is a simplified example showing ways of grouping requirements. The RBS should include every identified functional requirement; in large systems these can number in the hundreds or even thousands.

The RBS provides a common reference for everyone working on the project. Often a requirement will pertain to multiple system components, which means that multiple project teams will be working to meet that requirement. The RBS enables these teams to coordinate efforts and avoid omissions or duplication.

System requirements provide general direction for the project, but they are high-level and not sufficiently detailed to tell the project team what it must design, build, or purchase to create the end-item system. Stipulations must be placed on each of the requirements; these are called system specifications.

System specifications

System specifications are derived from system requirements. They define the end-item and its subsystems, components, and processes in sufficient depth so that the project team will be able to design, build, and/or purchase those subsystems and components.

System specifications are the basis for specifications of lower-level subsystems, which are the basis for specifications of detailed design drawings for the lowest-level subsystems. From the system specification for an automobile, for example, specifications are derived for the auto's drive train, suspension, steering system, brake system, etc. The specifications for these lower-level components normally take the form of a drawing or, for a commercially available

“off-the-shelf” (OTS) item, a catalog number. The relation between user requirements and system requirements and specifications is illustrated in Example 4.4.

Example 4.4: System Specifications for Spaceship

The progression from user requirements to system requirements and from system requirements to system specifications is illustrated in Figure 4.6. At the top, the system requirement “Motor must provide enough thrust” is derived from the user requirement “Spaceship must reach 100 km”; in turn, the system specification “Motor must provide ≥ 88 kN thrust” is derived from the system requirement “Motor must provide enough thrust.” (Note, kN or kilonewton, is a measure of force, used here to measure rocket motor thrust.) System specifications tell the project team what it must do and targets it must meet. For example, besides the motor having a specific thrust, another specification, 4.1.1, says the motor will burn nitric oxide and rubber. Since there are no OTS motors that do this, the team will have to design and build one from scratch. The multiple arrows to each specification indicate that each system specification must satisfy system multiple requirements.

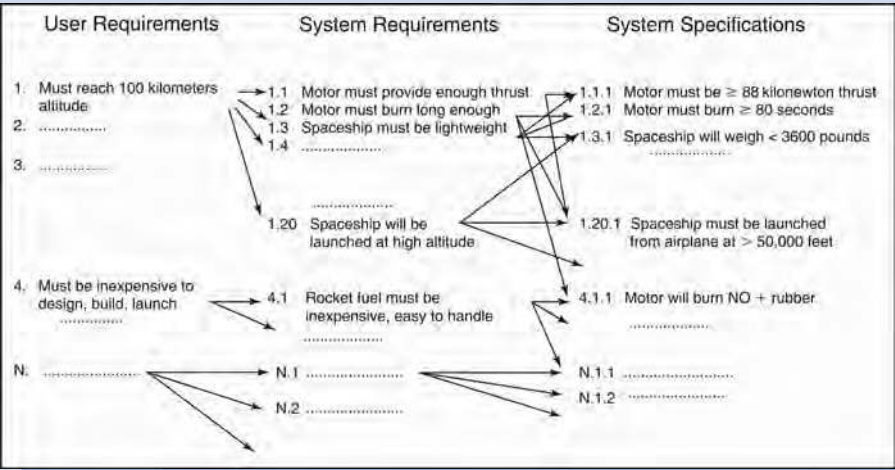


Figure 4.6
Relationships between user requirements, system requirements, and system specifications for spaceship.

Traceability

Developing clear specifications is important, but so is keeping track of their relationships to each other and to system requirements. Throughout the system development cycle numerous changes and tradeoffs will be made to the requirements that will each impact multiple specifications. For example, altering the spaceship weight (Figure 4.6, system requirement 1.3) will impact the spaceship’s required launch altitude (specification 1.20.1) and the motor’s required

thrust and burn time (1.1.1 and 1.2.1). Because weight impacts so many of the specifications, a designer cannot be cavalier about doing anything that might alter it. Any decision affecting weight must be assessed for the impact it will have on the specifications for launch and rocket motor. The ability to trace the effects of changes in some specifications and requirements to others is called **traceability**. A useful tool for this purpose is the **requirements traceability matrix**, described in Appendix A to this chapter.

See Chapters 10
and 13

The process of managing all of this—identifying requirements and specifications, tying them to physical components, tracing the impacts of changes, and controlling changes so requirements are met and do not conflict is called **requirements management (RM)**; it is part of **configuration management** and **change control**, discussed in Chapters 10 and 13. RM involves monitoring, revising as necessary, and updating requirements and specifications documentation. Big projects have numerous requirements, and someone should be responsible for RM, otherwise requirements tend to grow and change uncontrollably, leading to scope creep, rework, schedule and budget overruns and customer dissatisfaction.

System requirements and specifications are written by and for project subject-matter specialists—systems analysts, programmers, engineers, product and process designers, technical consultants, etc.—and they address all areas of the project—design, fabrication, installation, operation, and maintenance. They should be set to meet but not exceed customer baseline specifications, which are high-level specifications the customer can understand.

4.5 Alternate Approaches to Definition and Execution

When the project end-item or problem solution is well understood and requirements can be readily defined, the definition process can proceed as described, stage by stage according to the life cycles shown in Figures 4.1 and 4.7. As mentioned, a project where the work flows in this fashion is called the **traditional** or **waterfall methodology** (or, what the PMI calls the **predictive approach** since, presumably, project work and outcomes are all predictable). The

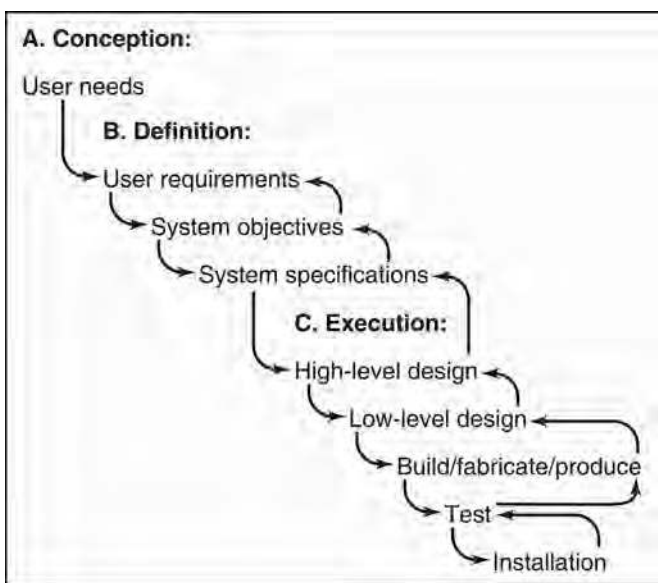


Figure 4.7
Development cycle
(waterfall process)
for complex systems.

project advances like a waterfall, “cascading down” from one stage to the next. For projects where the requirements can be defined early and are unlikely to change much (e.g., designing and constructing a new building or airplane), this methodology is usually appropriate. The problem is, when a project is not fully understood and the requirements are vague or likely to change, then the steps will have to be repeated. The waterfall process can accommodate this (back-arrows in Figure 4.7), although not efficiently because it often requires redoing work already done and is costly and time-consuming.

So, what to do when the end-item or solution is not well understood (or is even unknown) and the requirements cannot be well defined? The answer: use a different approach, one where the definition of requirements, and the design and testing of the system happen through a *repetitious* and more *trial-and-error* process. The requirements and end-item *evolve* through a series of **iterations**, where, in effect, the stages of definition, design, development, and testing are *repeated*. Each iteration delivers a partial yet stand-alone and functional result, plus the knowledge to improve the requirements and the system. A project life cycle that goes through iterations like this is called **agile** or **APM methodology** (what the PMI calls an **adaptive approach**—i.e., where the project life cycle is adapted to the state of knowledge about the end-item or solution), as described next.

Iterative design-testing and rapid prototyping

The evolution of the requirements and the end-item through iterations can occur in different ways. One way is through the APM approach, which is described in depth in Chapter 14, another is through use of a **prototype**, which is an early running model of a system or component built for purposes of learning about it, demonstrating its performance and functionality, or proving its feasibility. The prototype is built according to initial requirements—details about which might be vague—and is tested. Based upon the test results, the requirements and the prototype are modified and tested again. This cycle might be repeated many times before something emerges that completely satisfies the customer.



See Chapter 14

When the system is new to the market or even new to the world, it can be very difficult to define requirements and conceptualize the system. The system that the customer “sees” might be quite different from the one the developer sees, yet without a physical or working model the difference might not be apparent. Requiring the customer and developer to specify and sign-off on requirements early in the project only intensifies the problem. It forces them to commit to decisions before they reach a mutual understanding about the requirements. Prototyping helps them to see the results of their requirements in actual use.

In a process called **rapid prototyping**, a rudimentary, intentionally incomplete model of the product that is initially somewhat simple and inexpensive is produced.⁷ The rapid prototype (RP) model represents *key parts* of the system, *not* the complete system, and is somewhat easy to create and modify. The customer can experiment with the RP to assess the system’s functionality and determine any necessary modifications or additions. After iterations of experimenting and modifying requirements, the final requirements and design concept are *firmed up*. Product engineers and architects use 3-D physical scale models of products and buildings to see and experience how the end-item will function and what it will look like. They know that a physical model is always better than drawings or lists of requirements for conveying the look, feel, and functionality of a design. Drawings and requirements tell the development team what is expected, but the RP process ensures that the drawings and requirements are finalized only after the customer has accepted them as represented by the RP model.



See Chapter 14 Chapter 14.

Software developers use a variation of RP when they create products through iterations, starting with a product that meets high-level, crude requirements, and then refining it to incorporate requirements that are more detailed and better suited to the customer's needs. Based upon customer feedback about early releases, developers *home in* on the requirements and resulting software product that best satisfies the customer. This is described more in

Team involvement in definition

As requirements and the project plan are being developed, questions arise, “How do you know when the requirements are complete?” and “How do you keep everyone in the project focused on those requirements?” The problem is especially tricky when the project involves numerous people and teams, and spans months or years. Part of the answer is: make the system and project definition a *team effort* incorporating the perspectives of everyone who has or will have a significant stake in the project—customers, users and operators, suppliers, and functional areas such as engineering, marketing, manufacturing, customer service, and purchasing. The more these individuals and groups have a hand in defining requirements and the project plan, the better the plan will account for the requirements, needs, and interests of all stakeholders throughout the system life cycle. Everyone in the project should be working to the same set of requirements—a master requirements document, RBS, or equivalent. Any additional, necessary requirements should be derived from and compatible with this master document.

In product development projects, an effective way to generate product requirements is at an offsite workshop for all the key project stakeholders, including functional groups, users, and suppliers. Beginning with a list of customer/user needs or requirements, the team develops the system requirements (lacking adequate customer/user requirements, the team develops them too). For complex systems, a better approach is to *create a team* comprised of all key stakeholders to define requirements. The term for this is **concurrent engineering**, which implies the *combined* efforts of key stakeholders to define requirements to the satisfaction of everyone. The term is somewhat misleading since the approach involves not just engineering but marketing, purchasing, finance, quality, and more. Concurrent engineering has a close parallel in software development, where **DevOps teams** (developers working closely with operators, users and customers) create software to meet the needs of those who operate, interface with, and maintain the system. (Case 14.2 in Chapter 14 describes a DevOps team gone awry.)



See Chapter 14

Concurrent engineering teams are also called **design-build teams** because they combine the interests and involvement of designers, builders, and users into a single effort.

Example 4.5: Design-build Teams at Boeing⁸

At one time in the Boeing factory, the production plant was on the main floor, and the engineering group was upstairs. Whenever a problem occurred in the plant, engineers just walked down to look. Today, Boeing employs thousands of people at several locations, and such easy interaction is not possible. Like other large corporations, as Boeing grew, its finance, engineering, manufacturing, and planning units evolved into “silos,” each with strong self-interests and little interaction with the others. In the development of the 777 commercial aircraft, Boeing wanted to change that and implemented the “design-build team” concept, or DBT. Each DBT includes representatives from all involved functional units, customer

airlines, and major suppliers. The concept emerged from one question, “How do we make a better airplane?” The answer required not simply a good understanding of aircraft design and manufacture, but also knowledge of aircraft operations and maintenance. To capture such knowledge, customers, manufacturers, and designers joined together early in the project to discuss ways of incorporating all their objectives into the aircraft design.

The formation of DBTs mirrored the physical breakdown of the major subsystems and subcomponents of the airplane. For example, the wing was divided into major subsystems such as wing leading edge and trailing edge and was then further broken down into components such as inboard flap, outboard flap, and ailerons; responsibility for each subsystem and component was handled by a DBT.

The project required 250 DBTs, each with 10 to 20 members and run like a little company. The teams each met twice weekly for a few hours, following a preset agenda coordinated by a team leader, and involved people from airlines, finance, production, and quality.

Since most components in an airplane interact with numerous others (i.e., they interface), most participants in the program had to be assigned to multiple DBTs (to ensure their components would work with other DBT’s components). The manufacturing representative, for instance, belonged to 27 DBTs. His duty was to tell engineers what would happen when their elegant designs met with the realities of manufacturing processes and assembly line and maintenance workers, and he offered suggestions that would improve the airplane’s maintenance. One suggestion concerned the cover on the strut-faring that holds the engine to the wing. The faring would contain many electrical and hydraulic components that maintenance personnel would need to access, but design engineers had not noticed that repairing these components would require removal of the entire faring. But the manufacturing rep noticed and suggested adding a big door on each side of the faring. This improved access to the components inside and greatly simplified the repair work of maintenance personnel.

Concurrent engineering teams are discussed more fully in Chapter 15.



See Chapter 15

4.6 Project Communication

Everything described thus far involves communication. Projects are group efforts that involve many people, teams, and organizations—stakeholders—whose needs, ideas, suggestions, decisions, plans, and progress must be communicated. Some project managers claim they devote 80 percent of their time to communicating; in fact, that number might be low since it’s hard to imagine aspects of project management that do not involve communication. Part of project definition and planning is anticipating and preparing for all that communication.

Communication plan

The methods and measures to be used for all communication throughout the project should be thought out before the project begins and, ideally, laid out in a **project communication plan**. For larger projects, the plan would be included as part of the execution plan. The plan might specify:

- Expected reports and documents, their content, frequency, and who is responsible for them.
- Expected meetings, updates, and reviews, their itineraries, advance preparations, time limits, attendance policy, and who will lead.
- A tentative schedule for formal reviews and milestone meetings showing the participants, purpose, and agendas.
- Important points of contact (who's-who among customers, contractors, and other stakeholders).

The table in Figure 4.8 represents part of a communication plan, showing the expected meetings and reports, and participants for each. Not shown, the plan might also specify other meetings such as for safety and kicking-off and closing-out project stages. The plan would be supplemented with details about the What, Where, When, and How for each kind of meeting and report. The plan also would acknowledge important external stakeholders, reports they will receive, and their preferred communication modes (this information might instead be



See Chapter 16 included in the **stakeholder engagement plan**, discussed in Chapter 16.)

The communication plan should be distributed to everyone on the project team and to all relevant stakeholders outside of the project team, and reviewed before the project begins. To make sure they understand the required documentation and the content and format of each, the plan should include examples of good and bad documentation from past projects. Much of this can be posted online. During the project, team members and external stakeholders will be in transition, so the plan must be continuously updated.

Role/type	Meeting/ report	Status meeting (frequency)	Status meeting minutes (frequency)	Business feasibility	Information request	Technical feasibility	Business brief	Project plan (frequency)	Problems and issues (frequency)	Business study	Use case analysis	System architecture	Detailed technical design	Other
Client		X	X	X			X	X	X	X	X			
Relationship manager			X	X	X	X	X	X	X	X	X	X		
Business analyst	X	X	X				X	X	X	X	X	X		
Project manager	X	X	X	X	X	X	X	X	X	X	X	X		X
Client project team	X	X	X				X	X	X	X	X			
IT project team	X	X						X	X	X	X	X		X
Client director			X	X			X		X	X				X
IT director			X	X	X	X	X	X	X	X	X	X		
Project sponsor			X	X			X			X				
IT VP			X	X						X				
Architect	X	X	X	X	X	X	X	X	X	X	X	X		X
Security/audit	X	X	X			X			X	X	X	X		
Internet operations	X	X	X			X		X	X	X	X	X		X
Intranet operations	X	X	X			X		X	X	X				
Legal/corporations communication			X	X		X			X	X				
Other														

Figure 4.8
Sample communication plan.

Details about communication for monitoring and controlling projects are discussed in Chapter 13, and for working with stakeholders and virtual teams in Chapters 16 and 17, respectively.



See Chapter 13,
16 and 17

Informal communication

Much communication in projects happens informally, person-to-person, and through the grapevine. Informal communication is not especially dependable, garbles messages, and doesn't guarantee that people will get the information they need, but nonetheless it is largely beneficial. Some theorists posit that a vast network of informal communication is essential for any organization to perform well since it fulfills social and work needs and tends to convey information more rapidly than formal methods.

Managers cannot control informal communication, but they can influence and take advantage of it. One way to bolster informality is by removing status barriers and inspiring casual conversations between managers and workers. Some companies insist that everyone—from the president on down—avoid titles, speak on a first-name basis, and maintain an “open door” policy. Offices are physically arranged to encourage informal communication by removing walls and partitions, putting chairs and desks in “team groupings,” and adjusting the placement of lounges to encourage informal chats. These practices are especially prevalent in companies that encourage innovation and practice agile project management.

4.7 Summary

There are good reasons why the project life cycle approach is used in so many kinds of projects. First, it emphasizes continuous planning, review, and authorization. At each stage results are examined and used as the basis for decisions and planning for later stages. Second, the process is goal oriented—it strives to maintain focus on user requirements and system objectives. Third, in each phase the risks are assessed and reduced: knowledge gaps are filled in; mistakes and problems are caught early and corrected before they get out of control; if the environment changes, timely action can be taken to modify the system or terminate the project. Finally, user requirements and system requirements are always in sight and activities are done so that they are coordinated and occur at the right time and in the right sequence. The project life cycle approach is an essential aspect of project management—regardless of differences in methodologies for its execution (waterfall, agile, or hybrid).

The front-end phases of a project—conceptualization and definition—are important to the viability and success of the project. What is surprising is that in many projects user and systems requirements definition receive relatively little attention. The impetus is to begin preparing the project plan—without even having clearly defined the end result of the project! Project definition and system definition go hand in hand; only in cases where there is much latitude in terms of what the customer wants, when she wants it, and how much she is willing to pay can a project succeed in the absence of good requirements. In the more usual case (the customer is more demanding, and the schedule and budget are constrained) success is predicated on a well-defined description of what the result must be and do—the user requirements and the system requirements.

Projects involve numerous meetings, reviews and documented reports for communication, planning, assessment, and control. To ensure nothing important is overlooked, a project communication plan is prepared early in the project specifying the kinds of reports

and meetings, and details about the contents, formats, schedules, participants, and points of contact for each.

At the conclusion of Phase B, companies sometimes employ a phase-gate, i.e., a formal review or checklist to ensure that the project is on track and ready to move on to Phase C, execution. Assuming all the requirements of Phase B have been met, the project is given the green light to proceed.

APPENDIX A: STAGES OF SYSTEMS ENGINEERING⁹



See Chapter 2

The systems engineering methodology described in Chapter 2 follows a series of stages that parallels that of the project life cycle and systems development cycle. A misnomer, really, systems engineering is not “engineering” in the same context as other engineering disciplines. Rather, as described earlier, it is a logical *process* employed in the evolution of a system from the point when a need is first identified, through the system’s planning, design, construction, and to ultimate deployment and operation by a user. The process, outlined in Figure 4.9, has two parts, one associated with the system’s *development and production* (Stages 1 through 4), the other with the system’s *utilization* (Stage 5).

Stage 1. Needs Identification and Conceptual Design¹⁰

This stage is analogous to Phase A of the project life cycle where the main tasks are to define stakeholder needs and requirements, do a feasibility analysis, and perform high-level requirements analysis, system-level synthesis, and system design review. The result of this stage is a **functional baseline design**—a list of high-level requirements and high-level functions for the intended end-item system.

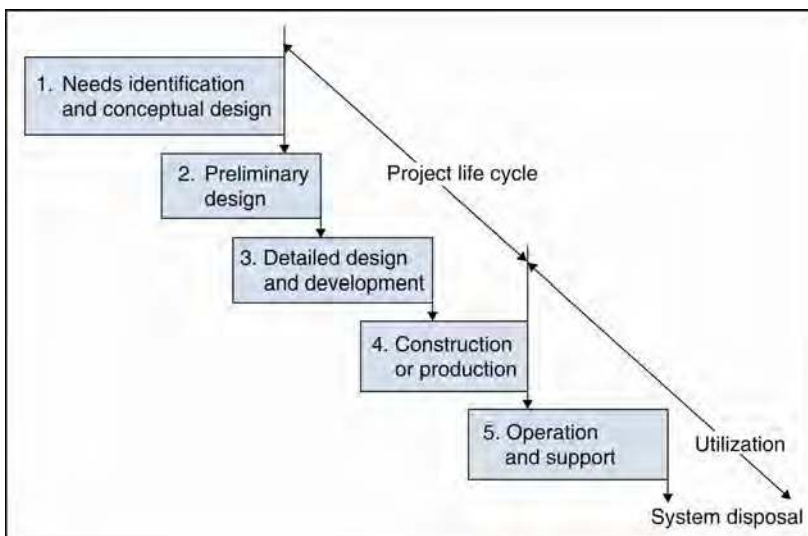


Figure 4.9
Stages of the systems engineering process.

Stakeholder and needs identification

Systems engineering deals with poorly defined problems. The customer may feel that something is wrong or may need something new but be unclear about the problem or need, or what the system should look like or do. Sometimes it is not clear who has the problem or need. The first step is to identify the stakeholders who will be affected by or able to impact the system. Identifying the “customer” is not trivial: it might be an organization, however, within the organization only certain parties have the authority to make decisions regarding the system, or will use, operate, or be impacted by it. These parties must be singled out and their needs identified.

Developing a clear concept of the need or problem begins by asking basic questions:¹¹

1. How did the problem or need arise?
2. Who believes it is a problem or feels the need?
3. Is this the root problem or need, or is it a manifestation of a deeper problem?
4. Why is a solution important? How much money (or time, etc.) will it save?
5. How important is the need? Would resources be better applied to another need?

Answers to these questions lead to a preliminary description of a system that addresses the need or problem, including its expected performance, cost, and schedule. The customer reviews the description and perhaps redefines the need, in which case the contractor must redefine the system description. The process continues until parties agree on the need definition and proposed system.

Requirements definition

High-level requirements specify everything important about the system—its objectives, life cycle, operational modes, constraints, and interfaces with other systems. As discussed earlier, they should also address all the stakeholders—producers, suppliers, operators, and others who will ultimately use and benefit from, manage, maintain, and otherwise impact or be impacted by the system—and reflect their interests and perspectives. These include, for example, corporate customers who are interested in the system’s market, capacity, and operating and capital costs; operators who are interested in its performance, durability, reliability, and parts availability; and users who care about its comfort, safety, and usability.

In systems engineering practice, the initial requirements are stated in the language of the stakeholders and compiled in a list called the **stakeholder (or user) requirements document (SRD)**. Anyone reading the SRD should be able to readily understand the mission and application of the intended system. The project should not be started until the primary stakeholders have reviewed and endorsed the SRD.

Example A4.1: SRD for the Spaceship¹²

As an example, let us revisit the X-Prize/SpaceShipOne project described in Chapter 1. The X-Prize competition’s criteria were to send a reusable vehicle that could carry three people into space twice within 2 weeks. Besides winning the X-Prize, the goal of developer Burt Rutan and customer Sir Richard Branson was to develop technology that would enable low-cost space tourism. Among the constraints were a relatively small budget and a small development company with limited resources. Hence, the SRD would include developing a spaceship with the following requirements:

1. Can attain minimally 100 km altitude.
2. Carries three people.
3. Provides comfortable flight.
4. Is relatively inexpensive to design, build, and launch.
5. Can be turned around in 2 weeks or less.
6. Is inherently safe to operate.

Feasibility

The next step is to identify high-level (system-level) alternative ways to meet the needs, objectives, constraints, and requirements. The alternatives are evaluated in terms of costs, risks, effectiveness, and benefits using studies and models; the most feasible solutions are recommended to customers and supporters.

Requirements analysis

With approval of the project and system-level alternatives, the next step is to specify what the system must do (the functions it must perform) to be able to meet the requirements in the SRD. For example, the requirement that the spaceship “provide comfortable flight” implies system requirements that the spaceship’s cabin temperature, humidity, and pressure all function at “comfortable” levels throughout the flight. This implies that the spaceship will be equipped to perform the necessary functions to make this happen. Whereas the SRD specifies the system in terms of stakeholder wants or needs, the system requirements tell the designers the *functions* the system must perform and the physical characteristics it must possess to meet the SRD. This process of defining requirements, called **requirements analysis**, results in a document called the **system specification**, described later. Requirements analysis for physical systems addresses three kinds of requirements: *functional*, *performance*, and *verification*.

Functional requirements

Functional requirements specify the functions that the system (its hardware, software, and related services, etc.) must perform to meet all the requirements in the SRD, including those for system support, operation, and maintenance. A popular tool for analyzing and defining functional requirements is the **functional flow block diagram, FFBD**, illustrated in Figure 4.10. Each block represents a function that the system must perform to satisfy requirements. As illustrated, each function is defined in greater detail by decomposing it into sub-functions; for example, as shown function 3 is logically comprised of five sub-functions, 3.1 through 3.5. In the *conceptual design* stage, the decomposition of functions into smaller, better-defined sub-functions proceeds only to the next level (e.g. subdivides function 3 into 3.1–3.5). Later, in the *preliminary design* stage, the decomposition will resume and continue to whatever level necessary to arrive at the best possible requirements definition. In the figure, this is shown by decomposing function 3.5 into functions 3.5.1–3.5.4.

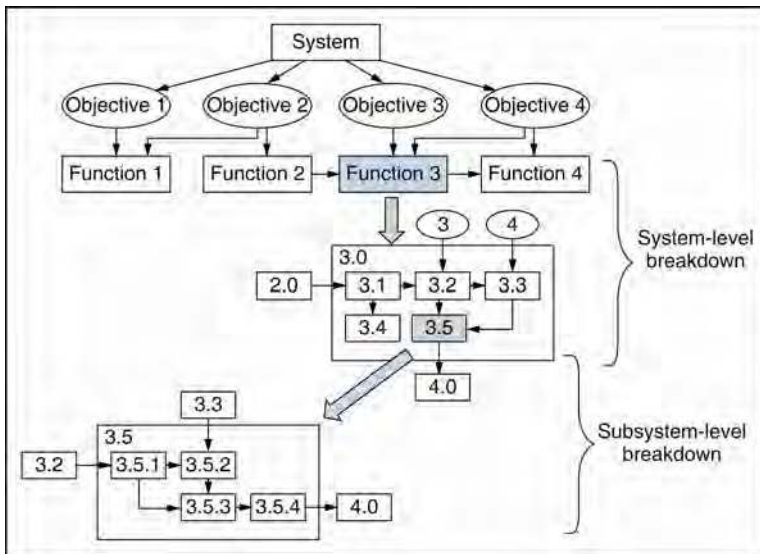


Figure 4.10

FFBD for decomposing system-level functions into lower-level functions.

Notice the numbering scheme used in Figure 4.10: each function has a unique identifier that enables it to be traced to the original system-level function, e.g. function 3.5 contributes to function 3.5, which contributes to function 3. This **traceability** of functions is essential because throughout the system life cycle numerous changes will be made to components and functions, and for each change it is necessary to know the impact on higher-level and lower-level functions. This helps prevent mistakes that could lead to later problems. For example, the cryogenic tanks in the Apollo 13 spacecraft were originally designed to operate at 28 volts. Later, the spacecraft design required that certain controls be changed to 65 volts. This involved changes to numerous components, including to the cryogenic tanks, but somehow the linkage between tanks and controls was missed, and the changes were never made. During the mission this mistake caused a tank to explode, which ruined the mission and nearly cost the lives of the three Apollo astronauts.

Example A4.2: Functional Requirements Breakdown for the Spaceship

Figure 4.11 shows a portion of the FFBD for the spaceship, and decomposition of the system-level functions that address stakeholder requirements 3 and 5. The other system-level functions would be decomposed as well.

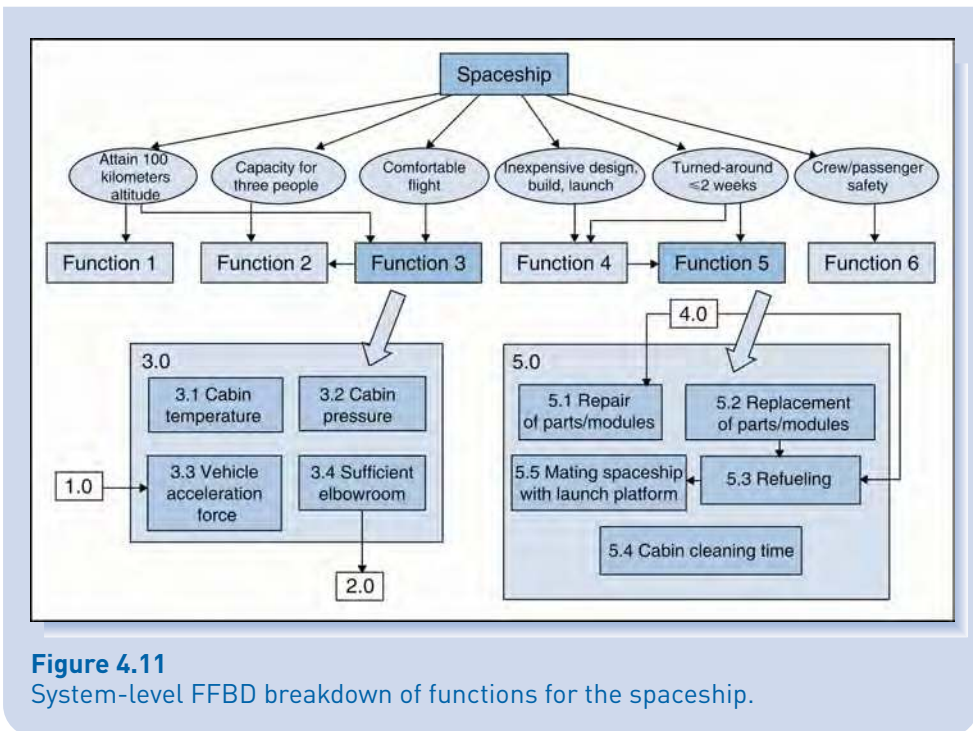


Figure 4.11

System-level FFBD breakdown of functions for the spaceship.

Performance and verification requirements

Associated with each functional requirement are performance requirements and verification requirements. Whereas a functional requirement states *what* the system must do, a **performance requirement** states *how well* it must do it. Performance requirements are usually specified in physical parameters such as speed, acceleration, weight, accuracy, power, force, or time. They are the targets on which designers set their sights. For example, the stakeholder requirement “provide comfortable flight” has many functional requirements, including some regarding cabin temperature and pressure. The associated performance requirements for these might be:

1. Cabin temperature: 75–85 degrees F.
2. Cabin pressure: 4.2–3.2 psi.

Accompanying each performance requirement is a set of **verification requirements**; these specify procedures and tests to verify that the performance requirement has been met. For example, verification requirements would specify the kinds of tests to prove that cabin temperature and pressure will remain at the required performance levels during all phases of spaceflight.

Synthesis

Up until now the systems engineering process has been focused on *top-down analysis*, resulting in a big list of functional, performance, and verification requirements. The next step, *synthesis*,

looks at relationships among the system-level requirements and alternative ways of satisfying the requirements. One question is, can these requirements be satisfied using existing, **off-the-shelf (OTS)** designs and products, or will it be necessary to employ new and innovative designs or technologies? An OTS item is one that can be readily purchased or built; if it meets the requirements, an OTS item is often preferable to a newly designed one because it is readily available and usually costs less. Sometimes there is no OTS item, and to create a new design that meets the requirements would be overly costly, risky, or time-consuming; in such cases the requirements must be revised.

The result of synthesis is called the **system specification**, which is a comprehensive list of all the functions the new system must satisfy plus a firm or tentative solution (to be developed or bought) for each function. The system specification serves as a guide for designers in the later stages of preliminary and detailed system design.

Example A4.3: System Specification for Spaceship Motor

A decision must be made about the kind of rocket motor the spaceship will have. Among the functional requirements for the motor are:

- 1.1 Must provide thrust of x.
- 4.1 Cost of fuel and fuel handling must be economical.
- 5.3 Refueling procedure must be simple.
- 6.1 The fuel and the fuel system must be inherently safe.

A check of existing OTS rocket motors used to launch satellites shows that none fit the requirements; all are too costly to fuel and operate and are somewhat dangerous. Hence, a new rocket motor must be developed—one that will be safe, simple, and inexpensive to fuel and operate, and provide the necessary thrust. Experiments reveal a promising solution: a motor that uses ordinary rubber as the fuel and nitrous oxide (laughing gas) as the oxidant; both materials are stable, safe, inexpensive, and easy to handle. The decision is made to adopt the technology and design and build a completely new motor. Thus, one system specification for the spaceship (of many hundreds) is that the rocket motor burns nitrous oxide and rubber.

The system specification is reviewed and checked against the functional requirements at a formal meeting. When approved, it becomes the **functional baseline** or template for all subsequent design work.

Stage 2. Preliminary Design¹³

The purpose of the **preliminary design** stage is to translate system-level functional requirements into design requirements for the subsystems. This stage roughly corresponds with project life cycle Phase B. Studies are performed of the high-level elements comprising the system, and the system-level requirements are *allocated* among the subsystems.

Grouping of functions: architecture and components

The next step is to group the identified functions and requirements according to the physical **architecture** of the system. In general, the term “architecture” refers to the major components in a system and how they are configured or arranged to satisfy the functions of the system; for example, the architecture most people have in mind for a bicycle is:

- *Major components:* two wheels, frame, seat, pedals and chain, handlebar.
- *Configuration:* wheels attached at ends of frame; front wheel pivots on frame; seat mounted on frame between wheels; pedals attached to chain; chain linked to rear wheel; etc.

Another case is Example A4.5.

Example A4.5: Architecture of the Spaceship

The spaceship will have airplane features of a fuselage and wings, but also spacecraft features of a rocket motor and ability to maneuver in space. Unlike an airplane where the cabin and fuselage walls are the same, the cabin in a spaceship is a separate “pressure vessel” fitted inside the fuselage. The spaceship architecture will include the following subsystems:

- Fuselage: structure containing or attached to other subsystems (cabin pressure vessel, hydraulics, avionics, motor, fuel system, wings, etc.).
- Cabin pressure vessel contains seats, storage space, instruments and flight controls, and environmental control system.
- Rocket motor: main propulsion system, fuel system, motor controls.
- Avionics: aviation electronics; computers, subsystems for communication, navigation, flight controls, auxiliary power system, etc.
- Wing/aerodynamic surfaces: main wings, tail, and hydraulic/electronic actuators.
- Landing gear: gear doors, braces, skids or tires, brakes.

Sometimes the architecture “looks right,” sometimes not. Often, to satisfy unique requirements, designers are forced to stray from the commonplace architecture, the result being a “funny-looking” architecture. Each major component will perform a major function or set of system-level functions as listed in the functional baseline.

A large system will have numerous components, all of which must be documented and monitored throughout the system’s life cycle of design, production, and usage. This documenting and tracking, referred to as **configuration management**, is necessary to ensure that any changes in the design, production, or usage of the components does not alter or degrade the end-item system’s ability to meet the functional requirements. Configuration management utilizes **traceability** to prevent snafus such as the voltage change that caused the Apollo 13 accident mentioned earlier. It pertains to not only major subsystems, but to any items identified as risky, costly, or critical to performance. Every major subsystem or component that will be documented and tracked throughout the system’s life cycle is referred to as a **configuration item** or **CI**.

Requirements allocation among the components

As of this point, the design consists of (1) a list of the functional requirements and (2) a high-level design of the system—the major subsystem or components (the CIs). The next step is to “allocate” the functional requirements to the CIs, which means to assign responsibility for each functional requirement to one or more of the CIs. The purpose here is to ensure that every functional requirement will be addressed (and hopefully satisfied) by at least one of the subsystems or CIs. The resulting allocations are shown in an **allocation matrix** or **traceability matrix**. Shown in Figure 4.13, the matrix columns are the subsystems responsible for meeting the requirements; the rows are the requirements that the subsystems must fulfill.

With this allocation the transition from functions to physical items accelerates. Since each of the CIs represents something that will ultimately be a physical item—a piece of hardware, software, or both, assigning functional requirements to CIs represents a transition in thinking from *what must be done* (e.g., travel 100 km above the Earth) to *how it will be done* (with a spacecraft that has a fuselage, cabin, wings, and engine, configured in a certain way).

Notice in Figure 4.13 that responsibility for some requirements is shared by multiple CIs. For example, the weight of the system (requirement 1.5) is shared by all the CIs. That is, the spacecraft weight is the sum of the weights of all the CIs; if the weight of any one is changed,

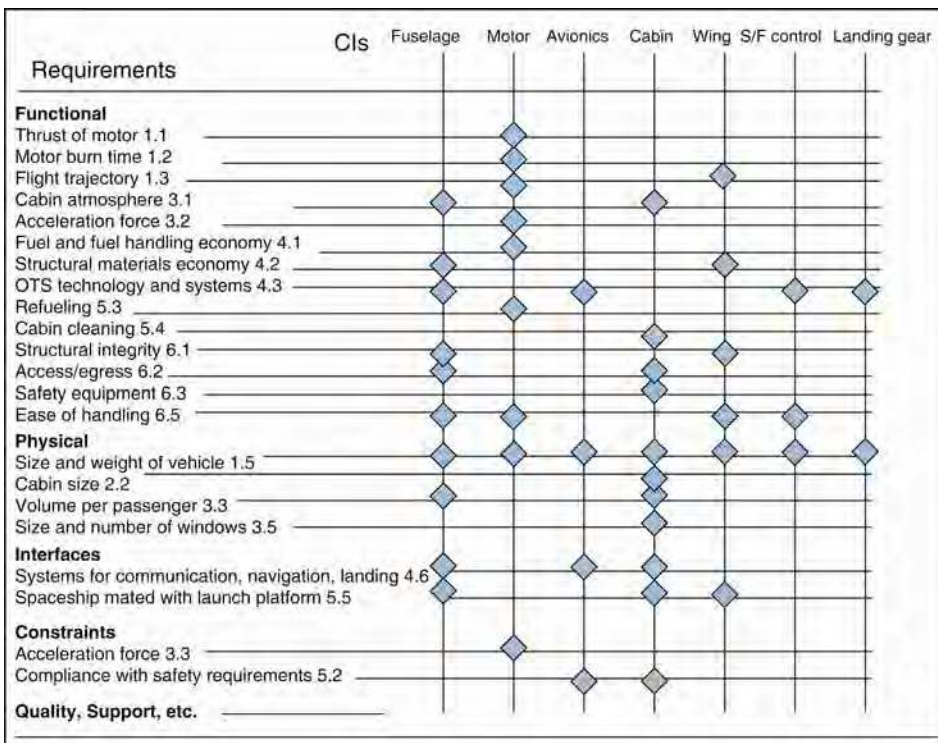


Figure 4.13

Allocation (traceability) matrix.

Source: Adapted from Falconbridge R. and Ryan M. *Managing Complex Technical Projects*. Boston: Artech, 2003; 78.

so is the weight of the spacecraft. If the maximum loaded weight of the spacecraft is set at 3,600 kg (about 7,937 lb), the CIs must be designed so that all of them combined will not exceed that weight.

Example A4.6: Allocation of Weight among CIs

Question: How do you design and develop all the CIs so that the total weight (shared requirement) does not exceed 3,600 kg? Answer: Estimate the percentage of the total spaceship weight that each configuration item (CI) should account for and set that as the “target” design weight for the CI. For example, allocate, say, 30 percent of the total system weight to the fuselage and contents, 20 percent to the motor, 20 percent to the wings, 10 percent to avionics, and 10 percent for everything else. Hence, the fuselage target weight would be $0.30 \times 3,600 \text{ kg} = 1,080 \text{ kg}$, the motor target weight is $0.20 \times 3,600 \text{ kg} = 720 \text{ kg}$, and so on. Since achieving targets is critical, each is designated as a **technical performance measure**, or **TePM**, which means that as the CIs are being designed, their estimated and actual weights are carefully compared to the targets. If during the project it becomes clear that a target cannot be achieved (as will surely happen), then the allocations are readjusted. If, say, the weight of the motor cannot be held to its target but must be increased by 30 kg, then either the allotted weights for other subsystems must correspondingly be reduced, or the target spaceship weight increased by 30 kg. Throughout the design process, it will be necessary to adjust the CI targets and allocations guided by the TePM process. This process is described in Chapter 13.



See Chapter 13

Interfaces

None of the subsystems function independently; they all rely on the outputs of other functions and, in turn, provide inputs to still others; in a word, they **interface**. Part of the preliminary design process is to identify all interfaces in the system and establish requirements for the interfaces. A main source of information about interfaces is FFBDs. For example, the FFBD in Figure 4.12 shows that function 5.5 *receives* input from functions 5.3, 5.4 and 4.6.6 and *provides* input to functions 8.6.3 and 9.3. Each arrow represents an interface—a connection or “flow” between functions. The connection can be:

- Physical—mechanical connections, physical joints and supports, pipes.
- Electronic—analog or digital signals.
- Electrical—electric energy.
- Hydraulic/pneumatic—liquid or gas.
- Software—data.
- Environment—temperature, pressure, humidity, radiation, magnetism.
- Procedural—completion of a procedural step so another next step can begin.

Identifying the interfaces is necessary for setting requirements on the inputs and outputs of every subsystem and element. For example, since the spacecraft’s fuselage contains the motor and supports the wings, neither wings nor motor can be designed without considering the fuselage’s design, and vice versa. The requirements for each interface (e.g. allowable

maximum or minimum flow or physical strength) are set by a design team that includes representatives from subsystems or functions at both sides of the interface.

Synthesis and evaluation

Designing each of the CIs and its subsystems and elements involves choosing among design alternatives and, again, deciding whether to use an OTS design or product, or to develop a new design from scratch.

The selection of alternatives in the preliminary design stage must consider the synthesis of components—the impacts of each design decision on other components and the overall system. Following is an example.

Example A4.7: Tradeoffs in Designing Cabin Size

The weight requirement for a spacecraft is important because the greater the weight, the more thrust (power) required of the rocket motor to propel the vehicle into space and the greater the load-carrying capacity of the mothership to carry it aloft. At some point early in the conceptual design the maximum weight will be set, and thereafter every

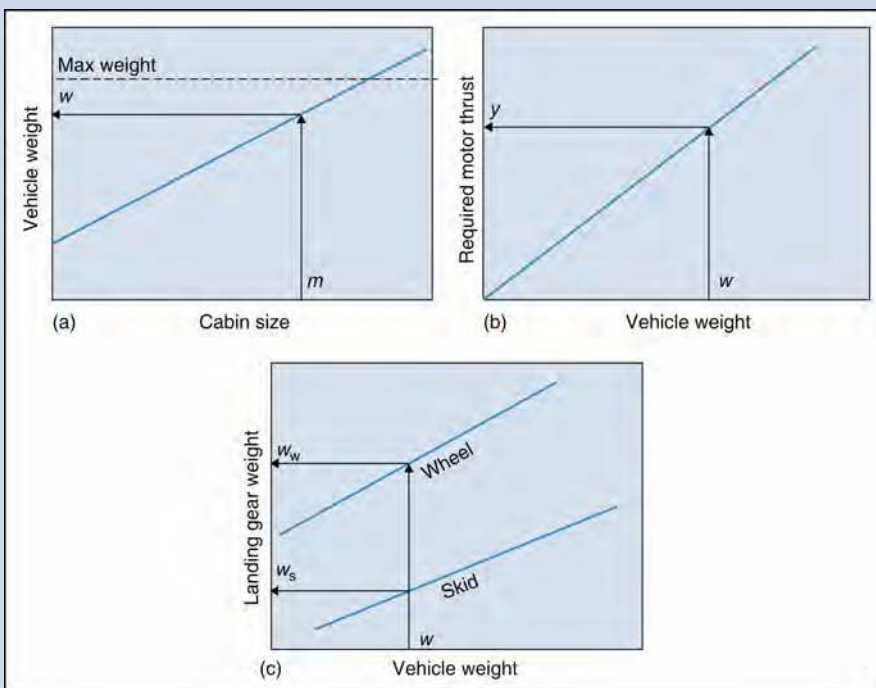


Figure 4.14

Relation between cabin size, vehicle weight, rocket thrust, and landing gear weight.

effort will be made to find ways to reduce it. Consider below some tradeoff decisions that designers face.

How big should the cabin be? In general, the cabin should be roomy enough to hold three people, instruments and controls, and stowage; a bigger cabin would be more comfortable for the occupants but would also weigh more. Suppose a cabin of volume m is chosen, which will result in an estimated weight of w for the spaceship. Also suppose that to propel a vehicle of weight w into space will require a rocket motor with thrust of y (Figure 4.14, top diagrams). Note that if the cabin size is increased, then the thrust of the rocket motor must also be increased — unless weight somewhere else in the spaceship can be reduced.

Now consider the impact of vehicle weight on another decision: landing gear. The more the vehicle weighs, the stronger the required gear; but all else being equal, the stronger the gear, the heavier the gear. If the weight of a typical wheeled landing gear, strong enough to support the vehicle, is deemed too high, then an alternative must be considered, such as a skid (Figure 4.14, bottom). The skid has no wheels and weighs less than a wheeled gear. If the skid meets other functional requirements, then it would be chosen over a wheeled landing gear.

Such tradeoff decisions will be necessary for all the CIs and other components. As decisions are made, a design evolves that meets the requirements. The form and configuration of the CIs is set, and the system's physical appearance takes shape. By the end of the preliminary design stage the system architecture will have been established, and all system-level requirements allocated among the major subsystems (CIs). Combined, the architecture and allocated requirements form the “allocated baseline” design (see example, Figure 4.15).

Stage 3. Detailed Design and System Development

The detailed design stage involves further description of subsystems, assemblies, components, and parts of the main system and support items. It roughly corresponds to tasks performed in Phase B and early Phase C in the project life cycle.

Everything up to this point has been analytical in nature. With the detailed design, the development process moves from “concepts on paper or computer”—the SRD, system specifications, FFBs—to a design that is *ready to build*. Decisions are made about whether subsystems and components will function manually or automatically; whether components will be electronic, mechanical, or hydraulic; whether inputs–outputs will be manual, mechanical, electronic, and so on. Available OTS components are selected based on surveys or laboratory tests, and newly developed components are tested experimentally using models that enable designs to be verified by trial and error. Models or mockups for components (**breadboards**) are used to develop pieces of equipment that will subsequently be mated and integrated into the overall system. A **prototype** (nearly complete system) is assembled for the purposes of developmental testing and evaluating the overall system in terms of satisfying requirements. Different type models and mockups for testing are described in Chapter 10.

System development and design testing and evaluation includes:¹⁴

1. Checking the operation of subsystems when combined in the complete system.
2. Evaluating the validity of design assumptions.



See Chapter 10

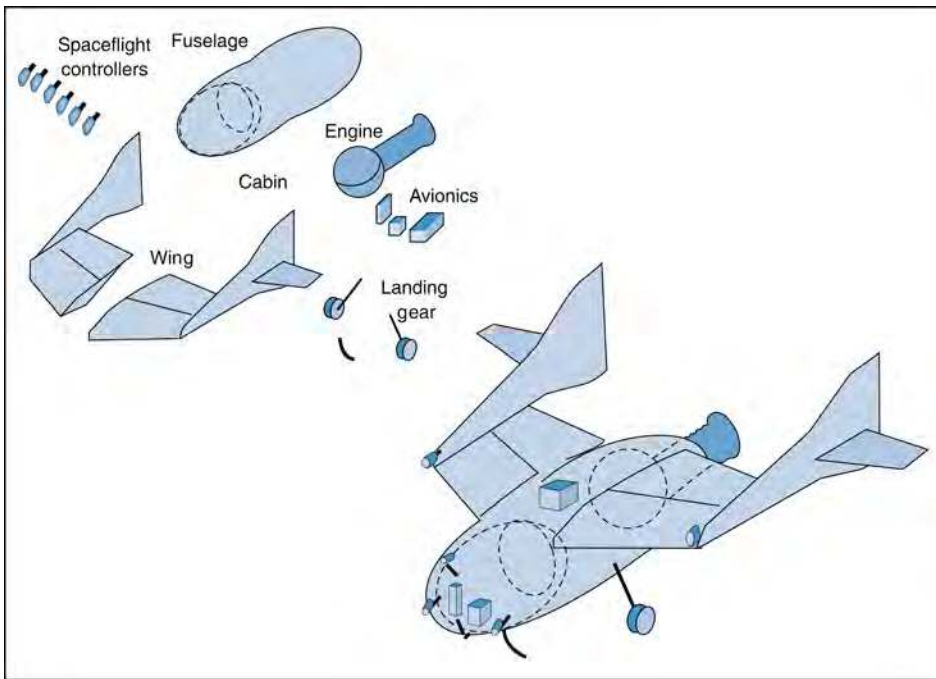


Figure 4.15

Pictorial representation of major subsystems (CIs) and allocated baseline design. (The “funny looking” architecture derives from the spaceship having to meet many requirements. On re-entry, the wings rotate upward, making the spacecraft one big airbrake that floats to earth like a shuttlecock, thus avoiding high speed and high temperature. Nearer to the ground the wings tilt back and airplane-like the ship glides to a landing.)

3. Paying close attention to the interfaces:
 - a. feedback and “cross talk” among subsystems
 - b. adjustments and calibrations
 - c. serviceability and maintenance.

The system is checked under a variety of conditions and operational modes. Problems previously overlooked often become known during these tests. Designs are modified to eliminate deficiencies and improve the system.

Example A4.8: Testing SpaceShipOne

Numerous ground and flight tests of SpaceShipOne resulted in changes, for example:

- In one test flight the spacecraft began to pitch wildly and only with great difficulty was the pilot able to regain control. Engineers diagnosed the cause as being a too-small

tail, which they quickly redesigned. (Problem was, the small company did not have a wind tunnel in which to test it. Undeterred, they mounted the tail assembly on a Ford pickup truck and checked it by racing up and down the runway.)

- The nose skid showed excessive wear after tests and had to be replaced by one with a stronger material.

When there is not enough time or money to build test mockups, then the first few manufactured models are subjected to testing and design evaluation. Gradually, as modifications are made and the design is approved, full-scale production begins. Design and development testing is phased out and replaced with quality control to ensure the end-item system as produced will conform to design specifications.

At this time, the methods, resources, and capability to produce the system (process design) are also addressed. This involves the design of new (or redesign of old) facilities and manufacturing processes, selection of specific materials and pieces of equipment, and preparation for production control, quality testing, manufacturing tooling, product transportation, personnel hiring and training, and data collection and processing.

Stage 4. System Fabrication, Construction and/or Production

Analogous to the latter part of Phase C, in Stage 4 the system is (1) mass produced, (2) produced in limited quantities with different features, or (3) built as a single item. This stage begins as soon as the design is approved and “frozen.” The stage involves acquiring materials and controlling production/construction to uphold performance, quality, reliability, safety, and other requirements.

Stage 5. System Operation and Support

Stage 5 completes the systems engineering process. Analogous to Phase D, the customer maintains and operates the system until it wears out or becomes obsolete. The system developer might continue to support the system in several ways: assisting in deploying, installing, and checking out the system; assisting in day-to-day operation or field service and maintenance support; modifying or enhancing the system to ensure continued satisfaction; or providing support in closing out and disposing of the system at the end of its life cycle. The last way, system closeout and disposal, should be considered during the system’s early design to minimize its degradation of the surrounding environment. The design of nuclear reactors and mines for metals and coal, for example, must account for the way each will be shut down. Their closeout must include measures to restore the land, clean up waste, and remove toxins from soil and water, which can be expensive, time-consuming, and take years or decades.

Example A4.9: Product Launch of SpaceShipOne

Development of SS1 and its support systems took almost four years, after which Burt Rutan announced his intention to capture the X-Prize and begin flight-testing (Figure 4.16).

In May 2004, Mike Melville piloted SS1 on a test beyond 100 km, making him the world's first civilian astronaut. On October 29 he again flew the craft into space, and less than 2 weeks later so did pilot Brian Binney, winning the \$10 million X-Prize for the SS1 team (Figure 4.17). Today SS1 hangs on display at the Smithsonian Air & Space Museum in Washington, DC. A bigger spaceship, SS2, and a bigger mothership, WK2, have since been developed for use by Sir Richard Branson's commercial "spaceline," Virgin Galactic, which hopes to operate a fleet of them soon. (See <https://www.virgingalactic.com/>.)

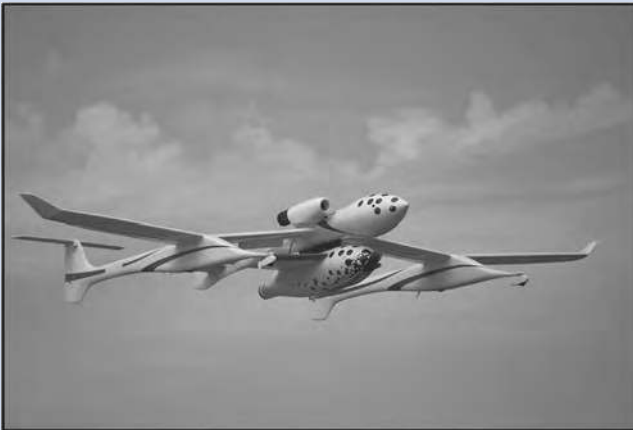


Figure 4.16
SS1 beneath
mothership White
Knight.

Photo courtesy John
Nicholas.



Figure 4.17
Designer Burt Rutan
(center) and pilots
Mike Melville (left)
and Brian Binney.

Photo courtesy John
Nicholas.

APPENDIX B: QUALITY FUNCTION DEPLOYMENT¹⁵

Quality function deployment, QFD, is a methodology for defining requirements and, specifically, for translating customer needs into system or product characteristics, and specifying the processes and tasks needed to produce the system or product. As demonstrated in numerous applications QFD not only yields end-item results that meet customer needs, but it does so in less time and at a lower cost than traditional development methodologies. QFD was developed by Mitsubishi's Kobe Shipyards in 1972, adopted by Toyota in 1978, and has since been implemented by companies worldwide.

House of Quality¹⁶

QFD mandates that the project team articulates the means by which the product or system being designed will achieve customer requirements. The process starts with customer requirements or market needs and then uses a planning matrix called the *house of quality* to translate the requirement and needs into technical requirements. The structure of the house is shown in Figure 4.18.

- The left side of the matrix lists “what” the customer needs or requires.
- The top of the matrix lists the design attributes or technical requirements of the product; these are “how” the product can meet customer requirements.
- Additional sections on the top, right, and bottom sides show correlations among the requirements, comparisons to competitors, technical assessments, and target values.

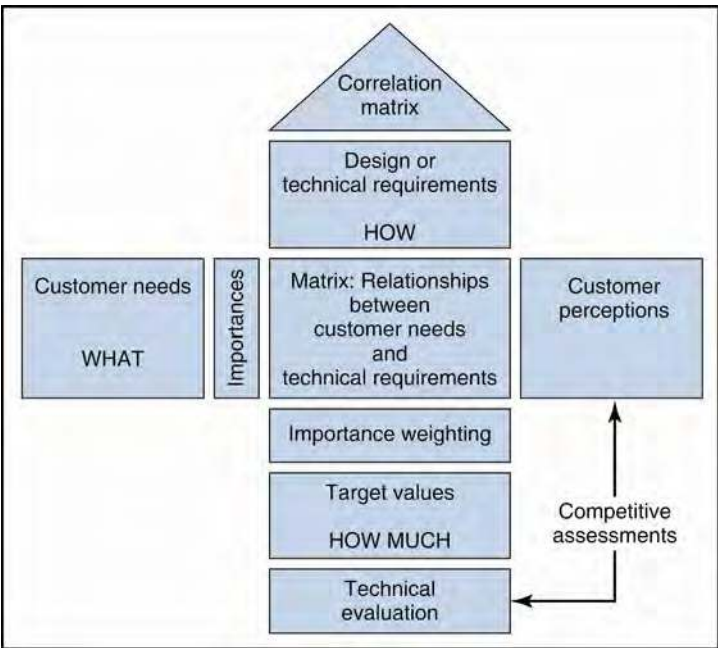


Figure 4.18
Structure of the
house of quality.

Features of the house of quality are illustrated below in Example A4.10.

Example A4.10: House of Quality for a TV Remote-Control Switch

Figure 4.19 is a portion of the house of quality matrix for the design of a television remote-control (RC) switch. The house is interpreted as follows:

- Rows (customer requirements): these show what customers think is important about the product. They are the product “whats.”
- Importance to customer: The requirements have been ranked 1–6 by customer preference; “multifunction buttons” is rated the highest; “RC easy to see/find” the lowest.
- Columns (technical requirements): These are the *requirements or attributes* of the product, the ways that the product meets customer requirements. They are the product “hows.”

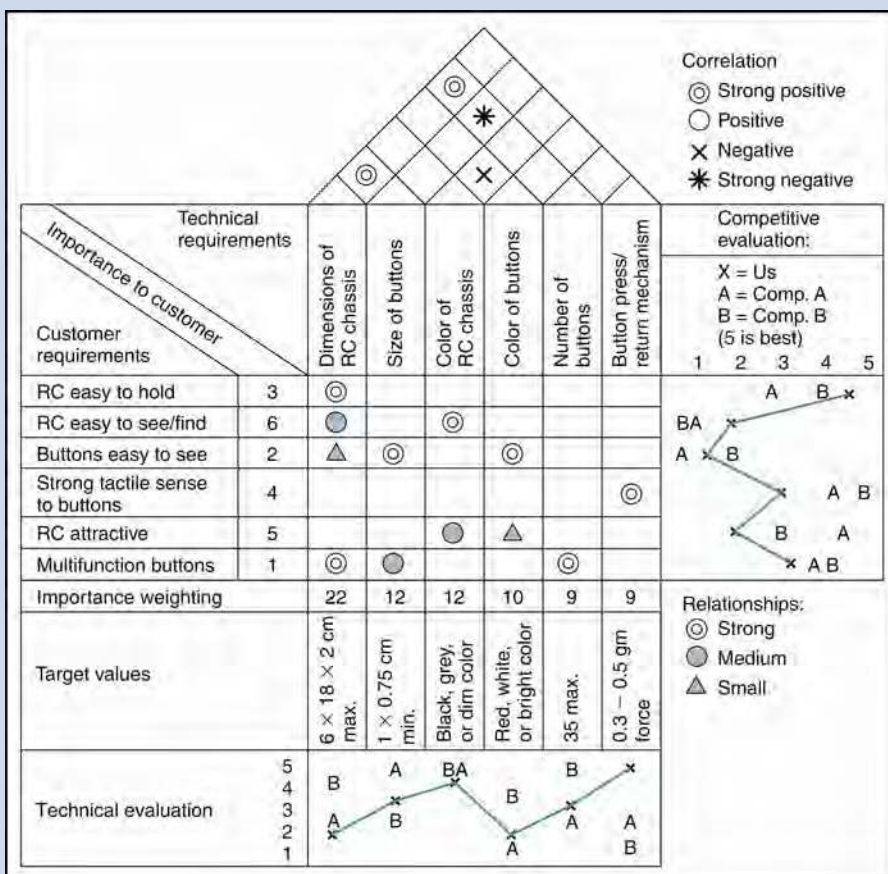


Figure 4.19
House of quality for television remote-control switch.

- Central matrix: Contains symbols that show the strength of the relationship between the what's and the how's (strong positive, positive, negative, strong negative). For example, "buttons easy to see" has a strong positive relationship to the size and color of the buttons, and a positive relationship to the size of the remote-control chassis. Note that each relationship has a numerical weighting (small = 1, medium = 3, strong = 9).
- Importance weighting: The weights of the symbols in each column are summed to determine the relative importance of the technical attributes. Thus, the most important technical attribute is "dimensions of the RC" (weight = $9 + 3 + 1 + 9 = 22$), followed by "size of buttons" and "color of RC chassis" ($9 + 3 = 12$ each).
- Gabled roof: This contains the correlations among the technical attributes. For example, "dimensions of the RC chassis" has a strong positive correlation with "size of buttons" and "number of buttons"; "size of buttons" has a strong negative correlation with "number of buttons" (smaller buttons allow more buttons; larger buttons allow fewer).
- Target values: The numerical or qualitative descriptions (in the "basement" of the house) are design targets set for the technical attributes. One target of the design, for example, is to keep the dimensions of the RC within "6 × 18 × 2 cm."
- Technical evaluation: The graph (in the "sub-basement") compares the company (x = "us") against two of its competitors, A and B, on the technical attributes. For example, the company's current product does relatively poorly on the attributes of RC dimensions and button color, but fares well on chassis color and return mechanism. These evaluations are based on test results and opinions of engineers.
- Competitive evaluation: The graph on the right rates the company and its competitors in terms of customer requirements. These ratings are based on customer surveys. For example, customers think the company does best in terms of the RC being "attractive," but worst in terms of it being "easy to hold."

The house of quality suggests areas in which designers might focus to gain a market niche. For example, the rating on the right in Figure 4.19 indicates that no company does particularly well in terms of "buttons easy to see," even though customers rank that requirement second in importance. A requirement that customers rank high, yet on which all companies rank low suggests a feature that could be exploited to improve a company's competitive standing. The company making the RC, for example, might try to improve the button visibility by increasing button size and/or using bright colors.

The house provides a systematic way of organizing, analyzing, and comparing the "hows" with the "whats," and prevents things from being overlooked. It justifies where to devote time and money, and where to refrain from adding resources. Still, the results of QFD are only as good as the data that goes into the house. At minimum, competitive evaluations require two perspectives: the customers' viewpoints regarding how the product compares to the competition, and the engineers' and technicians' views regarding how well the product meets technical requirements. The data comes from sources such as focus groups, tests of competitors' products, and published reports.

An important aspect of requirements definition is to determine priorities—to distinguish between the *critical few* and *trivial many* aspects of the end-item system so as to ensure that the critical ones are done correctly. As an example, a computer printer might have as many as 30 different design features that affect print quality, but the most important feature is the fusion process of melting toner on the page, which is a function of the right combination of temperature, pressure, and time. Focusing on temperature, pressure, and time narrows the design emphasis

to the relatively few most important technical parameters. These parameters become the ones for which designers seek the “optimum” values. Once these values have been set, the analysis moves on to identify key factors in the manufacturing process necessary to achieve those values.



Review Questions

1. When does the project manager become involved in the project?
2. What is the purpose of the kickoff meeting? When is the meeting held and who runs it?
3. How is the project team created?
4. Briefly describe the contents of a project execution plan.
5. Describe phased project planning.
6. What are user requirements, system requirements, and system specifications? Give examples. How are they related?
7. What are functional requirements? What are the performance requirements? Give examples.
8. What are “non-functional requirements”? Give examples.
9. Describe the process of developing user requirements and system specifications.
10. What problems are associated with requirements definition? What are ways to minimize these problems?
11. What is the purpose of specifying priorities and margins in defining requirements?
12. Describe concurrent engineering.
13. Describe the stages of systems engineering in Figure 4.9. Think of some projects and describe the stages of systems engineering in these projects.
14. Distinguish the following: functional requirements, performance requirements, and verification requirements. Give an example of a functional requirement and its associated performance and verification requirements.
15. What is meant by the term “traceability”?
16. Think of a simple system like a mousetrap, tape dispenser, or can opener. Draw a simple high-level functional flow block diagram for it. If possible, decompose each of the functions into sub-functions.
17. Briefly define the purpose of quality function deployment (QFD).
18. What is the source of customer needs or requirements that appear in the house of quality?
19. Think about the following or use whatever consumer research material available to you to define customer needs or requirements for the following:
 - a. a “good” college course
 - b. toaster (or other home appliance of your choosing)
 - c. cellular telephone
 - d. coffee mug for your car.

For each, define a corresponding set of physical or technical characteristics. Using the format of Figure 4.20 construct a house of quality matrix and show the relationship between the technical characteristics and customer requirements. Use the matrix in each case to “design” or suggest what the ideal product or service would be like or look like.

20. What is the purpose of the project charter? What is included in the charter?
21. To what situations does agile project management apply? How does agile differ from “waterfall”?
22. What is the purpose of the communication plan? What is the content of the plan?

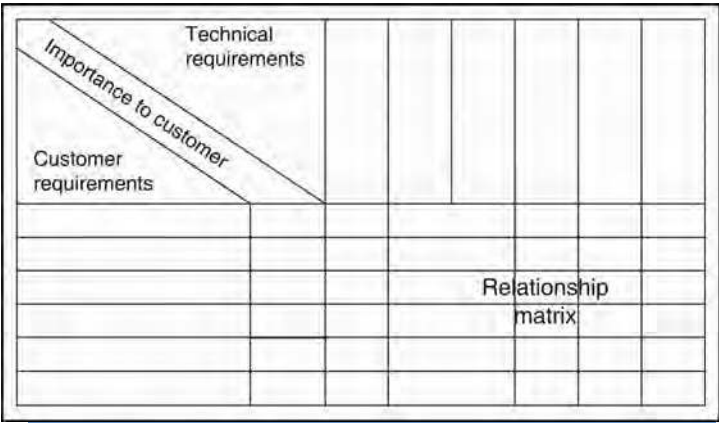


Figure 4.20
QFD matrix.



Questions About the Study Project

1. Did the project have a kickoff meeting? What happened there?
2. How did the project manager become involved in the project? Was she selected as project manager before or after the proposal was completed?
3. How was the project team formed?
4. Were there user requirements? How were they defined? Were they “well-defined” requirements?
5. Were there any system requirements? Were they clear and utilized by the project team?
6. Were there any system specifications and performance requirements? If not, how did the project team know what was required of the end-item?
7. Did the project have a project execution plan? If so, describe the contents. If not, how did the team know what they were supposed to do (tasks, schedules, responsibilities, etc.)?
8. Describe the process of creating the project plan.
9. Did different stakeholders participate in defining the requirements and creating the project plan?
10. Did the project manager encourage open, informal communication? If so, in what ways?
11. Was QFD or a similar process used to define requirements and/or create the project plan?
12. What is your overall impression about how well the definition phase was conducted in the project, and of the quality of the system requirements and project plan?

CASE 4.1 STAR-BOARD CONSTRUCTION AND SANTARO ASSOCIATES: REQUIREMENTS SNAFU

Star-Board Construction (SBC) was the prime contractor for a large skyscraper project in downtown Manhattan. SBC was working directly from drawings received from the architect, Santaro Associates (SA). Rob Santaro, owner and chief architect of SA considered this building similar to others he had designed, although one difference he failed to notice was the building’s facing, which was to consist of large granite slabs—slabs much larger than anything he or SBC had prior experience with.

Halfway into the project, Kent Star, project manager for SBC, started to receive reports from his site superintendent about recurring problems with window installation. The windows were pre-manufactured units made according to SA's specifications. The granite facing on the building was to be installed according to specifications that allow for dimensional variations in the window units. The architect provided the specification that the tolerance for each window space should be 1/2-inch (that is, the window space between granite slabs could vary as much as 1/4 inch larger or smaller than the specified value). This created a problem for the construction crew, which found the slabs too big to install with such precision. As a result, the spacing between slabs was often too small, making it impossible to install window units. Most of the 2,000 window units for the building were already manufactured so it was too late to change their specifications, and most of the granite slabs were hung on the building. The only recourse for fitting window units into tight spaces was to grind away the granite, which would be expensive and delay completion of the building.

QUESTION

What steps or actions should the architect and contractor have taken before committing to the specifications on the window units and spacing between granite slabs that would have prevented this problem?

CASE 4.2 REVCON PRODUCTS AND WELBAR, INC.: CLIENT-CONTRACTOR COMMUNICATION

Revcon Products manufactures valves for controlling the water level in industrial tanks. It had concentrated on products for the construction industry (valves for newly installed tanks), but now wanted to move into the much larger and more lucrative replacement market. Whereas annual demand for new valves is about 100,000, it is about 1 million for replacement valves. The company envisioned a new valve, the Millennium Valve, to gain a share in the tank-valve replacement market. Revcon's objective was to design and produce the Millennium Valve to be of superior quality and lower cost than the competition. It decided to outsource the development and design of the new valve, and prepared an RFP with these objectives and requirements:

Product objectives:

- Innovative design to distinguish the Millennium Valve from competitors' valves.
- Price-competitive but offer greater value.

Market (user) requirements:

- Ease of installation
- Non-clogging
- Quiet operation
- Ease in setting water level
- Adjustable height.

Revcon sent the RFP to four design companies and selected Welbar, Inc., primarily based on it being the lowest bidder. Welbar's proposal was written by its sales and marketing

departments and revised by senior management, but with no input from industrial designers, engineers, or anyone else who would be on the project. Welbar had no prior experience with industrial water valves, but its sales team saw Millennium as an opportunity to earn profits and align with a major equipment manufacturer. The marketing department prepared time and cost estimates using standard tasks and work packages from proposals for old projects.

The Welbar design team for the Millennium Valve project was headed by Karl Fitch, a seasoned engineer, and included two industrial designers and two engineers. His first task was to research the valve market and talk to contractors, plumbers, and retailers. Karl reviewed Welbar's proposal and concluded that it had omitted several critical steps, and that its cost was substantially underestimated.

Karl divided the project into small work packages and prepared a Gantt chart. During the project, the design concept, work tasks, and schedule had to be changed many times. Welbar engineers were frustrated at Revcon's constant harping that the valve be low-priced and have functional superiority, and that the project be speedy and low-cost. During the project, Welbar engineers learned that to design such a valve required more resources than had been budgeted. Because of all the changes, Welbar exceeded the budget and had to request additional funds from Revcon four times. A major problem occurred when Welbar delivered a working model to Revcon. Because the model description in the proposal was vague, Revcon expected the model to be an almost-finished product, whereas Welbar understood it to be a simple prototype to demonstrate functionality. Welbar had to spend extra time and money to bring the prototype up to Revcon's expectation. To compensate, Welbar crammed project stages together. When the design stage fell behind because of the prototype, Welbar went ahead and prepared production-ready models, but the prototype later demonstrated that the production models could not be produced.

Eventually Welbar was able to design an innovative valve, although it required substantial retooling of the factory and cost 50 percent more to produce than expected. In the end, Revcon spent twice as much time and money on development as expected and was not able to price the product low enough for it to be competitive.

QUESTION

What happened to this project? What are the factors that contributed to Revcon's failure to get the product it wanted? For each factor, discuss what might have been done differently.

CASE 4.3 LAVASOFT.COM: INTERPRETING CUSTOMER REQUIREMENTS

Lavasoft Company is developing new website software for a corporate client. The project starts when two Lavasoft staffers meet with the client to create a list of user needs and requirements, which they then turn over to the Lavasoft design team.

The project manager, Lakshmi Singh, feels that the system best suited to the user's needs is obvious, and to address them she creates bullet points and flowcharts. She then presents these to the design team and asks if anyone has questions. Some people are concerned that the approach as stated by the bullets and charts is too vague, but Lakshmi assures them that the vagueness will subside as details of the system are defined.

To reduce outside interference, the team works in relative isolation of other development teams in the company. Daily the team is forced to interpret the bullet points and

high-level charts and to make design decisions. Whenever there is disagreement about interpretation, Lakshmi makes the decision. The team creates a list of detailed system specifications, and the project is considered on schedule. Upon working on the specifications, however, issues arise concerning the system's compatibility with the client's existing site. Further, some of the specifications call for technical expertise that the team lacks. Also, in a review of the original user needs, the team discovers that some specifications are unrelated to the needs, and that for some needs there are no specifications.

The team drops some specifications and adds new ones. This requires eliminating some of the existing code, writing new code, and retesting the system, which puts the project behind schedule. Resistance grows to changing the specifications further since that would require even more recoding and put the project further behind. Lakshmi adds people to get the project back on schedule. Eventually the system is ready for installation, although it is 2 months late. Because additional people were added to the project team, Lavasoft does not make a profit. Because the specifications were incorrect, the system is not fully compatible with the client's website and Lavasoft must continue to work on it and introduce "fixes."

QUESTIONS

1. What went wrong with the project?
2. Where were mistakes made in the project initially?
3. How were problems allowed to persist and go uncorrected for so long?

CASE 4.4 PROPOSED GOLD MINE IN CANADA: PHASED PROJECT PLANNING

July 12: Peter's firm acquires the rights to an ore body in the Canadian Shield region. The firm is considering developing a new mine there, and Peter is responsible for proposing a project plan to the board in September. The mine will take a few years to reach full production, and there is much uncertainty as to the price of gold when that happens. Peter includes in his proposal a history of the gold price (Figure 4.21).

August 2: Peter meets with Bruce, a mining engineer with 20 years of experience in Australian gold mines, and Sam, a geologist who a few years earlier did exploratory work on gold deposits in the Canadian Shield region. They discuss known facts about the ore body, the likelihood of unforeseen geological phenomena that could jeopardize mine development, production figures that might be achieved, and production costs and technical problems that might arise in extracting gold from the ore. A quick calculation shows that 300,000 ounces of gold per year at \$700 per ounce would be very lucrative, but 150,000 ounces at \$400 per ounce, 3 years from now, would not and could ruin the company. Current information about the ore body is inadequate, so it will be necessary to drill exploration holes to learn more about the geology of the area.

Peter summarizes: "To the best of our knowledge, we could produce anywhere between 150,000 and 300,000 ounces a year. The capital cost for developing the shaft will be C \$150 million to \$260 million, and annual operational costs could be \$60 million to \$100 million. Exploration to provide information on the ore body would require drilling 200 exploration holes at a cost of \$1.2 million to \$1.6 million. Rock samples from these holes will be analyzed in a laboratory to determine the gold content."

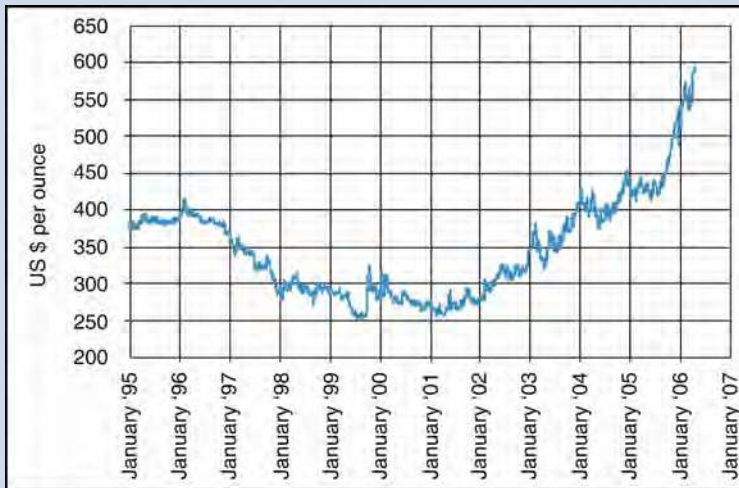


Figure 4.21
Gold price.

Peter instructs Sam to review the data from his previous exploration work and to prepare a report of his recommendations concerning future exploration. He is authorized to spend up to \$25,000 to do this. They agree that, should the exploration holes yield good results, a “demonstration shaft” will be sunk to haul out a sample of 30,000 tons of ore to be processed to extract gold. Results from this demo would increase confidence about the amount of gold present, reduce uncertainty about processing the ore, and provide a good indication of potential yields. They estimate that the demo shaft and analysis would cost \$18 million to \$25 million, some of which, however, could be deducted from the cost of the full-fledged mine—should it go ahead. Only if these results are positive—and the gold price is relatively high and stable at that stage—would the full-fledged mine be authorized.

QUESTIONS

1. List the phases of the project and indicate the minimum and maximum cost of each phase as foreseen in August 2024.
2. “While estimates for the distant future are very ‘broad brush,’ it is always possible to make relatively accurate estimates for the imminent phase of a project.” Explain.
3. Describe how each of the proposed project phases will help reduce the project’s risk.
4. Comment on the problem that, once money has been allocated to the process, people might become “hooked” into the project and tempted to go ahead regardless of high risks.
5. How would you determine the value of accurate estimates for the number of ounces that could be mined and for costs?
6. Would you trust any internal rate of return or net present value estimates created now?

REFER TO CASE 2.3, SANTA CLARA COUNTY TRAFFIC OPERATIONS SYSTEM AND SIGNAL COORDINATION PROJECT

QUESTION

The INCOSE Transportation Working Group determined that a requirements traceability matrix, which was not used in the project, would have—were it used—aided in technology-related decision-making during construction and reduced the number of change requests. Based on the limited information provided in the case, discuss the applicability of the traceability matrix to this project.

For a case related to communication, see Chapter 12, Case 12.3, FORMAL AND INFORMAL AGREEMENTS IN CONTRACTUAL MATTERS



See
Chapter 12

Notes

1. For advice for naming projects see Gause D. and Weinberg G. *Exploring Requirements: Quality Before Design*. New York, NY: Dorset House; 1989, pp. 128–134.
2. Association for Project Management (APM) *APM Body of Knowledge*, 7th edn. Buckinghamshire, UK: Association for Project Management; 2018.
3. Kay R. *An APMP Primer*. Lul.Com Self Publishing; 2010.
4. This section is adapted from Merrow E. *Industrial Megaprojects*. Hoboken, NJ: John Wiley; 2011.
5. See Frame J.D. *Managing Projects in Organizations*. San Francisco, CA: Jossey-Bass; 1988, pp. 146–151.
6. Hajek V. *Management of Engineering Projects*, 3rd edn. New York, NY: McGraw-Hill; 1984, pp. 35–37; Whitten N. *Managing Software Development Projects*, 2nd edn. New York, NY: John Wiley & Sons; 1995, pp. 250–255.
7. Connell J. and Shafer L. *Structured Rapid Prototyping*. Upper Saddle River, NJ: Yourdon Press/Prentice Hall; 1989.
8. Portions adapted from Sabbagh K. *Twenty-First Century Jet: The Making and Marketing of the Boeing 777*. New York, NY: Scribner; 1996.
9. Sources: (1) Falconbridge R. and Ryan M. *Managing Complex Technical Projects: A Systems Engineering Approach*. Boston, MA: Artech House; 2003, pp. 9–93; (2) Blanchard B. and Fabrycky W. *Systems Engineering and Analysis*. Upper Saddle River, NJ: Prentice Hall; 1981, pp. 18–52; (3) Kossiakoff A., Biemer S., Seymour S., and Flanagan D. *Systems Engineering: Principles and Practice*, 3rd edn. New York: Wiley; 2020; (4) INCOSE (Ed) *INCOSE Systems Engineering Handbook*, 5th edn. New York: Wiley; 2023; (5) Chestnut H. *Systems Engineering Methods*. New York, NY: John Wiley & Sons; 1967, pp. 1–41; (6) Jenkins G. *The Systems Approach*. In Beishon J. and Peters G. (eds), *Systems Behavior*, 2nd edn. London, UK: Harper & Row; 1976, pp. 78–101.
10. Falconbridge and Ryan, *Managing Complex Technical Projects*, pp. 29–65.
11. Jenkins, *The Systems Approach*, p. 88.
12. The SpaceShipOne (SS1) examples in this book illustrate concepts. While there is much factual information about the project available from published sources, information about the actual design and development of the spaceship is confidential. SS1, the X-Prize, and the stakeholders described are all true-life, however for lack of information portions of this and subsequent examples are hypothetical. Information for this and other examples of SS1 are drawn from news articles and the SS1 website at Scaled Composites, <https://scaled.com/2024/06/21/twenty-years-spaceshipone/>.
13. Adapted from Falconbridge and Ryan, *Managing Complex Technical Projects*, pp. 67–96.
14. Chestnut, *Systems Engineering Methods*, p. 33.
15. Sources for this section: Bounds G., Yorks L., Adams M. and Ranney G. *Beyond Total Quality Management*. New York, NY: McGraw-Hill; 1994, pp. 275–282; Hauser J. and Clausing D. The house of quality. *Harvard Business Review* May–June; 1988: 63–73.
16. Portions of this section adopted from Nicholas J. *Competitive Manufacturing Management*. Burr Ridge, IL: Irwin/McGraw-Hill; 1998, pp. 428–434.

Chapter 5

Project Execution and Closeout

So much effort goes into the definition phase—preparing the project plan and defining the end-item, all of it in preparation for what is to come next: the execution phase.

Project execution is what most people—project outsiders—witness or read about: it is a building being constructed, a new vaccine being developed, a rocket engine being test fired, or employees and facilities being moved to a new company headquarters. To them, the word “project” means doing *the physical work* of creating an end-item or end result. Of course, project insiders also recognize the many things that had to happen first—*before* execution, the necessary pre-work in the conception and definition phases, without which there would be no project execution and, for that matter, no project!

Execution is generally the longest and most expensive phase of a project (for a megaproject, it typically costs over a billion dollars and lasts several years). For many projects, the start of execution is the tipping point—the point of no return.

What happens during execution and the stages and details of work depends entirely on the end-item—skyscraper, space vehicle, pharmaceutical drug, software application, etc. This chapter provides an overview of what happens in *many* projects during execution, especially projects aimed at delivering physical end-item systems. It describes the most common stages and steps. With a little imagination you might see how these stages and steps occur in other kinds of projects as well—for example in research programs, corporate audits, motion-picture productions, or large-scale litigations. The point is, most projects course through execution in a pattern similar to that described in this chapter.

5.1 Phase C: Execution

The **execution phase** typically includes the **stages** of detail design, production/build, and implementation, though, as mentioned, the stages will differ depending on the project. The typical stages in a hardware development project are design, development, and production; in construction projects they are design and build; and in consulting projects they are background research and investigation, report compilation, and presentation. Many companies utilize customized project methodologies with their own unique project phases and stages; these are discussed in Chapter 18. All projects that produce a physical end-item—a product, structure,



See Chapter 18

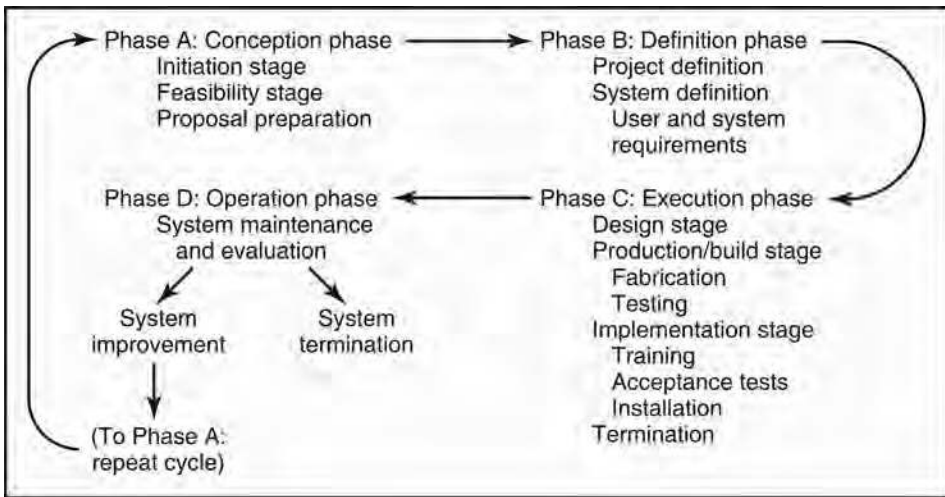


Figure 5.1

Phases and stages of systems development life cycle. Phases A, B, C = project lifecycle.

or system—also include an implementation stage wherein the end-item is handed over to the user. This chapter looks at the stages of detail design, production/build, implementation, and project closeout and termination (Figure 5.1).

Given its importance and the significant resources devoted to it, execution often initiates with a formal **kickoff meeting** and, sometimes, a project **charter** document that authorizes upcoming work and expenditures; both are discussed in Chapter 4.

The stages of execution typically occur in the sequence described in this chapter, and in traditional TPM projects, the bulk of each stage occurs just once, as implied in Figure 5.1. In agile and hybrid projects, however, the stages occur repeatedly in iterations, as described in



See Chapter 4



See Chapter 14 Chapter 14.

5.2 Detail Design Stage

In the **detail design stage**, system specifications are converted into plans, graphics, or drawings. The outputs of this stage are pictorial forms—blueprints, flow charts, schematic diagrams—or models showing the end-item system's components, dimensions, relationships, and configuration.¹

During this stage the system is divided into tiers of subsystems, components, and parts. Various design possibilities for elements at each tier are reviewed for compatibility with each other and with elements at higher-level tiers, and ability to meet system specifications and cost, schedule, and performance requirements. The breakdown into tiers and components uses tools mentioned elsewhere (block diagramming, Chapter 2; requirements breakdown structure, Chapter 3; and work breakdown structure, Chapter 6).

The design process is composed of two interrelated activities. First is preparation of a **functional** (or logical) **design** that specifies the functions of system and its components, the relationships of the components, and flows of information, electricity, fluids, etc.—whatever moves through the system. The purpose of this design activity is to ensure that the system's



See
Chapters 2,3,
and 6

components individually and collectively provide the functionality necessary to satisfy the system's performance requirements. The functional design is largely *conceptual* as represented by flowcharts and schematic diagrams. It is the thrust of systems engineering (Chapter 2) and system definition/FEL-3 (Chapter 4).

Second is preparation of a **physical design** that represents what the actual system and its components will *look like* and *be made of*—their sizes, shapes, relative positioning, and materials. This design activity results in engineering, manufacturing, architectural, and other types of drawings and models that reveal details necessary to later fabricate, assemble, and maintain the system. The physical design sometimes reveals limitations in the functional design because of space, cost, materials, appearance, or other considerations, in which case the functional design must be redone.

Design often follows an evolutionary, trial-and-error process as illustrated in Figure 2.7 in Chapter 2 and Figure 4.7 in Chapter 4. A trial design is prepared, modeled, and tested against system requirements. If it fails, the design is modified and retested. This design–build–test iteration happens in virtually all projects for developing new systems.

When the end-item system is complex, the iteration occurs in many places for elements and subsystems throughout the system, and necessary changes in one will have a ripple effect on the others. For instance, one subsystem might have to be enlarged, which robs space from another subsystem, which then must be moved somewhere else, and so on. Uncontrolled, the result is an almost never-ending series of redesign iterations, as illustrated in Example 5.1.



See Chapters 2
and 4



See Chapters 2
and 4

Example 5.1: Design Complexity in the Channel Tunnel²

One of the mandated requirements for the English Channel Tunnel project was that trains running through it must be resistant to fire damage for at least 30 minutes; this would enable every train car to be capable of making it out of the tunnel with a fire raging in it. But the frame of a normal train car would deform from the heat and the train soon would become immobile, so special metal alloys would have to be used. This however would make the trains heavier, 2,400 tons instead of 1,600 tons, and would require heavier locomotives needing six axles instead of four. The locomotives would have to be specially designed, and because they needed more power, the tunnel's power system would have to be increased, too.

Design and production/build do not always occur as discrete, sequential stages, but sometimes overlap. In other words, building a part of the system commences as soon as *some* of the design is completed, then building another part begins as soon as more of the design is completed, and so on. The system is built while it is being designed—a practice referred to as **fast-tracking**. Fast-tracking is common in the construction industry: the foundation is being dug and steel raised even though the roof and interior are still being designed. The practice speeds up work and can save up to a year on a major construction project, but it can be risky. Design problems often surface only after the details have been worked out, but by then portions of the building will have been fabricated and might have to be rebuilt—adding to time and costs. The usual sequential or “slow tracking” method takes longer but allows time to discover and resolve design problems before construction begins.

Also, as mentioned, the design sometimes results in a “tentative” or incomplete end-item design that is built and tested, and, sometimes, installed and operated. This process is

repeated, each time incorporating what was learned from the previous iteration—a trial and error, learn as you go approach whereby the design evolves to satisfy the requirements.

Interaction design³

Why is it that so many software-based products are difficult to use and contain obscure or irrelevant features that most people don't need or want? Examples are software products for home computers, mobile phones, and apps and entertainment systems—all which contain numerous features and functions that most people do not need and never learn to use. Yet, in an effort to continuously “improve” the product, developers keep adding more features, a process that leads to “bloatware”—software with excessive and unneeded functionality. Compare, for instance, all the things you could do with your word-processing software or mobile phone with the few features you actually use. Not only do such products contain too many features, but they inter-mix seldom-used features with often-used ones, making the product more difficult to understand and use. In the eyes of customers, they are too complex.

Complex systems have always existed, but in the past, they were operated by trained personnel. Farm and construction equipment, aircraft, trains, and electrical turbine generators are complex, but they are used by trained personnel, not the average person. Commercial products (software, electric vehicle, mobile phone) are complex too, but they are used by amateurs, not skilled operators.

Complexity and bloatware happen when product goals and user requirements are poorly defined, no one ensures the design meets user requirements, and user–system interaction is not a key design issue. They also happen when design is *controlled* by engineers and programmers, people who are technically astute but tend to be ignorant of **interaction design**, which is how users interact with the product's functions. Whenever programmers or marketing managers want to add another product feature, they are contributing to bloatware and ignoring its impact on the average end-user.

The project manager and systems engineer must retain control over the design process, especially the interaction design. This starts with knowing the end-users—their wants, aptitudes, and skill levels, reflecting these while defining user requirements, and thereafter considering how every decision that influences the function and operation of the product will affect the end-user.

Controlling design



See Chapters 10 and 13



See Chapter 13

Project reviews, discussed in Chapters 10 and 13, are scheduled to occur at key milestones. Ideally, they are headed by experts to ensure the functional design satisfies requirements, and the final design meets the users' needs and budget.

Throughout the design stage, changes to earlier designs might be necessary due to new technology, technical obstacles, or shifting requirements. Since these inevitably require alterations to planned work, the project manager must monitor the project for changes; determine how the changes impact work plans, schedules, and budgets; and relay these impacts to stakeholders. All this is handled through a **change control system**, as described in Chapter 13.

Design changes tend to increase project costs, but as shown in Figure 5.2, design costs are typically but a small fraction of production costs. Consequently, prolonging this stage to get the design right (the first time) tends to be far less costly than changing the design or fixing design-related problems that crop up later during production. But the design stage cannot be

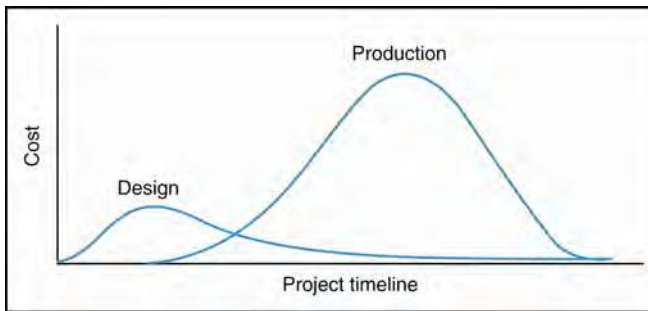


Figure 5.2
Relative costs for design
and production.

allowed to continue indefinitely, and sometimes the project manager imposes a **freeze date** after which no non-essential design changes are allowed.

Project management is responsible for ensuring that design efforts and outcomes are adequately documented. Everyone in the upcoming production/build and later stages should be able to access and understand the end-item design, its features, functions, configuration, strengths, and limitations, all as documented in specifications, drawings, models, etc.

Planning for production/build and later stages

Throughout the design stage the project manager is looking ahead and planning for the next stage: production/build. This planning addresses all aspects of production/build—the needed tools, equipment, and materials; procedures for assembly, testing, packaging, and so on—and includes preparation of a detailed production/build schedule.

5.3 Production/Build Stage

With detailed designs in hand, the project team is now ready for the **production/build stage**. For mass-produced items, this means the system is ready to produce or manufacture; for one-of-a-kind items, it means the system is ready to build or assemble. The main activities in this stage are system fabrication, testing, and planning for implementation.

System fabrication and testing

System fabrication, which is another term for assembly, building, or construction, begins as soon as design work has been sufficiently completed. Components prepared by contractors and suppliers are assembled into the final end-item. As in earlier stages, the project manager monitors the work, coordinates efforts among departments/subcontractors, and tracks progress against project plans, budgets, and schedules. During this stage the project manager shares responsibility with the manager of manufacturing, construction, or fabrication in managing tasks such as releasing work orders; monitoring, inspecting, and documenting progress; comparing planned versus actual results; and taking corrective action.

During system fabrication, work quality is constantly assessed. Quality control is not, per se, the responsibility of the project manager, although the quality of the final system is. Because of that the project manager must ensure that other managers have implemented the



See Chapter 10

quality plan (discussed in Chapter 10), which ensures the project's quality objectives will be met.

Throughout the production/build stage, numerous **tests** might have to be performed on the components, subsystems, and the final end-item to ensure everything conforms to requirements (aspects of testing are discussed in Chapter 10). The project manager oversees preparation of test plans which are included in the production/build plan and ensure that the necessary tests resources will be available, that tests will be performed as scheduled or necessary, and test results will be documented and filed for later reference.

Planning for implementation



See Chapter 4

With phased project planning, discussed in Chapter 4, details of the project plan are filled in as the project progresses and more information becomes available. During each project stage, a detailed plan for the next stage is prepared. Although preparing the implementation plan should start very early in the project, every effort is made to complete the plan so implementation can begin as soon as the production/build stage finishes.

Implementation is the process of turning the system over to the user. Two main activities in implementation are **installing** the end-item system in the user's environment and **training** the user to operate the system. The implementation plan must ensure that the necessary **side items** (discussed later) will be available in time for both sets of activities.

In so-called "brownfield" projects where a new system is to replace an existing system, the plan must address the *strategy* to be used for orchestrating the replacement, including the:

1. Approach for converting from the old system to the new system.
2. Approach to phasing out the old system and reassigning personnel.
3. Sequencing and scheduling of implementation activities.
4. Acceptance criteria for the new system.

An initial implementation plan might have been developed as part of the initial project execution plan. If such a plan exists, then a more detailed implementation plan is prepared with the participation of customers and other stakeholders. As the plan is being prepared, the project team accumulates materials for preparing the user to operate and maintain the system; these include manuals for system operation, repair, testing, and servicing; training materials and simulators; manuals for training the trainers; and schematic drawings, special tools, and equipment for servicing and support—among the so-called side items mentioned previously.

Agreement must be reached with the customer about how and when the project can be closed out—that is, how and when the customer will consider the system acceptable and the project completed. Misunderstandings about this, such as "acceptance only after modification," can cause a project to drag on indefinitely; to prevent this, user requirements defined early in the project should include conditions or criteria for customer acceptance of the system. This will be discussed next.

5.4 Implementation Stage

In the **implementation stage**, the end-item system or other deliverable is turned over to the user for operation. Sometimes implementation happens in an instant; sometimes it takes much longer. Take a clock. If it is a simple clock on your mobile device, you might learn how to use it in a minute. If it is a nuclear clock such as the one used by the US Naval Observatory,

you might need to attend several days of training. If it is a clock to replace an existing clock connected to a timing device that controls lighting in a large skyscraper, you might need to develop a strategy for swapping the new/old clocks to minimize inconveniencing people in the building. There can be many issues associated with implementation, starting with user training and acceptance testing.

User training

User training informs the user how to operate, maintain, and service the system. At one extreme, training is a simple instruction pamphlet; at the other, it is an extensive, ongoing program with a hefty annual budget. User training starts with determining the training requirements—the type and extent of training required. This will dictate the kinds of training materials needed (manuals, videos, simulators); personnel to be trained (existing or newly hired personnel); training techniques to be used (classroom, online, independent study); training schedule (everyone at once, in phases, or ongoing); and staffing (contractor, user, or subcontracted training personnel). Users should review and approve all training procedures and documents before training begins and provide feedback afterward to improve the training. Training often begins with the contractor's trainers training the user's trainers, who then take over the training.

User training should address the issue of how the new system will fit into the user's environment. It should provide an overview of the system's objectives, scope, and operation, and how the system interfaces with the user organization. This will enable users to understand the new system within their environment and integrate it with existing systems. Anything new can create fear, stress, and anxiety, and one aim of training should be to reduce or eliminate these emotions regarding the newly implemented system.

User acceptance testing

Before or during installation the end-item is subjected to a series of **user acceptance tests**. The results of these tests determine whether the system can be adopted as is, needs modifications or adjustments, or should be rejected.

While similarities exist, user acceptance tests differ from tests conducted by the contractor project team during earlier project stages. Although those tests are intended to validate the system design (discussed in Chapter 10), they should be rigorous enough to exceed the user acceptance test criteria. Nonetheless, the contractor should be prepared to make modifications should the results of user tests call for them.  See Chapter 10

Ideally users perform acceptance tests with minimal assistance from the project team. In cases where they cannot, the project team must act as surrogate users and make every effort to test the system just as the user would, which means taking on the role of someone who, often, is devoid of system-related technical expertise. Lack of user participation in these tests can lead to later problems, so the project team should insist that, minimally, the user is on hand to witness the tests.

System installation and conversion

System installation and **conversion** is part of implementation and conducted according to the implementation plan. At this time, equipment is installed, tested, fine-tuned, and deemed operable to the fulfillment of requirements.

Virtually all new systems are, in a sense, designed to substitute existing systems, so picking the strategy for replacing the old system with the new is of major importance. This process is called **conversion** and three possible conversion strategies, illustrated in Figure 5.3, are:

- **Parallel installation:** both new and old systems are operated in parallel until the new system is sufficiently proven.
- **Pilot operation:** the new system is operated in a limited capacity until proven, and then is fully phased in as the old system is phased out.
- **Cold turkey (Big Bang—immediate replacement):** in one fell swoop, the new system is moved in, and the old one is moved out.

Selecting a conversion strategy is no simple matter; it involves considerations of costs, risks, and logistics. The first strategy, parallel installation, seems safest: if the new system fails, the old one is still there. But it is also the most expensive because two complete systems must be simultaneously operated and fully staffed. With the second strategy, pilot operation, the costs and risks are low, and staff can be trained in stages. But pilot operation might not be representative of full system operation, and only after the new system has been completely phased in (and the old one phased out) will problems become apparent. The last strategy is the fastest and potentially least costly, but it is also the riskiest and raises issues about when to train staff to operate the new system (since they are busy operating the old) and what happens if the new system fails.

Prior to installation, the project manager updates all plans and schedules, gains approvals for revisions, and renews commitments with the customer. Implementation is a high-stress

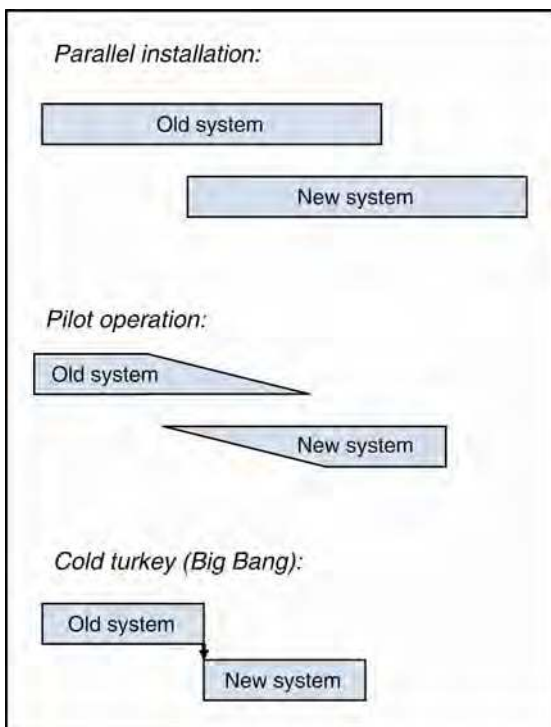


Figure 5.3
Three system conversion
strategies.

stage, and the project manager and team must be patient with users and sensitive to their questions, concerns, and fears.

Once the new system has been installed, the contractor continues to monitor it and perform tests to ensure it was installed properly, operates as expected, and interfaces smoothly with other systems in the user environment.

Side items

The installation, operation, maintenance, and monitoring of the contract end-item is often contingent upon availability of necessary ancillary articles called **side items**. These items include special tools, instruments, spare parts, reports, drawings, courses of instruction, and manuals. Side items are no less important than the main end-item, for without them it would be impossible to produce, operate, or maintain the end-item. Side items are usually provided by subcontractors and can range from the simple and mundane to the complex and novel. The former is exemplified by an online operating manual for a network server, the latter by a high-fidelity computer simulator for training operators of a nuclear power plant. Simple or complex, successful completion of side items is necessary for successful completion of the project.

Side items are deliverable contract items, and their cost may contribute to a significant percentage of total project cost. Yet perhaps because they are deemed “side” items, the time and effort required to develop and produce them is often underestimated, resulting in a delay in implementing the end-item and closing the project.

Although side items are often developed and produced by other companies, the project manager is responsible for ensuring that they have all been identified, contractors are developing and producing them, and they will be ready for use when the main end-item has been completed. The project manager must be certain that the scope of side-item work is well understood and qualified personnel or contractors are assigned to meet their requirements.⁴ Side items should be considered as part of the necessary contracted work, not extraneous or superfluous, and included in the WBS, project schedule, and budget.

5.5 Project Termination and Closeout

By the time the end-item has been installed, project team members will be eager to move on to new work. Managers eagerly shift emphasis to upcoming projects and might give little attention to termination and closeout. **Termination** means, simply, ending the project; **closeout** means ending the project and ensuring there are no outstanding issues, work, claims, and so on. As common sense would indicate, the way a project is concluded should receive no less attention than any other project activity. In fact, the method of termination can ultimately determine the project’s success or failure.

Termination can occur in a variety of ways, the best way being in a planned, systematic manner. The worst ways are abrupt cancellation, slow attrition of effort, or siphoning off resources by higher priority projects. Some projects are allowed to simply “limp along” until they fizzle out. Getting rid of such “white elephants” is discussed in Chapter 19.  See Chapter 19 Unless formally terminated, a project can drag on indefinitely, sometimes from neglect or insufficient resources, sometimes intentionally for lack of follow up work. In the latter case, workers remain on the project payroll even after work has been completed. Until a project is officially terminated, work orders remain open and labor charges continue to accrue.

Reasons for termination

The seeds of project success are sown early: since success depends in large part on customer acceptance of the project results, the project manager must make sure during project definition that the acceptance criteria are clearly defined, agreed upon, and documented, and any changes made during design or production/build are approved by the contractor and customer.

There are many reasons why a project is not successfully completed. A project might be aborted when the financial or other losses from termination are considered less than the losses expected from completing the project. The project might be perceived as a “white elephant” with low payoff or likelihood of success. The customer might simply change his mind and no longer want the project end-item.

Projects are also terminated because of changes in markets or technology, unsatisfactory technical performance, poor quality of materials or workmanship, violation of contract, or customer dissatisfaction with the contractor. Many of these factors are the contractor’s fault and could have been avoided had project management exercised better planning and control, shown more respect for the customer, or acted more ethically. Both sides suffer: the user requirements are left unmet, and the contractor’s technical competency and managerial ability are cast in doubt.

Termination and closeout responsibilities



See Chapter 12

Like all other project work, the project manager is responsible for planning, scheduling, and controlling termination and closeout activities. Much of this activity falls under the heading of **contract administration**, discussed more fully in Chapter 12.

Archibald lists the following responsibilities:⁵

- A. Planning, scheduling, and monitoring closeout activities:
 - Obtain and approve termination plans from involved functional managers.
 - Prepare and coordinate termination plans and schedules.
 - Plan for transfer of project team members and resources to other projects.
 - Monitor completion of all contractual agreements.
 - Monitor the disposition of any surplus materials and project equipment.
- B. Final closeout activities:
 - Close out all work orders and contracts with subcontractors for completed work.
 - Notify all departments of project completion.
 - Close the project office and all facilities occupied by the project organization.
 - Close project books.
 - Ensure delivery of project files and records to the responsible managers.
- C. Customer acceptance, obligations, and payment activities:
 - Ensure delivery of end-items, side items, and customer acceptance of items.
 - Notify the customer when all contractual obligations have been fulfilled.
 - Ensure all documentation related to customer acceptance as required by contract has been completed.
 - Expedite any customer activities needed to complete the project.
 - Transmit formal payment and collection of payments.
 - Obtain from customer formal acknowledgment of completion of contractual obligations that release the contractor from further obligation (except warranties and guarantees).

Responsibility for group C above, particularly for payment and contractual obligations, is shared with the project contract administrator, who is responsible for company–client negotiations and contracts. The last activity, obtaining formal customer acknowledgment, may involve claims by the contractor when the customer failed to provide agreed-to data or support, or requested items beyond contract specifications. Such cases entitle the contractor to compensation.

Before the project is considered closed, the customer reviews the results or end-item with the contractor to make sure everything is satisfactory. Items still open and in need of attention, and to which the contractor agrees, are recorded on a list, sometimes called a **punch list**. The contractor checks off the items on the list as they are rectified.

Example 5.2: Punch List for the Channel Tunnel⁶

Five months before the scheduled completion date of the Channel Tunnel, the punch list still contained over 22,000 items. Incredibly, by one day before the scheduled handover of the tunnel to its owner/operator, that number had been whittled down to only a hundred. The problem was, the contract allowed for *no* (zero) items on the punch list, and any open items would void the agreement and stop payment. A simple solution would be to delay the handover until the remaining items were fixed, which was estimated to take only a week. But few things associated with the Channel Tunnel were simple. Invitations for the handoff ceremony had already gone out, and preparations for the big gala celebration had been completed. Besides, a syndicate of some 200 banks located around the world had financed the project, and any proposed delay in the handover would require their approval.

What followed was a series of frenzied, harried negotiations via telephone and fax that lasted throughout the night. By dawn, the bank syndicate had agreed to amend the contract. The gala signoff ceremony went off as planned, complete with fireworks, champagne, a choral group, and a Dixieland jazz band. The ceremony—attended by corporate executives and project managers from the Channel Tunnel's ten prime contracting companies plus a thousand other guests—was a minor project in itself.

Delivery, installation, and user acceptance of the main contract end-item does not necessarily mean the project is closed. As mentioned, project closeout might be dependent on the availability of necessary side items, without which the customer would not be able to operate or maintain the main end-item.

The importance of doing a good job at termination cannot be understated; neither can the difficulty: in the rush to finish the project and the accompanying confusion, it is easy to overlook, mishandle, or botch. The termination responsibilities previously listed should be systematically delegated and checked off as completed. Ultimately, the project might be consummated in a formal project **closeout meeting** where the contractor, owner, sponsor, or customer agree that the terms of the contract have been met and delivery of the end-item is complete.

Closing the contract and negotiated adjustments

In many projects, at closeout the contractor receives payment for only a portion of the total project cost, say 80 to 90 percent, and the remainder is contingent on the end-item's performance, the contractor's compliance with contractual agreements, or the quality of the working

relationship with the contractor.⁷ The contractor might have to repay the customer for any advances received, or the customer might have to pay the contractor for work started but not completed or accepted.

These payment adjustments are considered **post-acceptance** issues because they occur after the customer has accepted the major end-item. If the delivered end-item is satisfactory but found to not perform fully to the contracted specifications, or to be defective after a trial period due to design or production inadequacies, or is delivered late, the contractor might have to pay a **negotiated compensation** to the customer. If the final end-item is found to be deficient after installation or delivery, the contractor might be obligated to provide on-site user support, at no additional fee, to remove the deficiencies.

Sometimes the customer or contractor seeks to negotiate aspects of the contract price or completion date after the project is completed. The US government retains the right to negotiate overhead rates after it receives the final price on cost-plus contracts. Likewise, a contractor sometimes seeks to negotiate a revised completion date on the contract after the project is completed—usually because it overran the scheduled date and wants to salvage its reputation.

It should be noted that all the above applies not only to contractor–customer agreements, but to contractor–subcontractor agreements as well.

5.6 Project Summary Evaluation

Among the final activities of the project team after project closeout is to perform a formal evaluation. This final **summary evaluation** affords project and company management the opportunity to learn from past successes and mistakes. Without a summary review, there is a tendency to mentally suppress problems encountered and to understate the impact of errors or misjudgments—“Things weren’t really so bad, were they?” Project summary evaluation reviews and assesses the performance of the project team and the end-item system. It addresses the questions “What happened, what were the results, and what if anything remains to be done?” Decidedly, its purpose is not to find fault or pass blame on anyone. It is the central theme of the “post-project” and “confirm benefits” stage of the PRINCE2 project methodology. Two forms of summary evaluation are the post-completion project review and the post-installation system review.

Post-completion project review

The **post-completion project review** (also called a **lessons learned meeting** or, perversely, a **postmortem**) is a summary review and assessment of the project, conducted by the project team soon after project closeout—early enough so that project team members are still around to participate and remember what happened.⁸ In long-duration projects, the review might happen after each project phase and before people forget what happened. It is an important task for which funds and time should be included in the project budget and schedule. Post-completion reviews are one way companies try to continuously improve projects through lessons learned from past projects—an opportunity that many companies forego.

The post-completion project review should evaluate:

1. Initial project objectives in terms of technical performance, schedule, and cost; and the soundness of objectives given the needs and problem the end-item system was to resolve.
2. Changes in objectives and reasons for changes, noting which changes were avoidable and which not.

3. The activities and relationships of the project team throughout the project, including the effectiveness of project management; relationships among top management, project team, functional departments, and the customer; and customer reactions and satisfaction.
4. The involvement and performance of all stakeholders, including subcontractors and suppliers, the client, and external support groups.
5. Expenditures, sources of costs, and profitability.
6. Areas of the project where performance was particularly good, noting reasons and identifying processes that worked especially well.
7. Problems, mistakes, oversights, and areas of poor performance, and the causes.
8. A list of lessons learned and recommendations for incorporating them into future projects.

The review happens in a half- or day-long meeting with representatives from all functional areas that substantially contributed to the project. To encourage openness and candor, the managers of these areas should not be at the meeting. An outside facilitator might be selected to guide the review and ensure it is comprehensive and unbiased. At the meeting, participants independently make notes on what went right and wrong with the project; they then share their notes and create a **lessons learned log** or **lessons learned register** providing recommendations for future projects. These are formally shared with stakeholders, others on the project team, and project, functional, and senior managers.

The review seeks to determine lessons that may be applied to future projects, not to criticize or place blame. The results are documented in a **project summary report**, which becomes the authoritative document on the project and describes the project, its evolution and outcomes, the project plan, what worked, and what did not. Because projects affect different parties in different ways, opinions of the customer, project team, and upper management should be listed separately.

The project summary report becomes the reference for project-related questions that might arise later. Thoroughness and clarity are essential since people who worked on the project usually will not be available later to answer questions. The report is retained in a project library, and its lessons learned and recommendations are promoted in other projects, sometimes by the PMO. The post-completion review is a way to capture and reapply knowledge to future projects—a tool for project **knowledge management**, discussed in Chapter 18.



See Chapter 18

Example 5.3: Microsoft Postmortems⁹

Product development projects at Microsoft often conclude with a written postmortem report that is circulated to team members; senior executives; the directors of product development, coding, and testing, and to the highest levels of management, which for major projects includes the company president. A report can take as much as six months to prepare and range in length from under ten pages to over one hundred pages. Its purpose is to describe what worked well in the project, what did not, and what could be done to improve future projects. Descriptive information is also included such as *the size of the project team*, *duration of the project*, *aspects of the product* (size in thousand-lines-of-code [KLOC], languages and platform used), *quality issues* (number of bugs per KLOC, type and severity of bugs), *schedule performance* (actual versus planned dates), and the *development process* (tools used, interdependencies with other groups). Functional managers prepare the initial draft and then circulate it via email to other team members for comment.

Post-installation system review

The post-completion project review focused on the *project*. Several months after installation, a review of the performance of the *end-item* system in the user environment and under normal operational conditions should also be done. This **post-installation system review** serves multiple purposes, such as revealing potential or necessary enhancements to the system, and providing information related to the system's operation and maintenance. Based upon the original user requirements, the post-installation system review attempts to answer the questions:

- Now that the system is fully operational, is it doing what it was intended to do and meeting user requirements?
- Is the user getting the expected benefits from the system?
- What changes, if any, are necessary for the system to better fulfill user needs?

It is important that the evaluated system is *unaltered* from the one delivered. Frequently the user makes modifications and improvements to the system after installation; there is nothing wrong with this per se, but it means the system has been physically or functionally altered from the one installed, which must be considered when evaluating its performance.

During the review the evaluation team might discover elements of the system in need of repair or modification. Design flaws, operating problems, or necessary enhancements that could not have been foreseen earlier sometimes become obvious once the system has been in operation.

The results of the review are summarized in a report that describes the system's performance versus the requirements, any maintenance problems, and suggested possible enhancements. The post-installation system review and the project summary review are filed together and retained as references for planning future projects.

5.7 After the Project—Phase D: Operation

Beyond project termination, what happens next depends on whether the end-item or deliverable is a *physical* system that must be operated and maintained (e.g., a product, machine, or operating procedure), or is a *service* or *event* for which there is nothing physical to operate (e.g. a rock concert, company relocation, corporate merger, or audit). In the former (i.e., the project results in a physical system or product), the systems development cycle enters **Phase D, operation**.

The contractor can remain involved with the customer and the system in the operation phase in two ways: (1) through ongoing evaluation, maintenance, or repair of the system, or (2) through initiating a new project to enhance or replace the original system, perhaps years later.

System evaluation, maintenance, and repair

The contractor may perform evaluation of the end-item system as part of the original contract agreement, either as a one-time scheduled activity in the form of a post-installation review, or as a warranty arrangement whereby the contractor reviews and maintains the system for a pre-specified time period. The contractor might sign an “extended” agreement with the customer to provide preventive maintenance and system repairs, upgrades, and overhauls on a continuing basis, and assign **system representatives** and technicians to the customer site to

perform these services on a scheduled basis or as requested by the user. Providing services for system **maintenance, repair, and overhaul** is commonly referred to as **MRO**.

Enhancing or replacing the system

When the customer wants to enhance or replace the originally contracted system, a new project emerges. From the original contractor's perspective, this is an *extension* to the original project, although it typically involves a whole new contract.

There are two kinds of project extensions: discretionary and essential. **Discretionary extensions** are requested by the customer or proposed by the contractor to improve the operation, performance, or convenience of the original project end-item. The environment remains the same, but new and better ways have appeared that can improve the system. The other kind, **essential extensions**, are compulsory; without them the system will cease to operate effectively or become obsolete. An end-item that is no longer adequate because the environment has changed or the design has become deficient must be enhanced or replaced.

The decision to expand, enhance, or replace a system marks the beginning of a new systems development cycle, one that might be initiated with a new RFP and new proposal. The extension itself becomes a new project. Humankind engages in few dead-end projects; each spurs others, and the systems development cycle keeps, so to speak, rolling along—hence the term “cycle.”

In many cases, however, the system is not enhanced or replaced; it is **terminated**. For example, if the end-item is a nuclear power plant, **system termination** means closure of the plant (typically many decades after it was built) and includes rehabilitation of the plant site—itsself a big, lengthy, expensive project (analogous to the Rocky Flats case in Chapter 20). See Chapter 20. Indeed, such considerations about termination during project feasibility is one reason some projects are never approved.



5.8 Ethics and the Dark Side of Projects¹⁰

We have addressed the life cycle of the project and its end-item product, and, starting with the next chapter, will address practices used to plan and control projects. Before moving ahead though, we need to address the assumption that whatever happens in project management is ethical and appropriate, while acknowledging that in reality that it isn't always.

Projects exist to provide value and benefits, although it must be said that not all project stakeholders benefit from the project, and sometimes certain ones must pay the costs so that others can receive the benefits. In project management, the emphasis is on “satisfying the customer and meeting customer requirements,” although the unsaid part of that statement is “and do it in a way that is moral and meets universal ethical standards.” Unfortunately, projects, like all human endeavors, don't always meet those standards. The **dark side of projects** can be defined as any illegal or unethical phenomena associated with projects. Examples include modern slavery, corruption, exploitation of the natural environment, and cut-throat organizational politics. It also includes bad behavior outside of projects that impacts their performance or affects how projects are initiated, governed, managed, and executed.

What is considered the “dark side” can be controversial depending on customs and laws. Some project practices are clearly dark side, for example, bribery, since it violates both local and international laws. Others are controversial, for example the Kafala system, which is a variety of laws and practices, adopted in diverse ways in different countries to manage immigration and work visas. Although it has been criticized as favoring modern slavery and worker

exploitation, some countries consider the Kafala system legal and ethical. Another example is sexism. In some countries it is okay to discriminate by gender, but according to international standards it is illegal or unethical. A third example is tax evasion where big companies pay little or no taxes to the countries wherein they operate or sell products; although this is lawful according to the international legal framework, it is considered unethical in many countries because it gives nothing back to the countries that support the companies. Although certain practices are obviously on the dark side, others are dark too, such as engineers being forced to make design decisions that they know are unsafe yet are economically justified and not illegal.

Here is a brief rundown of practices that put projects on the dark side.

- **Modern slavery** is the “recruitment, movement, harboring or receiving of people through the use of force, coercion, abuse of vulnerability, deception or other means for the purpose of exploitation.”¹¹ According to one conservative estimate, over 40 million people were in modern slavery as of 2017. An example is the Qatar 2022 World Cup and projects to build infrastructure and the Khalifa Stadium in Doha. Workers lived in crowded and unsafe conditions, received low pay, had limited access to justice and health care, and could not exit the country legally; reportedly, 6,500 of them died. Unfortunately, modern slavery seems to be somewhat common, especially in global megaprojects.
- **Corruption** refers to anything unlawful. An example is the Rio Olympics in 2016, which began with a \$2 million bribery to buy votes to host the Olympics and continued with more bribery to build the Olympic Park and its infrastructure. Brazil by far is not an exception; the annual costs of corruption across the globe is estimated at \$3.6 trillion.
- **Money laundering** is attempting to conceal the source and ownership of proceeds received from unlawful activities, and to access and control them thereafter. It is prominent in big construction projects and can be associated with illicit funds obtained through bribes and income derived from fraud, modern slavery, and tax evasion.
- **Conflict of interest** is a situation where one’s private interests conflict with their responsibilities. Unlike other dark sides, conflict of interest does not require malicious intent; it can arise whenever people must make decisions that are incompatible with their vested interests. It is common in projects where decision makers must account for the demands of multiple stakeholders who have differing vested interests in the project.
- **Sexism and gender discrimination** refers to negative stereotypes about women and other genders, and that these groups are discriminated against in terms of salary and access to jobs, promotions, and power. Studies show, for example, that women are underrepresented in project management roles in male-dominated professions such as construction and engineering, and in upper management positions everywhere.
- **Greenwashing** means promoting a project’s environmental (green) virtues when, in fact, it has few, none, or is even harmful. The term pertains beyond environmental matters and includes broadcasting misinformation that promotes anything about the project, its practices, or outcomes that is untrue or misleading. A solar-panel farm project might be “green,” but it is greenwashing when it subjects workers to unsafe work conditions or leaves the local citizenry with a degraded living environment.

Although not universal, dark side practices are, unfortunately, common in certain industries, localities, and regions of the world. The discussion here offers no solutions to delimiting these dark practices or mending their consequences; rather, the intent is to raise awareness of these practices in the hope that managers will work to resist them or ameliorate their consequences whenever possible.

5.9 Summary

The execution phase includes the stages of *detail design*, *production/build*, and *implementation*. During the design stage, the system concept is subdivided into tiers of subsystems, components, and parts, and for each design, schematics, and models are created. The result is a *functional* design that represents the functions of the end-item system and its components, and a *physical* design that shows the configuration, dimensions, materials, and other physical aspects of the end-item system and its components. During design, the emphasis remains on meeting the requirements of the end-users and avoiding ancillary, mostly unneeded features that result in bloatware, i.e., overly complex and user-unfriendly products. In most projects, design costs are a small fraction of production/build costs; thus, prolonging this stage to get the design right is usually less costly than changing the design later in production/build to get it right. System design logically precedes production/build, although in a practice called fast-tracking the stages are overlapped: building begins after only some of the design has been completed.

In the production/build stage the main activities are fabrication and testing. Components and the end-item system are assembled and tested to ensure that requirements for the system are met. The project manager is responsible for coordinating activities, controlling changes, and assuring the quality of the end-item by its conformance to the design and system requirements. Throughout this stage, the project manager is preparing for the next stage: implementation.

Implementation is when the end-item system or other deliverable is completed and turned over to the user. Among the principal tasks during implementation are user training, user tests of acceptance, and system installation and conversion. The user is trained to operate, maintain, and service the system, and then performs tests on the end-item system to determine if it is acceptable. Actual implementation of the end-item system and conversion from the old system to the new system requires a strategy; the three main ones are parallel, pilot, and cold turkey. Side items necessary for the operation and upkeep of the end-item system, often provided by other contractors, must be planned for and fully available upon or before implementation.

The last stage of execution is project termination and closeout. Every project should be terminated through a formal closeout procedure; this includes the closing of all work orders, project facilities, project books; and ensuring delivery of and customer acceptance of all end-items. The project is not considered closed until the customer has reviewed and approved everything promised about the end-item.

Following project close-out, a summary evaluation should be conducted. This evaluation has two parts: a post-completion project review (postmortem) to assess the effectiveness of the project organization and identify lessons learned and recommendations for future projects, and a post-installation system review to assess the end-item system's performance and determine possible maintenance or enhancements needs. The documented results of the project review and the system review are filed together as reference for future projects.

Project termination/closeout marks the end of the project and the end of the project life cycle. But in terms of the system development cycle, there might be one more phase remaining, operation, which is that phase where the system is operated and maintained by the user. The contractor of the system can remain involved with the customer during the operation phase by (1) agreeing to ongoing monitoring and upkeep of the system, or (2) conducting a new project to enhance or replace the system. In the former, the contractor continues to work with the customer in the assessment, maintenance, and/or repair of the system, either on a scheduled basis or as requested by the user. In the latter, the customer hires the contractor to enhance the end-item system or replace it with something entirely new—the beginning of a new end-item and a new project.

Like all human endeavors, projects can have dark sides—practices that are unlawful, unethical, inappropriate, or immoral; they include modern slavery, corruption, greenwashing, and gender bias. Although not universal to all projects, the fact that they exist in some projects must be acknowledged and, wherever possible, resisted.



Review Questions

1. What is the practice of “fast tracking” or “design/build?” What are the associated potential benefits and dangers?
2. What happens during the design stage? Who is involved? What do they do? What is the role of the project manager? How are design changes monitored and controlled?
3. What is the role of interaction design in product design and development?
4. What does the plan for production/build include?
5. What happens during the production/build stage? How is work planned and coordinated? Who oversees the work?
6. What is the distinction between the project end-item and project side items? What role does the project manager have regarding each?
7. How is the system implemented? Describe the important considerations for turning the system over to the user.
8. Discuss user training and why it is sometimes included in the implementation stage.
9. How is the project end-item tested and checked out for approval?
10. Describe the different strategies for installing or converting over to the new system.
11. What are the reasons for project termination? How can termination for reasons other than achievement of project goals be avoided?
12. What is involved in planning and scheduling the project termination?
13. What is the role of the project manager and contract administrator in receiving customer acceptance of the work and final payment?
14. What are side items? Give examples not used in this book. How can they delay project completion?
15. What kinds of negotiated adjustments are made post-acceptance to the contract? Why would a user or contractor want to specify the terms of a contract *after* the project is completed?
16. What is a punch list?
17. What are project extensions and how do they originate? How is a project extension managed?
18. What are the differences between the two kinds of project summary reviews: the post-completion project (or postmortem) review and the post-installation system review? Describe each.
19. Describe what happens during the operation phase. What role does the systems development organization (contractor) play in this phase?
20. List practices that clearly put projects on “the dark side.” Give five examples from recent news reports illustrating these practices.



Questions About the Study Project

As appropriate, answer the above review questions 2, 5, 7, and 9 regarding the study project. Also answer the following questions:

1. How was the project terminated? Describe the activities of the project manager during the final stage of the project and the steps taken to close it out.
2. If the end-item is a building or other “constructed” item, how was it turned over to the user? Describe the testing, acceptance, training, and authorization process.
3. How was the contract closed out? Were there any side items or negotiated adjustments to the contract?
4. Did any follow-up projects grow out of the project being investigated?
5. Describe the project summary review (postmortem). Who prepared it? Who was it sent to? How was it used? Where is it now? Show an example (or portion of one).
6. Was there a post-installation review of the product or project output? When? By whom? What did they find? Did the client request the review or was it standard procedure?
7. What happened to the project team after the project was completed?
8. Did the contractor or project manager remain involved with the customer and end-item through an extended agreement?

CASE 5.1 SLU INFORMATION CENTRAL BUILDING

Construction of the new Information Central building at South Land University (SLU) has just been completed. SLU administrators and managers at Finley Construction Company, the building’s prime contractor, are pleased with the results. Besides meeting schedule and cost targets, the building and its equipment, including a variety of computer and technical gadgetry intended to augment learning, appear to have met all the technical requirements. Much of the technology is leading edge, and some of it is being applied for the first time in a learning/teaching environment by SLU. By all accounts the project is a success.

After reviewing and confirming that all of Finley’s obligations for the project have been met, Jack Krackower, the project manager, meets with Sharon Holden, SLU’s vice president of finance, and Ramat Ghan, SLU’s vice president of facilities, to finalize details for project termination and payment. The meeting goes well and ends with discussion of future projects at SLU and Finley’s possible involvement. After the meeting Jack returns to his office, whereupon the director of Finley’s PMO asks him if he planned to do a post-completion project review. “Nope,” quipped Jack, “no need to. The project was a success, and everything went as planned.”

A few months later, Sharon and Ramat give a final presentation on the project to SLU’s president, reporting that it met all the technical and building requirements, the schedule, and the budget. In fact, they say, given the positive outcome of the project, some of the new technology in the building should be installed in other campus buildings and Finley hired to oversee it. “Not so fast,” says the president. “I’ve heard reports that students and faculty find the new technology confusing, difficult to use, and maybe

irrelevant. In fact, some rooms in the building are almost vacant for lack of use. Other rooms are crowded, but students go there to socialize or relax, not to take advantage of any sophisticated learning technologies. I don't know what the problem is—if it's with the technology or with the way Finley handled it."

QUESTIONS

1. Comment on Jack's neglect to conduct a post-completion project review. Is a review unnecessary whenever a project is considered a success?
2. Is the project really a success? What kind of follow-up steps should Finley and SLU have done after the project was completed?

Notes

1. Design output is normally catalogued in a master record index or data pack that lists all drawings, material specifications, process specifications e.g. for materials heat treatment, welding, etc. One guide for specification practices is MIL-STD 490A.
2. Fetherston D. *The Chunnel*. New York, NY: Times Books; 1997, pp. 198–199.
3. Cooper A. *The Inmates Are Running the Asylum: Why High-Tech Products Drive Us Crazy and How to Restore the Sanity*. Indianapolis, IN: Sams; 1999.
4. Hajek V. *Managing Engineering Projects*, 3rd edn. New York: McGraw-Hill; 1984, pp. 233–240 describes monitoring and support side items for both engineering hardware and computer software projects.
5. See Archibald R. *Managing High-Technology Programs and Projects*, pp. 235–236 and 264–270, for a complete checklist of closeout activities.
6. Fetherston D. *Chunnel*. New York: Time Books; 1997, pp. 372–375.
7. Hajek V. *Managing Engineering Projects*, pp. 233–240 describes monitoring and support side items for both engineering hardware and computer software projects.
8. Williams T. *Post-Project Reviews to Gain Effective Lessons Learned*. Newton Square, PA: Project Management Institute, 2007; Whitten N. *Managing Software Development Projects*, 2nd edn. New York: John Wiley & Sons, 1995, pp. 343–357.
9. Cusumano M. and Selby R. *Microsoft Secrets*. New York: Free Press; 1995, pp. 331–334.
10. Primary source for this section is Locatelli G., Konstantinou E., Gerald J. and Sainati, T. The dark side of projects: dimensionality, research methods, and agenda. *Project Management Journal* 53(4); 2022: 367–381. <https://doi.org/10.1177/87569728221103911>. Accessed Feb. 14, 2024. Unless noted otherwise, all cited statistics and their sources come from this article.
11. Locatelli et al. The dark side of projects, p. 375.

Part III

Systems and Procedures for Planning and Control

6	Basic Project Planning Techniques	175
7	Project Schedule Planning and Networks	228
8	Advanced Project Network Analysis and Scheduling	272
9	Project Costs and Budgeting	317
10	Project Quality Management	351
11	Project Risk Management	384
12	Project Procurement Management and Contracting	429
13	Project Monitoring and Control	456
14	Agile, Lean, and Hybrid Project Management	500

Beyond defining project objectives and requirements, project management involves forming a project organization or team, identifying the necessary work tasks and the resources to do them, and providing leadership to get the work done. Project objectives and requirements need to be articulated into detailed plans, schedules, and budgets. The project must then be managed to ensure the plans and schedules are carried out as intended.

Over the years an impressive collection of methods has been developed to help project managers define, plan, and direct project work. The next nine chapters describe these methods, which include techniques and procedures for identifying, scheduling, and budgeting project activities, assessing risks, and monitoring and controlling work to achieve project quality, time, and cost requirements.

Much of this part of the book focuses on **traditional project management (TPM or waterfall)**, and tools for planning and controlling projects where the solution or end-item is well understood, and the project progresses sequentially through stages of the project life cycle. Projects that are less well-understood and must be executed iteratively use **agile (APM)** and **hybrid project management** approaches; planning and control tools for these projects are discussed in Chapter 14. Important to note is that most of the principles and many of the tools from TPM also apply to APM, although in modified forms.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Chapter 6

Basic Project Planning Techniques

Every project is somewhat unique because it is aimed toward a result or end-item that is, in some way, unique. Because of its uniqueness, basic questions about the project must be answered before work can begin. Answering these questions so that the project will achieve its goals is the purpose of **project planning**. This chapter gives an overview of the project planning process and covers the topics of project scope and work definition, estimating, and elementary scheduling.

6.1 Planning Process

Once the project has been approved and the contract signed, the project can begin. But before any actual work begins, the system requirements must be defined and a project plan prepared. For internal projects, a project charter is prepared and sent to stakeholders to announce the project.

The project manager, if not already assigned or involved, is now identified to oversee the planning process, which will result in a plan that elaborates on earlier plans as prepared for the proposal, business case study, or charter.

Because each project is unique, there never is an a priori, established way for exactly how the project should be done. Each project poses new questions, and the planning process requires answering them, roughly in this order:

1. **What is the desired end result?**
Define the project **objectives**, **scope**, and **system requirements**. These specify the project deliverables, end-items, products, or other sought results, plus the time, cost, and performance targets.
2. **How will the result be achieved?**
Define the **work activities**, tasks, or jobs to be done to achieve the objectives and requirements. These will include everything necessary to create and deliver the end-item and deliverables, including planning, control, and administration activities.

3. **Who will do it?**
Specify the **project organization**—the individuals, departments, contractors, and managers who will perform and manage the work, and specify their **responsibilities**.
4. **When and in what order?**
Create a **schedule** showing the timing of work activities, deadlines, and milestone dates.
5. **How much?**
Create a **budget** and **resource plan** to fund and support the project.
6. **How well?**
Specify methods for **monitoring** and **controlling** project work to keep it conforming to the project plan.

The result will be a project plan. Since the plan specifies what will happen in Phase C, project execution, we will refer to it as the **project execution plan**. This and the next several chapters discuss the plan in detail.

6.2 Project Execution Plan

Project planning begins early in the project life cycle—in most cases with preparation of the proposal. During proposal preparation, a proposal team or rudimentary project team is organized, and the team prepares a brief summary plan for inclusion in the proposal. They prepare the plan using the same, albeit more abbreviated, procedures as used later to develop a more elaborate and detailed project execution plan. The difference between a proposal summary plan and a project execution plan is that the former is aimed at the customer, while the latter is aimed at the project team and is, therefore, more detailed and comprehensive.¹ The plan in the proposal includes estimates for the project duration, cost, and needed resources—just enough information about the project and its price to enable the customer to make a decision. In contrast, the project execution plan lays out details of what is to happen in the project and will be used as a baseline and roadmap to *guide* the project team throughout project execution. As discussed in Chapter 4, the plan might contain details only for the immediate upcoming phase of the project, about which the most is known, with additional details for later project phases to be added as the project progresses and more about the project becomes known.



See Chapter 4

Contents of execution plans

Contents of execution plans vary depending on the size, complexity, and nature of the project. Figure 6.1 shows a template for a typical plan as outlined in Chapter 4.² Depending on the customer and type of project contract, the plan might require additional items not shown;³ in small, low-cost projects it might exclude some of the items, being careful not to overlook the crucial ones. It is good practice to review every item in the template even if only to verify that some are “N/A” (not applicable). An example of the project execution plan for the LOGON project is at the end of the book in Appendix C.



See Appendix C



See Chapter 3

You might notice similarities between sections of the project plan and contents of the proposal, as displayed in Chapter 3, Figure 3.6. That’s because the proposal, after revisions to reflect updates, agreements, and contract specifications, sometimes becomes the project execution plan. More often, however, the proposal serves as only an outline, and the execution plan is much more expansive. The execution plan is aimed at the project team—to tell them what must be done in the project, who will do it, when, for how much, and so on.

Project Execution Plan

I. Scope, Charter, or Statement of Work

Overview description of the project oriented towards management, customer, and stakeholders. Includes a brief description of the project, objectives, overall requirements, constraints, risks, problem areas and solutions, master schedule showing major events and milestones.

II. Management and Organization Section.

Project organization, management and personnel requirements.

- A. Project management and organization: key personnel and authority relationships.
- B. Manpower: Workforce requirements estimates: skills, expertise, and strategies for locating and recruiting qualified people.
- C. Training and development: Executive development and personnel training necessary to support the project.

III. Technical Section. Major project activities, timing, and cost.

- A. High-level **user requirements** and **system requirements**.
- B. **Work breakdown structure**: Work packages and detailed description of each, including resources, costs, schedules, and risks.
- C. **Responsibility assignments**: List of key personnel and their responsibilities for work packages and other areas of the project.
- D. **Project schedules**: Generalized project and task schedules showing major events, milestones, and points of critical action or decision.
- E. **Budget**: Control accounts and sources of financial support: Budgets and timing of all capital and developmental expenses for work packages and project.
- F. **Quality plan**: Measures for monitoring quality and accepting results for individual work tasks, components, and end-item assemblies.
- G. Areas of uncertainty, and **risk plan**: Risk strategies, contingency and mitigation plans for areas posing greatest risk.
- H. **Health, safety, and environmental (HSE) plan**: To address HSE and sustainability matters affecting project workers, the community, and society arising from the project and its end-results.
- I. **Communication plan**: Expected meetings, reports, participants, formats—formal and informal.
- J. **Human resource plan**: Project organization, roles, responsibilities, staffing, and training.
- K. **Work review plan**: Procedures for periodic review of work, what is to be reviewed, by whom, when, and according to what standards.
- L. **Testing plan** (may be included in work review plan): Listing of items to be tested, test procedures, timing, and persons responsible.
- M. **Change control plan**: Procedures for review and handling of requested changes or de facto changes to any aspect of the project.
- N. **Documentation policy/plan**: List of documents to be produced, format, timing, and how they will be organized and maintained.
- O. **Procurement policy/plan**: Policy, budget, schedule, plan, and controls for all for goods, work, and services to be procured externally.
- P. **Implementation plan**: Procedures to guide customer conversion to or adoption of project deliverables.

Figure 6.1
Template for Project Execution Plan.

As illustrated in Example 6.1, developing a project execution plan can be an evolutionary, cross-functional process.

Example 6.1: Developing a Project Plan for LOGON Project at Iron Butterfly Company

Iron Butterfly Company (IBC) is a medium-sized engineering and manufacturing firm specializing in warehousing and materials handling systems. It purchases most of the subsystems and components for its product systems, and then combines them to meet customer requirements. The company was awarded a contract by the MPD Company to design and build a system to place, store, retrieve, and route shipping containers. The system, called the Logistical Online System, LOGON, is to be developed and installed at MPD's Chicago distribution center. Iron Butterfly is responsible for design, assembly, and installation of the system. Two of its contractors, CRC and Creative Robotics, will provide the computer and robotic drone systems plus assistance with their installation and checkout. Frank Wesley, the IBC project manager, is in charge of preparing the project execution plan.

Most of the project plan for LOGON originated in IBC's project proposal. In preparing the proposal, engineers from IBC, CRC, and Creative Robotics together designed a system that covered MPD's requirements. The design included schematics, operational specifications, and a bill of materials. Design managers at CRC and Creative Robotics estimated the labor expertise needed and costs for parts and labor. Frank and his engineers prepared a work breakdown structure and estimates for IBC's time and costs. They combined these with CRC's and Creative Robotics' estimates to arrive at a plan, schedule, and price for the proposal.

After winning the contract, Frank met with his project engineer and managers from the fabrication, software, and purchasing departments to review the design, plan, costs, and schedules in the proposal and prepare a detailed execution plan. This plan contained similar information to the proposal but had been updated and expanded to include schedules for procured materials and parts, plans for labor distribution across work tasks, a task responsibility matrix, a detailed work breakdown structure and a budget and master schedule.

The LOGON project execution plan evolved in stages: it was initially created during proposal preparation but then was expanded and modified after contract signing. In many projects, however, particularly for large, complex systems, the proposal serves only as a reference and the bulk of project planning happens *after* the contract is signed (i.e., in Phase B, Definition). Often, project planning is a significant effort that requires substantial time and labor, a project in itself.

Learning from past projects

Oftentimes organizations look at each project as being too unique and ignore the lessons of past projects.⁴ No project is totally unique, so in developing the project plan it makes sense to refer to earlier, similar projects, their plans, procedures, successes, and failures. Ideally the project manager receives planning assistance in the form of lessons learned, best practices, suggested methodologies, and even consultation, all that rely upon experience from past

projects. Sometimes this assistance is provided through the project management office (PMO), described in Chapter 18. Lessons learned and best practices are compiled by the PMO from the **post-completion project reviews** (or **postmortems**) of past projects, described in Chapter 5.



See Chapters 5
and 18

6.3 Scope and Statement of Work

Project planning starts with defining the objectives, deliverables, and major tasks of the project; in combination these determine the overall size of the project and the range of work it encompasses, the concept of **project scope**. Determining the project scope happens during project conception, first when the project is initiated, then during preparation of the RFP and the proposal, and then again during project definition. Each time, user needs and requirements are compared to time, cost, resource, and technology constraints to determine what the project should and can encompass. This scope-setting process is called scope definition.

Scope definition

Scope definition involves specifying the breadth of the project and the span of its outputs, end-results, or deliverables. Results or end-items to be produced or delivered by the project are termed **inclusions**, meaning they are *included* in the project. To ensure clarity, any results, items, or conditions *not* to be included in the project, called **exclusions**, are also defined. For example, a building construction project might exclude the building's landscaping and interior decorating. Distinguishing exclusions (customer responsibilities) from inclusions (contractor responsibilities) prevents misunderstanding and false expectations.

Scope definition focuses primarily on determining outputs and deliverables, not time and cost. Of course, time and cost delimit or dictate the potential deliverables; as such, they might be accounted for as “constraints” in the scope definition.

The outcome of scope definition is a **scope statement** that describes the main results, end-items, or deliverables of the project. The scope statement might also include criteria for acceptance of the deliverables, assumptions, and constraints (to provide rationale for why the project has these deliverables and not others), functions to be fulfilled by the deliverables, brief background about the problem being addressed or the opportunity being exploited, project objectives, user requirements or high-level specifications, and high-level project tasks or major areas of work. The input information needed for scope definition includes a set of user needs and requirements, a business case or other expression of needs, and assumptions and constraints. Ideally, the principal subsystems and components of the end-item will have been identified and serve as inputs.

Besides everything to be included and things to be excluded, the scope statement sometimes lists outcomes or consequences to be *avoided*, such as negative publicity, interference with other systems, pollution, or damage to the natural environment. Rather than repeat the detailed requirements and specifications, the scope statement normally references the documents that contain them.

For a unique project concept, the preliminary scope statement defined during project initiation might be somewhat vague; it must, however, be expanded and clarified during project definition while detailed plans for the first phase of the project are being developed. For *programs*, scope statements are developed for the overall program and for the individual projects that comprise it.

Once the scope statement has been approved, it becomes a controlled document that can be modified only through a formal **change control** process, described in Chapter 13.



See Chapter 13

Example 6.2: Scope Statement for the LOGON Project

The RFP for the LOGON project (see Appendix A, back of this book) sent by Midwest Parcel Distribution Company (MPD) specifies “The Contractor shall be responsible for furnishing expertise, labor, materials, tools, supervision, and services for the complete design, development, installation, checkout, and related services for full operational capability of the LOGON system.” It also specifies the technical performance requirements for the system, as well as project exclusions, i.e., “Removal of existing storage, placement, and retrieval equipment will be performed under separate contract.”

Upon receiving the RFP, Iron Butterfly Company (IBC), one of the proposing contractors, decided that the best way to meet MPD’s needs was with a system that employs robotic drone transporter units for placing and retrieving containers as instructed by a neural-network system. IBC analyzed MPD’s technical and budget requirements, and after a preliminary system design effort created the following scope statement for its LOGON proposal:

1. Project background: (short description of MPD’s Chicago distribution facility, and of the purpose and objectives of the LOGON system).
2. Description of the work to be done: design, fabrication, installation, test, and checkout of a transport, storage, and database system for the automatic placement, storage, and retrieval of standardized shipping containers.
3. Deliverables and main areas of work:
 - a. Overall system: create basic design. Reference requirements A and B.
 - b. Racks and storage-bucket system (termed “Hardware A”): develop detailed design. The storage-bucket system is Model IBS05 adapted to requirements C.1 through E.14.
 - c. Robotic drone transporter units and track system (termed “Hardware B”): develop detailed design. RBU is Model IBR04 modified to meet requirements F.1 through G.13.
 - d. Neural-network, database, and robotic-controller system: develop software specifications. Reference requirements H.1 through H.9 and K.3.
 - e. Hardware A and Hardware B: procure software, subassemblies, and components. Reference requirements K.1 through L.9.
 - f. Hardware A and Hardware B: fabricate at IBC site. Reference requirement M.
 - g. Overall system: install and check-out at MPD site. Reference requirement Y.

Items 3a-g represent deliverables for different stages of the project; associated with each are specific requirements (i.e., “Reference requirements”) listed in separate documents appended to the scope statement. For example, detailed design as noted in points 3b and 3c includes reference to requirements C.1–E.14 and F.1–G.13. The requirements must be comprehensive enough to enable subcontractors to produce the specified systems and components. Elsewhere the scope statement lists exclusions, either as noted in the RFP or identified by IBC.

The scope statement is the reference document for project stakeholders; it becomes the basis for making decisions about resources needed for the project, and, later, determining whether required or requested changes to work tasks and deliverables fall within the agreed-upon scope. A common tendency in projects is **scope creep**, which means the project grows

due to changes in the number and/or size of deliverables. Scope creep, if not controlled, can lead to runaway project budgets and schedules.

The scope statement appears in many places—project proposals, charters, and plans. Often the scope statement is incorporated into the statement of work.

Statement of work (SOW)

The **statement of work** is an overview description of the project; it can include the scope statement but often goes far beyond that, describing, for example, specifications and requirements; deliverables; schedule for deliverables; management procedures for communication and handling risks and changes; project budget; and key personnel responsible for administrative and work tasks. The SOW is effectively a high-level version of the project execution plan.

The term SOW and its usage are commonly associated with **contracted projects**, and the SOW appears in documents associated with the contracting or procurement process. The RFP, proposal, contract, and project execution plan all contain SOWs, each an updated, expanded, or more refined version of the SOW in previous documents. The project charter might also contain a SOW.

6.4 Work Definition

Once project objectives and deliverables have been set in the scope statement, the next step is to translate them into specific, well-defined work activities; that is, to specify the tasks and jobs that the project team must do. Particularly for large, unique projects it is easy to include unnecessary activities or overlook necessary activities. To ensure that every necessary activity is identified and clearly defined and none are missed, a work breakdown structure is prepared.

Work breakdown structure (WBS)

As described in earlier chapters, the project end-item or deliverable can be thought of as a system that consists of subsystems, which themselves consist of smaller components, and so on. A project can be viewed similarly, and a way to subdivide it into smaller elements is called the **work breakdown structure**. The purpose of the WBS is to divide the project into “pieces of work” or **work packages**. Dividing a project into work packages helps in preparing time and cost estimates, schedules, and budgets and in assigning work task responsibilities.

Creating a WBS begins with dividing the overall project into major categories or deliverables. These categories are then divided into subcategories that, in turn, are each subdivided. With this level-by-level breakdown, the scope and complexity of work elements at each level get smaller. The objective is to reduce the project into many small work elements, each so clearly defined that the project can be easily planned, budgeted, scheduled, and monitored.

A typical WBS consists of three levels:

Level	Element Description
1	Project
2	Sub-project or deliverable
3	Work package

Level 1 is the total project. Level 2 is the project broken down into several (usually four to ten) major elements, sub-projects, or deliverables. These sub-projects must conform to the deliverables or work areas specified in the scope statement, and all of them when combined must comprise the *total project scope*. Each sub-project is then broken down into activities at Level 3. If a further breakdown is necessary, that occurs at Level 4. When the project is part of a **program**, another level is added at the top and the levels are renumbered: Level 1 is the program, Level 2 the project, and so on.

In the table above, the term **work package** appears at Level 3, but that is for illustration only. The PMI uses the term work package to refer tasks at the *bottom level of each branch of the WBS*, whatever level that might be. Besides work package, the terms **activity**, **task**, and **deliverable** are also used to refer to work in the project. Throughout this book, the terms work package, activity, and task will be used interchangeably, although the term work package has a more special meaning, described later. The actual number of levels in a WBS varies by project, as do the element descriptions at each level. Note the different levels and element descriptions in the WBS in Figure 6.2, with elements at the bottom of each branch termed “work package.”

For a small project it makes sense to break down the work to Level 2 or maybe 3. For a large project, however, this would result in a WBS too big to be practical, so it makes better sense to break down the work to Level 3 and then, for each work package create a separate document that gives details about the work package (see Example 6.4). All the work package documents can be combined into a **WBS list** or **WBS dictionary** that includes for every work package everything necessary to perform the work.

The WBS process happens somewhat naturally, starting with the list of user and system requirements. These requirements suggest the main system, end-item, or deliverables of the project and the major subsystems and components; they also say which of these results will be met externally (by suppliers/subcontractors) and which internally. These major subsystems and components are boxes on the WBS. Those boxes are then logically subdivided into smaller components of the system and the work tasks to create or acquire them. For technical and engineering projects, the WBS should include all the **configuration items (CIs)** and major components of the system, as well as the work tasks to design, develop, build, and test them.⁵

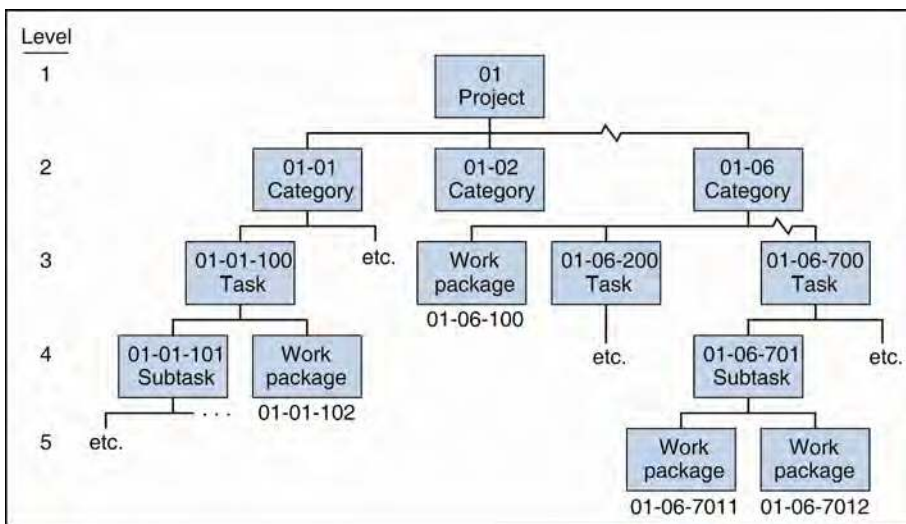


Figure 6.2
Elements of a WBS.

The WBS becomes the basis for assigning project responsibility and contracting. For contracted work, responsibility for each sub-project or activity is assigned to a contractor through a contract agreement between the contractor and the project manager. For internal projects, responsibility for each sub-project is assigned to an in-house department through an agreement between the department manager and the project manager.

To avoid unnecessary complexity, the number of levels in the WBS should be limited. For a large project, a five-level WBS might be necessary, but for most small projects a three-level WBS is adequate. To organize and help track project activities, each work element is coded with a **unique code identifier**, where the identifier at each level is based on the identifier at the level above it. In Figure 6.2 Project “01” has six categories numbered 01–01 through 01–06; then, for example, category 01–06 has seven tasks numbered 01–06–100 through 01–06–700. The project manager establishes the identifier scheme.

Figure 6.3 illustrates ways to create the WBS for constructing a house. The top part of the figure (a **hardware, product-oriented WBS**) shows the main project end-item

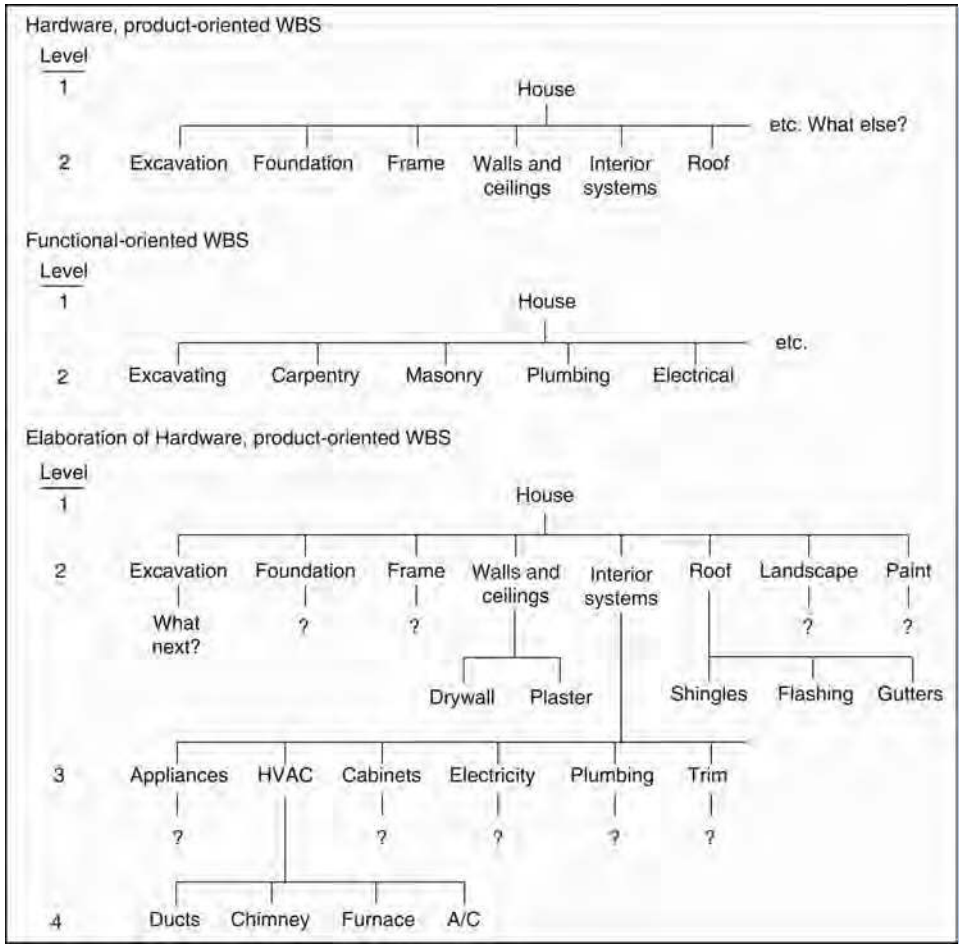


Figure 6.3
Example WBS for house project.

(Level 1) and the physical components of the house to be included (Level 2). By subdividing a project in this way—according to physical products or deliverables, it is easy to attach performance, cost, and time requirements to each item, and to assign responsibility for meeting those requirements. That is, creating a WBS assists in preparing the project schedule and budget. The bottom part of Figure 6.3 shows how the product-oriented WBS would be subdivided into four levels.

Sometimes the WBS or its portions are divided by work tasks or functions (a **function-oriented**, rather than product-oriented WBS). For example, the middle part of Figure 6.3 shows the project subdivided according to the functions of excavating, carpentry, and masonry. Functions or tasks that apply to deliverables or to *integrating* multiple deliverables (e.g., management overhead, design, engineering, training, inspection) should be identified as separate work packages. The WBS can also be broken down by project phase or geographical location. Whether to use a product- or function-oriented or other form of breakdown in the WBS is a matter of preference or is stipulated by the customer or contractor organization.

During the WBS process, the question “What else is needed?” constantly arises. Supplementary or missed elements are identified and added to the WBS at appropriate levels. For example, the bottom WBS in Figure 6.3 does not include blueprints, budgets, and work schedules, even though a house cannot be built without them. These are deliverables associated with designing the house and managing the project and could be included in the WBS by inserting boxes for “Design” and “Project Management” in Level 2, and then inserting “blueprints” under Design and “budget and work schedules” under Project Management at Level 3, respectively.



See Chapter 4

The concept of **traceability**, mentioned in Chapter 4, applies directly to the WBS. One test for the completeness of the WBS is to compare the list of project objectives and high-level requirements with work packages in the WBS. Every objective and requirement should be traceable to at least one work package. If an objective or requirement cannot be traced to a work package, then it will likely not be met. The reverse also applies: every work package should be traceable to at least one objective or high-level requirement. If it cannot, the question is, why is it in the WBS?

Figure 6.4 exemplifies the WBS for a large engineering project where the main system and many of its subsystems and components must be developed, built, integrated, and tested from scratch. Notice, some portions of it are hardware, product-oriented (vehicle and facilities), while others are function-oriented (test/evaluation and project management/systems engineering).

As creation of the WBS progresses, the question always arises “should we do this work ourselves or should we procure it” (meaning buy it or contract someone to do it for us)? Most projects involve at least some procured goods, materials, or services, and in some projects most everything is “procured” and virtually nothing is produced “internally.” Whether project work should be done internally or procured from outsiders is the result of a make-or-buy analysis of the project end-item and its subsystems, components, services, or deliverables identified in the WBS. The process by which this happens and how it is integrated into project planning is



See Chapter 12 the subject of **procurement management** and Chapter 12.

In general, the larger and less standardized the project, the easier it is to overlook something, and the more valuable the WBS process is to avoid that. In large projects the initial WBS is usually rather coarse and shows only major products or work functions and aspects of each to be allocated to specific contractors. Before work commences, however, details of each product or function must be more fully developed in the WBS.

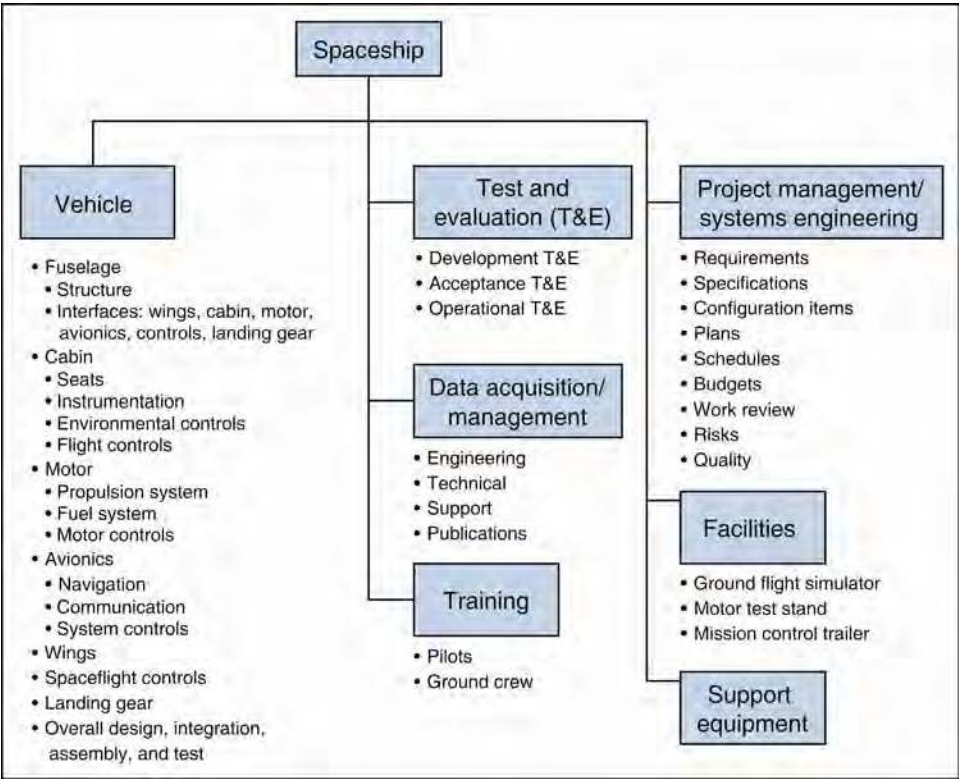


Figure 6.4
WBS for spaceship project.

Example 6.3: Process of Developing the WBS for the LOGON Project

Project manager Frank Wesley and his staff meet several times in brainstorming sessions to create the WBS for LOGON, first during proposal preparation to sort out key deliverables and define the project scope, later during project definition to update the WBS and break down the work packages into greater detail. They first “rough out” the major categories of work and deliverables in the Level 2 breakdown from the SOW and requirements (described in Example 6.2) and identify the responsible functional areas.

After contract signing, Frank meets with managers from the functional areas that will be contributing to the deliverables in the Level 2 breakdown (Figure 6.5). The managers then meet with their technical staffs to prepare Level 3 and, where necessary, Level 4 breakdowns.

The WBS in Figure 6.5 is part function-oriented (basic design, procurement, etc.) and part product-oriented (Hardware Part A, Hardware Part B, software, etc.). Where necessary, Level 2 items have been subdivided into Level 3 items, and Level 3 items into Level 4 items. The boxes at the bottoms of the branches are “work packages,” denoted by letters in parentheses. Notice that the designated work packages are at different levels (some at Level 2, some at Levels 3 or 4); this is because each branch of the WBS is developed separately.

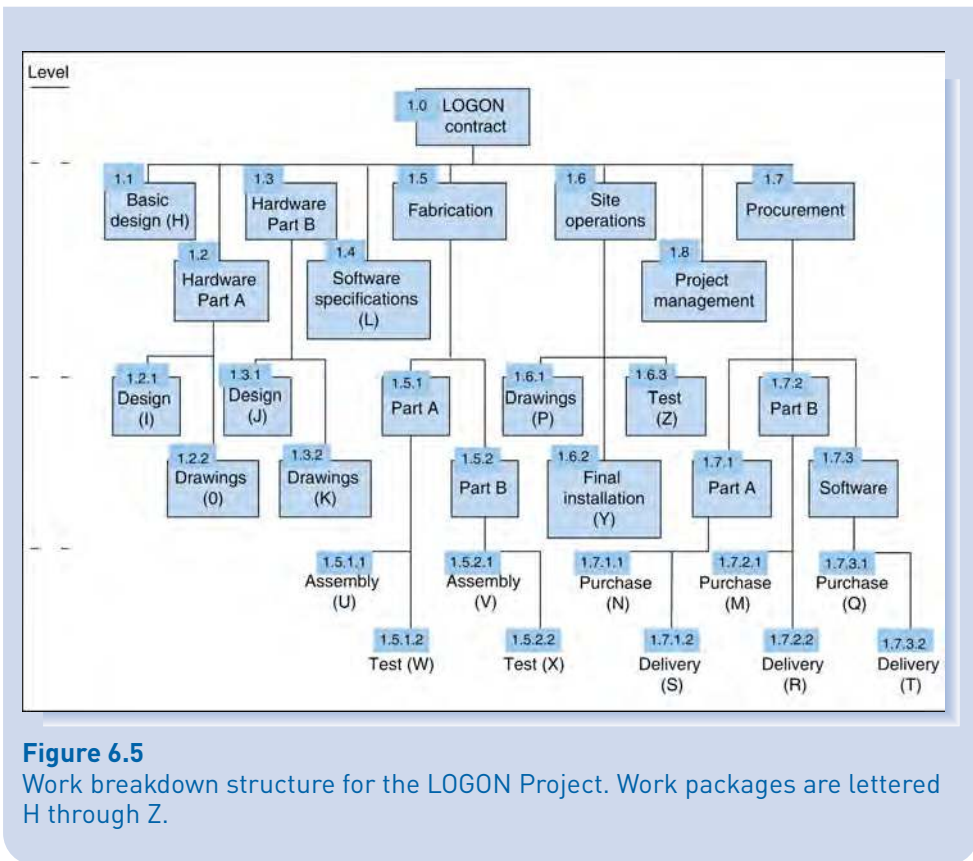


Figure 6.5

Work breakdown structure for the LOGON Project. Work packages are lettered H through Z.

Work packages

Detailed work planning and control occurs at the work package level, and each work package should provide enough information so that it is “well defined.” For a work element to be well defined, ideally it should include the following:

1. **WBS code identifier.**
2. **SOW** for the work task or activity.
3. **Resource requirements:** labor, equipment, facilities, and materials needed for the task.
4. **Time and schedule:** estimated time to perform the task and completion date.
5. **Costs:** estimated resources, management, and related expenses for the task.
6. **Assumptions and constraints.**
7. **Responsibility:** parties, individuals, or job titles responsible for doing and/or approving the task.
8. **Outcomes:** requirements, specifications, and associated deliverables, end-items, or results for the task.
9. **Inputs:** preconditions or predecessors necessary to begin the task.

Inputs	Task	Outcomes
Predecessors	Statement of work	Deliverables
Preconditions	Time	Results
Resources	Cost	
Requirements/specifications	Responsibility	
	Quality assurance	
	Risk	

Figure 6.6
Properties of a work package.

- 10. **Quality assurance:** entry, process, and exit conditions to which the task must conform as specified in the quality plan.
- 11. **Risk:** uncertainty about the time, cost, and resources associated with the task.
- 12. Additional information as necessary, for example safety.

These properties are summarized in Figure 6.6. If any of them cannot be defined for a given work element, then the work element is too broad and must be broken down further. When all or most of the properties can be defined for an element, the element is considered “well-defined” and, by definition, a **work package**. All of the information for a work package would be included in the **WBS dictionary**.

But the level of work breakdown must not continue so far as to result in an unnecessarily large number of work packages. During the project each work package becomes the focal point for planning and control and, as such, involves paperwork, schedules, budgets, and so on. Thus, the more work packages, the more time and cost needed to manage them.

Example 6.4: Work Package Definition for LOGON Project

The LOGON project was divided into 19 work packages—for easy reference the **boxes lettered H through Z** in **Figure 6.5**. Below exemplifies the properties for one of them: Work Package X, Test of Hardware. Note how the properties correspond to those shown in Figure 6.6.

- 1. Code 1.5.2.2
- 2. **Statement of work:** perform checkout, operational test, and corrections as necessary for sign-off approval of four Batman drone transporter units, Model IBR04.
- 3. **Resource requirements:**
 - Labor: FT commitment, 3 weeks for test manager, 2 test engineers, 3 technicians.
 - Procured materials: materials for mockup; all other materials on hand.
 - Facility: test room 1405 at Iron Butterfly for 3 weeks.
- 4. **Time and schedule:** 3 weeks scheduled; (time critical) start December 2; finish December 23.

5. **Costs** (Control account RX0522):

Labor:	Manager, 75hrs + 25% OH	=	\$9,750
	Engineers, 1125 hrs + 25% OH	=	\$135,000
	Technicians, 1125 hrs + 25% OH	=	\$112,500
	Material:		\$70,000
	Subtotal		\$327,250
	10% G&A		\$32,725
	Total		\$359,975

6. **Assumption:** Contractor has successfully completed gen-AI review of transporter.
7. **Responsibilities:** oversee tests: B.J., manager of robotic assembly. Approve test results: O.B., manager of Fabrication Department.
Notify of test status and results: J.M., project engineer; F.W.N., site operations.
8. **Deliverables:** four tested and approved Batman robotic transporters, Model IBR04.
Refer to specifications.
9. **Inputs:** predecessor: assembly of Batman robotic transporter (work package V); pre-conditions: test room setup for robotic transporter.
10. **Quality assurance:** refer to entry, process, and exit conditions for work package X in the LOGON quality plan.
11. **Risk:** RBU will fail test requirements because of assembly/integration problems/errors. Likelihood: low. Contingency reserve: additional week included in the schedule.
12. **Specifications:** refer to test document 2307 and LOGON contract spec sheets 28 and 41.
13. **Work orders:** none, pending.
14. **Subcontracts/purchase orders:** no subcontracts; P.O. 8967–8987 for track tests.

Details for this and all other work packages can be found in the LOGON WBS dictionary.

WBS templates

Companies that routinely perform similar kinds of projects might utilize standardized WBS **templates** at Level 2 or Level 3. The template is based upon experience from many similar kinds of project. Even with a template, however, it is good practice to remember that every project is somewhat unique, and that such uniqueness might not become apparent until Level 3 or 4. Hence, the full WBS for a project should never be a mere template or complete copy of the WBS from another project, no matter how similar the projects seem. Nowhere is the saying “the devil is in the details” more appropriate than in projects, and the WBS is the tool for identifying the details wherein the devil might be.

Ideally work packages represent jobs of about the same degree of effort and of relatively small cost and short duration compared to the total project. For example, DOD/NASA guidelines specify that work packages should be a maximum of 3 months duration and not exceed \$100,000 cost. But these are simply guidelines since work package cost and duration depend on many factors such as project size (smaller projects have smaller work packages).

Each work package represents a contract or agreement with a contractor, supplier, or internal functional unit. Although several functional or contracting units might share responsibility for a work package, ideally each work package has only one party with primary responsibility for it.

Integrated execution project plan and the WBS process

The execution plan is an **integrated plan** where all elements of the plan (Figure 6.1)—requirements, work tasks, schedules, budgets, risk, quality, communications, procurement, and so on—are interconnected, and in the words of the Project Management Institute, are **aligned** with commitments, funding, resources, risks, and stakeholder needs. Once created, the plan enables managers to track the project and assess the impacts of actions or problems in some elements of the plan on other elements.

To better describe what an integrated execution plan is, we can compare it to what it is not, which would be: (1) a list of work packages or tasks seemingly created without regard for user requirements; (2) a budget that does not account for the resources required in the work packages; and (3) a schedule where the tasks do not match up with the work packages in the WBS or budget. To the outsider, it seems that five people who never talked to each other each had produced, respectively, a list of requirements, work tasks, needed resources, a schedule, and a budget. Amazingly, that sometimes happens, with the result being that the requirements, work tasks, resources, schedules, budgets, and so on are seemingly independent and unrelated.

One noticeable feature about an integrated project plan is that the *same list of work packages* or tasks reappears throughout different elements of the plan. The same work tasks developed in the WBS appear in schedules, budgets, and other elements of the plan.

In several ways, the process of creating the WBS and the resultant work packages form the basis for integrated project planning and, later, integrated project control.⁶

1. Managers, subcontractors, and others to be responsible for the project are identified during the WBS process and involved in defining the work. Their involvement helps ensure completeness of work definition and gains their commitment to the work.
2. Work packages in each phase are logically related to those in earlier and later phases; this ensures that predecessor relationships are met and no steps overlooked.
3. Work packages identified in the WBS become the basis for budgets and schedules. The project budget is the sum of budgets for the work packages plus overhead and indirect expenses. The project schedule is a composite of the schedules for the work packages.
4. The project organization is formed around the work packages, with resources and management responsibility assigned to each work package.
5. The project is managed by managing people working on the individual work packages.
6. The project is controlled by controlling the work packages. During project execution, work completed and costs accrued are compared to schedules and budgets for the work packages, indicating which work packages need corrective action.

The creation and use of an integrated project plan is a systems approach to management—recognition that a project is a system of interrelated work elements that must be defined, budgeted, scheduled, and controlled to achieve overall objectives.

6.5 Project Organization and Responsibilities

Integrating WBS and project organization

During the WBS process, each work package is associated with the area in the project organization that will have functional or budgetary responsibility for it. An example is the LOGON project and its contractor, Iron Butterfly Company, represented in Figure 6.7: on the left is the company organization structure; across the top is the project WBS. For contracted project work, i.e., performed by external parties, the organization structure on the left would consist of contractors and suppliers instead of departments. (The matrix can also include other important stakeholders who need to be notified or give approval, e.g., the customer.) The box at the intersection of each department and work package is called a **control account** or **cost account**. Each account represents assigned responsibility for a particular work package or portion of it to a department. Just like a work package, each account includes requirements, a schedule and budget, resource needs, deliverables, and a manager or supervisor responsible for it. Each account integrates the WBS with the project organization and represents an agreement or contract with departments or contractors to fulfill work package requirements. Control



See Chapter 9

Responsibility matrix (RACI)

The individuals responsible for work packages are shown in a chart called a **responsibility matrix**. Figure 6.8 is the responsibility matrix for the LOGON project. Rows represent the work packages or major project tasks identified in the WBS. Columns represent the persons, groups, departments, contractors, or other stakeholders who are responsible for the tasks or need to be notified about project matters. Letters within the matrix symbolize the kind of responsibility: primary (ultimate accountability for the work package); secondary (provide assistance or help); notification (must be informed about the work package's status); and approval (authority to approve or reject work package deliverables). The matrix enables everyone associated with the project to easily see who is responsible for what. This helps avoid people overlooking tasks or shirking responsibility and “passing the buck.” The matrix can be modified to signify any other conceivable kind of responsibility. For instance, a similar kind of matrix is a **RACI chart**, which stands for Responsible, gives Approval, and can or should be Consulted or Informed. (Sometimes A means accountable, although it would seem that anyone who is responsible is automatically accountable, though not vice versa!) Note that for each task in Figure 6.8 one and *only one* person is assigned *primary* responsibility (P). The same should be true for all tasks in all projects.



See Chapter 17 in Chapter 17.

6.6 Estimating and the Systems Development Cycle⁷

Planning and estimating are intertwined processes. Before plans, schedules, and budgets can be created, estimates are needed for how long project activities will take (duration), how much

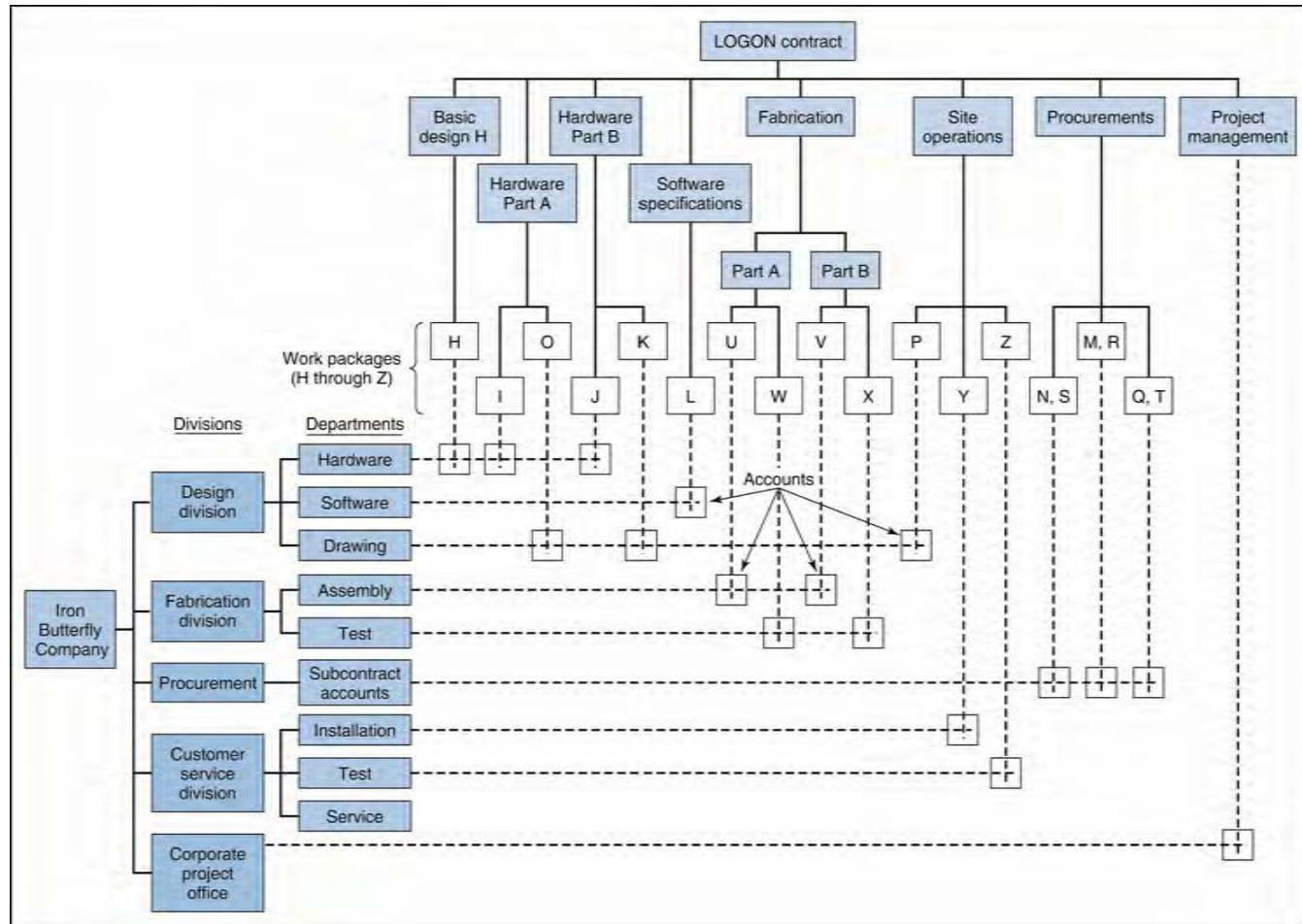


Figure 6.7
Integration of WBS and project organization.



Iron Butterfly, Corp.
Elegant design. Built to last.

Persons Responsible

Responsibility Code

P Primary responsibility
S Secondary responsibility
N Must be notified
A Must give approval

Project Task or Activity	Project Manager																												
	Project Engineer																Fabrication Manager												
	Design						Drawing				Software				Site Operations				Assembly A				Assembly B				B.V.-Purchasing		
	FW.	J.M.	SE.H.	RL.Q.	P.J.	D.V.R.	R.I.P.	Q.E.M.	P.V.P.R.	D.M.N.	RL	LS.F.	LLL.	J.R.S.	D.V.Q.	F.W.N.	J.M.M.N.	LOT.	A.U.A.	D.A.R.	BOB.	EN.	G.G.F.	R.T.T.	B.V.L.	B.J.	T.T.Y.	H.R.D.	
Project coordination	P	S																			S								
Project development	A	P																											
Project design	A	P					A					N				A					N								N
H Basic design	N	A	A	P	S	S	N					N			N														
I Hardware design A			A		P	S																							
J Hardware design B			A	P		S						N																	
K Drawings B							A	S	P												N								
L Software specs	N	A										A	P	S	S														
M Parts purchase B	N																				A								P
N Parts purchase A	N																				A								P
O Drawings A							A			S	P										N								
P Installation drawings							A	P		S					N														
Q Software purchase	N																				A								P
U Assembly A	N																				N	A	P	S	S				
V Assembly B	N																				N					A	P	S	
W Test A	N														N						A	P							
X Test B	N														N						A					P			
Y Final installation	N														A	P	S	S											
Z Final test	N														A		P		S										

Figure 6.8

Responsibility matrix for LOGON project (with initials of persons responsible).

they will cost (expenses), and what will be needed to do the work (resources). Although estimates for the entire project depend on estimates for the work packages that comprise it, sometimes estimates are needed *only* for the entire project. For example, during project conception and *before* a project is approved, project sponsors and customers want to know what the project will cost and how long it will take; these estimates, often based on costs for similar, completed projects, will be used in early discussions and proposals for the project. The question is, given whatever value the project is supposed to deliver, will it be worth the project’s estimated time and cost? The concepts and methods described in this section are applicable to estimating the entire project and individual work activities within it. Chapters 7 and 8 on network scheduling provide additional ways to estimate project duration using the **critical path** and **critical chain** methodologies. Chapter 9 discusses additional matters pertaining to cost estimating.



See Chapters 7, 8 and 9

Although estimates for project time and cost are needed even before the project begins, the estimating process happens throughout the project life cycle. As mentioned, the first estimate is made during project conception. At that time, little information about the project is available, so estimates are the least reliable that they will ever be. Uncertainty about the project cost and duration may be large, as illustrated by the largest “region of time–cost uncertainty” on the **S-curve** in Figure 6.9. How much the project will *really* cost and how long it will *really* take are open to question. As a rule, the better-defined the project requirements and scope, the better the estimates, which highlights the importance of clarifying everything during the Conception and Definition phases (and, for big projects, the so-called front-end loading part, FEL) as described in Chapters 3 and 4. Looking at other, similar (analogous) projects, estimates are made based upon standards of what will be needed in the project to do the job—labor time, materials, equipment, and so on. When there are no similar projects or



See Chapters 3 and 4

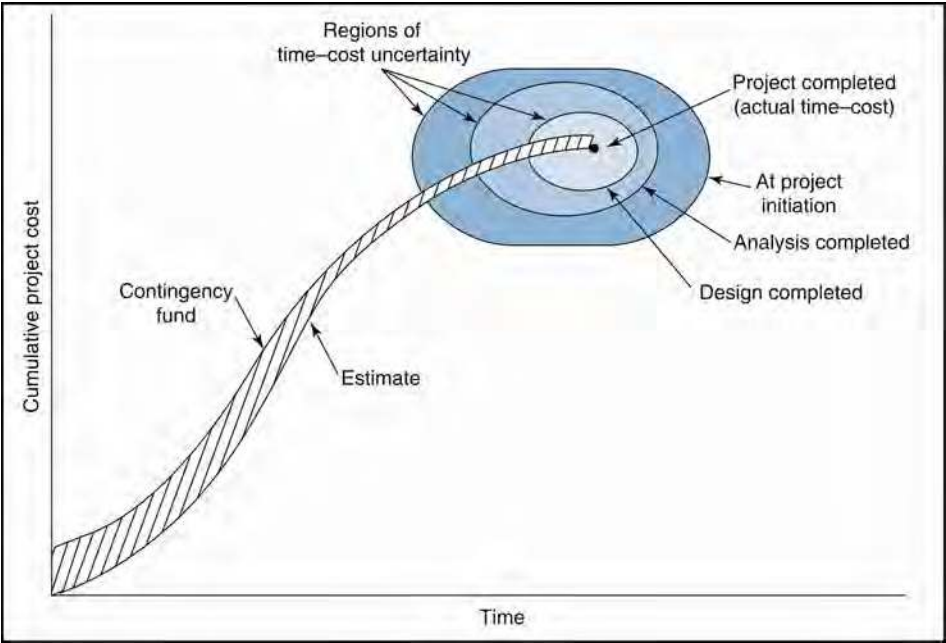


Figure 6.9
Time-cost graph (S-curve) showing cumulative project cost and regions of time-cost uncertainty.

standards, initial estimates are largely “guesstimates” and might end up being nowhere close to actual costs.

If the project is unique and ill-defined, uncertainty in estimates often dictates that contracts are of the cost-plus kind, covered in Chapter 12. As more aspects of the system and project are defined, the material costs, labor times, and labor rates can be nailed down, and estimates become more reliable. When the system and project are routine, the estimates are more reliable, and contractors are willing to accept fixed-price or incentive-type contracts (again, in Chapter 12). Sometimes the awarding of contracts is delayed until designs are more complete and when somewhat accurate estimates are possible. This of course requires contractors to do a lot of front-end work without assurances that they will be awarded the job.



See Chapter 12

Contractors required to bid before they can attain reliable estimates must include a **contingency** (or **reserve**) amount in the estimate to cover the uncertainty, although typically such a fund is included to cover uncertainties even when estimates are considered reliable.

As a project moves into the middle and later phases—as work is being completed and money is being spent, the estimates become more accurate, as reflected in the shrinking time–cost uncertainty regions in Figure 6.9. With decreasing uncertainty, the amount in the contingency fund is reduced. For example, a contingency starting at 15 percent of the base project estimate might be decreased halfway in the project to 8 percent, then to 3 percent at the three-fourths mark, and then to 1 percent near completion to cover minor corrections at project close-out.



See Chapter 4

As discussed in Chapter 4, in the **phased** or **rolling-wave planning** approach a detailed plan is developed only for the most-immediate, upcoming stage of the project, and that plan will include time and cost estimates and commitments for that stage. Concurrently, every attempt is made to look ahead and develop realistic estimates for the remainder of the project.

Once developed and approved, estimates for the project are used to set targets and create schedules and budgets, which become **baselines** against which project progress and performance will be measured. It is thus bad practice to keep changing estimates and targets during the project because that destroys their purpose—to provide baselines against which to measure progress and control time and costs. As described in subsequent chapters, however, sometimes adjustments to schedules and budgets are necessary to account for estimates that were unrealistic, or unanticipated changes occur in the project or the environment.

6.7 Estimating Process

Estimate versus target and commitment

The words “estimate,” “target,” and “commitment” are sometimes misconstrued as synonymous. They shouldn’t be. Whereas an **estimate** is an attempted *realistic assessment* based upon known facts and stated assumptions about the work, resources, constraints, and so on, the **target** is a *goal or desired outcome*, and the **commitment** is a pledge, promise, or contract to deliver an outcome that meets the target. But confusion often occurs. For example, the project manager creates a cost *estimate*, but the customer mistakenly takes it to mean a target and the project manager’s *commitment* to meet that target. Alternatively, a customer lacking the necessary information sets a target completion date and then *expects* the project team to commit to finishing by that date. Other than by chance, the estimate will likely not be the same as the target or commitment. Estimates are based on available information and assume “normal conditions,” e.g., the cost estimate would include the current normal price plus increases from inflation. Targets, initially set by the customer, are based upon either the estimates—if they are felt to be realistic and practical, or by necessities (e.g., to meet a deadline or fixed budget).

Commitments are made when the contractor feels the targets are realistic and can be met. Never should the estimate be simply based on the target, and never should it be assumed to be a commitment.

Accuracy versus precision

Accuracy represents the closeness of an estimated value to the actual value: the accuracy of a \$99,000 estimate for a project that actually costs \$100,000 is very good. In contrast, **precision** is the number of decimal places in the estimate. An estimate of \$75,321 is more precise than one of \$75,000 (though is inaccurate if the actual cost is \$100,000). Accuracy matters more than precision: the aim is to derive the most accurate estimate possible.

Sometimes accuracy can be improved by employing a **three-point estimate**, which is just the average of three estimates—optimistic (a), pessimistic (b), and most likely (m). A variation of this is a *weighted average estimate*, for example E as determined in the PERT approach (described in Chapter 8), which is:

$$E = \frac{a + 4m + b}{6}$$



See Chapter 8

Classifying work activities and costs

When the project scope is well-defined, the cost estimating process begins by breaking the project down into work stages such as design, development, and fabrication, or into work packages from the WBS. The project team, including members from involved functional areas and contractors, meet to discuss the work stages or packages and receive specific assignments.

The team reviews all work tasks in the project and classifies them as either **off-the-shelf (OTS)** or **developmental**. OTS work implies use of something that already exists, meaning it is not new but is an adaptation or repetition of existing or similar systems or activities. Estimating for OTS is straightforward, based on known prices or records of material and labor time and costs. In contrast, developmental work implies starting from scratch, sometimes using trial and error, and where the work and resources needed are uncertain. Overruns in developmental work are common, so it is often beneficial to make use of OTS designs and technology wherever possible.

Estimated costs are classified as **recurring** and **nonrecurring**. Recurring costs happen more than once in the project and are associated with periodically repeated activities such as quality assurance and testing. Nonrecurring costs happen once and are associated with development, fabrication, and testing of one-of-a-kind items or procurement of special items.

For many reasons, project estimates tend to be inaccurate, sometimes for lack of information, other times due to factors that lead to project **escalation** or changes in the project's scope, duration, costs, and outcomes, some of which are discussed in Chapter 9. Often the estimates are inaccurate for lack of information about the project scope and details about the work. In fact, accurate estimates are not possible without a well-defined scope, which is why planning and estimating sometimes happen in a **phased, rolling wave** fashion. A mature design of the project deliverable (product) and a mature project scope definition are necessary for accurate estimates. It is expected that the accuracy of estimates would improve with phases of the project, the worst being at conception, better during definition, and the best at the end of execution, which corresponds to the decreasing time–cost uncertainty shown in Figure 6.9.⁸



See Chapter 9

6.8 Estimating Methods

In very few projects, the project manager does the estimating, but in most projects the estimating is delegated to estimating specialists or to those who will be responsible for the work. The project manager coordinates the estimating efforts, aggregates the estimates, and presents the final estimates to management or the customer. Although this typifies the estimating process, the actual approach depends on the project organization, the information available, and the required accuracy of the estimate. Most estimates come from variants of five methods: expert judgment, analogy, parametric, cost engineering, and Delphi.

Expert judgment

Expert judgment is an estimate provided by an expert—someone who from breadth of experience or expertise can provide a good ballpark estimate. It is a “seat of the pants” estimate with no claim to accuracy, used whenever lack of information precludes more-rigorous cost analysis. Expert judgment is usually restricted to estimates made during the conception phase and for projects that are poorly defined or unique and for which there are no previous, similar projects to compare.

Analogy estimate

An **analogy estimate** is developed by reviewing previous, similar projects. Often such an estimate is handy as a fast reality check for estimates derived from detailed planning, although it is most often used early in the project life cycle before project details are known. The method can be used at any level: overall project time and cost can be estimated from the times and costs of analogous projects; work package time and cost can be estimated from analogous work packages. The time and cost for a similar project or work package are analyzed and adjusted for differences between it and the proposed project or work package in terms of scale, locations, dates, and so on. If, for example, the analogy project was performed 2 years ago and the proposed project is to commence a year from now, the cost estimate must be adjusted to account for inflation and price changes during the 3-year interim. If the analogy project was in California and the proposed project will be in New York, the cost estimate must be adjusted to account for regional differences in labor and material costs. If the scale (scope, capacity, or performance) of a proposed activity is twice that of the analogy activity, then time and costs from the analogy must be “scaled up” accordingly. But twice the size does not mean twice the time and cost, and the size–time and size–cost relationships must be uniquely determined.

Example 6.5: Estimating Project Costs by Scaling an Analogy Project

So-called process industries such as petrochemicals, breweries, and pharmaceuticals use the following formula to estimate the costs of proposed projects:

$$\text{Cost (proposed)} = \text{Cost (analogy)} [\text{Capacity (proposed)} / \text{Capacity (analogy)}]^{0.75}$$

where “proposed” refers to a new facility and “analogy” to an analogous facility. In practice, the exponent varies from 0.35 to 0.9, depending on the kind of process and equipment used.⁹

Suppose a proposed plant is to have a 3.5 million cum (cubic meter) capacity. Using an analogy project for a plant with 2.5 million cum capacity and a cost of \$15 million, the estimated cost for the proposed plant is

$$\$15 \text{ million} [3.5/2.5]^{0.75} = \$15 \text{ million} [1.2515] = \$18.7725 \text{ million}$$

Because the analogy method involves comparisons to earlier, similar projects, it requires historical information about prior projects. Companies that are serious about using the method gather documentation and retain it on a database that classifies time, costs, and other information according to type of project, work package, activity, and so on. When a new project is proposed, the database is accessed for details about similar projects and work packages. Of course, the basic assumption in the analogy method is that the analogy chosen is *valid*; sometimes that is where things go awry.

Example 6.6: A Case of Costly Mistaken Analogy¹⁰

In the 1950s and 1960s when nuclear power plants first appeared in the US, General Electric and Westinghouse, the two main contractors, together lost a *billion* dollars in less than 10 years on fixed-price contracts because they had underestimated costs. Neither had expected to make money on these early projects, but certainly they had not planned to *lose* so much either. The error in their method was assuming that nuclear power plants are analogous to coal power plants—for which the marginal costs get smaller as the plants get larger. But nuclear power plants are not like coal power plants. For one thing, they require more safeguards. When a pipe springs a leak in a coal plant, the water is turned off and the plant shuts down until the leak is fixed. In a nuclear plant the water cannot be turned off and the plant shut down; the reactor continues to generate heat and if not cooled will melt, cause pipes to rupture, and disperse radiation. The water-cooling system needs a backup system, and the backup system needs a backup. Typical of complex systems, costs for nuclear plants increase somewhat exponentially with plant size—but in the early years of nuclear power nobody knew that.

Parametric estimate

Parametric estimates are derived from empirical, mathematical relationships or models. The method can be used with an analogy project to scale costs up or down (Example 6.5), or it can be applied directly without an analogy project when time and costs are a function of system or project “parameters.” The parameters can be physical features such as area, volume, weight, or capacity, or performance features such as speed, rate of output, power, or strength. The method is especially useful when early design features are first being set and an estimate is needed quickly, but its accuracy depends on the amount and accuracy of information it uses:

when the information consists of known facts, accuracy is high; when it consists of rough estimates (i.e., the estimate is based on other estimates), it is uncertain.

Example 6.7: Parametric Estimate of Material Costs

Warren Warehousing Company, a facilities contractor, needs a quick estimate of the material cost of a new facility. Company engineers investigate the relationship between several building parameters and the material costs for eight recent projects comparable in terms of architecture, layout, and construction material. Using the method of least squares (a topic covered in textbooks on mathematical statistics), they develop the following multiple regression model that relates material cost (y , in terms of US \$) to floor space (x_1 , in terms of 10,000 sq. ft.) and number of shipping/receiving docks (x_2) in a building:

$$y = 201,978 + (41,490)x_1 + (17,230)x_2$$

The least squares method for this model indicates that the standard error of the estimate is small, which suggests that the model provides somewhat accurate cost estimates when compared to the actual cost of each of the eight projects.

A proposal is being prepared to construct a 300,000 sq. ft. facility with two docks. The estimated material cost using the model is thus:

$$y = 201,978 + (41,490)(30) + (17,230)(2) = \$1,481,138.$$

Cost engineering

A **cost engineering** estimate is derived from a detailed analysis of individual cost categories at the work package or activity level. (In some countries, cost engineering is referred to as **quantity surveying**.) It often provides the most accurate cost estimates of all the methods, but it also is the most time-consuming and requires detailed work-definition information that, often, is not available early in the project. It begins with project activities or work packages (e.g., as created in the WBS), and divides each into cost categories. For small projects like Example 6.8 the approach is simple and straightforward.

Example 6.8: Cost Engineering Estimate for a Small Project

The project manager is preparing a cost estimate based upon the WBS for the project. Given the eight work packages identified in the project WBS, he creates a Gantt chart schedule (discussed later). Three labor grades will be working on the project; for each work package he determines which labor grades will be involved, and for each grade he estimates the number of labor hours per week needed. The shaded boxes in Figure 6.10, which represent

work packages arranged by date on the project schedule, show the hours per week per labor grade. For example, in weeks 1 and 2 Activity A will need 30 hours of labor grade 1, and 10 and 20 hours, respectively, of labor grade 2.

For each work package he also estimates the cost of material, equipment, supplies, subcontracting, freight charges, travel, and other non-labor costs. Table 6.1 summarizes the labor hours from Figure 6.10 and non-labor costs.

Table 6.1 Labor hours and non-labor costs.

Work Package	Total Hours per Labor Grade			Non-Labor Costs			
	1	2	3	Material	Equipment	Subcontracts	Other
A	60	30		\$ 500			
B		20	60		\$1,000		
C		40			500		\$500
D	105	30			500		
E		60				\$4,500	
F	40	60		8,000	1,000	5,000	500
G	100	40		1,500			500
H		70	40		1,000		1,500
Total	305	350	100	\$10,000	\$4,000	\$9,500	\$3,000

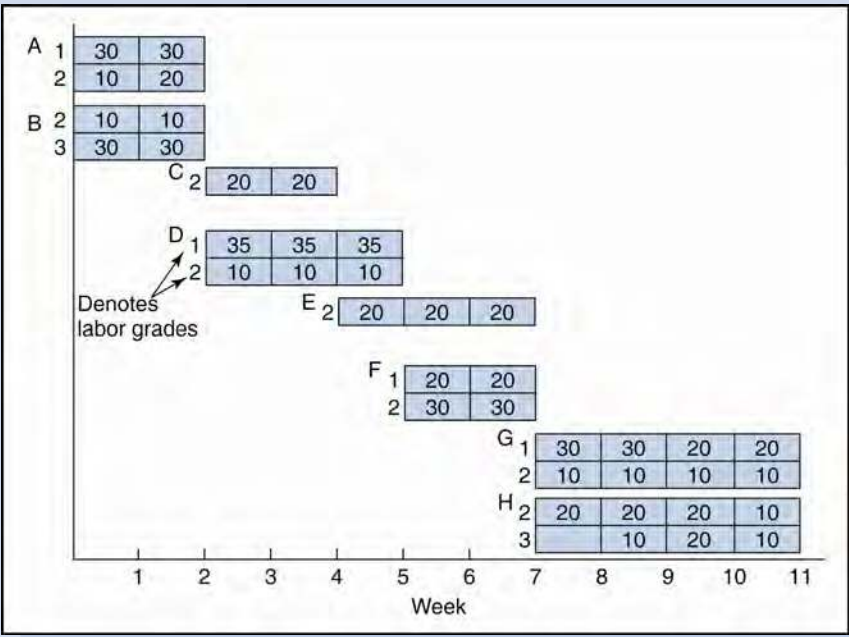


Figure 6.10
Gantt schedule showing hours allocated to work packages by labor grade.

The sum of the non-labor costs is \$26,500. The hourly rates for labor grades 1, 2, and 3 are \$10, \$12, and \$15, and the overhead rates are 90 percent, 100 percent, and 120 percent, respectively. These overhead rates are *added* to the labor cost. (Determining overhead rates is discussed later.) Thus, labor-related costs are:

$$\begin{aligned}\text{Grade 1: } 305(\$10(100\% + 90\%)) &= \$5,795 \\ \text{Grade 2: } 350(\$12(100\% + 100\%)) &= \$8,400 \\ \text{Grade 3: } 100(\$15(100\% + 120\%)) &= \$3,300 \\ &\underline{\$17,495}\end{aligned}$$

Therefore, the preliminary estimate for labor and non-labor costs is \$17,495 + \$26,500 = \$43,995. Suppose the company adds 10 percent to cover general and administrative expenses; this puts the cost at \$43,995(1.1) = \$48,395. If a 10 percent contingency fee is also added, the total cost estimate for the project is \$48,395(1.1) = \$53,235.

At the work package and lower levels, detailed estimates are sometimes derived with the aid of standards manuals and tables, which contain time and cost information about the labor and materials needed for particular tasks. In construction, for example, the numbers of labor hours to install an electrical junction box or a square foot of wall section are both standards. To determine the labor cost of installing junction boxes in a building, the estimator multiplies the required number of boxes times the labor standard per box, and then multiplies that by the hourly labor rate. For software development, an estimator might apply the industry standard of one person-year to create 2,000 lines of bug-free code.

For larger projects, the estimating procedure is roughly the same as illustrated in Examples 6.7 or 6.8 although more involved. First, the manager of each work package subdivides the work package into “basic” areas of work such as “engineering” and “fabrication.” Supervisors in each area then estimate the hours and materials needed to do the work. An engineering supervisor might further divide work into tasks such as structural analysis, computer analysis, layout and installation drawings, and manuals, then for each task develop an estimate of the task duration and the labor grade or skill level involved. Similarly, a fabrication supervisor might subdivide the work to identify the needed materials (steel, piping, wiring), hardware, machinery, equipment, insurance, and so on, then estimate how much (quantity, size, length, weight, etc.) of each will be needed. Estimates of time and materials are determined by reference to previous, similar work, standards manuals, documents from analogy projects, and rules of thumb (e.g., “one hour for each line of code”).

The supervisors submit their estimates to the work package manager who checks, revises, and then forwards them to the project manager. The project manager and professional estimators on the project staff review the submitted estimates to be sure that nothing was overlooked or duplicated, everyone understood what they were estimating and used correct estimating methods, and allowances were made for risk and uncertainty.¹¹ The estimates are then aggregated as shown in Figure 6.11 and converted into dollars using standard wage rates and material costs (current or projected). The project manager then adds in a project-wide overhead (to cover project management and administrative costs) and company-wide overhead (to cover general company expenses) to arrive at a total-project-cost estimate. The accumulation of work package estimates (upward arrows in Figure 6.11) such as this to derive the project estimate is called the **bottom-up** approach.

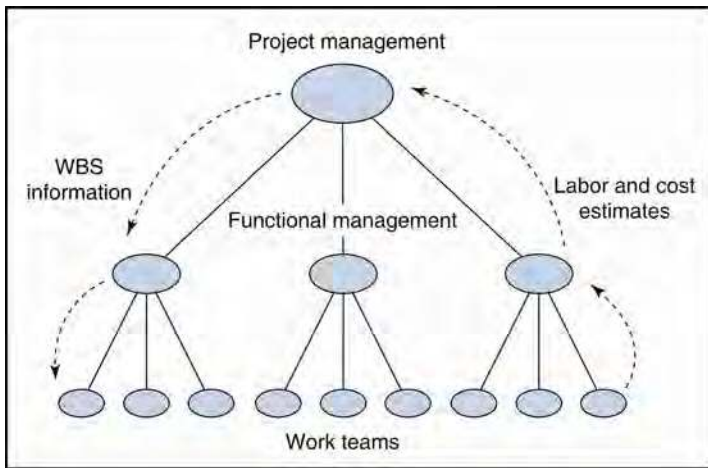


Figure 6.11
The estimating process.

Delphi technique

Delphi technique refers to a group survey technique for combining the opinions of several people into a single judgment. It is an application of **expert opinion** whereby the opinions of multiple experts are combined. The technique comprises a series of structured questions and feedback reports. Each participant is given a series of questions (e.g., how long do you think the project will take to complete and what do you think it will cost?), to which the participant responds, giving opinions and reasons. Everyone's responses are then summarized in one shared report. Seeing others' opinions, the participants are invited to modify their opinions. Because written responses are anonymous, no one feels pressured to conform to others' opinions. If people change their opinions, they must explain the reasons why; if they do not, they must also explain why. The responses are summarized and shared. The process continues until the group reaches a collective opinion about the estimates. In a variation called **wideband Delphi**, only those who provided the highest and lowest estimates in each round are asked to explain why. Studies have proven the Delphi technique to be an effective way of reaching consensus.

Contingency amount

To offset the uncertainty of cost estimates, contingency amounts (also called **contingency fund**, **contingency reserve**, or **management reserve**) are added to estimates and budgets; generally, the more complex or poorly defined the situation, the greater the required amount. Contingency amounts can be developed for individual activities or work packages, or for the whole project. **Activity contingency** is an amount estimated to account for "known unknowns" in an activity or work package, i.e., sources of time and cost increases that could or likely will occur; they include allowance for design changes and increases in the scope, size, or function of the end-item, and delays due to problems such as weather. Later, when the project budget is established, the contingency amount should be placed in a special budget, subdivided

according to work packages, and strictly controlled by the project manager. When the sum of the activity contingencies is added to the project cost, the result is the **base estimate** for the project cost.

Senior managers, the program manager, or sometimes the project manager add yet another amount to the base estimate, a **project contingency**, to account for “unknown unknowns”—external factors that affect project costs but cannot be pinpointed. Examples include unforeseen fluctuation in exchange rates, shortages in resources, and changes in the market or competition. In smaller projects, the project manager controls usage of this contingency; in larger projects, the program manager or senior management does. Adding the project contingency to the base estimate gives the **final cost estimate**, which is the most likely cost for the project.

Besides the activity and project contingencies, the corporation might set aside yet another amount to cover overruns. This amount, the **overrun allowance**, is added to the final cost estimate to yield a cost where, as a rule, the probability of exceeding it is less than 10 percent. The overrun allowance is controlled by the program manager or corporate managers and ordinarily is not available to the project manager without approval.

To account for uncertainty in time estimates, a contingency called a **schedule reserve** or **time buffer** is added to the estimate. For example, for a project estimated to take 6 months, a one-month time buffer might be added to the schedule, so the committed project duration would be 7 months. Everyone would work to complete the project in 6 months, but they have one extra month in case something goes wrong. Schedule reserves and time buffers are discussed in Chapters 7 and 8.



See Chapters 7
and 8

Top-down versus bottom-up

Estimating costs can be done in two ways: top-down or bottom-up. Top-down refers to estimating costs by looking at the project as a whole. A **top-down** estimate is typically based upon an expert opinion or analogy to other, similar projects. **Bottom-up** refers to estimating costs by first looking at elements of the project—individual activities, work packages, and end-item components. The cost for each work package or end-item element is estimated separately, then the costs for all elements are added to derive the total project cost. Example 6.8 is a bottom-up approach; Example 6.5 is a top-down approach. The approaches can be used in combination: portions of a project that are well-defined can be estimated bottom-up; other less-defined portions can be estimated top-down. The cost of each work package can also be estimated either way—by breaking the package into small elements and estimating the cost of each (bottom-up) or by making a gross estimate from analogy or expert opinion (top-down). The bottom-up method provides better cost accuracy than the top-down method but requires more information and is more time-consuming. Important to note is that while project cost can be estimated bottom-up, project duration—summing the durations of the activities in the project—cannot. Ways to estimate project duration are described in the next chapter.

Use of top-down or bottom-up cost estimating corresponds somewhat to the project life cycle. In project conception, the cost estimate might consist of no more than a top-down “ball-park” number to suggest the order of magnitude of the project duration and cost. The method involves little effort, but the estimate is usually of low accuracy and not much confidence is attached to it.

As the proposal is being prepared in the conception phase, the time and cost estimates are often based on the top-down method while looking at previous, analogous projects and compensating for lessons learned and differences between projects. The estimate’s accuracy

depends on the validity of the analogies and ability to distinguish differences and areas of uncertainty.

During the definition phase (and sometimes also during conception, depending upon the availability of reliable data information), cost and time estimates are often prepared using the bottom-up approach. This method provides a fairly accurate estimate of the cost figures needed to establish the project budget and control accounts, discussed in Chapter 9.



See Chapter 9

Reconciling estimates

The project manager submits the final estimate to company management or the project customer, plus forecasts showing the effects of potential escalation factors such as inflation or inclement weather. Management or the customer compares the estimate against its own **gross estimate** (the goal or target) and either accepts it or mandates a revision. If the gross estimate is larger, the project manager reviews the work package estimates for oversights or over-optimism. If the gross estimate is smaller, the project manager looks for ways to reduce the estimates.

Reducing time and costs

What happens if the final estimate must be reduced? No one in the project will want to give up their share of the schedule or budget and lose funding or staff. Non-managers especially (engineers, scientists, systems analysts) are often unaware of the constraints and will resist cutbacks. Through diplomacy and negotiation, the project manager tries to convince everyone to look for ways to reduce the time or costs. Failing that, she must convince them to accept any imposed reductions, such as an across-the-board cut on all estimates. Across-the-board cuts are generally poor practice because they fail to account for judgmental errors or difficulties in just a few aspects of the estimate and penalize managers who tried to produce fair estimates and were honest enough not to pad them. Such indiscriminate cuts induce everyone to pad estimates for their own protection. Sometimes another way to reduce the estimate is to reduce project scope, since cutting work reduces both time and costs.

Suppose you are the project manager, and the gross estimate is too small to do the project. There are two courses of action: either undertake the project and attempt wholeheartedly to meet the targeted gross estimates, or hand the project over to another manager. If you decide the former, you should document and report your disagreement to top management and, later, search for ways to shorten the project or reduce costs to meet the targets. If the contract is cost-plus, additional costs will be reimbursed. If it is fixed-price and the budget is so underfunded as to require cutting corners or stalling the project, then you might recommend that the project be cancelled (or another person be appointed project manager). This is the professional thing to do, although it might cost you your job.

6.9 Scheduling

The next step after work definition and estimating is to *schedule* the project work tasks. A project schedule shows the timing for work tasks and when specific events and project milestones should occur.

Events and milestones

Project plans are similar to roadmaps: they not only show you what you must do to get to where you want to go, but they enable you to see the progress you have made along the way. The roadmap includes signposts called **events** and **milestones** to show how much progress you have made. Passing the final event means you have reached a major milestone: project completion.

Events and milestones should not be confused with work packages, activities, or tasks. A work package or task consumes resources and time; it is the *process* of doing something (such as driving a car to get somewhere). In contrast, an *event* signifies a moment in time. In a project, events represent the *start* or *finish* of something (equivalent to beginning a trip or arriving at a destination).

In project schedules, each work package or task is depicted as a line segment, with the two ends of the segment representing the events of starting and completing the task. For example, in Figure 6.12 the line segment “Task A” represents the task and the time to do the task; events 1 and 2 represent the moments when Task A is started and finished, respectively. In project schedules, each event is associated with a specific calendar date (day, month, and year).

Two particular kinds of events in projects are “interface” and “milestone.” An **interface event** denotes the completion of one task and simultaneous start of one or more subsequent tasks. Event 4 in Figure 6.12 is an interface event: it represents the simultaneous completion of Task B and start of Task C. An interface event often represents a change in responsibility: one individual or group completes a task and another starts a new next task.

A **milestone event** represents a major project occurrence, such as completion of a phase or several critical or difficult tasks, an important approval, or availability of crucial resources. Milestone events are measures of significant progress. Approvals of system requirements, preliminary design, detailed design, or completion of major tests are considered milestones: they signify the project is ready to proceed to the next stage of work. Failure to pass a milestone is usually a bad omen for the project budget and schedule.

Kinds of schedules

Two common kinds of schedules are the project schedule and the task or activity schedule. Project managers and upper management use the **project schedule** (project master or execution

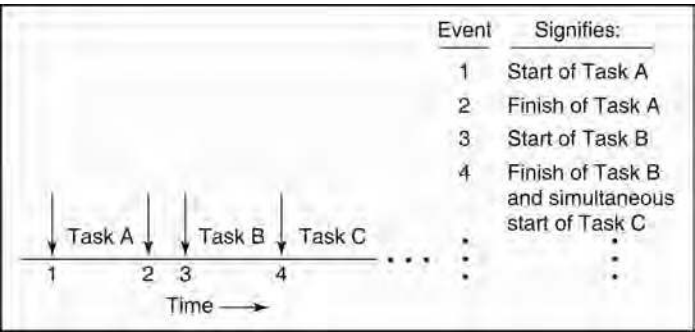


Figure 6.12
Relationship between tasks and events.

schedule) to plan and monitor the entire project. This schedule shows all the major project activities, but not much detail about each. It is first developed during project initiation and refined thereafter. Managers develop the project schedule in a top-down fashion. First they schedule the tasks as identified in the scope statement or WBS. Later, they refine the schedule, taking into account the detailed task schedules as developed by functional managers. When the project is performed in phases, the schedule for each phase must be sufficiently detailed to enable management to authorize work on each phase to begin.

A **task** or **activity schedule** shows the specific tasks or activities necessary to complete a work package. It is created for those working on the activities (lower-level managers and supervisors) and enables them to focus on those activities without being distracted by work with which they have no interaction. Task schedules are prepared by functional managers or subcontractors but incorporate the interface and milestone events shown on the project master schedule. Project and task schedules are displayed in many ways, including Gantt charts.

6.10 Planning and Scheduling Charts

Gantt charts

The simplest and most common scheduling technique is the **Gantt chart** or *bar chart*, named after the management consultant Henry L. Gantt (1861–1919). During World War I Gantt worked with the US Army to find a way to visually portray the status of the munitions program. He realized that time was a common denominator among most elements of a program plan and that it could be used to assess program progress by viewing each element’s status with respect to time. His approach, which came to bear his name, became widely adopted in industry.

The chart consists of a horizontal scale divided into time units—days, weeks, or months—and a vertical scale showing work elements—tasks, activities, or work packages. Figure 6.13 shows the Gantt chart for the LOGON project. Listed on the left-hand side are work packages; along the bottom are work weeks. The beginning and ending of each bar on the chart represent the starting and completion dates of a work package.

Preparation of the Gantt chart happens after a WBS analysis has identified the work packages or tasks to be scheduled. As each work package is identified in the WBS, the manager or others estimate how long it will take and determine its prerequisites or predecessors. The work packages are then listed in time sequence, considering which of them must be completed before others can begin.

As an example, consider the first nine work elements in Figure 6.13 (work packages H through P). Here, as in every project, there is a **precedence relationship** between the tasks—some tasks must be completed before others can begin, and this relationship must be determined before the tasks can be scheduled. (These are the **predecessors** mentioned as “inputs” in the earlier discussion of work package definition.) Suppose that during the WBS analysis for LOGON it was determined that: before work tasks I and J can be started, task H must be completed; before tasks K, L, and M can be started, task J must be completed; and before task N, O, and P can begin, task I must be completed. This is another way of saying that H is the predecessor of I and J, I is the predecessor of N, O, and P, and so on.

Before these can be started. . .	This must be completed
I, J	H
N, O, P	I
K, L, M	J

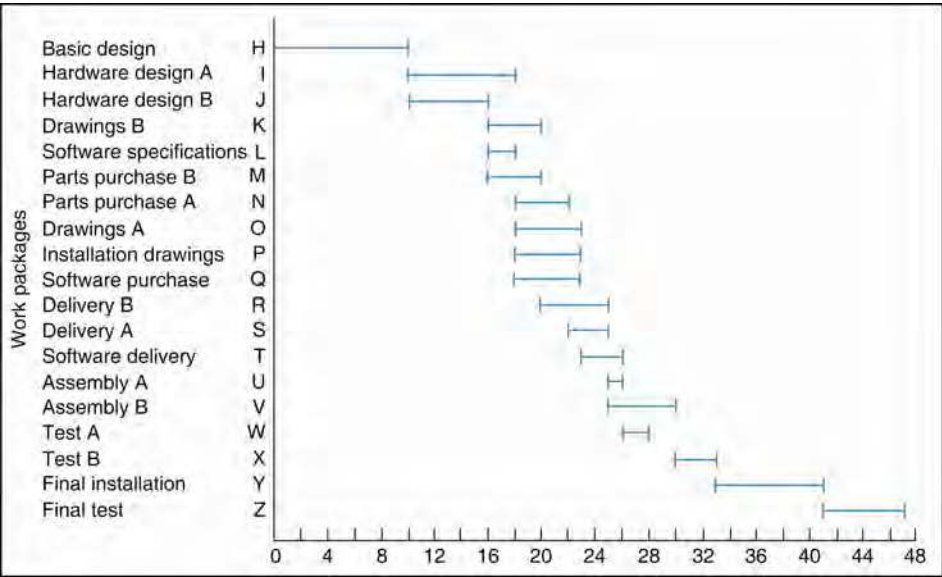


Figure 6.13
Gantt chart for LOGON project.

This sequencing logic must be maintained to create the Gantt chart. Thus, as shown in Figure 6.14 (and given the times shown for the work packages), only after task H has been completed—i.e., after Week 10—can tasks I and J be started; only after task J has been completed—after Week 16—can tasks K, L, and M be started; and, only after task I has been completed—after Week 18, can tasks N, O, and P be started. As each new work task is added to the chart, care is taken to locate it after completing all its predecessor tasks. This example uses work packages as the tasks being scheduled, but any unit of work can be scheduled.

After the project is underway the Gantt chart becomes a tool for assessing the status of individual work elements and the project as a whole. Figure 6.15 shows progress as of Week 20, the “status date.” The heavy portion of the bars indicates the amount of work that has been completed. The thinner part represents work unfinished or yet to be started. This method is effective for showing which tasks are behind or ahead of schedule. For example, as of Week 20 task N is on schedule, task O is ahead of schedule, and tasks K, L, M, P, and Q are behind schedule; L is the furthest behind because it should have been completed but has not been started yet.

When the Gantt chart is used like this to monitor progress, the information it reflects must be the most current possible and updated daily or at least weekly. Tracking progress is important for identifying and rectifying problems, and posting progress is a way to keep the team motivated.

Hierarchy of schedules

In large projects with many work elements, a **hierarchy of schedules** is used, as illustrated by the three levels in Figure 6.16. The top or project-level schedule shows sub-projects within a project, the intermediate-level schedule shows major activities within a sub-project, and the

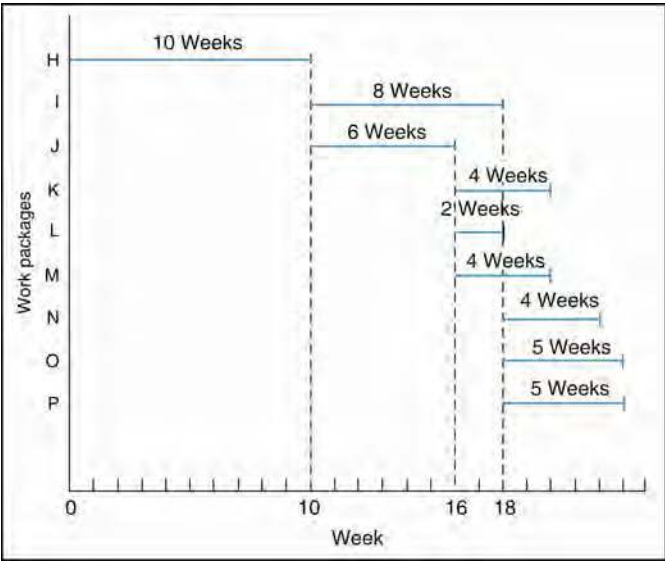


Figure 6.14
Creating a Gantt chart.

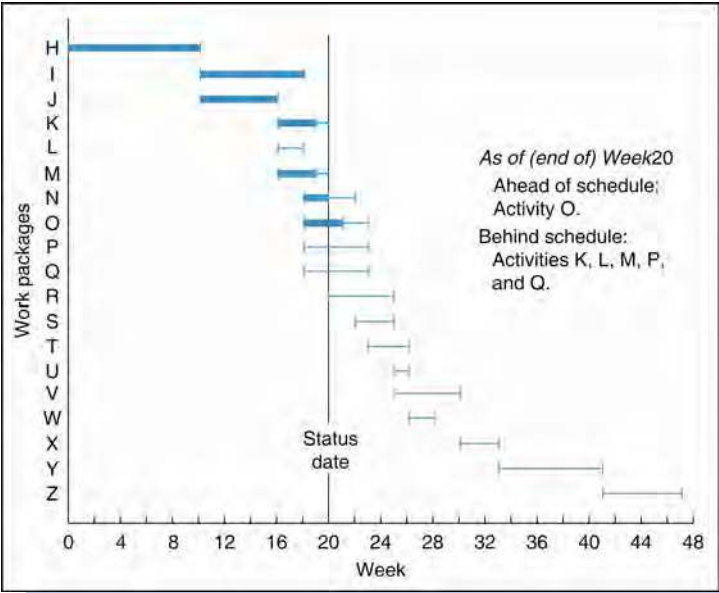


Figure 6.15
Gantt chart for LOGON project showing work progress as of week 20.

bottom or task-level shows work packages or smaller tasks. Milestones and target dates are displayed at any level, though the highest level might be a **milestone schedule**, a simple timeline with milestones.

Each level of schedule expands on the details of the schedule at the level above it. Intermediate- and bottom-level schedules are used for project and functional managers to plan

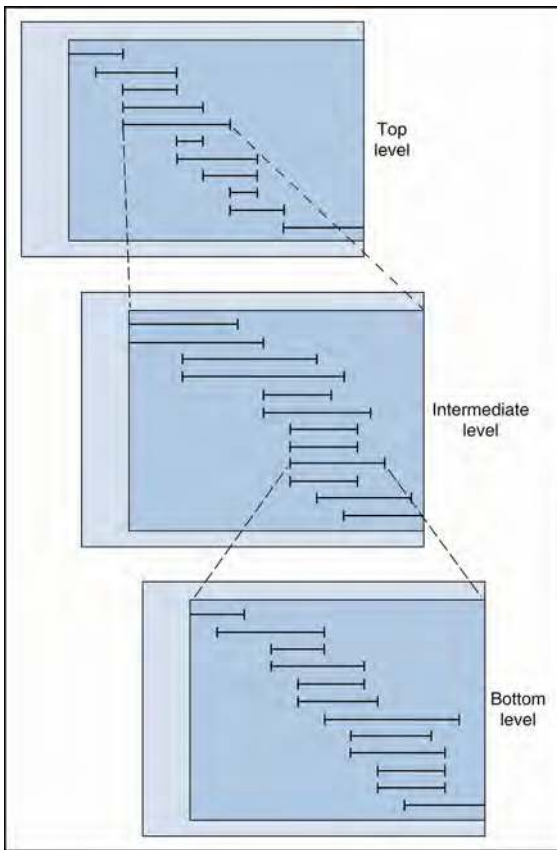


Figure 6.16
Hierarchy of schedules.

labor and resource allocations. Bottom-level schedules are the most detailed, showing the daily (and even hourly) schedules of the tasks within work packages. These are employed by work package leaders and correspond to the task schedules mentioned earlier. Figure 6.17 is a multilevel schedule, showing both the higher-level project activities (denoted by “summary” bars) as well as the detailed tasks within each activity (denoted by “activity” bars).

As a rule, lower-level, more-detailed schedules are created closer to when they are needed and when such details are better known—the phased, rolling-wave, planning approach described in Chapter 4.



See Chapter 4

Disadvantages of Gantt charts

A disadvantage of Gantt charts is they do not necessarily show the dependencies between work elements—for example, the effects of one work element falling behind schedule on other work elements. As described, some work elements depend upon others before they can begin; if those others are delayed then so will the elements that depend on them and, possibly, the entire project. Gantt charts alone provide no way of determining how delays in some work elements impact other elements and the project. This potential limitation is addressed in the next chapter.

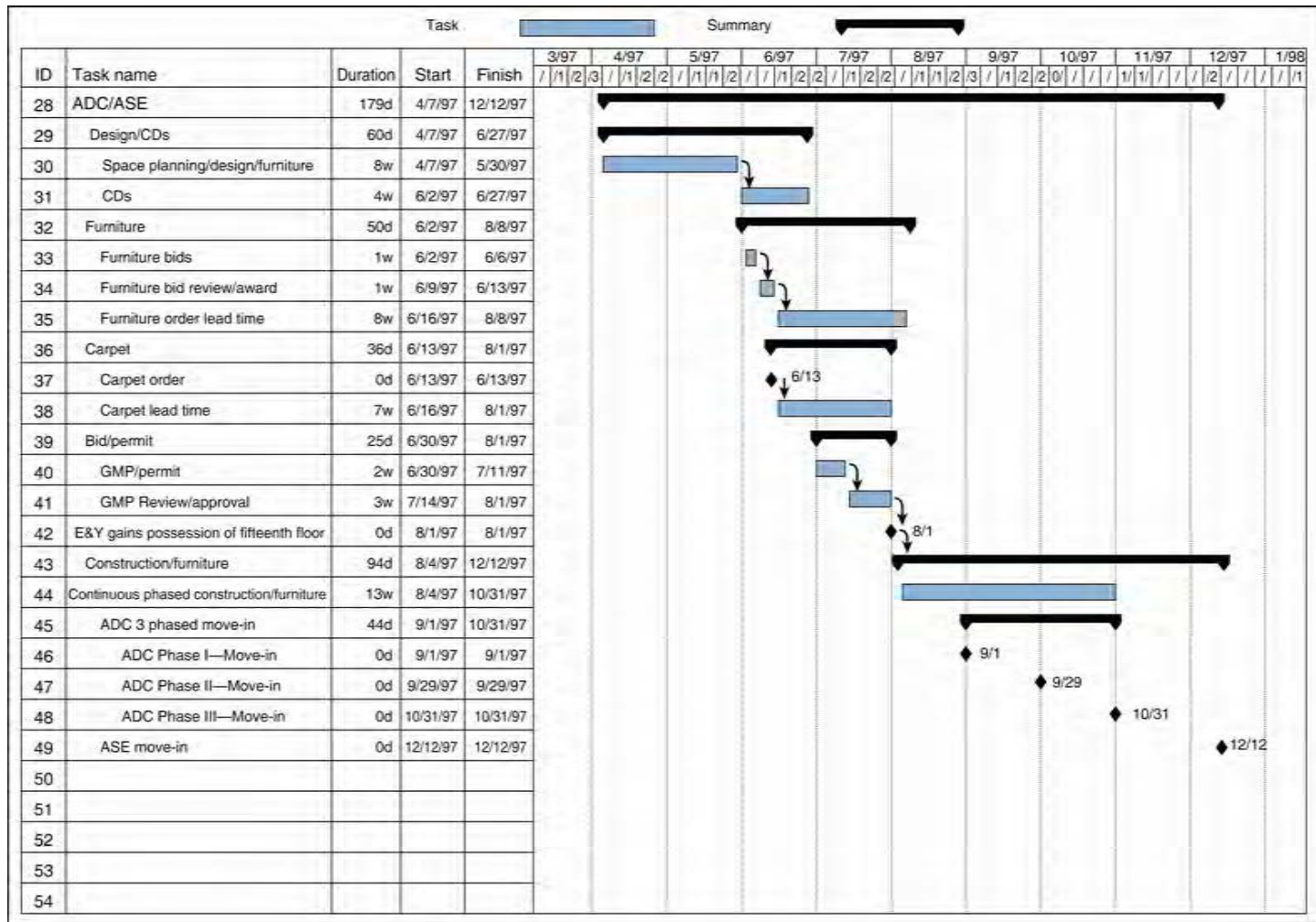


Figure 6.17
Multilevel schedule.

6.11 Summary

The purpose of project planning is to determine the way in which project goals will be achieved—what must be done, how it will be done, by whom, when, and for how much.

The project scope statement and WBS are ways that managers and planners answer the question “What must be done?” The scope statement outlines the main areas of work to be done and the deliverables or end-items. It appears commonly in two places, the SOW or the project charter. The SOW is a summary description of the project used for contracted work; it appears in the RFP, proposal, contract, and project execution plan. The charter is a document used for internal projects to describe, announce, and formally authorize the project.

The WBS process subdivides the project into work packages or other work elements, each small enough to be well understood and planned. Most elements and functions of project management—scheduling, budgeting, resource allocation, tracking, and control—are subsequently carried out with reference to the WBS and work packages.

The responsibility matrix integrates the project organization with the WBS; it prescribes which units and individuals, both internal and subcontractors, have project responsibility and the kind of responsibility for each. It is valuable for achieving consensus, ensuring accountability, and reducing conflict among project participants.

Part of work definition is estimating the time and cost of the project and its work packages and activities. The accuracy of an estimate is partly a function of the stage in the project life cycle when the estimate is prepared; the further along in the cycle, the easier it is to produce accurate estimates. However, good estimates are needed early in the project, and accuracy can be improved by clearly defining the project scope and objectives, and by subdividing the project into small tasks and work packages. In general, the smaller the work element being estimated and the more standardized the work, the greater the accuracy of the estimate. Among estimating methods—expert opinion, analogy, parametric, Delhi, and cost engineering, the last tends to be the most accurate. The cost estimate for the overall project is the sum of the estimates for all the work elements plus overhead costs.

Project schedules show the timing of work and are the basis for resource allocation and performance tracking. Depending on the amount of detail required, different types of schedules are used: project-level schedules show high-level milestones; lower-level schedules show the activities needed to complete individual work packages. The most common form of schedule is the Gantt chart, a visual planning tool that is effective for showing when work should be done and whether work elements are behind or ahead of schedule.

In many projects, software, as discussed in the following appendix, performs planning and control functions such as scheduling, resource management, budgeting, tracking, cost control, and performance analysis. Many utilize web-based technology, which provides the benefits of ready accessibility at remote sites, ease of usage, and reliability and currency of information. Entering the scene is AI to assist project managers with simple daily tasks and in predicting outcomes and making decisions.

The concepts and techniques in this chapter are foundation tools for planning and scheduling. The next two chapters look at additional scheduling techniques. Later chapters address the role of the WBS, work packages, and schedules in budgeting, risk management, procurement, and project control.

APPENDIX: INFORMATION MANAGEMENT AND ARTIFICIAL INTELLIGENCE

The formal planning and control methods described in this book require information that is, or should be, available in any project. What that implies is the need for some sort of system to

collect, organize, store, process, and disseminate information. We refer to that system as a **project management information system (PMIS)**—manual, digital, or some combination.

Software

Methods for project planning, control, change control, and configuration management (topics discussed in later chapters) require processing and integrating hefty amounts of information, especially for large projects. Computers are good at that, so PMIS software has become essential for project planning and control. In fact, without software, it would be difficult to do much of the analysis required to plan and control large projects.

Numerous kinds of PMIS software packages are available that vary widely in capability, flexibility, and price. Simpler PMIS software packages are limited in what they can do but usually are good at whatever that is; once mastered, it is time to upgrade to more sophisticated software.

PMIS capabilities

The following is a rundown of the analytical capabilities, outputs, functions, and features offered by PMIS software. It is important to note that among the many available software packages, most have only some of these capabilities and perform the most basic functions.

Scheduling and resource management

Virtually all project software performs scheduling using methods discussed in the next two chapters. They vary by type of procedure utilized, number of allowable activities, format for activities (some use a WBS scheme), and quality and clarity of outputs (e.g., network diagrams, Gantt charts, tabular reports, etc.). Most perform the basic functions, but they vary in analytical sophistication.

Budgeting

Software varies significantly in the way it handles costs and generates budget and cost summary reports. In some, cost and expense information are not treated explicitly; in others, cost accounting is a major feature. For large projects, software should have a cost and budgeting module (like the project cost accounting systems described in Chapter 9) that is integrated with modules for planning, scheduling, procurement, and tracking.



See Chapter 9

Managing multiple projects and project portfolios

Some software allows data to be pooled from several concurrent projects for multi-project analysis, planning, and control. Some software provides a **dashboard** overview (described later) of all projects so managers can distinguish projects that are performing well from those experiencing problems—an important capability for project portfolio management, discussed in Chapter 19.



See Chapter 19

Cost control, performance analysis, and change control

Here is where project software differs the most. To perform these functions, the topic of Chapter 13, a system must be able to compare actual costs and work completed to budgeted



See Chapter 13

costs and planned work. The most sophisticated software packages “roll up” results and allow aggregation, analysis, and reporting at all levels of the WBS. They also permit updating of plans to reflect actual finish dates and costs, plus help manage the impacts of change requests. Software with simulation capabilities allows managers to integrate network, budget, and resource information and ask “what if” questions under various scenarios.

Some project management software is compatible with and tie into existing databases for payroll, purchasing, inventory, ERP, cost-accounting, and other PMISs; some can be used with popular modeling, risk analysis, data base management, and artificial intelligence (AI) systems. Among available applications of these systems are change control, configuration management, responsibility matrixes, expenditure reports, cost and technical performance reports, and technical performance summaries (all discussed in this book). Many allow access through a browser to various business applications and databases utilizing Internet technology.

Web-enabled project management and Internet of Things¹²

Many software products take advantage of web-enabled technology to display plans and reports on interactive websites. Putting information on a project website or other Internet or intranet network affords the benefits of rapid and easy communication between workers, and immediate access to information that is current and is communicated in real time. The use of web-enabled software to allow people and groups to share knowledge and make decisions through file sharing, wikis, and blogs is sometimes called **project management 2.0** or **PM 2.0**.

With web-browser integrated project management software, teams can report progress and retrieve assignments through their own individual web pages. Managers can aggregate information from scattered worksites to get an overview of the entire project.

Intranets and group productivity¹³

An **intranet** is a private computer network that uses Internet standards and protocols for communication among people within an organization. It provides access to a shared pool of information from computers in the organization and is accessible only by organizational members and other authorized parties, though access can be extended to trusted external parties through an extended network or **extranet**. With an intranet, users can access **group productivity software** to meet and store reports, calendars, and schedules. Information in these documents is easily located using **document-sharing tools** such as file hosting services, newsgroups, chat rooms, and electronic white boards, some of which can be used to share pictorial information about product design requirements and descriptions.

Today, project meetings at desktop locations occur daily via software such as Zoom and Microsoft Teams that allow sharing of spreadsheets, text documents, presentations, graphics, and engineering schematics. Through online **discussion forums** and **chat rooms**, teams share information and view each other’s contributions.

At Boeing, for example, and other engineering firms, designs are stored electronically and are available to anyone who needs them; whenever a design is changed, everyone who needs to know is notified. Everyone with a computer and browser can participate in meetings: engineers at contractors in Kansas having trouble with a mockup send video images to designers in Seattle who can see the mockup, assess the problem, and offer suggestions. Managing technol-



See Chapter 17 **ogy-enabled meetings and virtual teams** is discussed in Chapter 17.

A note about messaging and email: they are important communication tools, but any experienced project manager will tell you that when it comes to making important decisions, there is no substitute for face-to-face or phone meetings.

Social project management software¹⁴

Social project management, SPM, is a response to the global shift to digital workplaces, virtual teams, and social networking at sites like LinkedIn and Facebook. It contrasts with PM 2.0, which enables project team members to participate through web-based tools such as blogs, wikis, and online file-sharing but has limited capability in terms of project management. Among those limitations are (1) users working on different projects require different accounts (and sometimes tools) for each; (2) free web tools are often unavailable for use by companies; (3) the tools do not integrate with company/enterprise software; and (4) the tools have little or no capability to aggregate information across projects.

SPM software accounts for the fact that project teams and members are part of a larger **project community**. It integrates all project teams and their stakeholders so that needed information and relevant knowledge is accessible to everyone. Team members working on different projects do not have to switch back and forth between accounts and tools for each project. SPM software is integrated with the organization's collaborative business platform and linked to its project scheduling software. Everyone has access to necessary information and project schedules.

SPM specifically addresses the importance of engaging the larger project community in a project—the concept of **stakeholder engagement**, discussed in Chapter 16. It maximizes engagement in the same way that social networks do, although unlike social networks, it is structured around the goals of the project community, not of each user or network of friends. SPM takes advantage of the skills and perspectives of individual team members and recognizes that everyone in the project community needs to be aware of each facet of a project and potentially has something important to contribute.

A common way for project managers to provide information to the project community is with **activity streaming**, that is, to constantly stream information about what is happening in and around projects to involved stakeholders. Streaming ensures that even the largest distributed teams have access to current information. The following example shows how.



See Chapter 16

Example A6.1: Dealing with Emergent Issues¹⁵

Before web-based technology, a project team member with an issue would notify the project manager, who would schedule a series of meetings to discuss the issue. With PM 2.0, the team member posts the issue online and requests help from someone (or anyone) to resolve it. She might also request estimates for time, cost, and other impacts. The project manager then reviews the posted solutions, chooses one, and assigns it to someone. With SPM, the team member posts the issue on SPM software, which broadcasts it to the project community; like social media, this produces a flurry of social connections that eventually leads to a (possibly previously unknown) subject matter expert, who posts the solution for everyone to see and comment on. The project manager sees all this in almost real time and makes an informed decision.

Ideally, SPM software is integrated with the organization's other collaboration platforms so users can see and use email, calendaring, social business, and SPM without switching platforms.

Dashboards

Access to information about a project's status and progress is commonly available via desktops and laptops using a so-called **dashboard**, which displays in one place information at a high level about how well all projects underway are doing. By “clicking” on a particular project, the dashboard zooms in to reveal detailed information about project key metrics for schedule and cost performance. Using green, yellow, and red flags, the dashboard highlights whether the project and individual activities are performing well, marginally, or poorly. Responsibility for updating these metrics usually falls to the project office or project management office, PMO,



See Chapter 18 as described in Chapter 18.

Internet of Things

The **Internet of Things (IoT)** is a network of physical items such as mobile devices and electronic gadgets that have sensors, software, and actuators that enable them to communicate, interact, and exchange data with one another via software, which includes the project management software discussed previously, and virtual assistant software, discussed later. Through these devices and software, the project manager knows what is happening, as it is happening, and receives alerts about problems. For example, IoT sensors and devices can:

- Track the location of assets, equipment, and materials to provide information for materials management and trigger alerts when they are moved to unauthorized areas.
- Monitor the weather and provide information to adjust the work environment to improve worker comfort and prevent equipment damage.
- Track equipment vibration, temperature, sound, and other operation indicators to predict equipment maintenance needs.

IoT provides clients, contractors, and other stakeholders with the ability to communicate directly with each other and with IoT devices and software via a common platform.

The advent of IoT and everything else discussed in this section makes one thing clear: the more familiar project managers are with project management software, including web-enabled and internet-based technologies, data analysis, and even AI, the better they will be able to do their jobs. This is true for most project managers, but especially those in large, complex projects.

Artificial intelligence¹⁶

Artificial intelligence refers to computer software that emulates human thought and learning to perform complex tasks. Although AI's usage has been somewhat restricted to large, complex projects to date, its application can be expected to expand as more AI commercial products become available. Delimiting factors in the use of AI include the requirement for large, quality datasets to “train” AI systems and familiarity with how to use those systems. AI encompasses a broad range of methods and applications, many that are practical for both large and small projects.

The two main categories of AI are **applied AI**, which are tools to make predictions and decisions and perform tasks, and **generative AI**, which are tools to create new content such as text, music, and images. The different AI tools are often used in combination. We'll look at applied AI tools first; these include **machine learning (ML)**, **natural language processing (NLP)**, and **rules-based learning (RBL)**.

Machine learning

Machine learning uses mathematical models or algorithms to predict or classify outcomes based on correlations identified in historical data. In project management, the data would be about past projects, and the algorithms would identify patterns in that data. In a sense, the algorithm “learns” by looking at what happened in past projects. Neural network algorithms are commonly used in ML, so-called because they learn in ways similar to the human brain. Once an algorithm has been created and validated, it can be used to predict outcomes for future projects. The algorithms learn in diverse ways: supervised, unsupervised, and reinforcement.

Supervised learning. The algorithms learn from input–output “training data” pairs. The input training data might be features of past projects, and for each project, the output would be the project’s outcome. For supervised learning, each project’s outcome is **labeled**, meaning it has been evaluated as being, e.g., “successful” or “unsuccessful.” The algorithm searches the data to identify project features that differentiate successful from unsuccessful projects, which gives it the ability to predict whether a future project will be successful. Say, for example, the input data for numerous construction projects includes “unexpected buried cable” and for most of those projects the outputs were labeled “project delay.” The algorithm would learn to associate buried cable with project delay, and for future projects would predict a delay whenever buried cable is suspected. The prediction here might be obvious, but ML models can identify patterns and make predictions beyond what humans can.

Unsupervised learning. The training data is not labelled (it specifies project features, actions, or situations, but not outcomes) and the algorithm must develop labels on its own, which it does by clustering similar things together. For example, it might identify from the data a cluster of projects that, in hindsight, far exceeded the budget, even though that was not revealed in the data. For future projects, it would predict that projects with the same features as the clustered ones would also exceed their budgets.

Reinforcement learning. Through a series of trial-and-error actions, the algorithm learns which action sequences best achieve the goal. Current applications of RL are in gaming, autonomous systems like robots, and natural language processing (discussed next) wherein the algorithm learns to “anticipate” spoken or written words and their intent, as explained next.

Natural language processing

Natural language processing (NLP) software treats words as data and uses an algorithm to understand them and offer responses through voice or text. An AI chatbot is the simplest example: it receives spoken queries, converts the words into meaning, interprets the intent, and provides a response. For example, when a person types “RF,” the algorithm will anticipate “RFP” (before the person enters “P”) and what the person wants to know about or do with RFPs. As the person enters more letters and words, the system will refine its anticipation and home in on what the person wants *before the person even enters it*.

Some NLP systems go beyond simple responses and can converse like humans. Some can discern people’s attitudes or emotions from their spoken or written words using so-called “sentiment analysis,” which detects a person’s sentiment—satisfaction, stress, anger, or frustration. This is useful in stakeholder engagement to determine ideal ways to communicate with stakeholders based on their sentiments.

Rules-based learning

Whereas ML relies on a statistical analysis of data, RBL starts with a set of prewritten rules through which the algorithm searches to determine the combination that provides the best result.

For example, given a large set of proposals, an RBL system would select the best of them based on established guidelines and criteria. An advantage of RBL is that the logic behind its results is understandable since the rules it follows are known, which is not the case for ML wherein the logic behind predictions is unknown. A drawback of RBL is that the rules must be specified in advance, whereas ML discovers the “rules” and incorporates (but does not define) them into the algorithm. The previously described PMIS software packages for scheduling, budgeting, and control are all rule-based: they rely on the *logic of mathematics*. The large majority of AI models use ML, not RBL, which, curiously, makes them rather poor at solving problems that require math!

Generative AI

Just as ML models rely on large datasets to identify patterns and make predictions, generative AI (GenAI) relies on large datasets and neural network models (called **large language models** and **deep learning**) to analyze and comprehend text, sound, and pictures. But rather than make predictions, these models *generate* text, documents, pictures, and so on that simulate human language, reasoning, and creativity. For project management, neural network algorithms learn to recognize the data patterns and intricacies in human language and their associated meaning, and to create text that is almost indistinguishable from human-created text.

Data for AI

The training data used by AI software to build algorithms must be the correct data, in the proper format, and clean—i.e., not corrupted, and not irrelevant or anomalous. It must also exist in a readable form with formats and data fields structured to be compatible. For example, data in documents such as lessons learned, risk logs, project control metrics, and change requests from different projects must all be stored in the same format and data fields. Commonly, data used for different purposes and by different systems is stored in various locations and formats that are not compatible; to be accessible and understood by AI tools, it must be cleaned, manipulated, and stored in a common location and format.

For supervised learning, the training data must be categorized and labeled by outcome. Also, for a type of document (e.g., lessons learned, scope statement, issues log, risk log, or change request) it must be consistent across all projects. For example, data about project issues must be the same for every project. Also, the data must be *linked*, meaning that training data from lessons learned, risk logs, budgets, and other documents for a given project must all be linked to that particular project.

For all that data to be consistent and available for use, it must first be captured in the same way for every project, which means the data collection methods and documents for all projects must be the same in terms of content and format. This will require setting and enforcing standards for the content and format of those documents, which usually falls to the PMO.

In general, the more training data the better. Too little data will result in a model that is *underfitted*, which means it is too simple to provide accurate predictions. Organizations that are large or are associated with large organizations have access to ample data and can create their own algorithms. Small organizations might not, although for them, an alternative is commercial AI tools that are already trained and available from developers or on the Internet. This approach, however, is of dubious value when training data for the tools comes from projects irrelevant to those for which the tool will be applied. For example, data from software development projects would not create valid models for wind farm projects.

In addition to historical data, implementing AI for project management requires software for scheduling, budgeting, and so on, as described earlier, to provide the *logic of how things work* in

projects, such as network task dependencies and the critical path. The logic should also consider the interactions and trade-offs among aspects of the project such as schedule, cost, and risk.

AI applications

Currently the most popular applications of AI in project management are prediction, virtual assistance, and text review and generation.

Prediction. Project managers rely on predictions to make decisions in many scenarios; they include:

- **Project screening and selection:** given data about a proposed project, AI can predict the probability that the project will be successful.
- **Project monitoring during execution:** based on current data about a project's performance, AI can assess the probability that the project will be completed on time or on budget.
- **Gating process:** at each gate, AI can predict the probability that the project will pass the next and subsequent gates.
- **Issue management:** during project planning, AI can predict *potential issues* that might arise in a proposed project. Potential issues can then be treated as risks and added to the risk register.
- **Change control:** a change request has a ripple effect whereby the change produces many consequences, which AI can predict much more readily than humans. AI can also predict, for a proposed project, possible future changes that should be addressed in the project plan.

Virtual Assistants. A **virtual assistant (VA)** is NLP software that recognizes speech, attaches meaning to it, and provides a response. Familiar examples are Clickup, Google Assistant, Siri, and Alexa. With a VA tool linked to the project database, a project manager can inquire about a document such as an RFP and get answers without reading the document. When VA software is linked to scheduling software, the project manager can ask about rescheduling a task. Beyond yes or no, the response might suggest whatever is necessary in terms of resources or other tasks to reschedule the task. A VA is also useful for:¹⁷

- Changing tasks on a calendar.
- Getting information that otherwise would require searching a web browser.
- Monitoring the status of projects or individual tasks based on KPI metrics.
- Making and receiving phone calls, emails, or text messages.
- Scheduling meetings and providing reminders.

VA is especially important for virtual and international teams since it operates 24/7, can perform translations, and is available no matter where teams are located.

Text review and generation. Project managers use GenAI to review, summarize, and simplify reports, and to generate text and documents, for example:

- Given the required structure and format, it can create initial drafts of documents such as RFPs, proposals, contracts, and project plans.
- It can condense information from a voluminous source into a concise summary.
- It can create a single document that summarizes information from multiple sources—reports, emails, meeting notes, and so on.
- Provided with data on markets, costs, risks, and benefits, it can create robust business cases.

Bottom line

Many of the project management methods and tools described in this book require time-consuming effort, e.g.: tools for planning—creating requirements, WBSs, schedules, budgets, risk plans, stakeholder plans, etc.—and control—creating status reports, issuing work orders, monitoring KPIs, dealing with risks and issues, handling change requests, etc. This is one reason the tools are sometimes avoided or used poorly. Many involve repetitive tasks that can be automated using software as previously described (e.g., for project scheduling) and do not need AI. The value of AI, says Bourdeau, an AI expert in project management, comes not from speeding up repetitive tasks but in changing the way *entire projects are managed*.¹⁸

In implementing AI, care must be taken to ensure the predictions and suggestions it provides encompass everything important to a project. For example, an ML model might predict that the best way to speed up a project is to execute two tasks in parallel that are currently in sequence. But the project manager must ask, did the model account for whether the activities are on the critical path? Did it consider the resulting impact on risk, work quality, needed resources, stakeholder reactions, and inputs from other activities? To address such questions, the model must incorporate **project logic** as defined, for example, by scheduling software, and incorporate standard project management practices, for example, as prescribed by professional societies like PMI, IPMA, and APM. The model must seek to *optimize the project*, not particular aspects of it, which means it must be linked to software that address multiple aspects of the project—schedule, budget, risk, and so on.

Like all technology, AI comes with potential pitfalls. For example,

- Some VAs are always listening; although intended for use only when activated with a user's voice, they may accidentally capture private conversations or information without the user knowing. The systems must comply with security and data protection regulations.
- Few project managers have the time or ability to use or create AI models, or to provide the necessary data and ensure it is in a usable form. Until commercial AI software advances to a sophistication and ease-of-use level beyond today's, the PMO must provide the software (create or buy one) and the data and support to use it.
- Applications of AI must be integrated with existing project management procedures, and project managers and staff must be trained in these applications.
- AI models learn from training data; if the data is inadequate, incomplete, or biased, AI-based predictions and decisions will be biased or wrong.
- Use of data that is considered personal, confidential, or intellectual or trademarked property is an important ethical and legal matter that pervades AI's usage and, in every case, must be scrutinized.

Fitting the software to the project

In the words of Palla, the purpose of project software is to “get the right information to the right person at the right time so the right decision can be made for the project.”¹⁹ Firms often use multiple kinds of software—say, Microsoft Project for smaller projects and Primavera for large ones. Big companies with big databases also use AI ML tools, but VA tools are used in all-size companies. Some industries have developed special software to meet their special needs, as Example A6.2 illustrates.

Example A6.2: Digital Twin and Building Information Modeling (BIM)²⁰

A **digital twin** is a virtual replica of a physical entity such as a building or community. It uses real-time data and physics-based simulations to provide an operational model of a physical entity that behaves in the same way as its real-world counterpart. A digital twin has many applications from concept and design to continuous monitoring, fault detection, maintenance planning, and testing of new ideas or “what if” scenarios.

Building information modeling (BIM) refers to software for three-dimensional modeling and for building design and construction management. Its purpose is to create a digital model of an entity such as a building, and to integrate this with project management information for use in the building’s construction and upkeep. Architects, engineers, and contractors use it for projects such as highway engineering, urban planning, and rail and offshore facilities.

A key feature of BIM is the plethora of information it provides—information that can be associated with almost every aspect of a project—costs, schedules, and performance ratings. Designers and managers can analyze and preview the physical structure they intend to create, the construction process to be used, as well as management of the structure throughout its life cycle; additionally, project managers can access information to track and control the building’s construction. While BIM methodology requires more upfront analysis, modeling, and planning than other, less-comprehensive design and management approaches, it is claimed to reduce costs, shrink schedules, increase reliability, and improve quality and coordination.²¹

While BIM and digital twins are similar, there are big differences. Real-time sensors and big data, plus physics-based simulation provide a digital twin much greater application over a typical BIM model. A digital twin not only looks like the physical “entity” it represents, e.g., a building, it behaves like the real building and responds to operational conditions and even the building’s occupancy. Thanks to physics-based simulations, it can make predictions for various scenarios and optimize the operational performance of the entity. Also, the scope of a digital twin can extend beyond individual entities. Whereas BIM typically focuses on one structure, a digital twin can encompass multiple interconnected buildings or communities to simulate how they will look and interact in different scenarios. This enables the entities to be better designed, built, and operated.



Review Questions

1. What questions need to be answered whenever a new project is planned? What are the steps in the planning process that answer these questions?
2. What is the purpose of a project execution plan? At what stage of the project should this plan be prepared?
3. Can a project be undertaken without an execution plan? What are the possible consequences?

4. Which aspects of the execution plan might be eliminated for projects with small budgets? Which might be eliminated for short duration projects (few weeks or months) with few tasks?
5. A section for addressing “Risk and uncertainty” is often left out of the project execution plan. What are the potential pitfalls of doing this?
6. What is the purpose of the project scope statement? What information is used to create the scope statement? How is the scope reflected on the WBS?
7. What is the statement of work? In what documents does the SOW appear?
8. What are similarities and differences between the SOW and the project charter?
9. Think of a somewhat complicated endeavor you are familiar with and develop a WBS for it. (Examples: wedding, high school reunion, questionnaire survey, movie, or stage play, etc.). Now repeat for a complicated job you are not familiar with. At what point do you need assistance from “functional managers” or other specialists to continue the breakdown?
10. How do you know in a WBS when you have reached a level where no further breakdown is necessary?
11. Could the WBS in Figure 6.5 have started with different Level-2 elements and still result in the same work packages? In general, can different WBS approaches give similar results?
12. In what ways is the WBS important to project managers?
13. What is the role of functional managers in developing a WBS?
14. What is the impact of altering the WBS after the project has started?
15. What should a “well-defined” work package include?
16. What is the relationship between the WBS and organization structure? In this relationship, what is the meaning of a “control account”?
17. Figure 6.8 shows some possible types of responsibilities that could be displayed on a responsibility matrix. What other kinds of responsibilities or duties could be displayed?
18. Construct a responsibility matrix using the WBS you developed in question 9. In doing this, consider the project organization and the managerial/technical staff to be assigned and their duties.
19. What function does the responsibility matrix serve in project control?
20. Could a responsibility matrix seem threatening to managers and others? Why?
21. Why are accurate cost estimates so important, yet so difficult, in project planning? What are the implications and consequences of overestimating costs? Of underestimating costs?
22. What is the purpose of a contingency fund (management reserve)? How is the contingency fund used and controlled?
23. Describe the different kinds of contingency amounts and the purposes each serves.
24. Explain the difference between a cost estimate and a cost target. What are the problems in confusing the two—in using cost targets as cost estimates?
25. Explain the difference between accuracy and precision. Give two examples that illustrate the difference.
26. For each of the following estimating methods, briefly describe the method, when it is used, and the estimate accuracy it provides:
 - a. expert opinion
 - b. analogy
 - c. parametric
 - d. cost engineering
 - e. Delphi

27. Describe the process of using the WBS to develop cost estimates. Discuss “top-down” versus “bottom-up” estimating. How are work package estimates aggregated into total project cost estimates?
28. What is the role of the functional units and subcontractors in cost estimating?
29. Distinguish an event from an activity. What problems can arise if people on a project confuse these terms?
30. Distinguish an interface event from a milestone event. Give some examples of each. When is an interface event also a milestone event?
31. How are project-level and task-level schedules prepared? What is the relationship between them? Who prepares them?
32. Construct a Gantt chart similar to the one in Figure 6.13 using the following data:

Task	Start Time (wks)	Duration (wks)
A	0	5
B	6	3
C	7	4
D	7	9
E	8	2
F	9	8
G	12	7

When will task G be completed?

33. How must the Gantt chart you drew in question 32 be changed if you were told that C and D could not begin until B was completed, and that G could not begin until C was completed? What happens to the project completion time?
34. Is the Gantt chart adequate for planning and controlling small projects?
35. In a hierarchy of schedules, how does changing a schedule at one level affect schedules at other levels?
36. How do you decide when more than one level of schedule is necessary?
37. If a hierarchy of schedules is used in project planning, explain whether there should be a corresponding hierarchy of plans as well.
38. What is the difference between PM 2.0 and social project management?
39. In what ways can artificial intelligence be used in project management? What are the drawbacks/limitations of AI?



Questions About the Study Project

1. Describe the project execution plan for your project (the plan developed at the start of the project). What is the content? Show a typical execution plan.
2. Who prepared the plan?
3. At what point in the project was the plan prepared?
4. What is the relationship between the execution plan and the project proposal? Was the plan derived from the proposal?
5. Is there a project scope statement? Who prepared it? Do major areas of work and deliverables of the project correspond to the scope statement?
6. Is there an SOW or project charter? Describe its purpose and contents.

7. How, when, and by whom was the work breakdown structure (WBS) prepared? Describe the process used in preparing the WBS.
8. Where in the WBS is project management included?
9. Was the work package concept used? If so, describe what a work package includes. How are work packages defined?
10. How were ongoing activities such as management, supervision, inspection, and maintenance handled in the WBS? Was there a work package for each?
11. How were responsibilities in the WBS assigned to the project organization (i.e., how did the functional areas become involved in the project)?
12. How were individuals assigned to the project? Describe the process.
13. Was a responsibility matrix used? Show an example.
14. How and by what methods were estimates created for project time and cost?
15. How were activities in the WBS transferred to a schedule? Who prepared the schedules?
16. Show examples of project-level and task-level schedules. Who prepared each? How were they checked and integrated?
17. How were project time costs estimated? Who was involved? Describe the process.
18. When did estimating take place? How were estimates checked and accumulated? How were they related to the WBS?
19. How often and when were time and cost estimates revised during the project?
20. How were overhead costs determined? What basis was used for establishing overhead cost rates?
21. How were estimates tallied to arrive at a total project cost estimate? Who did this?

CASE 6.1 BARRAGE CONSTRUCTION COMPANY: SEAN'S WBS

Sean Shawn was recently appointed project planner at Barrage Construction Company, which specializes in custom-made garages. He worked for 2 years in the HR department while completing his MBA and now has a desk in the newly created project office. Barrage is considering branching out to building standard two-car and three-car garages as well as its usual customized garages, and asked Sean to determine the feasibility of moving into this market. Skimming a book on project management he discovered the WBS concept and decided it would be helpful for developing cost estimates for the standard garages. He had never worked on a garage construction project but felt he knew the process from having talked to company employees. He drew the WBS in Figure 6.18. To estimate costs for each work category in the WBS he reviewed company cost records from three recent two-car garage projects that he thought were similar to standard garages, computed the average, and then apportioned the costs among the categories in the WBS. The company had no actual cost records for a three-car garage, so as an estimate he increased the estimate for the two-car garage by 50 percent. When he summed the costs for all the categories, he arrived at a total of \$83,000 for a two-car garage and \$104,500 for a three-car garage. Compared to competitors, he discovered, these costs were 10 percent higher than their prices. However, because his estimates had been based on custom garages, he believed they might be at least 20 percent higher than for standard garages. He thus reduced his estimate by 20 percent and concluded that Barrage would be able to price its garages competitively and still make a 10 percent profit.

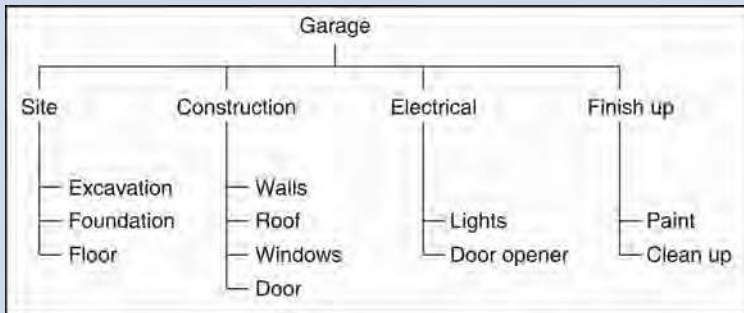


Figure 6.18
WBS for Sean's garage project.

QUESTION

What is your opinion of Sean's approach to creating a WBS and estimating project costs? Please elaborate.

CASE 6.2 STARTREK ENTERPRISES, INC.: DEVA'S PROJECT PLAN

Deva Patel, project manager at Startrek Enterprises, Inc., is planning and coordinating the company's move to a new building currently under construction. Deva wants the move to commence as soon as the building is ready for the estimated June 1 occupancy—still 2 months away. The entire move, which will affect four departments and 600 people, is to be completed within 1 week. Because timing is critical, Deva starts her planning by preparing a Gantt chart. At the project level she draws a bar 7 days long, then subdivides it into three major categories: (1) pack office supplies, equipment, and furniture (3 days allotted); (2) move everything (2 days allotted); and (3) unpack and arrange it at new location (2 days). She then estimates the total number of boxes, equipment, and furniture that will have to be moved in 2 days, gives the estimate to a moving contractor, and receives a price quote. To assist in packing and unpacking boxes and equipment, Deva intends to hire temporary workers. She estimates the number of workers needed and asks a temp agency for a price quote.

Deva shows the completed plan to her manager for review. It consists of the Gantt chart and a budget based on the price quotes from the moving company and the temp agency.

QUESTIONS

1. What do you think about Deva's approach to scheduling work and estimating the costs?
2. If you were Deva's manager, would you consider her plan comprehensive?
3. How would you prepare a plan for the move and what would your plan include?

CASE 6.3 WALTER'S PROJECT PLAN

Walter has just been assigned to manage a project—his first experience as a project manager. The project involves developing an end-item that must meet a lengthy list of requirements, but after reviewing the project SOW and requirements list, the first thing he wonders is who is going to be on his project team. He asks his manager, who gives him the names of three people who are available to work on the project.

Next, Walter starts thinking about what each of the three people on the team will do. He feels that for a project to be successful, team members should each be assigned to tasks where they are the most qualified or experienced. Since he has worked with these people before, he knows a little about their individual expertise. He sits down to prepare a list of tasks for each person; as he considers each person, he thinks about things that need to be done in the project and selects those things he thinks are best suited to the person. When he is finished creating the lists, he sees that person A has 11 tasks and persons B and C have 4 tasks and 5 tasks, respectively. To balance the workload, he takes 4 tasks from person A and splits them between the other two. He is pleased because he feels, with 7, 6, and 7 tasks, respectively, the team members will each have roughly the same amount of work.

On each list he then arranges the tasks in the approximate sequence in which to do them.

The next day Walter meets with the team and gives them the lists of tasks. He asks that they estimate the time they will need to do each of the tasks and that they meet as a group to figure out how their tasks are interrelated and create a Gantt chart. He feels that by requiring team members to estimate task times and create their own schedule, the estimates and schedule will be realistic and accurately reflect the timing of the project.

Walter stops by his manager's office and eagerly reports that his project plan is almost done and will consist of a Gantt chart and lists of responsibilities for project team members.

QUESTIONS

1. Discuss Walter's approach to (a) defining work (creating task lists), (b) creating the schedule, and (c) assigning responsibility.
2. What do you think of Walter's approach to "balancing the workload" among the team members?
3. Do you think the Gantt chart will realistically reflect work that must be done in the project? Do you think the project will be able to satisfy the SOW and requirements?
4. How else might Walter have gone about defining work tasks, creating the schedule, and assigning responsibility?

CASE 6.4 PLANNING THE BOCA IMPLEMENTATION AT KULCZYŃSKI PRODUCTS

Tomasz Grabowski was an IT analyst at Kulczyński Products but has been put in charge of installing the new Boca Business System. This is his first management position, but with 4 years of IT experience he feels confident he can do a good job. He will be leading a team of up to 12 IT professionals, 9 from Kulczyński and 3 from Boca Systems, the software contractor. So far only 3 of the Kulczyński team have been assigned to the project.

To plan for the project, he prepares a detailed list of the tasks he believes needs to be performed, shown below. He is proud of the list and thinks it is quite comprehensive.

Task list

1. Identify resources to be monitored.
2. Define users and workflow.
3. Identify event sources by resource type.
4. Define the relationship between resources and business systems.
5. Identify members of the implementation team.
6. Order the server hardware for production as well as test/quality assurance.
7. Order console machines.
8. Order prerequisite software.
9. Install test and QA servers and prerequisite software.
10. Install consoles and prerequisite software.
11. Install Boca Business Systems Manager on console.
12. Install production servers and prerequisite software.
13. Install console machines and prerequisite software.
14. Configure Boca Console server and verify connectivity.
15. For each resource type, do the following:
 - a. Extend the data model.
 - b. Configure the vector placement in the model.
 - c. Configure the Boca Enterprise Console rule to send events.
 - d. Associate tasks and URLs with object types.
 - e. Configure filtering, if appropriate.
 - f. Verify the event flow.
16. For each business system interface, do the following tasks:
 - a. Test Automated Business Systems file and XML definitions; verify resource inclusion.
 - b. Create databases on the history server.
 - c. Set up and test jobs on the database server to produce the database backup.
 - d. Set up and test jobs to copy backup databases to the history server.
 - e. Set up and test jobs to replicate events to the history server.
 - f. Install your request processor on the Boca Business Systems Manager database server for use by the problem and change request processing function.

- g. Optional: Update the System Configuration table to reflect request processor names along with processing options for the request processors.
 - h. Optional: Update the TLAP table to specify resource options for problem ticket creation.
17. Consider training a key group and have them train their peers.
 - a. Evaluate the addition and deletion of user IDs.
 - b. Establish a relationship between Boca Business Systems Manager and change management. Monitor system performance and adjust hardware as required.
 - c. Boca Business Systems Manager SQL server jobs.

The project is to begin in March and finish by November 30. Tomasz is not accustomed to working according to schedules, but the department manager says he ought to prepare one to make sure the project can be done on time. For the schedule Tomasz decides to create a “timeline” similar to one he saw in another project. He takes that timeline and modifies it to show 11 “work categories” that he believes represent the Boca project. He then estimates how long each work category will take to the nearest half-month. Arranging the categories and times on the timeline, Tomasz finds that the project will finish in January—two months late. He reviews the time estimates and reduces several of them enough to shorten the timeline by 2 months. Figure 6.19 is the finished timeline.

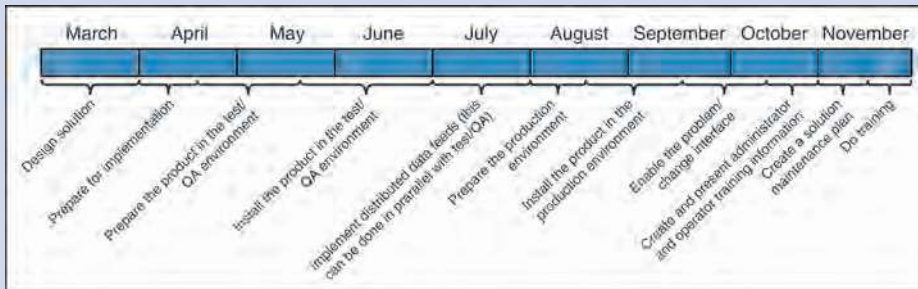


Figure 6.19
Boca implementation project timeline.

Tomasz intends to assign responsibility as follows:

- The Boca team will be responsible for everything on the task list with the word “Boca.”
- The Kulczyński team will be responsible for everything else on the list.

Just before the project begins Tomasz gives the Boca team and Kulczyński team each a copy of the task list and the timeline. He then explains, “Here’s the list of what we need to do. If you can do everything within the timeline, we should be able to meet the project deadline.”

QUESTION

Comment on Tomasz’s “plan.”



Notes

1. Some organizations use the term “project charter” to refer to an execution plan. Our preference is the more common usage, i.e., the charter is a somewhat brief document to announce and authorize the decision to undertake a project, while the execution plan is a comprehensive document to guide the project team through the project execution phase.
2. Contents of execution plans are listed in Cleland D.I. and King W.R. *Systems Analysis and Project Management*, 3rd edn. New York, NY: McGraw-Hill; 1983, pp. 461–469; Allen J. and Lientz B.P. *Systems in Action*. Santa Monica, CA: Goodyear; 1978, p. 95; Kerzner H. *Project Management*, 10th edn. New York, NY: Wiley; 2009, pp. 459–463.
3. See, for example, Cleland and King, *Systems Analysis and Project Management*, pp. 461–469.
4. Seymour Sarason in *The Creation of Settings and the Future Societies* (San Francisco: Jossey-Bass; 1972) argues the importance of knowing the beginnings, origins, and history of any new “setting” before initiating work, and to anticipate and prepare for possible problems, obstacles, conflicts.
5. In technical projects, the subsystems and components—the “configuration items” (CIs)—are identified during preliminary design studies in systems engineering, described in Chapter 2.
6. Cleland and King, *Systems Analysis and Project Management*, p. 258.
7. Portions of this section adapted from Harrison, F. *Advanced Project Management*. Hants, England: Gower, 1981; pp. 154–161.
8. Peurifoy, R. and Oberlander, G. *Estimating Construction Costs*, 4th edn. New York, NY: McGraw-Hill Book Co Inc.; 1989.
9. Dingle J. *Project Management: Orientation for Decision Makers*. London: Arnold/John Wiley & Sons; 1997, p. 105.
10. Pool R. *Beyond Engineering: How Society Shapes Technology*. New York, NY: Oxford University Press; 1997; Heppenheimer T.A. Nuclear power. *Invention and Technology* Fall 18(2); 2002: 46–56.
11. A complete discussion of the cost review procedure is given by Kerzner H. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, 10th edn. New York, NY: Van Nostrand Reinhold; 2009, pp. 592–595.
12. Portions of this section were prepared with the assistance of Elisa Denney.
13. Palla R. Introduction to micro-computer software tools for project management, *Project Management Journal* August; 1987: 61–68.
14. *Social Project Management: Engaging the Social Network to Deliver Project Success*. Trilog Group. August, 2012. <https://www.triloggroup.com/socialprojectmanagement/> Accessed Feb. 2, 2024.
15. *Social Project Management*, pp. 6–7.
16. Parts of this section are derived from Boudreau, P. *Applying Artificial Intelligence to Project Management*. Self-published; 2019. For the technically inclined, AI algorithms are explained in Russell S. and Norvig, P. *Artificial Intelligence*, 3rd edn. Upper Saddle River, NJ: Pearson Education; 2010.
17. Yasar K. and Botelho B. *What is a virtual assistant (AI assistant)?* TechTarget, <https://www.techtarget.com/searchcustomerexperience/definition/virtual-assistant-AI-assistant>. Accessed Feb. 10, 2024.
18. Boudreau, *Applying Artificial Intelligence to Project Management*.
19. Palla *Introduction to micro-computer software tools*.
20. Green E. BIM 101: What is building information modeling? *Engineering.com*, February 3, 2016. <https://www.engineering.com/bim-101-what-is-building-information-modeling/>; Harvard University Construction Management Council. *Introduction to Building Information Modeling*. <https://www.scribd.com/document/410004849/Harvard-Bim-Intro-1996>. Accessed July 3, 2024.
21. A leading case study is the stadium design for the Singapore Sports Hub. See *Coliseum*, Global Sports Venue Alliance. <https://www.coliseum-online.com/singapore-sports-hub-pioneer-using-bim> Accessed June 23, 2024.

Chapter 7

Project Schedule Planning and Networks

Project scheduling is an integral part of project planning, a sometimes trial-and-error process of adjusting work tasks to satisfy resource constraints while trying to meet project deadlines. Although a Gantt chart is good for communicating the project schedule, as a planning tool it is limited because it does not explicitly show the impact of delaying activities or shifting resources on the overall project. The network methods described in this chapter do not have this limitation; they clearly show what happens to the project when resources are altered or activities delayed. This chapter and the next discuss the most widely used network-based approaches to project scheduling and planning.

7.1 Network Diagrams

A **network diagram** shows project activities or tasks and their logical relationships—i.e., the *precedence* relationships or *dependencies* among the tasks. Figure 7.1 is a network diagram for “getting up and getting dressed” (for a male). The boxes represent activities or tasks, and the arrows between them show how the tasks are related and the order in which they should occur, e.g. put on shirt *before* tie, put on pants and socks *before* shoes, etc. (The diagram in Figure 7.1 is of course for illustration only; any such attempt to plan work in such detail would be a real time-waster!) Ordinarily the boxes in the network would be the activities or work packages as defined in the work breakdown structure (WBS). Depending on the desired detail, however, they can represent any level of work, including projects in a program or tasks in a work package.

Networks can also show events. As described in Chapter 6, an **event** represents an instant in time, an “announcement” that something has happened or will happen. Typically it signifies the start or the end of an activity. (An activity with a very short duration may also be regarded as an event.) An important event such as completion of a project phase is a **milestone**.

Two methods for constructing network diagrams are **activity-on-node (AON)**—also called the **precedence diagramming method (PDM)**—and **activity-on-arrow (AOA)**. Our discussion will center on the more commonly used AON method. The AOA method is addressed in Appendix A to this chapter.



See Chapter 6



See Appendix A

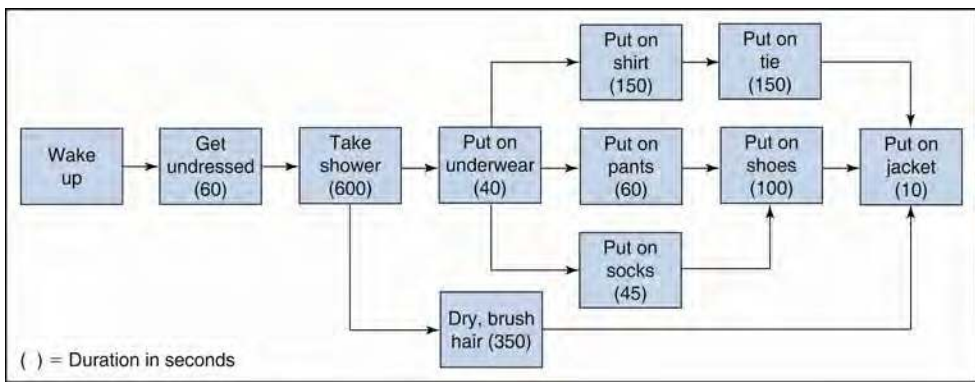


Figure 7.1
Network diagram for getting up and getting dressed.

Activity on node (AON) diagrams

Figure 7.1 is an AON network. Each box (or **node**) in the network is a work package, task, or activity, and inside each box is information about the activity.

To construct an AON network, start by drawing the first activity in the project (e.g. “wake up”). From that activity draw arrows to activities that happen next. As shown in Figure 7.1, activities are added one after another, in sequence or parallel.

Before you can actually create a network, you first must know the **dependency** relationships among the activities. For example, for each activity,

- What activities are its predecessors?
- What activities are its successors?
- What activities can be done at the same time as it?

In a network, every activity except the first has **predecessors**, which are activities that must precede or be completed ahead of it; in Figure 7.1, for example, “put on shirt” is a predecessor for “put on tie.” Similarly, every activity except the last has **successors**, which are activities that cannot begin until the current activity is completed; for example, “put on tie” is a successor of “put on shirt.”

There are four kinds of dependency relationships:

- **Mandatory:** the relationship between the activities cannot be reversed and the dependency cannot be eliminated. The relationship between “put on socks” and “put on shoes” in Figure 7.1 is an example.
- **Discretionary:** the relationship is a matter of choice. For example, “dry and brush hair” and “put on tie” *could* be done in either order. Sometimes a discretionary dependency can be eliminated and activities overlapped to speed up the project. This is called *fast-tracking*.
- **Internal:** the relationship is entirely between activities within the project and might be alterable. All the relationships in Figure 7.1 are internal.
- **External:** the relationship is between an activity in the project with factors, work, conditions outside the project and not on the WBS.

Sometimes only the **immediate predecessors** are used to construct the network. An immediate predecessor is an activity that *immediately* precedes another activity. For example, “wake up” and “get undressed” are both predecessors for “take shower” but only “get undressed” is an immediate predecessor. (The logic is, to “get undressed” you must “wake up” first.)

Given information on immediate predecessors, it is easy to construct a network. For example, the network in Figure 7.1 can be constructed solely from the information in Table 7.1. Start with the first activity in the project, the one with no immediate predecessor, which is “get undressed”; then connect to it the activity that has “get undressed” as its immediate predecessor, which is “take shower.” Next come “put on underwear” and “dry, brush hair” since “take shower” is their immediate predecessor. Continuing in this fashion, the result is the diagram in Figure 7.1.

Once the network is constructed it is easy to see which activities are sequential and which are parallel. Activities that have a predecessor–successor relationship (one follows the other) are called **sequential activities**; “take shower,” “put on underwear,” and “put on shirt” are examples. Two or more independent activities that can be performed at the same time are called **parallel activities**. “Put on shirt,” “put on pants,” “dry, brush hair,” and “put on socks” are parallel activities because they can be done all *at the same time* and in *any order*. Once the network has been completed, check the relationships among activities for completeness and logical consistency.

Another example is in Table 7.2. The network diagram for this project, shown in Figure 7.2, begins at Activity A (no immediate predecessors). Since Activities B and C both have A as their common immediate predecessor, both are connected directly to A. Then, because activities D and E have two immediate predecessors, B and C, they are connected to both.

In general, good practice dictates that a network should always have only one “start” and one “end” node, each a single place on the network to represent the start and end of the project. Whenever a project has multiple nodes at the start or end of the network, then a single node should be inserted before or after them, respectively. In Figure 7.2, for example, a single end node (with implied zero duration) has been inserted after Activities D and E. Without this node the mistaken understanding might be that the project ends upon completion of either Activity D or Activity E. The “end” node means that the project ends when *both* D and E are completed.

Table 7.1 Activities and immediate predecessors for Figure 7.1.

Activity	Immediate Predecessors	Duration (Seconds)
Get undressed	–	60
Take shower	Get undressed	600
Put on underwear	Take shower	40
Dry, brush hair	Take shower	350
Put on shirt	Put on underwear	150
Put on pants	Put on underwear	60
Put on socks	Put on underwear	45
Put on tie	Put on shirt	150
Put on shoes	{ Put on pants Put on socks	100
Put on jacket	{ Put on tie Put on shoes Dry, brush hair	10

Table 7.2 Activities and immediate predecessors for Figure 7.2.

Activity	Immediate Predecessor
A	–
B	A
C	A
D	B, C
E	B, C

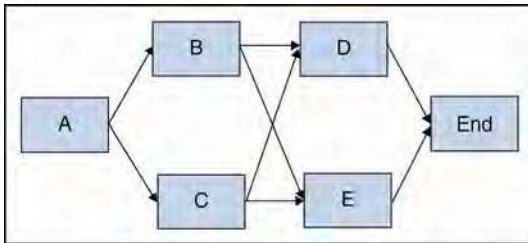


Figure 7.2
AON diagram corresponding to project in Table 7.2.

As a final example, Table 7.3 shows the immediate predecessors for the LOGON project using work packages from the WBS in Chapter 6. Figure 7.3 shows the corresponding network.

Tables 7.1, 7.2, and 7.3 for the examples show only the immediate predecessors for each activity. While it would have been okay to show all the predecessors for each activity in these tables, it would have been unnecessary. For example, had Table 7.2 shown all the predecessors for Activity D (A, B, and C), it would have been correct but also unnecessary because A is the predecessor for B and C, and, hence, listing A would have been redundant. The point: once dependencies have been thoroughly checked, only the immediate predecessors for each activity are needed to construct a network.



See Chapter 6

Creating a project network

Two methods for creating a project network are by hand and by computer. If done by hand, there might be several attempts and drafts. Even if done by computer, it is good practice to first sketch the network by hand, then proceed to enter the data into a computer. This affords the project manager an intuitive “feel” for the project. The activities can be clustered into higher-level subnetworks that represent, for example, sub-projects, or project phases. Project phases are normally conducted in series, although, as mentioned, discretionary dependencies can be eliminated so phases can be overlapped and **fast-tracked**. Even when phases overlap, however, it is still necessary to define their start and end points so the phases can be authorized and milestone payments approved.

In lengthy projects the network might show detailed activities only for the early phases, and rough clusters of activities for later phases. As a phase moves toward completion, details for activities in the next phase are added. This phased approach (called **rolling wave planning**

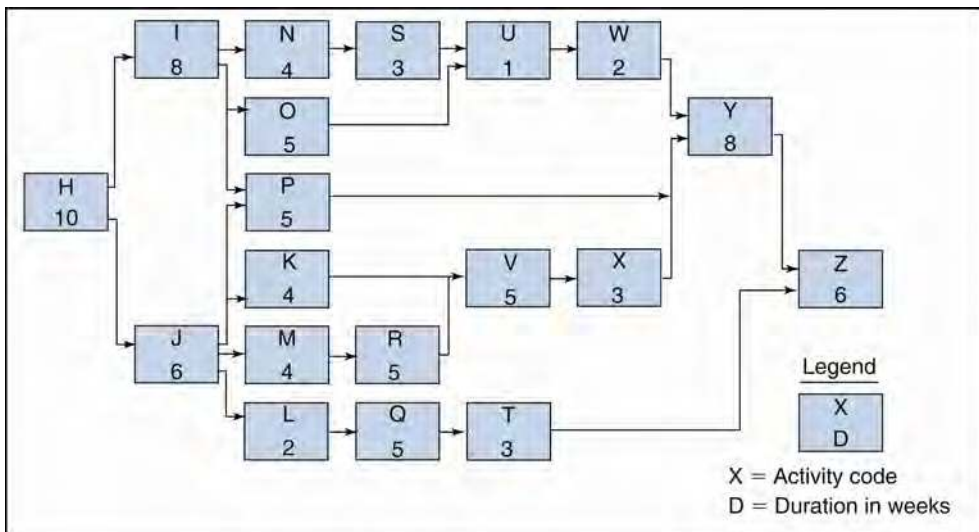


See Chapter 4

Table 7.3 Activities and immediate predecessors for LOGON project.

Description of Work Packages*	Immediate Predecessors	Duration (Weeks)
H Basic design	–	10
I Hardware design for A	H	8
J Hardware design for B	H	6
K Drawings for B	J	4
L Software specifications	J	2
M Parts purchase for B	J	4
N Parts purchase for A	I	4
O Drawings for A	I	5
P Installation drawings	I, J	5
Q Software purchases	L	5
R Delivery of parts for B	M	5
S Delivery of parts for A	N	3
T Software delivery	Q	3
U Assembly of A	O, S	1
V Assembly of B	K, R	5
W Test A	U	2
X Test B	V	3
Y Final installation	P, W, X	8
Z Final system test	Y, T	6

*Work packages from WBS, Figure 7.3.

**Figure 7.3**

Network diagram for LOGON project.

or **progressive elaboration** and discussed in Chapter 4) reduces the complexity of the network for a large project.

Computer software for creating networks is a convenience in small projects but a necessity in large projects. The resulting network should be reviewed for accuracy, omissions, and mistakes. As a rule, the network should be created only after a suitable scope statement and WBS have been developed (i.e., the list of work tasks should be created before—not while—the network is created). Afterward, the network can be used to create a Gantt chart, as explained later.

7.2 The Critical Path

Project networks are important tools for project planning and control. They are useful for determining *how long* the project will take (the **expected project duration**), when each activity should be scheduled to start and finish, and the *likelihood* of completing a project on time.

In general, the **expected project duration**, T_e , is determined by finding the *longest path* through the network. A “path” is any route comprised of one or more activities connected in sequence. The *longest path* in the network from the start node to the end node is called the **critical path**, and its length is the expected project duration. Should any activity that forms part of the critical path (called a **critical activity**) take longer than planned (due to delays, interruptions, lack of resources, etc.), the entire project will also take longer than planned.

This concept is illustrated in the following example. The firm of Kelly, Applebaum, Nuzzo, and Earl, Assoc. (KANE) is working on the Robotics Self-Budgeting (ROSEBUD) project. Figure 7.4 shows the network. For now, we will only refer to part (a) of Figure 7.4. The first phase in the project is systems design (Activity J), followed by the two parallel phases of

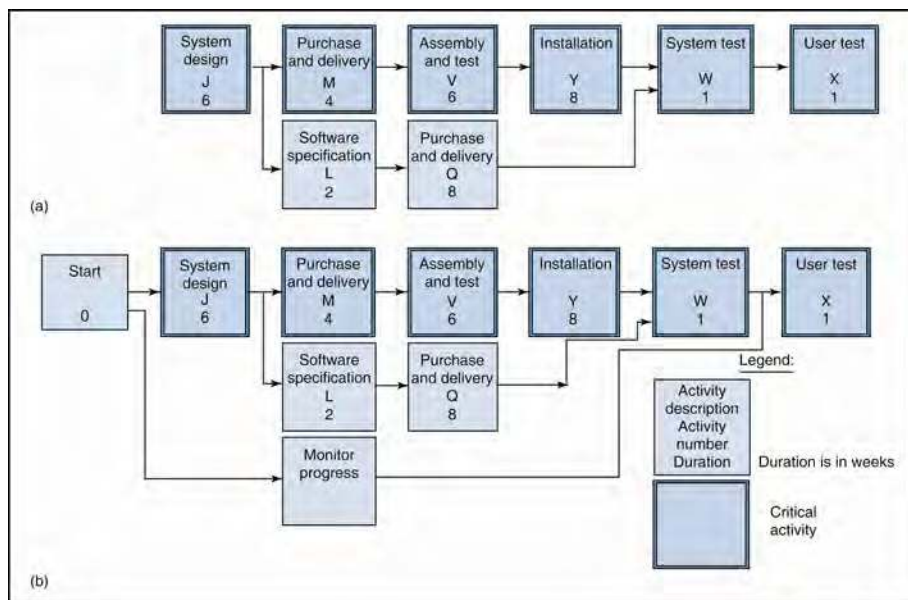


Figure 7.4
Network diagram for ROSEBUD project.

(1) purchase, assembly, and installation (Activities M–V–Y), and (2) software specification and purchase (L–Q). Both are followed by the last phase, system test and user test (W–X).

How long will this project take? The first activity, J, takes 6 weeks; after J has been completed, activities on the paths M–V–Y and L–Q can begin. The activities on path M–V–Y will take $4 + 6 + 8 = 18$ weeks, and the activities on path L–Q will take $2 + 8 = 10$ weeks. Because Activity J takes 6 weeks, path M–V–Y will be completed $6 + 18 = 24$ weeks, and path L–Q will be completed in $6 + 10 = 16$ weeks. The diagram implies that for Activity W to begin, both Activity Y and Activity Q must be finished. Thus, the earliest Activity W can begin is after 24 weeks. Activity W will be completed 1 week later, and Activity X will be completed 1 week after that. Thus, the ROSEBUD project duration (denoted as T_e) is $T_e = 24 + 1 + 1 = 26$ weeks.

Notice again from Figure 7.4(a) that there are two paths from the start node (J) to the end node (X). The shorter path J–L–Q–W–X is 18 weeks; the longer path, J–M–V–Y–W–X, is 26 weeks. In general, **the longest path—the critical path**—gives the project duration. The critical path is highlighted in Figure 7.4, and activities with the darker “framed” boxes are critical.

If it becomes necessary to reduce the project duration, any reduction in effort (i.e. reducing the time for an activity) must happen on the critical path. Shortening any critical activity by, say, 1 week would have the effect of shortening the project duration by 1 week. In contrast, shortening activities not on the critical path would have no effect on project duration. For example, had either L or Q been reduced by 1 week, then the Activity Q would be completed in Week 15 instead of Week 16; but since Activity W must still wait on completion of Activity Y—which won’t happen until after Week 24, there would be no change in project duration.

As mentioned, the critical path is important for another reason: any delay among the activities on the critical path will result in a delay in the project completion. Should any critical activity be delayed by, say, 1 week, the project completion will be delayed by 1 week. Note, however, that noncritical activities can be delayed somewhat without delaying the project. In fact, in the example, noncritical activities L and Q can be delayed by up to 8 weeks. This is because normally they will be completed in 16 weeks, which is 8 weeks earlier than the activities on path M–V–Y, or 24 weeks. In other words, although activities on path L–Q can be completed as early as 16 weeks, it is okay if they are completed as late as 24 weeks. Thus, the critical path shows the project manager which activities are most critical to completing the project on time. To prevent delays, the project manager should focus on the critical activities.

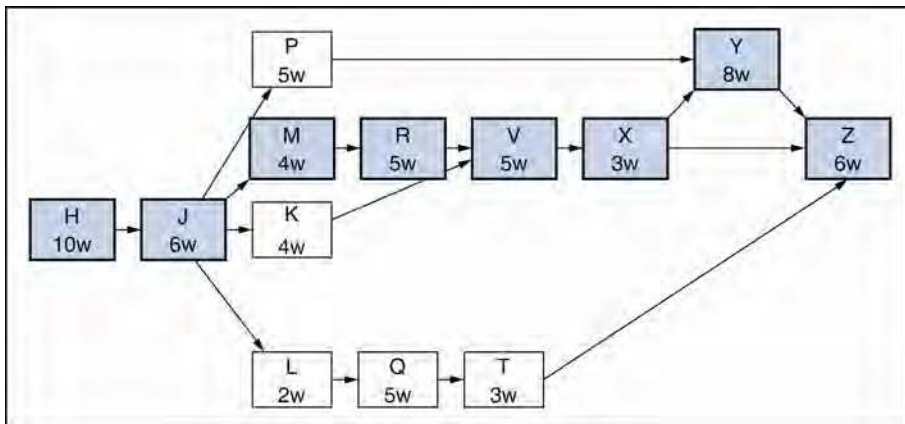
Although the critical path is important, that doesn’t mean the project manager can ignore noncritical activities. Whenever a noncritical activity is delayed, the path to which it belongs gets longer. When the length of a noncritical path grows to exceed the critical path, the former noncritical path becomes critical and the (former) critical path becomes noncritical! In other words, the critical path *changes*.¹ This change can happen without warning and leave the project manager focused on the wrong activities. One solution is to signal a warning when a noncritical activity is at risk of becoming critical; this is part of the **critical chain method**,

discussed in Chapter 8.

Figure 7.4(b) illustrates an activity that spans multiple other activities, called a **hammock**. The activity “Monitor progress” is a hammock because it covers every activity in the project except “User test,” implying that the project manager is responsible for monitoring progress on every activity except “User test.” The duration of a hammock is determined by the duration of the longest path of activities over which it spans; in Figure 7.4(b) this is $6 + 4 + 6 + 8 + 1 = 25$ weeks. Note, however, that although a hammock spans a portion of the longest path, it is not considered a critical activity. (The term hammock can also refer to a **summary activity**, e.g., a set of activities aggregated into one work package.)



See Chapter 8

**Figure 7.5**

Example network showing the critical path.

A final example network is Figure 7.5; it has four paths from start node H to end node Z:

- a. H-J-P-Y-Z 35 weeks
- b. H-J-K-V-X-Y-Z 42 weeks
- c. H-J-M-R-V-X-Y-Z 47 weeks
- d. H-J-L-Q-T-Z 32 weeks

The longest of the four paths is Path c (indicated by the “shadowed” boxes); hence c is the critical path and $T_c = 47$. (In Figure 7.5, notice the arrow between X and Z is unnecessary: if Z follows Y, and Y follows X, then obviously Z must follow X!)

Multiple critical paths

Can a project have more than one critical path? In a word: yes. Suppose the duration of Activity L in Figure 7.5 was 17 weeks instead of 2 weeks. In that case the durations of path M-R-V-X-Y and path L-Q-T would both be 25 weeks. The project would have *two* critical paths, both with duration 47 weeks, and the project would be delayed if a delay were to occur on *either* critical path. If, however, the project duration had to be *shortened* to less than 47 weeks, it would then be necessary to shorten *both* critical paths.

Early times: early start (ES) and early finish (EF)

Scheduling each activity in a project involves, at the minimum, specifying when the activity must be started and finished. The scheduling procedure depends on whether the project is assumed to start “at Time 0” or “on Day 1.” It makes a difference. The procedure described below is based on the simpler “Time 0” assumption; Appendix B to this chapter describes See Appendix B scheduling under the “Day 1” assumption.



The formula for computing the finish time given the start time and activity duration is:

$$\text{Finish time} = \text{Start time} + \text{Duration}$$

These start and finish times for an activity are represented on the network as “early times”: (1) the **early start time (ES)**, and (2) the **early finish time (EF)**. They represent the earliest possible times that the activity can be started and completed.

But the ES of an activity depends on the EFs of its immediate predecessors, which is found by summing the durations of all the predecessor activities along the paths leading to the activity in question. When an activity has more than one immediate predecessor, the ES for the activity will be determined by the EF of its latest immediate predecessor.

All this is shown in Figure 7.6. Suppose the ES for the first activity, H, is 0 (meaning the project starts at time 0). Since Activity H’s duration is 10 weeks, its early finish, EF, must be Week 10. This was determined from the formula

$$\text{EF} = \text{ES} + \text{Duration}$$

In Figure 7.6, ES is shown in the upper left of each node and EF on the upper right.

Given that Activity H’s EF is Week 10, the ES for Activity J will be Week 10 and the EF will be Week 16. Activity J is the only immediate predecessor for activities K, M, and L, so the ES for those activities will be Week 16. Activity V has two immediate predecessors, K and R, meaning it cannot start until both have been completed. The EF for Activity K is the length of the path H–J–K, or $10 + 6 + 4 = 20$; the EF for Activity R is the length of the path H–J–M–R or $10 + 6 + 4 + 5 = 25$. The ES for Activity V will depend on the later EF of its two immediate predecessors, which is R. Thus, ES for Activity V is Week 25.

The same happens at Activity P: it has two immediate predecessors, I and J. Since Activity I’s EF is 18 and Activity J’s EF is 16, Activity P’s ES must be Week 18. Activity Y has three

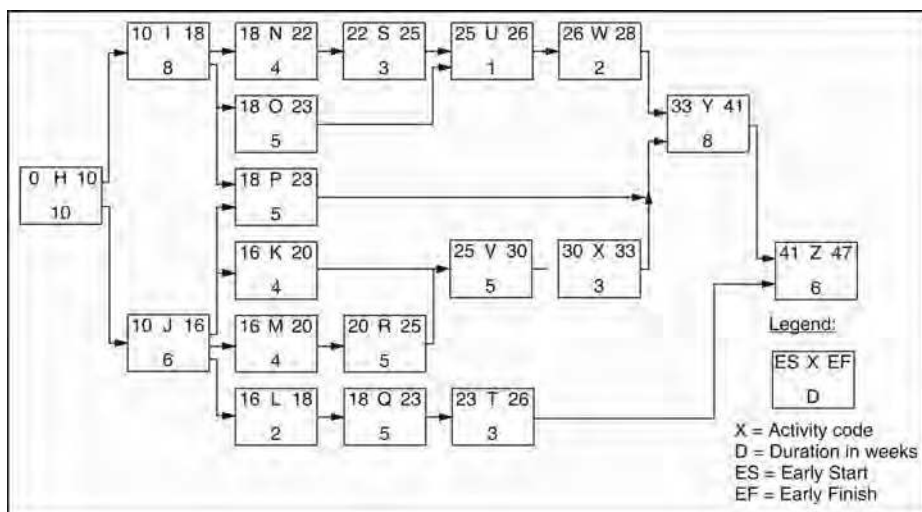


Figure 7.6
Example network showing ESs and EFs.

immediate predecessors, W, P and X; Activity X has the latest EF, 33, thus the ES for Activity Y is Week 33. Finally, the ES for Activity Z is 41, which is the latest EF of its immediate predecessors Y and T. Activity Z's EF is Week 47; this gives the expected project duration, T_e , 47 weeks.

In summary, ESs and EFs are computed by taking a “forward pass” through the network. When an activity has only one immediate predecessor, its ES is simply the EF of that predecessor. When an activity has several immediate predecessors, its ES is based on the *latest* EF of the immediate predecessors.

Late times: late start (LS) and late finish (LF)

As discussed earlier, a noncritical activity can be delayed without delaying the project; the question is, How much can it be delayed? To answer that we must determine the “late times,” that is, the latest allowable times an activity can be started and finished without delaying project completion. Just like the ES and the EF, every activity has a **late start time, LS**, and a **late finish time, LF**.

Refer to Figure 7.7. LS is shown on the lower left of each node, LF on the lower right.

To determine the late times, begin by assigning a **target completion date, T_s** , to the last node in the network. For projects that must be completed as soon as possible, the date for T_s is the same as the *estimated completion date, T_e* , calculated in a forward pass—the EF of the last activity. For projects with a **due date** set by the customer or the sponsor, T_s is the due date, not the calculated T_e value.

To determine the late times, start at the last activity in the network and make a “backward pass” through the network using the formula:

$$LS = LF - \text{Duration}$$

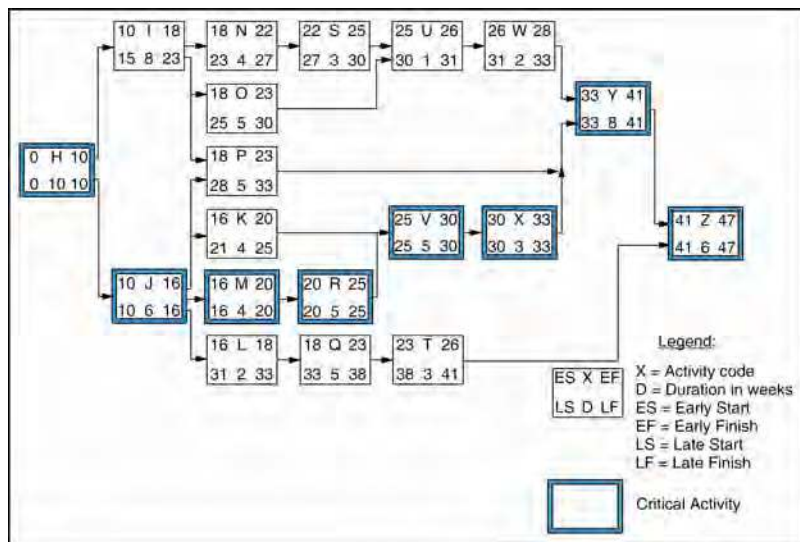


Figure 7.7
Example network showing LSs and LFs, ESs and EFs.

In Figure 7.7, start with Activity Z. If T_s is 47 weeks, then LF for Activity Z is 47 and LS is $47 - 6 = 41$; i.e., Activity Z must start in Week 41 for the project to end in Week 47. Continuing backward, LF for Activity Y and Activity T is 41 weeks, so LS for Y is $41 - 8 = 33$, and LS for activity T is $41 - 3 = 38$. Continue working backward like this through each path, computing LF and LS for each activity.

Whenever we encounter an activity that has multiple paths leading back to it (i.e., it has multiple **immediate successors**), it is the *earliest* (or *smallest*) EF of the immediate successors that determines the activity's LF. For example, Activity J has four paths leading back to it (four immediate successor activities):

- LS for Activity P = 28
- LS for Activity K = 21 weeks
- LS for Activity M = 16 weeks
- LS for Activity L = 31 weeks.

Since the smallest (earliest) LS is 16 for Activity M, the LF for Activity J must be 16; this is the latest time Activity J can be finished to enable all its successors to meet their late start times and, thus, complete the project by its target date of 47 weeks.

In summary, calculations for LFs and LSs start at the last node of the project network and work backward. When an activity has more than one path leading back to it, the smallest (earliest) value of LS among the immediate successors is the basis for determining the activity's LF. Having completed both forward and backward passes through the network, we now have the earliest possible and latest allowable scheduled times for every activity in the network. Once the forward and backward pass calculations have been completed the durations of hammock activities become evident.

Total slack

Notice that for most activities in Figure 7.7, the early times and late times are not the same. The difference between LS and ES (or LF and EF) is referred to as **total slack** (also called **total float** or, simply, *slack* or *float*) of an activity. Slack is the amount of allowable deviation between the latest when an activity must happen and the earliest when it can happen. It is the amount of time an activity can be delayed without delaying the project:

$$\begin{aligned}\text{Total slack} &= \text{LS} - \text{ES} \\ &= \text{LF} - \text{EF}\end{aligned}$$

In Figure 7.7 the total slack for Activity H using early start times is $0 - 0 = 0$ weeks; for Activity I it is $15 - 10 = 5$ weeks, and so on. Notice that activities on the previously identified critical path H–J–M–R–V–X–Y–Z have zero slack; hence, these activities cannot be delayed by any amount without delaying the project. (The case where critical activities *do* have some slack is discussed later.) The activities that *do* have slack (which, as it turns out, are the *noncritical* activities) can be delayed from their earliest possible dates by their slack time without delaying project completion. Total slack is shown in Table 7.4.

When activities lie in sequence on a path, a delay in earlier activities will result in a delay to later ones; this is the equivalent of reducing slack for the remaining activities. In Figure 7.7, for example, activities L, Q, and T all lie on the same path and all have the same slack of 15 weeks. But if Activity L is delayed 5 weeks, then activities Q and T will also be delayed 5 weeks

Table 7.4 LOGON project time analysis (from Figure 7.7).

Activity	Duration (Weeks)	Start Node		Finish Node		Slack		Note
		ES (start of week)	LS (start of week)	EF (end of week)	LF (end of week)	Total*	Free**	
H	10	1	1	10	10	0	0	CP
I	8	11	16	18	23	5	0	
J	6	11	11	16	16	0	0	CP
K	4	17	22	20	25	5	5	
L	2	17	32	18	33	15	0	
M	4	17	17	20	20	0	0	CP
N	4	19	24	22	27	5	0	
O	5	19	26	23	30	7	2	
P	5	19	29	23	33	10	10	
Q	5	19	34	23	38	15	0	
R	5	21	21	25	25	0	0	CP
S	3	23	28	25	30	5	0	
T	3	24	39	26	41	15	15	
U	1	26	31	26	31	5	0	
V	5	26	26	30	30	0	0	CP
W	2	27	32	28	33	5	5	
X	3	31	31	33	33	0	0	CP
Y	8	34	34	41	41	0	0	CP
Z	6	42	42	47	47	0	0	CP
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	

Total Slack* (7) = (4) – (3) = (6) – (5)

Free Slack** (8) = [(3) of earliest successor] – (5)

*Total slack is the spare time on an activity that, if used up, will delay successors and the completion of the project.

**Free slack is the spare time on an activity that, if used up, will not delay the start of succeeding activities.

and, thus, will have only 10 weeks of slack remaining, not 15. If, in addition, Activity Q is delayed 10 weeks, then Activity T will have no remaining slack and must be started immediately upon completion of Q. Having used up all their slack, Activities L, Q, and T would then all become critical activities. Once slack is used up, noncritical activities become critical activities, which means any further delays for these activities will delay project completion.

The practical implication of slack is that it gives the project manager flexibility regarding exactly when noncritical activities can be scheduled: they can be scheduled any time as long as it lies somewhere within the available slack—between the ES and LF times. Knowing the slack is important for managing resource workload. By starting some activities as early as possible and delaying others, the workload can be smoothed; this concept is discussed later. In general, when resources are sufficient, noncritical activities are usually scheduled as early as possible

(their ESs); this preserves slack and minimizes the risk of noncritical activities delaying the project. (Another method, called *critical chain*, schedules activities as late as possible; this is discussed in the next chapter.)

Notice that decisions about when exactly to schedule an activity require knowing both the late and early times for the activity. The implication is that a network analysis should be done *before* the Gantt chart is created. Most project management software develops networks and Gantt charts simultaneously, where the Gantt charts are based on early times. As mentioned, however, activities should *not necessarily* be scheduled according to the early times.

Free slack

While *total slack* refers to the amount of time an activity can be delayed without delaying the project, the term **free slack** refers to the time an activity can be delayed without delaying the start of any successor activity. Free slack of an activity is determined by the formula:

$$\text{Free slack for activity} = \text{ES (earliest successor)} - \text{EF (activity)}$$

For example, in Figure 7.7 Activity I has a total slack of 5 weeks but free slack of 0 weeks because any delay in it will delay the start of its successor activities N, O, and P. Activity O, on the other hand, has free slack of 2 weeks because its EF of 23 can be delayed to 25 without delaying the ES of its successor, Activity U, which is 25.

Knowing the free slack, managers can readily identify activities where slippages immediately impact other activities. When an activity has zero free slack, any slippage will cause at least one other activity to slip also. If, for example, Activity L slips, then so will Q and T, and teams working on Q and T must be notified of the delay.

As with total slack, the amount of free slack available to an activity assumes the activity starts at its ES time. Thus, the free slack for Activity O is 2 weeks, as long as Activity I, its immediate predecessor, is completed at EF = 18. If Activity I is delayed by any amount, then Activity O's free slack will be reduced by that same amount.

Free slack is important because many activities are scheduled to start as soon as possible, and resources are booked to be available on these dates. If an activity is delayed, it might delay other activities and disrupt the schedules of everyone who had planned to work on those activities. Moreover, such delays extend the period over which resources (e.g. equipment contracted at a daily or hourly rate) are needed and can lead to cost overruns.

Table 7.4 summarizes these concepts, showing ES, LS, EF, and LF, and total and free slack for the LOGON project in Figure 7.7. Notice that for activities on the critical path the total slack and free slack times are zero.

The effect of project due date

In discussing total slack, we assumed that the **target completion date**, T_s , was the same as the earliest **expected completion date**, T_e . But in fact, the target completion date can be set to be either later or earlier than T_e to reflect a customer's or supporter's wishes.

Setting the target date T_s to later than T_e has the effect of increasing total slack for every activity in the project by the amount $T_s - T_e$. Although no longer zero, the slack on the critical path will still be the smallest slack anywhere in the network. For example, if the target completion date

for the project in Figure 7.7 were increased to $T_s = 50$ weeks, then the total slack in Table 7.4 would be $50 - 47 = 3$ weeks for all critical activities and 3 additional weeks for all noncritical activities.

If T_s is set earlier than T_c , then total slack times everywhere in the project will be reduced by the amount $T_s - T_c$ and activities along the critical path will have *negative* slack times. The size of this negative slack is the amount of time by which the project duration must be reduced to meet the target completion date. (Note, altering T_s has no influence on free slack times: these depend on ES and EF times, both of which are affected the same amount when changing T_s .)

In general, projects must be completed either as soon as possible or by a predetermined due date. For projects that have to be completed as soon as possible, the project manager does a forward-pass calculation through the network, then commits to the computed T_c . For projects that must meet a predetermined due date, the project manager substitutes T_s at the last event, then works backwards through the network to compute late times and slack times, noting the feasibility of speeding up activities to eliminate negative slack times on the critical path.

7.3 Converting to Gantt Calendar Schedules

Using information from tables such as Tables 7.2 or 7.3 to create a network with activity start and finish times is a simple procedure that requires no management decisions and can readily be performed by computer software. To be usable, however, the times in the network must be converted into dates (day, month, and year) on either a Gantt chart or an actual calendar. But converting network times to a **Gantt** or **calendar schedule** is not a simple procedure and does require management decisions.

For starters, the Gantt or calendar schedule must account for **non-working time** such as weekends, holidays, and vacations. Figure 7.8 shows the LOGON project schedule as produced by Microsoft Project software and incorporating time off for weekends and holidays.

In addition, a calendar schedule must account for issues that require analysis and management decisions; examples include:

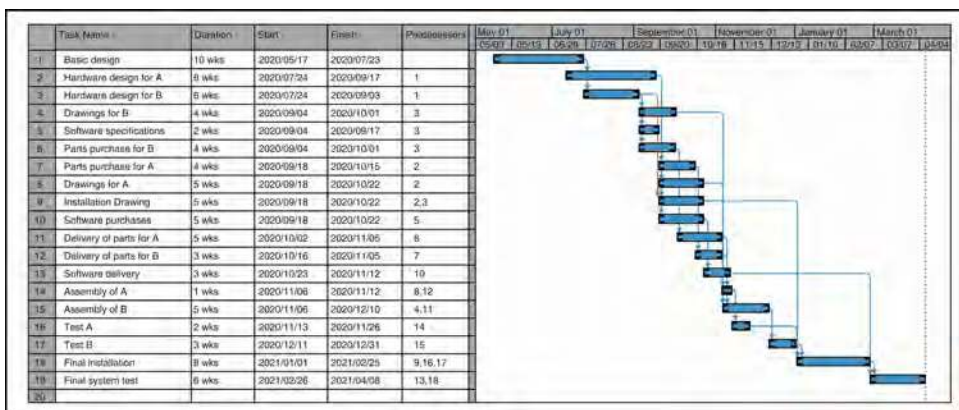


Figure 7.8
LOGON project schedule adjusted for holidays and weekends.

- **Resource constraints:** a work package is delayed because the necessary resources are unavailable or must be shared with other activities.
- **Cash flow:** procurement of an expensive piece of equipment must be delayed in order to defer cash outlay, improve cash flow, or await an exchange rate improvement.
- **Risk of changes:** a design activity is postponed due to changes in project scope or technology.
- **Logistics:** the acquisition of a bulky item for construction is delayed until space becomes available at the construction site.

Computer software can readily generate a project network, Gantt chart, or calendar schedule, but unless issues like those listed above are accounted for, the schedule will be infeasible, unworkable, or too risky. The point is, project scheduling involves more than merely creating a computer-generated version of the project network; it requires analysis and management judgment. Thus, the Gantt chart should be created only *after* a network analysis has established the early and late dates, and issues and constraints surrounding the project have been resolved.

7.4 Schedule Reserve

In common practice, the contractual or committed target completion time T_c is not simply the estimated completion time T_e plus allowance for non-working time, but rather is some time *after* that and includes a **schedule reserve**, also called a **management reserve**. This chapter has treated activity times and project durations as if they are fixed. Of course, each project is unique, and until it is actually completed its duration is only an estimate (a best guess). All time estimates (of projects and of the activities that compose them) are subject to uncertainty; the more unique the project, the larger the uncertainty. To account for that uncertainty a management schedule reserve or **time buffer** is added to the estimated duration. This reserve constitutes a “safety buffer” to accommodate any project delays. Time buffers are discussed



See Chapter 8 in Chapter 8.

7.5 Alternative Relationships and PDM²

The network scheduling procedures discussed earlier assume a sequential relationship wherein the start of an activity is predicated upon the completion of its immediate predecessors. Such is the case illustrated in the diagram in Figure 7.9 where Activity B starts upon completion of Activity A. This strict start-only-when-predecessors-finish relationship is called **finish-to-start, FS**. The limitation of this assumption is that it precludes those kinds of tasks that can be started when their predecessors are only *partially* (but not fully) completed. For example, when a company relocates to a new facility, the activity “move-in employees” would be able to start after *some* of the activity “move-in furniture” has been done; i.e., “move-in employees” can

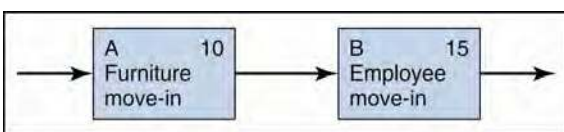


Figure 7.9
Example of FS relationship.

begin *before* its immediate predecessor “move-in furniture” has been completed. The **precedence diagramming method (PDM)** allows for this and similar such situations. Besides the usual FS relationship, PDM also permits other relationships such as **start-to-start (SS)**, **finish-to-finish (FF)**, and **start-to-finish (SF)**. It also allows for lags between the times when activities must be started or finished. These relationships are described next.

Start-to-start (SS)

In an SS relationship between two activities A and B, the start of B can occur at the earliest n days after the start of its immediate predecessor, A. This is diagrammed in Figure 7.10.

Using the example from Figure 7.9, suppose that “move-in employees” can begin 5 days after the start of “move-in furniture”; the network diagram and associated Gantt chart for the two activities would appear as in Figure 7.11. The n days delay is called **lag**—the start of B lags that of A by n days. (Confusingly it is sometimes also called **lead**, as “the start of A leads that of B by n days.”)

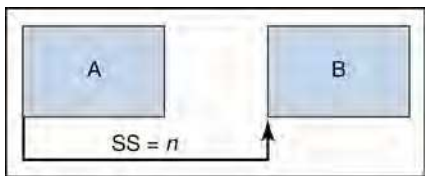


Figure 7.10

PDM representation of SS relationship with n day lag.

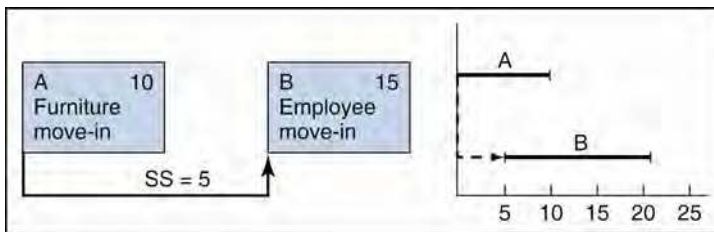


Figure 7.11

Example of SS relationship.

Finish-to-finish (FF)

In an FF relationship between two activities A and B, B will finish n days at the latest after A finishes. An illustration is in Figure 7.12 where the finish of “paint parking lines” (B) must occur within 5 days of the finish of “lay asphalt” (A). An FF relationship with zero lag means the activities must finish at the same time.

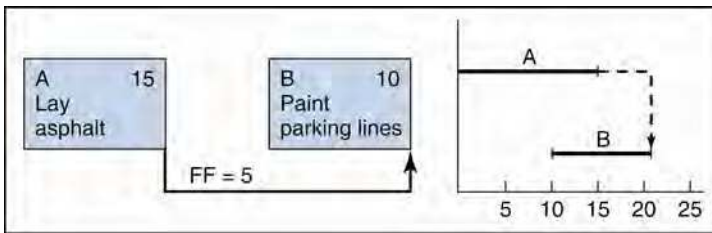


Figure 7.12
Example of FF relationship.

Start-to-finish (SF)

In an SF relationship, the finish of Activity B must occur at the latest n days after the start of Activity A. For example, “phase-out old system” (B) cannot finish until 25 days after “test new system” (A) begins. This is shown in Figure 7.13.

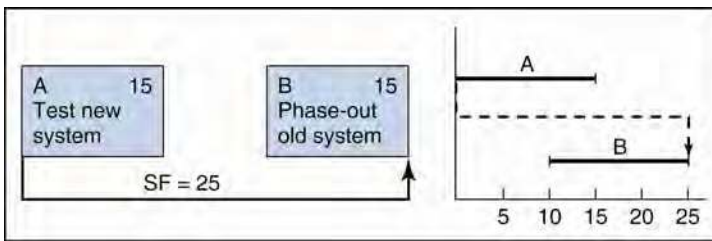


Figure 7.13
Example of SF relationship.

Finish-to-start (FS)

In an FS relationship, Activity B can start at the earliest n days after Activity A is finished. For example, “tear down scaffolding” (B) can start no sooner than 5 days after “plaster walls” (A) is finished. This is shown in Figure 7.14. Note that when $n = 0$, the FS relationship becomes the same as the traditional AON network method wherein the start of a successor coincides with the completion of its latest predecessor.

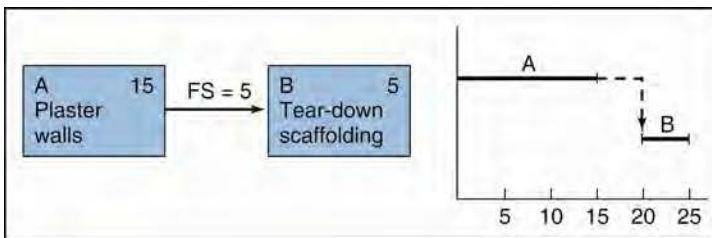


Figure 7.14
Example of FS relationship.

Multiple PDM relationships

Two PDM relationships can be used in combination. Having both SS and FF is a common case. Notice in the example shown in Figure 7.15 that because B must finish no later than 10 days after A finishes, the start of B must occur at day 10. But suppose B is an interruptible activity (i.e., work in B can be stopped and resumed later when, say, needed resources arrive). In that case, B could instead be started 5 days after the start of A and finish 10 days after A finishes. This is represented in Figure 7.16. The assumption is that the 15 days of work for B will be performed sometime within the 20 days allotted between days 5 and 25. Notice that the 20 days allotted for Activity B gives that activity two possible slack values, $LS - ES = 5$ or $LF - EF = 0$. PDM usually observes the smallest slack value, here 0.

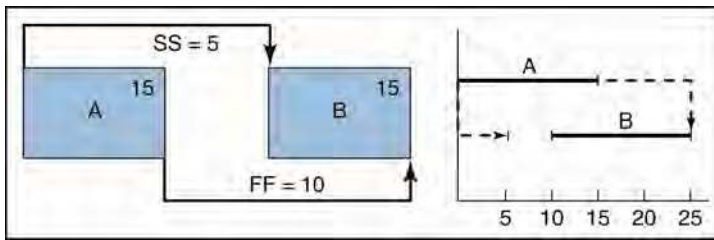


Figure 7.15
Schedule for non-interruptible activity B.

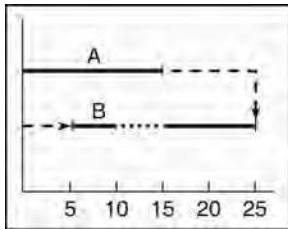


Figure 7.16
Schedule for interruptible activity B.

Example 7.1: PDM in ROSEBUD Project

Figure 7.17 shows the AON diagram for the ROSEBUD project and Figure 7.18 shows the corresponding **time-scaled network**, which is a form of Gantt chart or calendar schedule that explicitly shows dependencies among the activities.

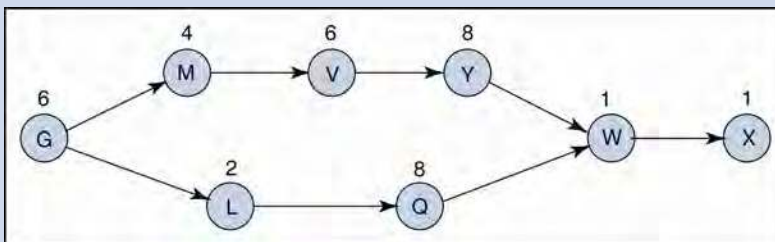


Figure 7.17
AON diagram for ROSEBUD project.

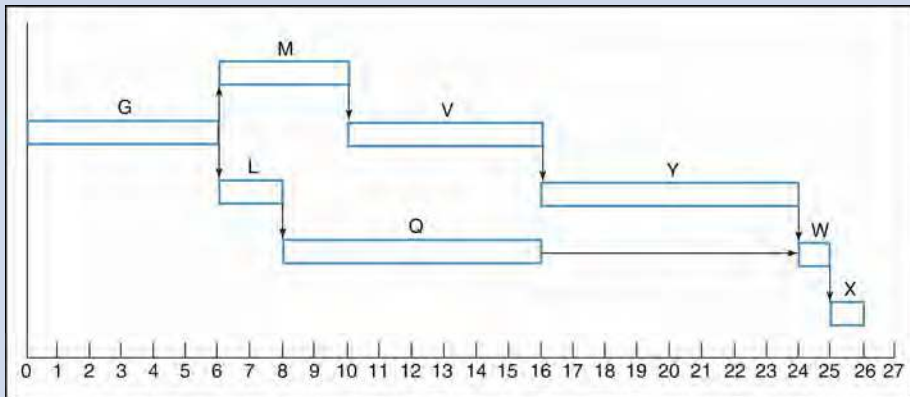


Figure 7.18
Time-scaled network for ROSEBUD project.

The network will now be altered to permit the following special relationships:

1. Activity L can begin 3 days after Activity G begins, but it cannot be finished until G is also finished.
2. Activity Y can begin 2 days after Activity V begins, but it cannot be finished until at least 6 days after V is finished.
3. Activity W can begin 5 days after Activity Y begins, but it cannot be finished until Y is also finished.
4. Activity X cannot be started until at least 1 day after Activity W is finished.

The PDM network in Figure 7.19 shows these relationships. Figure 7.20 shows the corresponding time-scaled network assuming earliest start dates and allowing for interruptible activities.

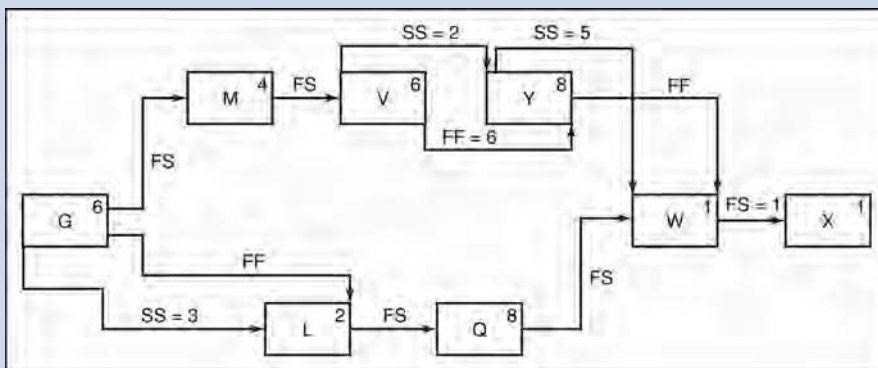


Figure 7.19
PDM network for ROSEBUD project.

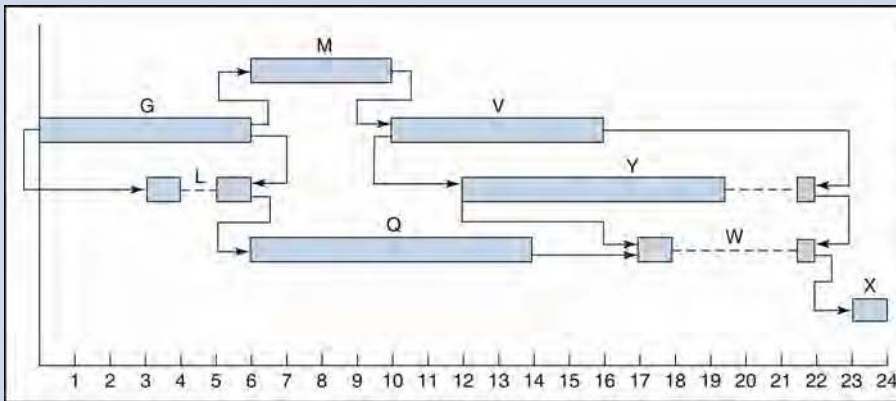


Figure 7.20
Time-scaled network for ROSEBUD project revised for PDM.

A traditional FS network can handle relationships where $FS > 0$ by creating artificial activities, but it has no way of incorporating SS, FF, or SF; thus, the obvious advantage of PDM is that it permits greater scheduling flexibility. The tradeoff is that PDM networks are more complex and require greater care both in their creation and interpretation. Because activities do not follow a neat FS sequence, finding the critical path and slack times is not so simple either. Complex precedence relationships also cause counterintuitive results. For example, in a simple network the way to reduce the project completion time is to reduce the duration of activities on the critical path; doing the same thing in a PDM network, however, does not necessarily shorten the project. In the previous example, the critical path is G–M–V–Y–W–X. Suppose we decide to reduce the time on Activity Y. Because of the precedence requirement that Y cannot finish sooner than 6 days before V finishes, the completion date of Y cannot be changed. Thus, any shortening of the duration of Y serves to *delay* the start date of Y. Because of the precedence requirement, delaying the start date of Y results in delaying the start date of W and, as a result, the start date of X. In other words, shortening critical Activity Y actually *increases* the project duration.

In general, interpreting a PDM network requires more care than ordinary AON networks, although such care is inconsequential when the network is created with project scheduling software.

7.6 Scheduling with Resource Constraints

While people think project scheduling means scheduling work tasks or activities, it is more important to think of it as **scheduling resources**. Every activity requires resources—people, equipment, material, working capital, etc., and whenever an activity is scheduled, that means resources *are scheduled* too. So far, we have assumed that such resources are always available when needed. But of course, resources are not always available, and when they are not, the project schedule must be changed to whenever the resources will be available. We now consider project scheduling when resources *are constrained* and the effect it has on workload and project duration.

Resource availability and project duration

In general, projects tend to be either **resource-constrained** or **time-constrained**. In a resource-constrained project, such as shown in Figure 7.21, the resources are limited and the project completion date is determined by the availability of those resources. In a time-constrained project, the project completion date is fixed and the schedule must be adjusted or the resources found to meet that date.

Often the availability of skilled workers, equipment, and working capital dictate whether activities can be scheduled at their early times or must be delayed. This is especially true when multiple activities requiring the same resource are scheduled for the same time; when the resources are not sufficient to satisfy the needs of all of them, some activities must be delayed. Figure 7.21 illustrates this: part (a) shows the network, and (b) shows the project schedule, not accounting for the resources. Suppose Activities B and C both require a resource that can be used by only one of them at a time, in which case, the schedule must be revised; (c) shows two alternative resource allocations, shifting the schedules of Activities B, C, and D; note that the result is to delay the project by one day.

Resource allocation, workload, and resource loading

The terms resource allocation, workload, and resource loading convey related but different concepts. **Resource allocation** refers to assigning one or more resources to an activity or project. **Workload** refers to the amount of work imposed on a resource. **Resource loading** refers to the amount of a particular resource needed to conduct all the activities in a project to which the resource is allocated. For an **individual resource** (such as a person), the workload can be specified either as a percentage of the resource's full workload potential or, more commonly, in units such as labor hours. For a **facility** or **labor category** (such as a department of workers

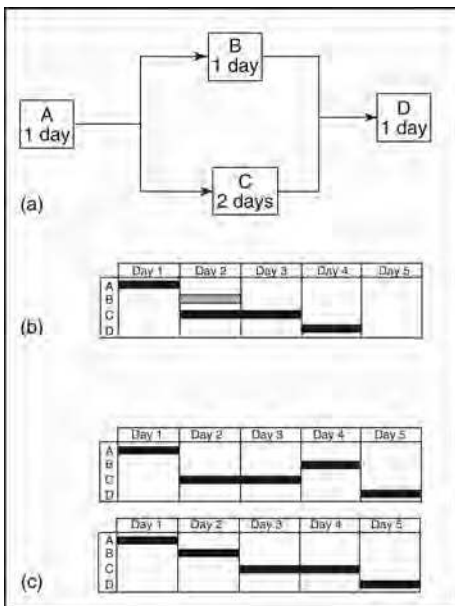


Figure 7.21

Effect of a constrained resource on schedule. (a) Network diagram; (b) Gantt chart where resource is unlimited; (c) Gantt charts where same resource is limited and must perform activities B and C.

with specific skills), the workload is specified in terms of number of workers. Since all the people in a labor category (such as “computer programmer”) seldom have exactly the same skills, ordinarily it is better to allocate a specific person (a specific programmer) rather than a labor category to an activity. The usual assumption when allocating people from a labor category is that everyone in the category is equally capable; often, though, after the work begins it becomes evident they are not.

The workload that an individual can handle in a year is computed as the number of working days (excluding holidays and all types of leave) times the number of productive (working) hours per day. Many companies have guidelines restricting the number of hours an individual should work on projects per week, month, or year. In a matrix organization (described in later chapters), functional managers are responsible for ensuring that each worker’s time is well utilized and her workload does not exceed a recommended maximum.

Workload is always from the perspective of the particular *resource*; in contrast, loading is from the perspective of the *project*. Loading is the number of hours, people, or other units of a particular resource needed at a given time in a project (or in multiple concurrent projects). Resource loading is important since virtually all resources are finite, and many are scarce. Thus, the resource loading (total amount of the resource needed for a project or projects at a given time) cannot exceed the quantity available. When resources are scarce, their allocation is constrained, and sometimes activities in a project must be rescheduled to accommodate the scarcity. The example in Figure 7.21 was such a case: Activities B and C require the same resource, but the resource cannot be used for both at the same time. Resources that are available in sufficient quantity can be ignored for scheduling purposes (breathable air is an example—except when the project is conducted under water or in outer space where air is limited).

The following sections consider two cases where the project schedule must be altered to accommodate resources. The first is called **resource smoothing** of a *time-constrained project*. In this case, there is enough of the resource to complete the project by a *set completion date*; however, the amount of the resource needed fluctuates throughout the project, making the resource difficult to manage. The goal of resource smoothing is to *smooth-out* the amount of the resource needed throughout the project without delaying the project. The second case, called **resource leveling**, is the situation mentioned before, the *resource-constrained project*—not having enough of a resource to do multiple activities at the same time. The goal is to satisfy the resource constraint even if the result is to delay the project.

Smoothing a time-constrained project

Because the loading for a particular resource depends on the amount of the resource needed by project activities and the start and finish dates of those activities, the loading tends to vary throughout a project. A common resource-loading pattern in a project is a steady buildup in the amount of the resource needed, a peak, and then a decline. Thus, relatively little of the resource is needed early and late in the project, but much is needed in the middle. This pattern is problematic for functional managers who oversee a fixed pool of workers and equipment because it results in the pool being either underworked or overworked. Certainly better would be a relatively *uniform* workload on the resource pool. This is the purpose of **resource smoothing**: to alter the schedules of project activities such that the resultant workload for a required resource is somewhat uniform throughout the project *without changing* the critical path or *delaying* the project completion date.

Figure 7.22 shows the schedule for the LOGON project and the resultant resource loading—here the resource being workers of a particular skill or trade (programmers, steel worker, etc.). The resource loading, bottom of Figure 7.22, is created from the schedule, top of Figure 7.22, and the weekly labor requirements in Table 7.5; it shows week by week the

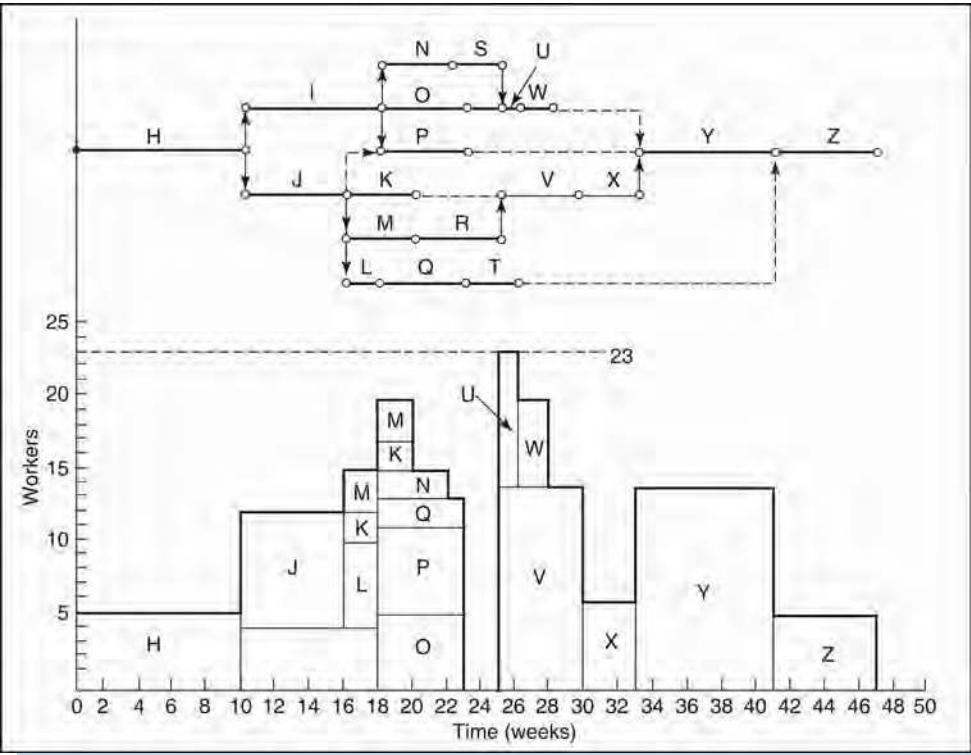


Figure 7.22
Schedule and corresponding worker loading for the LOGON project.

Table 7.5 LOGON project weekly labor and equipment requirements.

Activity	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Weekly Labor Requirements (workers)	5	4	8	2	6	3	2	5	6	2	0	0	0	9	14	6	6	14	5
Weekly Equipment Requirements (hours)	8	2	6	1	2	2	0	0	6	0	4	4	0	8	8	8	8	8	8

number of workers needed in the project. For example, for the first 10 weeks only activity H is scheduled, so the loading for those weeks is 5 workers (the weekly labor requirement for H). Over weeks 10–16, activities I and J are scheduled, so the loading, based on their labor requirements, becomes $4 + 8 = 12$, and so on.

From Figure 7.22, you can see that the loading for the LOGON project might pose a problem because it fluctuates so much, varying from 23 workers in Week 26 to zero workers in Weeks 24 and 25 (not a problem if activities R, S, and T are outsourced and do not require any workers). The problem facing the manager allocating these workers to LOGON is what to do with excess workers in slow periods and where to get additional workers in peak periods.

A way to handle the problem is to adjust the worker loading so it is more “level.” This is done by “juggling” activities—by taking advantage of slack times and delaying noncritical activities, so as to reduce workload peaks and fill in workload valleys. For example, the smoother workload in Figure 7.23 is achieved by delaying activities P and Q (and hence T) by 2 weeks and U (and hence W) by 5 weeks.

Splitting activities, multi-tasking, and hand-over points

In the previous example an even more uniform loading could have been achieved if activities were **split** and the pieces scheduled at different times. Whether this is feasible depends on whether a job, once started, can be interrupted and then restarted later. As discussed, project activities and work packages are defined during the WBS process, and these activities become the basis for establishing schedules, budgets, and so on. Once an “activity” has been defined in the WBS, it cannot be arbitrarily split later.

Although activity splitting can lead to a more uniform loading, the downside is that it can lead to wasted time and longer activity durations. Figure 7.24 illustrates how this happens. Uninterrupted, the activity starts slowly but then builds momentum as it moves ahead. Split into pieces, each piece starts slowly and never gains momentum. The sum of the durations of the pieces in (b) exceeds the duration in (a). The effect, called **multi-tasking**, leads to slower-paced work on average and extends the activity duration. The moral is, once an activity has been started, it is usually better to finish it uninterrupted.

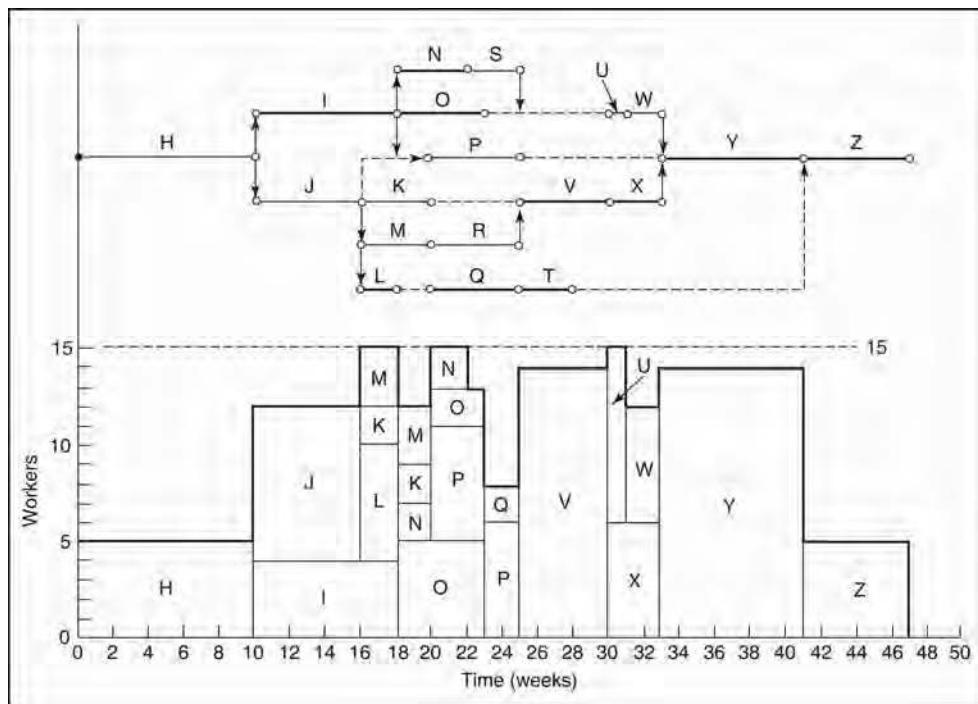


Figure 7.23

Schedule and smoothed worker loading for the LOGON project.

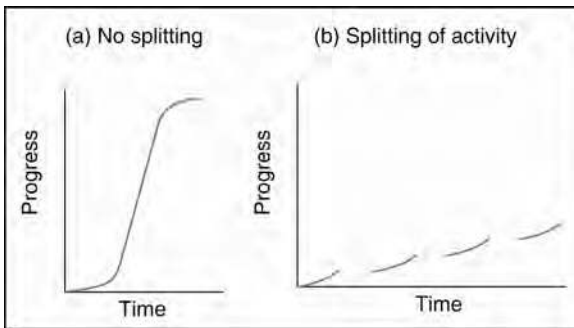


Figure 7.24
The effect of splitting of an activity on duration.

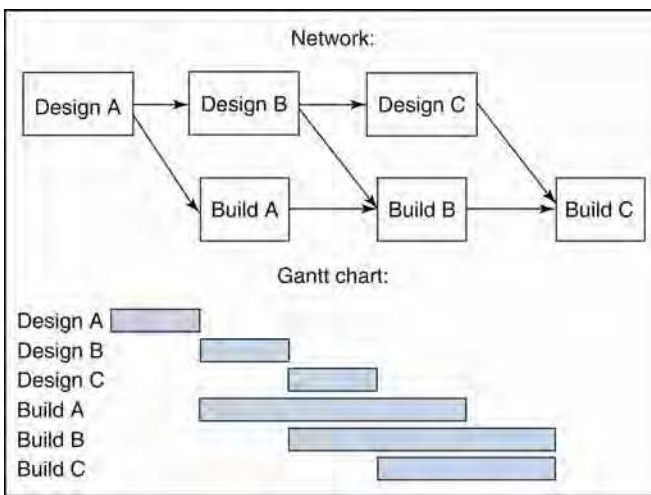


Figure 7.25
Multiple hand-over points of uninterrupted activities.

Multi-tasking, wherein the work is stopped and then resumed, should not be confused with work that continues uninterrupted but has multiple **hand-over points**. The hand-over concept is illustrated in Figure 7.25 where the design and build activities each progress uninterrupted (the Design A team hands over its work to the Design B and Build A teams), and where multiple hand-over points (called **laddering**) enable the build activity to start and continue well before the entire design activity (encompassing Design A + Design B + Design C) is completed. Although the activities appear to be split (Design A, Design B, Design C), in fact they are not since there is no time lag between them. The method shortens project duration and facilitates interaction between designers and builders.

Smoothing multiple resources

Smoothing is easy for a single resource but can be difficult for several resources employed simultaneously. Because work packages usually require resources from more than one functional unit or subcontractor, a schedule that provides a smooth loading for one unit may cause

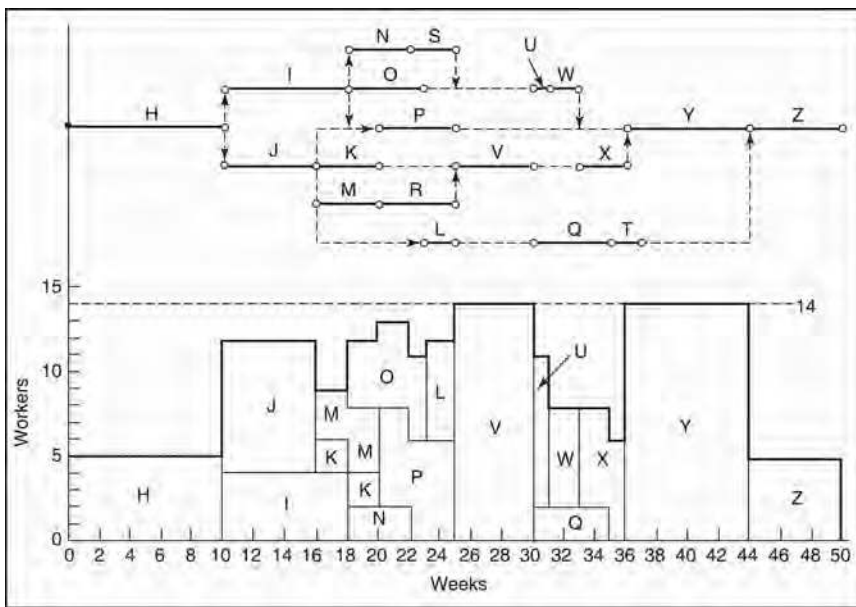


Figure 7.26
Equipment loading for the LOGON project.

overloading or difficult-to-manage situations for others. For example, based on the weekly equipment requirements for LOGON shown in Table 7.5, the schedule that provides the somewhat-smooth worker loading in Figure 7.23 yields the erratic equipment loading shown in Figure 7.26. An attempt to level the equipment loading by adjusting or delaying activities will disrupt the worker loading. As you can verify, the schedule in Figure 7.22 that produces the erratic loading for workers yields a relatively level loading for equipment.

It is impossible to completely smooth the load for all resources at once. The best results arise from applying the scheduling equivalent of the “Pareto optimum”; that is, schedule activities in the best interests of the project while trying to minimize the number of conflicts and problems in the departments and contractors that supply the resources. When considering multiple resources simultaneously, the focus is on leveling the “priority” resources—those where irregular loadings are the costliest to the organization or demoralizing to workers. The financial and social costs associated with hiring, overtime, and layoffs often dictate giving “human resources”—the workers—the highest priority. Many project software packages perform scheduling analysis that permit simultaneous leveling of multiple resources.

Delaying activities is one method to smooth resources; others include:

- Eliminate some activities or work segments (reduce project scope).
- Substitute some resources for others.
- Substitute high-resource activities with lower-resource activities.

For example, when the most qualified workers are not available, either eliminate the work that requires their expertise or use fewer qualified workers. These options, however, might compromise the scope or quality of the work and increase the risk of the project not meeting requirements.

Leveling a resource-constrained project

What happens when the number of personnel, pieces of equipment, or available working capital are strictly limited? This is a **resource-constrained project**. Activities in the project must be scheduled so that the loading of a particular resource to the project does not exceed the available maximum. The focus differs from time-constrained resource smoothing because the issue is the resource's *maximum* requirement, not its loading *variability*. As each activity is scheduled, the sum of its required resources plus the resources required for activities already scheduled for the same time must be checked against the maximum available. Mentioned earlier, the problem, called **resource leveling**, is more than just leveling of resources; it involves rescheduling or delaying jobs to when the resources are available. The objective is to balance the demand for resources with the available supply of the resources, *even if it results in changing the critical path and delaying the project*.

In the LOGON project, for example, suppose only 14 workers are available in any given week. The "leveled" schedule in Figure 7.23 resulted in a maximum loading of 15 workers. To reduce the loading to the 14-worker maximum, some activities will have to be delayed beyond their late start dates, which will delay the project beyond 47 weeks. With a problem like this something has to give because it is not possible to satisfy both the 14-worker limitation and the project deadline of 47 weeks. Figure 7.27 shows a schedule that satisfies the 14-worker constraint. It was determined by trial and error, making certain not to violate either the precedence requirements or the 14-worker limit. Notice that the project now requires 50 weeks to be completed because Activity X had to be delayed 3 weeks beyond its late start date.

As the example shows, a resource needed by multiple activities can dictate the project duration and override the critical path time. Consider another example from the LOGON project. Suppose one crucial resource is a technical inspector who has the skills to inspect a variety

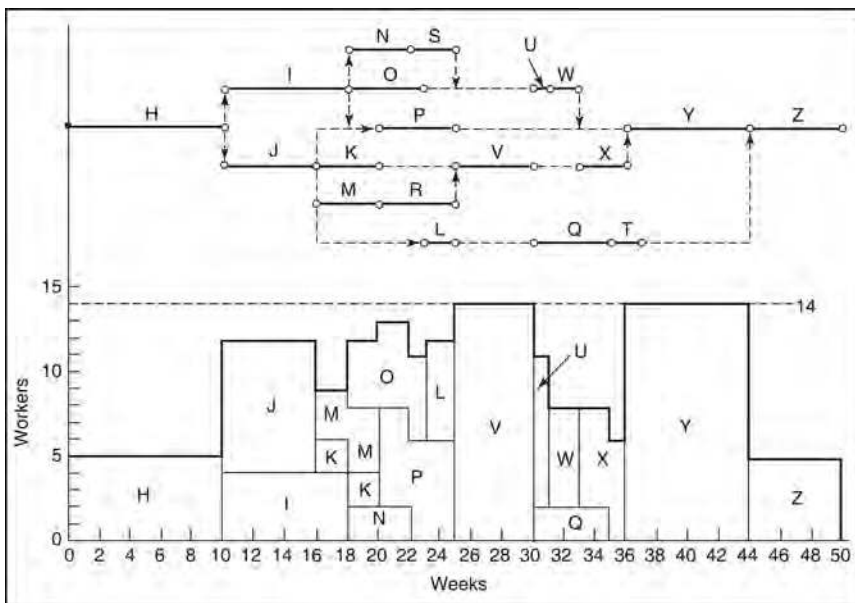


Figure 7.27

Schedule and corresponding worker loading for the LOGON Project with 14-worker constraint.

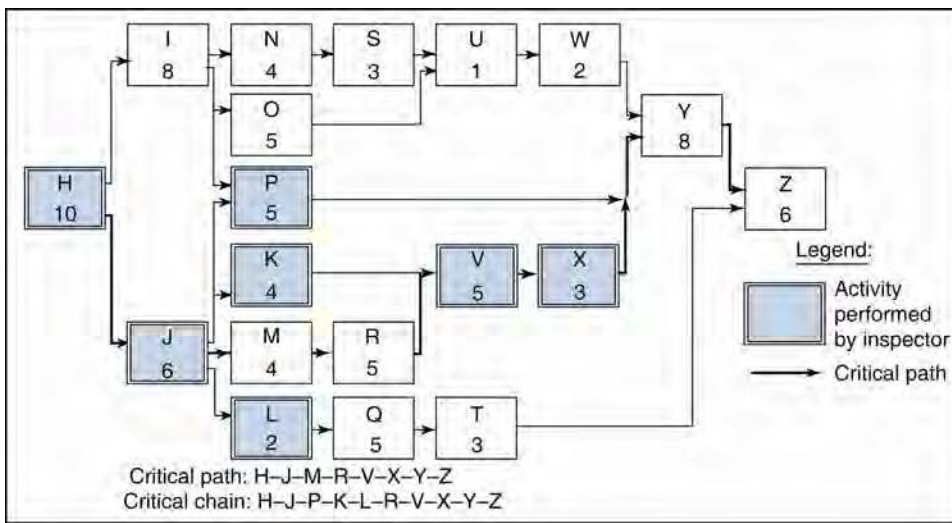


Figure 7.28

Activities in the LOGON project involving the resource of technical inspector.

of activities. Her work, however, is exacting, which prevents her from working on more than one activity at a time. Suppose the activities on which she will be working are H, J, P, K, L, V, and X. These activities are highlighted in Figure 7.28. Because she can work on only one of them at a time, the activities must be scheduled sequentially. Summing up the durations of these activities gives the time required for her to inspect all of them, 35 weeks. Add to this the times for the last two activities, Y and Z, and the total is 49 weeks. Thus, the project duration will be 49 weeks, not 47 weeks as determined by the critical path.

Goldratt calls the path connecting activities that require the same constrained resource the **critical chain** (here, H-J-P-K-L-V-X plus Y and Z) and distinguishes it from the **critical path** (H-J-M-R-V-X-Y-Z).³ Back in Figure 7.21, the critical path is A-C-D but the critical chain is A-C-B-D or A-B-C-D. The significance of this is that when activities must be performed sequentially due to a constrained resource, and when the sum of the durations of those activities, the critical chain, exceeds the length of critical path it is the critical chain—not the critical path—that determines the project duration. This is further discussed in Chapter 8.

Scheduling with constrained resources involves deciding which activities should receive resources and be scheduled immediately, and which can be delayed until resources are available.

The constrained-resource problem also occurs in multi-project organizations that draw resources from a common pool. To schedule activities for any one project, managers must account for the resources required by other, concurrent projects. The result is that schedules for some projects are determined in part by when resources will be freed-up from other, higher priority projects. Complex resource-constrained scheduling situations like this call for software-generated solutions, which typically use **heuristics** (simple rules that yield good results) to create schedules. Some of these heuristics are discussed in the next chapter.

7.7 Criticisms of Network Methods

Network methods have been criticized because they incorporate assumptions and yield results that are sometimes unrealistic. For example, they assume that a project can be completely



See Chapter 8



See Chapter 14

defined upfront in terms of identifiable activities with known precedence relationships. In many projects, however, not all work tasks can be anticipated or clearly defined at the start. Rather, the project “evolves” as it progresses, for which the agile methods described in Chapter 14 are well-suited.



See Chapters 4
and 6

A related problem is that activities and durations require periodic modification, one reason being that the network has many activities that are not well defined. This problem can be addressed by creating an initial “rough” schedule and then developing more-detailed schedules in a phased approach (discussed in Chapter 4) and by avoiding “proliferation” of activities, i.e. keeping the number of activities in the schedule to a minimum as prescribed in the work definition guidelines in Chapter 6.

In short, the failings of networks are actually inadequacies in project definition. It can be argued (and many project managers will attest) that network methods, though not perfect, offer a good approach for analyzing and creating project schedules, especially in TPM projects.

7.8 Summary

The advantage of networks is that they clearly display the interdependencies of project activities and show the scheduling impact that activities have on each other. This feature enables planners to determine critical activities and slack times, which is important for project planning and control. Knowledge of critical activities tells managers where to focus; knowledge of slack enables them to address the problems of non-uniform resource requirements and limited resources. The PDM method accounts for various relationships between project activities to better reflect the realities of project work. Network methods enable resources to be balanced or allocated to satisfy resources constraints or meet project deadlines.

The next chapter describes other well-known and more advanced network scheduling methods: PERT, simulation, time–cost tradeoff analysis (CPM), and critical chain project management (CCPM).

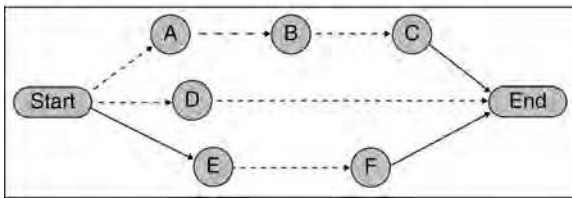
Summary list of variables

T_e	Expected project duration: the expected duration of the project based on the critical path.
T_s	Target project completion date: the contracted or committed date for project completion.
ES	Early start for an activity: the earliest feasible time an activity can be started.
EF	Early finish for an activity: the earliest feasible time an activity can be completed.
LS	Late start: the latest allowable time an activity can be started to complete the project on target.
LF	Late finish: the latest allowable time an activity can be completed to complete the project on target.
t	Activity duration: the most-likely or best-guess time to complete an activity.
$FS = n$	Finish-to-start: an activity can start no sooner than n days after its immediate predecessor has finished.
$SS = n$	Start-to-start: an activity can start no sooner than n days after the start of its immediate predecessor.
$SF = n$	Start-to-finish: an activity can finish no later than n days after its immediate predecessor has started.

- T_e Expected project duration: the expected duration of the project based on the critical path.
- FF = n Finish-to-finish: an activity can finish no later than n days after its immediate predecessor has finished.

Summary illustration problem

AON representation:



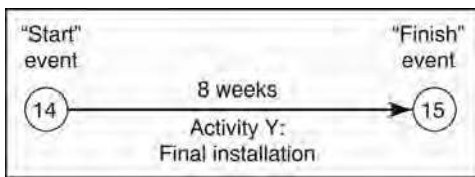
Activity	Time	Slack					
		ES	EF	LS	LF	Total	Free
A	2.0	0	2.0	0	3.0	1.0	0
B	5.0	2.0	7.0	3.0	8.0	1.0	0
C	2.0	7.0	9.0	8.0	10.0	1.0	1.0
D	5.0	0	5.0	5.0	10.0	5.0	5.0
E*	5.0	0	5.0	0	5.0	0	0
F*	5.0	5.0	10.0	5.0	10.0	0	0

*Activities on critical path

APPENDIX A: AOA DIAGRAMS

This chapter described the AON method of network diagramming. Another diagramming method is the **activity-on-arrow (AOA)** or **arrow diagramming** technique. The major feature that distinguishes AOA from AON is the way activities and events are denoted on the network. Figure 7.29 shows the AOA representation for one activity and its events: each activity is represented by an arrow between two nodes (the circles). As shown in Figure 7.29, the *nodes* represent the start and finish events for the activity, and the *arrow in between* represents the activity itself. The number inside each node identifies an event with its own unique identifier. In the example, the numbers 14 and 15 were chosen arbitrarily: node 14 represents the event “start Activity Y” and node 15 represents the event “finish Activity Y.”

The length of the arrowed line has no significance in AOA. As in AON networks, an AOA network should have only *one origin* event and *one terminal* event. All arrows generally point toward the right, toward the end of the network, and the arrows cannot double back.⁴

**Figure 7.29**

AOA representation for an activity and its start and finish events.

Now, as in the AON method, the activities follow a sequential order as defined by their immediate predecessors. When an activity has more than one immediate predecessor, the network must show it can be started only after all its immediate predecessors are completed. This is the purpose of a special kind of activity called a “dummy.”

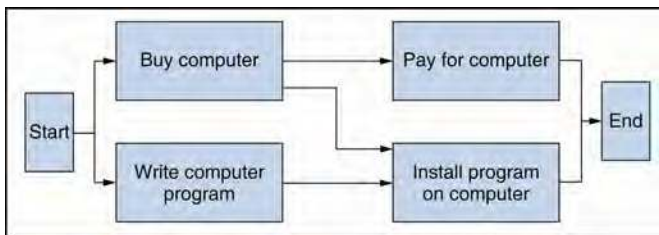
Dummy activities

A **dummy activity** is used to illustrate precedence relationships in AOA networks. It serves only as a “connector”—it is not a real activity and represents neither work nor time.⁵ As an example, a team must write a computer program, purchase a computer, and install the program on the computer. The specific activities and their dependencies are:

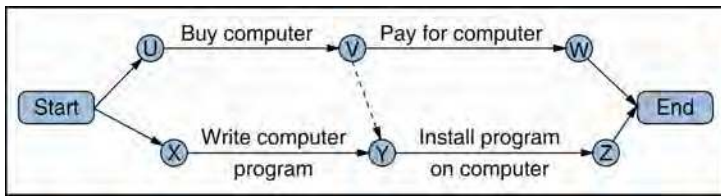
1. Buy computer
2. Write computer program
(Note: (1) or (2) can be done in either order.)
3. Pay for computer after buy computer (assume credit is okay!).
4. Install program on computer after both writing it and buying the computer.

The activities and their dependencies are illustrated in the **AON network** in Figure 7.30.

The **AOA network** for the same project is shown in Figure 7.31. Note that to show the dependencies “install program” after both “buy computer” and “write computer program” requires a dummy activity (the dashed arrow) between node V and node Y. This dummy links the start of “install program” to its two immediate predecessors, finish “buy computer” and finish “write computer program.” Note that the network has only one “Start” node and one “End” node—standard good practice.

**Figure 7.30**

AON diagram.

**Figure 7.31**

AOA diagram for AON diagram in Figure 7.30.

AON versus AOA

Since AON networks do not require dummies, they are easier to construct and interpret than AOA networks; thus, they are more popular. But because AOA diagrams use line segments (the arrows) to represent the flow of work and time, they can easily be converted into **time-scaled networks** that look like Gantt charts. Some project software packages create time-scaled networks (examples shown in Figures 7.18 and 7.20), and some create both AOA and AON network diagrams. The trend nowadays is to use AOA; AON is seldom used.

APPENDIX B: ALTERNATE SCHEDULING METHOD: PROJECT STARTS AT DAY 1

The scheduling technique illustrated in this chapter is the usual approach to introduce network scheduling. It assumes that the project *begins* at time zero and that a successor activity *begins* immediately upon completing all its predecessors. The method is simple and is mathematically correct.

For practical purposes, however, the method is incorrect. People speak of the “first day” of the project, not the “zeroth day.” Thus, they say, the project start time should be indicated as Day 1, not Day 0. Further, whenever activities are in series, each activity starts *on the period following* the completion of its predecessors, not in the same period. Thus, the network would show the early start time of an activity as being a day (or week) *after* the finish of its latest predecessor. Realistically this approach makes sense.

As an example, refer to Figure 7.32, which is Figure 7.7 revised for “Day 1” assumptions. Activity H is the first activity in the project and lasts 10 days. Using the Day 1 scheme for Activity H, $ES = 1$. In making the forward pass through the network, computationally

$$EF = ES + \text{Duration} - 1.$$

Thus, $EF = 1 + 10 - 1 = 10$. Now, the ES for Activity H’s successors, Activity I and Activity J, will be the next day, i.e., $ES = 11$.

Of course, using the Day 1 assumptions affects the late times too. Making the backward pass through the network,

$$LS = LF - \text{Duration} + 1.$$

For example, for Activity J and duration 6, if $LF = 16$ then $LS = 16 - 6 + 1 = 11$. The immediate predecessor of Activity J, Activity H, must finish the day before this, so $LF = 10$ for Activity H.

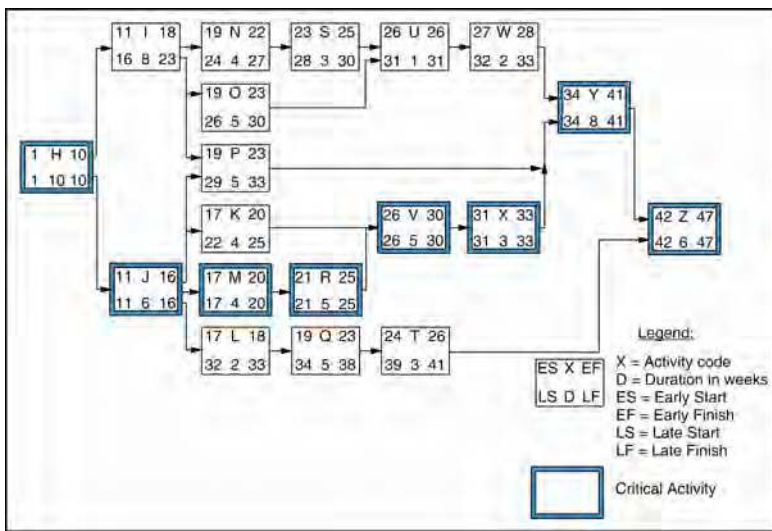
**Figure 7.32**

Figure 7.7 adjusted for “Day 1” assumptions.

The Day 1 scheme does not impact the computation of total slack, which remains the simple difference between early and late start times or early and late finish times. It does however change the computation of free slack:

$$\text{Free slack for an activity} = \text{ES (earliest successor)} - \text{EF (activity)} - 1$$

Project scheduling software uses calendar dates, not elapsed times, and the project start event will be indicated by the *date* of the first day (or week) of the project. Throughout the network, the start dates of successor activities will all be shown as the period (day or week) *after* the finish dates of their successors. In other words, scheduling software incorporates the Day-1 assumption.



Review Questions and Problems

1. What are the advantages of networks over Gantt charts?
2. Draw a high-level network diagram of college degree program, starting with enrolment and finishing with graduation. Indicate the courses, projects, and exams as well as precedence relationships where applicable.
3. How is a WBS used to create a network and what role does a scope statement play?
4. Can a Gantt chart be created from a network? Can a network be created from a Gantt chart? Which is the preferred way? Explain.
5. Why is it vital to know the critical path? Explain the different ways the critical path is used in network analysis and project planning.
6. Explain the difference between total and free slack.
7. Explain the difference between ES, EF, LS, and LF.
8. Consider each of the following projects:

- a. Composing and mailing (snail mail) a letter to a friend.
 - b. Preparing a five-course meal (you specify the main course and dishes served).
 - c. Planning a wedding for 500 people.
 - d. Building a sundeck for your home.
 - e. Planning, promoting, and conducting a rock concert.
 - f. Moving to a different house or apartment.
 - g. Developing, promoting, manufacturing, and distributing a new packaged food item.
 - h. Developing and installing a computerized information system, both hardware and software.
 - i. Remodeling a bathroom.
 - j. Adding a bedroom to a house.
- Now, answer the following questions for each project:
- k. Using your experience or imagination, create a WBS.
 - l. List the activities or work packages.
 - m. Show the immediate predecessors for each activity.
 - n. Draw the network diagram (using the AON scheme).
9. Draw the AON network diagrams for the following four projects:

a. Activity	Immediate Predecessors
A	–
B	A
C	A
D	B
E	D
F	D
G	D
H	E, F, G

b. Activity	Immediate Predecessors
A	–
B	A
C	A
D	B
E	B
F	C
G	D
H	D
I	G
J	E, F, H, I

c. Activity	Immediate Predecessors
A	–
B	A
C	–
D	–
E	D
F	B, C, E

d. Activity	Immediate Predecessors
A	–
B	–
C	–
D	C
E	A
F	B
G	E
H	F, G, J
I	A
J	D, I

10. Refer to Figure 7.1 in the text.
- a. If the person wants to get more sleep by waking up later, which of the following steps would be useful?

- i. Put socks on faster.
 - ii. Put tie in pocket and wear it later.
 - iii. Put shoes on faster.
 - iv. Buy a hair dryer that works faster.
 - b. Calculate the total float and free float of the activity “Put on socks.”
11. Eliminate redundant predecessors from the following lists so only *immediate predecessors* remain. (Example: In (a) below, the specified predecessors for G are B, D, C, E; but B and C are also predecessors for D and E, respectively, hence it is redundant to specify them. They can be removed, leaving only D and E, which are the immediate predecessors.)

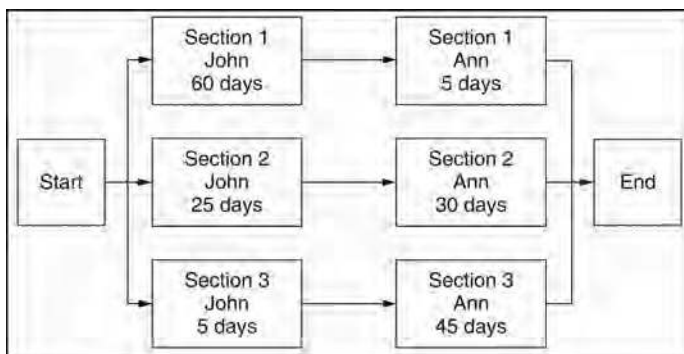
a.	Activity	Predecessors
	A	–
	B	–
	C	–
	D	B
	E	C
	F	A
	G	B, D, C, E
	H	A, B, C, D, E, F, G

b.	Activity	Predecessors
	A	–
	B	A
	C	A
	D	A, B
	E	A, B
	F	A, C
	G	A, B, C, D, E, F
	H	A, B, C, D, E, G

c.	Activity	Predecessors
	A	–
	B	–
	C	A
	D	A
	E	B
	F	B
	G	A, C
	H	A, B, D, E
	I	B, F
	J	C, D, E, F, G, H, I

12. Use Figure 7.4 (a) and (b) to draw Gantt charts for the ROSEBUD project.
13. Some projects have (a) fixed due dates and others (b) must be finished as early as possible, and the project manager commits to the completion date only after the project management team has scheduled the project. Explain how the backward pass differs for these two situations.
14. Explain how it is possible for the critical path to have non-zero slack. What is the implication of negative slack on the critical path?
15. In the development of a new (first of its kind) complex system, the design of a certain subsystem has large slack. Sufficient resources are available for either an early start or a late start. Discuss the pros and cons of early and late starts. Consider the risk of delaying the project and the risk of changes in the design, management focus, cash flow, and any other factors you can think of.⁶

16. What limitations of simple AON networks does PDM overcome? What limitations does it not overcome?
17. Give examples of applications of PDM. Take a project you are familiar with (or invent one) and create a PDM network.
18. For the PDM network in Figure 7.19, calculate ES, EF, LS, and LF for all activities.
19. To produce a three-section manual, John must write the text, after which Ann must prepare drawings and key in the final document. John can start with any section of the book (i.e., he doesn't have to start with the first section). The manual must be completed within 95 days. The network diagram below shows the precedence relationships and duration of each activity.



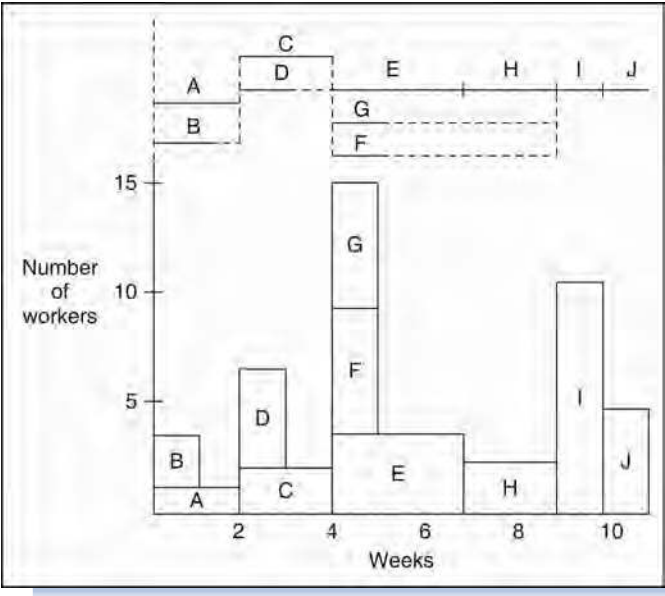
Draw a Gantt chart to show how the work can be done within 95 days. Take into account that both John and Ann are able to attend to only one task at a time.⁷

20. Why is leveling of resources preferred to large fluctuation of workload? What negative result could resource leveling cause?
21. Describe how resource leveling of a resource-constrained project differs from resource leveling in a time-constrained project.
22. The requirements for systems analysts and programmers for the GUMBY project are as follows:

Activity	J	M	V	Y	L	Q	Z
Predecessors	–	J	M	V	J	L	Y, Q
Duration (weeks)	6	4	6	8	2	8	2
Systems Analysts (weekly)	8	5	3	2	5	3	5
Programmers (weekly)	3	4	2	3	3	2	3

- a. Draw the network. Compute ESs, LSs, and total slack times.
- b. Then show the separate resource loadings for systems analysts and programmers, assuming early start times.
- c. Suppose the maximum weekly availability is eight systems analysts and five programmers. Can activities be scheduled to satisfy these constraints without delaying the project?

23. Level the resources for a project with the workload diagram below. In the time-phased diagram at the top of the figure, dotted lines indicate slack.⁸ Discuss pros and cons of the alternatives available.



24. Discuss the implications of resource allocation for organizations involved in multiple projects.
25. Show that the schedule in Figure 7.22 (which produced an erratic loading for workers) yields a more balanced loading for equipment than the one shown in Figure 7.26.
26. Suppose in Figure 7.19 everything is the same except Activity Y can start 4 days after Activity V starts, but cannot be finished until 6 days after Activity V is finished. Show how this changes the values for ES, EF, LS, and LF.
27. Redraw Figure 7.5 as an AOA diagram.
28. For each of the following predecessor tables:
- a. Draw a corresponding AON network.
 - b. Compute ES and EF for each activity.
 - c. Compute LS and LF for each activity. Find the critical path.
 - d. Determine the total slack and free slack.

a.				b.			
Activity	Predecessor	Duration		Activity	Predecessor	Duration	
A		6		A		3	
B		3		B	A	8	
C	A	9		C	B	9	
D	B	5		D	C	3	
E	B	4		E	B	2	
F	D	2		F	E, H	4	
G	E	8		G	A	6	
				H	G	5	
				J	D, F	1	

c.	Activity	Predecessor	Duration	d.	Activity	Predecessor	Duration
	A		9		A		10
	B	A	2		B	A, E	9
	C		8		C	B, N	15
	D	C	8		D	C	7
	E	B, D	7		E		5
	F	E	4		F	A, E	6
	G	C	4		G	K, F	7
	H	B, D, G	3		H	G	12
	J		6		J		12
	K	J	10		K	E, J	4
	M	G, K	3		L	K, F	11
	N	H, M	6		M	L	8
					N	E, J	7



Questions About the Study Project

1. Were networks used for scheduling? If so, describe the networks. Show examples. What kind of computer software was used to create and maintain them? Who was responsible for software data inputs and software operation? Describe the capabilities of the software.
2. At what point in the project were networks created? When were they updated?
3. What was used first to develop the schedule: (a) a table such as Table 7.3, (b) a network diagram, or was (c) the Gantt chart drawn first? Comment on the method used.
4. Was all detail planning done upfront or was a phased approach followed?
5. Was there a schedule reserve (buffer)? How was it determined and included in the schedule?
6. Were charts of resource workloads created?
7. If the project was done within a matrix structure, how did communication between the functional and project managers take place?
8. Did the functional manager(s) take responsibility for resource workloads?
9. Was resource leveling done?
10. Were there any complaints about unrealistic workloads?

CASE 7.1 NETWORK DIAGRAM FOR A LARGE CONSTRUCTION PROJECT⁹

The table below lists activities for constructing a bridge over an operational railway line. This project is similar to the bridge described in Case 11.3.

Activity No.	Activity Description	Duration (Months)	Immediate Predecessors
A	Detailed site investigation and survey	2	-
B	Detailed planning	6	A
C	Detailed design	6	B
D	Preparation of site	4	C
E	Relocate services	3	C
F	Re-align overhead track electrification	4	C, E
G	Access road and ramp construction	1	D
H	Piling	2	G
J	Construct foundations and abutments	3	H
K	Construct temporary supports to support bridge deck during construction	2	F, G
L	Fabrication planning of structural steel components	2	C
M	Manufacture structural steel components (off-site)	2	L
N	Transport structural steel components and erect on-site	1	M
P	Erect pylons and fill with concrete	2	J
Q	Construct main span deck on pre-cast concrete beams	3	H, K, N, P
R	Install stay-cables and lift the bridge deck off temporary supports	3	Q
S	Remove temporary supports	1	R
T	Electrical system installation	1	S
U	Roadway surfacing (paving)	2	S
V	Finishing and ancillaries	2	T, U
W	Commissioning—cut-over	1	V
X	Formal hand-over and ceremony	1	W
Y	Project sign-off	1	X
Z	Administrative closure	1	W
AA	Project End	0 (milestone)	Y, Z

1. Construct a network diagram for the project.
2. Do forward and backward pass calculations to determine early and late start and finish times.
3. Identify the critical path and the project duration.
4. Compute the total and free slack of each activity.

5. The following resources are required to perform the activities. Allocate the resources to the activities and indicate the workload on the resources. If needed, adjust the schedule.

Activity No.	Activity Description	Resources
A	Detailed site investigation and survey	Surveyors, Engineering, Project Manager
B	Detailed planning	Project Manager, Engineering, Construction, Contractors
C	Detailed design	Engineering
D	Preparation of site	Construction
E	Relocate services	Engineering
F	Re-align overhead track electrification	Engineering, Contractors
G	Access road and ramp construction	Construction
H	Piling	Construction, Contractors
J	Construct foundations and abutments	Engineering, Construction
K	Construct temporary supports to support bridge deck during construction	Engineering, Construction
L	Fabrication planning of structural steel components	Engineering, Manufacturer
M	Manufacture structural steel components (off-site)	Engineering, Manufacturer
N	Transport structural steel components and erect on-site	Transporter, Engineering
P	Erect pylons and fill with concrete	Construction, Engineering
Q	Construct main span deck on pre-cast concrete beams	Construction, Engineering
R	Install stay-cables and lift the bridge deck off temporary supports	Construction, Engineering
S	Remove temporary supports	Construction, Engineering
T	Electrical system installation	Construction, Engineering
U	Roadway surfacing (paving)	Contractor, Engineering
V	Finishing and ancillaries	Contractors, Engineering
W	Commissioning—cut-over	Project Manager, Engineering, Construction, Contractors
X	Formal hand-over and ceremony	Project Manager, Engineering, Construction, Contractors
Y	Project sign-off	Project Manager, Engineering
Z	Administrative closure	Engineering
AA	Project end	Project Manager

CASE 7.2 MELBOURNE CONSTRUCTION COMPANY, A

Bill Asher, scheduler for Melbourne Construction Company, has created a network of activities for a hotel project the company is planning. Figure 7.33 shows part of the network and the estimated number of days for each activity.

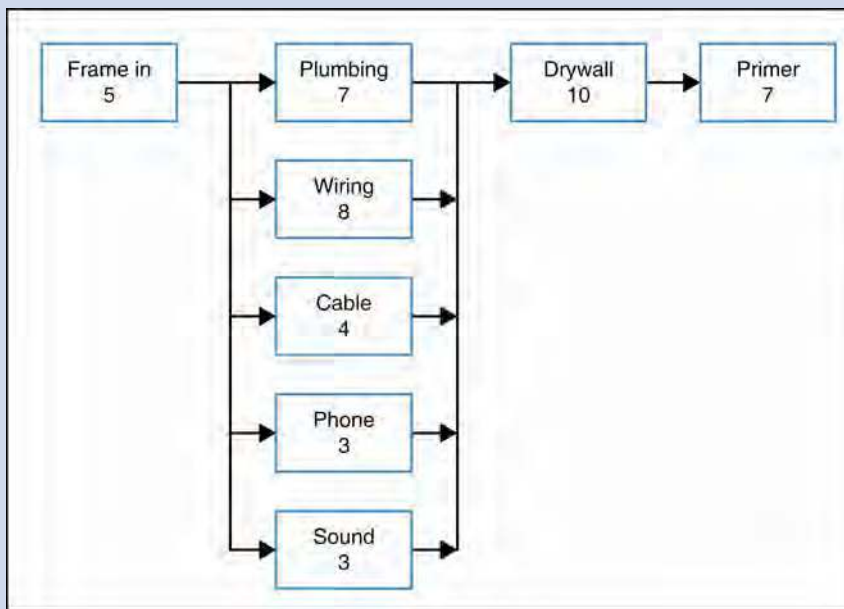


Figure 7.33
Partial network for hotel construction project.

1. What are the early and late start and finish times for all the activities?
2. What is the earliest this portion of the project will be completed?

Project schedulers often face constraints that originate outside the project (external dependencies). For example, materials might not arrive or a contractor might not be ready until a particular date, which imposes a constraint on when the work can start—a “start no earlier than” (**SNET**) date. At other times a customer, inspector, or someone else will require the work to be completed by a particular date—a “finish no later than” (**FNLT**) date. Bill faces such a situation. He has been informed that drywall boards will not arrive at the site until Day 15, i.e., drywall has a SNET date of 15. What effect will this delay have on the project schedule? Additionally, the owner of Melbourne Construction, Naomi Watts, wants to give the hotel owner a tour of the building, but not until all the walls have been primed. She has scheduled the tour on Day 29, which imposes a FNLT of Day 28 on Primer. Bill is now faced with this requirement plus the drywall delivery constraint. Is it possible to finish this portion of the project on Day 28? If not, what adjustments must be made to the work so it can be? Bill is meeting with Naomi to discuss the situation.

CASE 7.3 MELBOURNE CONSTRUCTION COMPANY, B

One way to speed up a project is to speed up activities on the critical path (discussed in the next chapter); another is to *overlap* activities. Refer to Case 7.2 above: the network diagram in Figure 7.33 shows the activities for *one floor* of the hotel and assumes *finish-to-start* relationships (FS), i.e., successors can start only upon completion of predecessors. Now suppose each floor is large enough so that the crew for any activity can begin work when the crews for its predecessor activities are only *partially* completed (a *start-to-start* relationship or SS), meaning the start of an activity *lags* the start of its predecessors by some specified amount; this allows activities normally in sequence to be overlapped. Ordinarily, an SS lag is used when successor activities are *slower* than immediate predecessor activities (so successors won't "catch up" with predecessors and have to wait on them). This is the case for the Plumbing and Wiring activities that succeed Frame-in, so Bill assigns an SS lag of 3 days between Frame-in and Plumbing and Wiring (meaning Plumbing and Wiring can start 3 days *after* Frame-in starts); he does the same for Cable, Phone, and Sound. This is shown in Figure 7.34. The same case applies to Drywall installation, which is slower than its immediate predecessors, so Bill assigns an SS lag of 4 days between Drywall installation and its predecessors.

When activities are to be overlapped but successors are *faster* than predecessor activities, a finish-to-finish (FF) lag can be used. Primer is faster than Drywall, so Bill inserts an FF lag of 3 days between them. This means Primer should finish no earlier than 3 days before Drywall finishes—also shown in Figure 7.34.

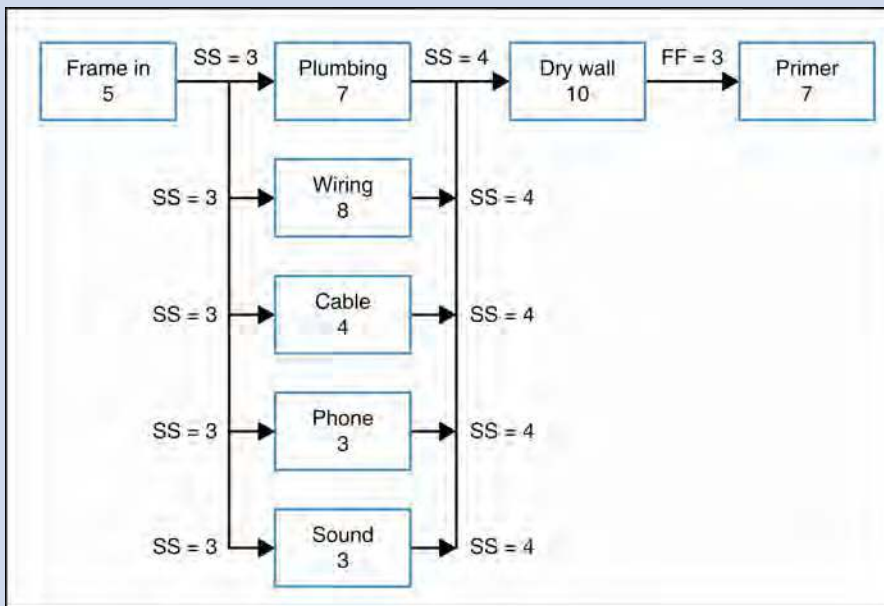


Figure 7.34
Network with lags inserted.

QUESTIONS

1. Prepare two Gantt charts, one using the original FS relationship and one using the SS and FF lags. Compare the charts. By how much do the lags speed up the project? Since Cable, Phone, and Sound take less time than Frame-in, what potential problem might occur in overlapping them with Frame-in?
2. Again refer to Case 7.2. Given that drywall board delivery will not happen until Day 15 and Naomi has scheduled a tour for Day 29, what is the effect of the SS and FF lags? Can Naomi conduct her tour as planned?

CASE 7.4 MELBOURNE CONSTRUCTION COMPANY, C

Bill Asher has created a network of activities for a three-story boutique hotel the company is planning to build on the Mornington Peninsula. Bill identified seven major activities for each floor. Figure 7.35 shows the network of activities for the three floors and the number of days estimated for each activity. Each activity will be done by a different subcontractor. As shown in the network, upon completing work on one floor, the subcontractor moves to the next floor.

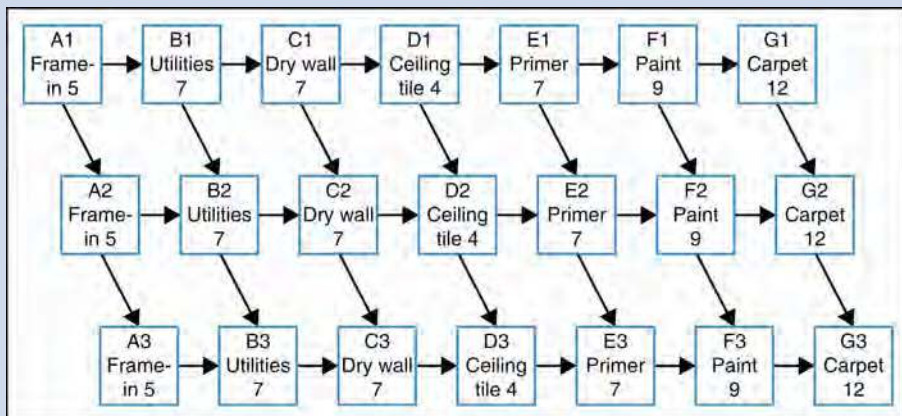


Figure 7.35
Network for three floors.

QUESTIONS

1. What is the critical path? Based on Bill's estimates, how long will it take to complete the three floors?
2. The activity times shown in Figure 7.35 are based on Bill's estimates of total labor hours per activity and an 8-hour workday. For example, Bill estimated that Frame-in will require 40 labor hours; given an 8-hour day, he came up with 5 days. Thus, all the times in Figure 7.35 assume that subcontractors assign a "crew" of one worker to each activity. According to these time estimates, it should

take 51 days to complete each floor. Bill's boss, Naomi Watts, says that 51 days per floor is too long, and that to complete the project on time each floor should take no more than 35 days.

Looking at the seven activities per floor, Bill sees that 35 days per floor could be achieved if **each activity took no more than 5 days**. He intends to point this out to his subcontractors.

3. Bill's estimates assume one worker per activity. How many workers (what crew size) should the contractors assign to each activity such that it will take at most 5 days?
4. Given the increased crew sizes, how long will it take to complete the project? Assume that a computed fractional day's duration is always rounded up to a full day.

Notes

1. Duncan, W. (ed.) Project Management Institute. *A Guide to the Project Management Body of Knowledge*, 1st edn. Newton Square, PA: Project Management Institute; 1996. The definition of the critical path in later editions of this document does not say that the critical path can change; that does not alter the fact that it does.
2. For more about PDM scheduling, see Dreger J.B. *Project Management: Effective Scheduling*. New York, NY: Van Nostrand Reinhold; 1992.
3. Goldratt E.M. *Critical Chain*. Great Barrington, MA: North River Press; 1997.
4. Loops are permitted in a special form of network analysis called GERT.
5. Adapted from Gordon G. and Villoria R. *Network-based Management Systems (PERT/CPM)*. New York, NY: John Wiley & Sons; 1967.
6. From Steyn H. (ed.) *Project Management: A Multi-disciplinary Approach*. Pretoria: FPM Publishing; 2003. Reproduced with permission.
7. Steyn, *Project Management*.
8. Steyn, *Project Management*.
9. Steyn, *Project Management*.

Chapter 8

Advanced Project Network Analysis and Scheduling

The scheduling methods discussed in Chapter 7 assume that activity times are known and fixed, even though in reality they are estimated and variable. This chapter discusses the implications of variable activity times on project schedules, and ways for shortening project duration, starting with the CPM method.

Example 8.1: The House Built in Fewer than 4 Hours¹

With virtually unlimited resources and meticulous planning and control, a project can be done *very* fast. An example is the Manukau, New Zealand Chapter of Habitat for Humanity (a nonprofit organization dedicated to eliminating poverty housing), which set a record for building a house: 3 hours and 45 minutes.

The project specifications included construction of a four-bedroom house on an established foundation (Figure 8.1). It incorporated prefabricated wall panels, wooden floor, roofing iron, ceilings, decks, and steps. Doors, windows, bath, toilet, plumbing, and the electrical system had to be installed and ready for use; walls, ceilings and window frames had to be painted; and carpets and curtains had to be installed. The specifications also included a path to the front door, mail letter box, installed clothesline, wooden fence around the yard, three trees planted, and a leveled lawn with grass. The new owners, Mr. and Mrs. Suafoa, watched the construction with their four children while CNN filmed the event. The house was inspected, passed all local building codes, and the keys were handed over to the family.

What made speedy completion possible? *First* were abundant resources: 150 people, mostly volunteers. *Second* was comprehensive and meticulous preparation: 14 months of planning, including many iterations of network analysis. The detailed plan was recorded on special task sheets so team leaders could hand over tasks from one to another without deliberation. With so many people and construction items at the site, workspace was at a premium. A crane was provided to lift the wooden roof frame onto the wall structure. *Third* was a systematic computerized method for planning, monitoring, and controlling the project that included the *critical chain method* and *time buffers* (explained later). The bathroom-



Figure 8.1
The house built in fewer than four hours.
Source: Photo courtesy of Habitat for Humanity.

fitting task was estimated to take 30 minutes but took 1 hour; the 30-minute overrun was absorbed in the time buffer. *Finally*, the project made use of suitable technology, including prefabricated walls and components.

8.1 Reducing Project Duration with CPM

With enough resources and meticulous planning, most any project can be shortened. But to avoid wasting resources, certain principles must be applied, and these are incorporated in the **critical path method, CPM**.² CPM is a mathematical procedure for estimating the tradeoff between project duration and project cost to determine the least expensive way to shorten project duration.

Time–cost relationship

CPM assumes that the time to perform a project activity can be varied, depending on the amount of resources (labor, equipment, etc.) applied; as more resources are applied to particular activities, the project duration is shortened. Adding resources speeds up the project, but it also increases the project cost. A major element of project cost is labor: a project can be sped up by working more hours or adding more workers, but either way the cost goes up.

Ordinarily, work on any given activity in a project is performed at a so-called **normal** (usual and customary) work pace—the “normal” point shown in Figure 8.2. Associated with this pace is the **normal time**, T_n , which is the time to do the activity under normal work

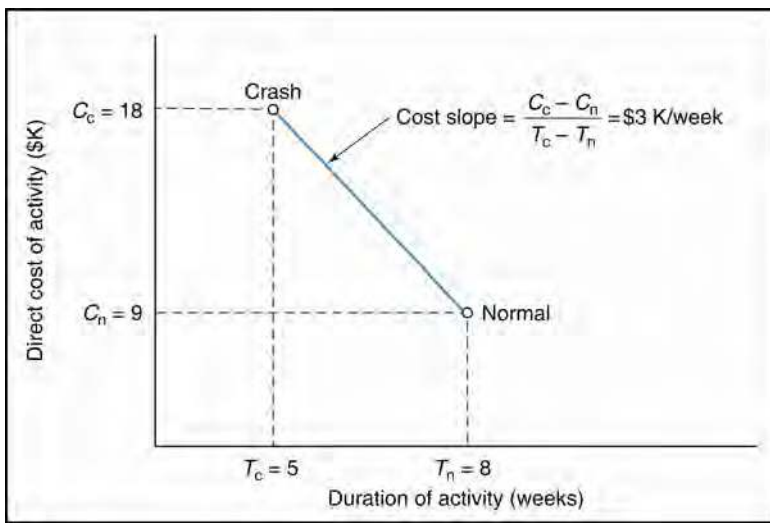


Figure 8.2

Time-cost relationship for an activity.

conditions, and the **normal cost**, C_n , which is the cost to do the activity in the normal time. (The normal pace is assumed to be the most efficient and thus *least costly* pace. Taking longer than normal will not reduce the cost.)

To reduce the time to complete the activity, more resources are applied in the form of additional personnel or overtime. As more resources are applied, the duration shortens but the cost increases. When the maximum effort is applied so the activity can be completed in the *shortest possible time*, the activity is said to be **crashed**. The crash condition (see Figure 8.2) represents not only the shortest duration, but the costliest as well.

Something to keep in mind is that some activities, called *process limited*, cannot be sped up; they require a specific amount of time, regardless of resources. Fermenting wine or curing concrete are examples.

As illustrated in Figure 8.2, the normal conditions and crash conditions define two theoretical extremes. The line connecting the points, called the **cost slope**, represents the time-cost relationship or marginal time-cost tradeoff for the activity. The time-cost relationship for every activity is unique and can be linear, curvilinear (concave or convex), or a step function. The nature of the actual time-cost relationship is usually unknown; thus, often it is assumed to be linear,³ in which case the formula for the cost slope is

$$\text{cost slope} = \frac{C_c - C_n}{T_c - T_n}$$

where C_c and C_n are the crash and normal costs, respectively, and T_c and T_n are the crash and normal times for the activity. The cost slope is how much it would cost to speed up or slow down the activity. In Figure 8.2, which represents the time-cost tradeoff for a particular activity, the cost slope is \$3K per week. That means that for each week the activity duration is *reduced* from the normal time of 8 weeks, the cost will *increase* by \$3K. Completing the activity 1 week earlier (from 8 weeks to 7 weeks) would increase the cost from the normal cost of \$9K to the

“sped up” cost of $\$9K + \$3K = \$12K$; completing it another week sooner (in 6 weeks) would increase the cost to $\$12K + \$3K = \$15K$; and completing it yet another week sooner (in 5 weeks) would increase the cost to $\$18K$. According to Figure 8.2, this last step puts the activity at the crash point, 5 weeks, the shortest duration for the activity.

Shorten the critical path

The cost-slope concept can be used to determine the least costly way to shorten a project. Figure 8.3 and Table 8.1 illustrate this with an example. Start with the preliminary project schedule by assuming a normal pace for all activities; therefore, the project in the figure can be completed in 22 weeks at an expense of $\$55K$. Recall from Chapter 7 that the project duration is the length of the critical path. In general, what that implies is that if you want to shorten the project, you must shorten the critical path. Because the critical path A–D–G is the longest path (22 weeks), to shorten the project it is necessary to shorten a critical activity—A, D, or G. Reducing an activity increases its cost, but because the reduction can be made anywhere on the critical path,



See Chapter 7

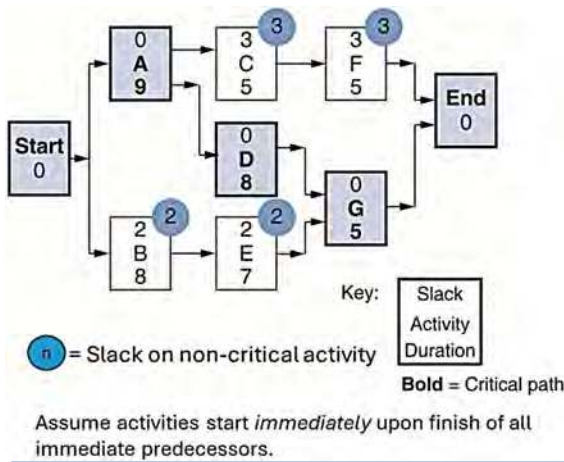


Figure 8.3

Time-cost trade off information for example network.

Table 8.1 Normal and crash duration and cost, cost slope, and slack.

Normal	Crash	Cost	Slack			
Activity	T_n	C_n	T_c	C_c	slope	given T_n
A	9	10	6	16	2	0*
B	8	9	5	18	3	2
C	5	7	4	8	1	3
D	8	9	6	19	5	0*
E	7	7	3	15	2	2
F	5	5	5	5		3
G	5	8	2	3	5	0*
Total		\$55K		\$104K		*Critical path

the increase can be minimized by selecting the activity with the smallest cost slope; this is Activity A. Reducing A by 1 week shortens the project duration to 21 weeks and adds \$2K (cost slope of A) to the project cost, bringing it to $\$55K + \$2K = \$57K$. This step does not change the critical path so, if need be, another week can be cut from A, resulting in project duration of 20 weeks and cost of $\$57K + \$2K = \$59K$.

In general, each time an activity is shortened, it is necessary to check for changes in the critical path. For example, look at the network in Figures 8.3 and 8.4. As Figure 8.4(a) shows, shortening A by two weeks uses up the slack on Path B–E, and the project now has two critical paths: A–D–G and B–E–G. Any further reduction in project duration must be made by shortening both paths (since shortening just one would leave the other at 20 weeks). The least costly way to reduce the project to 19 weeks is to reduce both A and E by 1 week, shown in

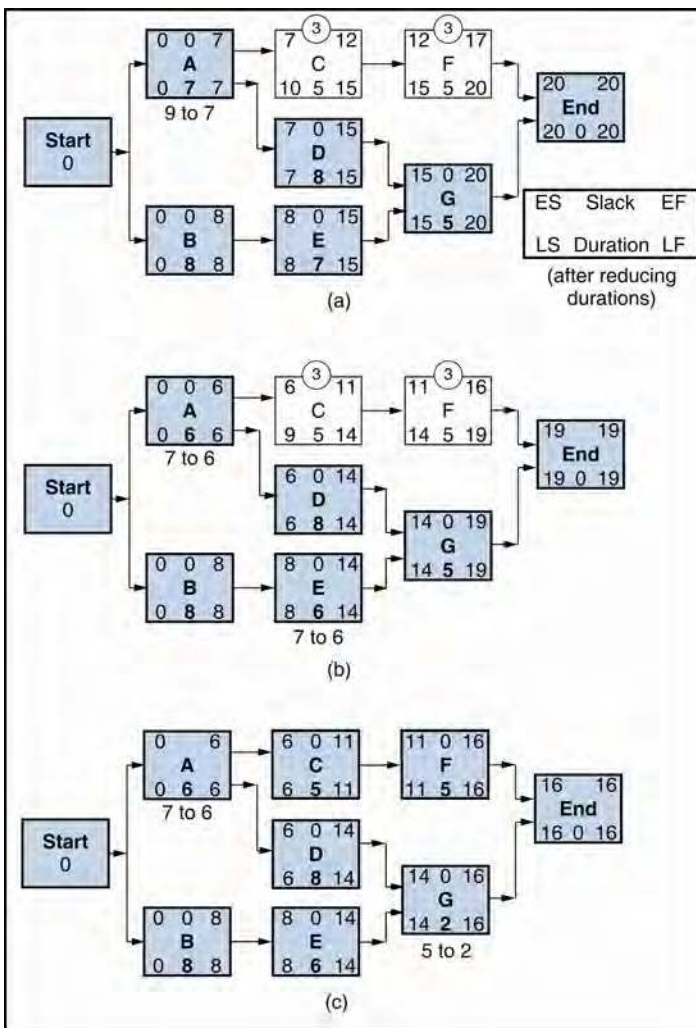


Figure 8.4
Process for reducing project duration.

Figure 8.4(b). The additional cost is \$2K for A and \$2K for E, so the resulting project cost would increase to \$59K + \$2K + \$2K = \$63K. This last step reduces A to 6 weeks, its crash time, so no further reductions can be made to A.

If a further reduction in project duration is desired, the least costly way to shorten both paths is to reduce G. In fact, because the slack on the noncritical path C–F is 3 weeks, and because the crash duration for G is 2 weeks (which means, if desired, 3 weeks can be taken out of G), the project can be reduced to 16 weeks by shortening G by 3 weeks, shown in Figure 8.4(c). This adds \$5K per week, or $3 \times \$5K = \$15K$, to the project cost. With this last step, all slack is used up on Path C–F, and all the paths in the network (A–C–F, A–D–G, and B–E–G) become critical.

Any further reductions desired in the project duration must shorten *all three critical paths* (A–C–F, A–D–G, and B–E–G). As you may wish to verify, the most economical way to reduce the project to 15 weeks is to cut 1 week each from E, D, and C, bringing the project cost up to \$86K. This step reduces the time of C to its crash time, the shortest possible project duration. The sequence of steps is summarized in Table 8.2.

Crashing the project: shortest project duration

The time–cost procedure described determines which activities to speed up, step-by-step, to reduce the project duration. This stepwise procedure will eventually lead to the shortest project duration and its associated cost. However, if we want to *directly* find the *shortest possible project duration* and avoid the intermediate steps, a simpler way is to simultaneously *crash* all activities at once. This, as shown in Figure 8.5(a), yields the project duration of 15 weeks. However, the expense of crashing all activities, \$104K (table in Figure 8.3) is unnecessarily high because, as will be shown, *not all* activities must be crashed to finish the project in the shortest time.

The project duration of 15 weeks is the length of the critical path. Because the critical path is the longest path, other (noncritical) paths are of shorter duration and, consequently, have no influence on project duration. Thus, it is possible to “stretch” or lengthen any noncritical activity by a certain amount without extending the project duration. In fact, noncritical activities can be stretched until all their slack is used up.

Just as reducing an activity’s time from the normal time increases its cost, so *extending* its time from the crash time *reduces* its cost. As a result, by extending noncritical activities the \$104K project crash cost can be reduced. To do so, start with those noncritical activities that will yield the *greatest* savings—those with the greatest cost slope. Notice in Figure 8.5(a) that because Path B–E–G has a slack of 5 weeks, activities along this path can be stretched by up

Table 8.2 Duration reduction and associated cost increase.

Step	Duration (T_e , weeks)	Activities on CP with Least Cost Slope	Cost of Project (K\$)
1*	22		\$55
2	21	A (\$2)	$\$55 + \$2 = \$57$
3	20	A (\$2)	$\$57 + \$2 = \$59$
4	19	A (\$2), E (\$2)	$\$59 + \$2 + \$2 = \$63$
5, 6, 7	18, 17, 16	G (\$5)	$\$63 + \$5, \$68 + \$5, \$72 + \$5 = \$78$
8	15	E (\$2), D (\$5), C (\$1)	$\$78 + \$2 + \$5 + \$1 = \$86$

* Duration and cost using normal conditions.

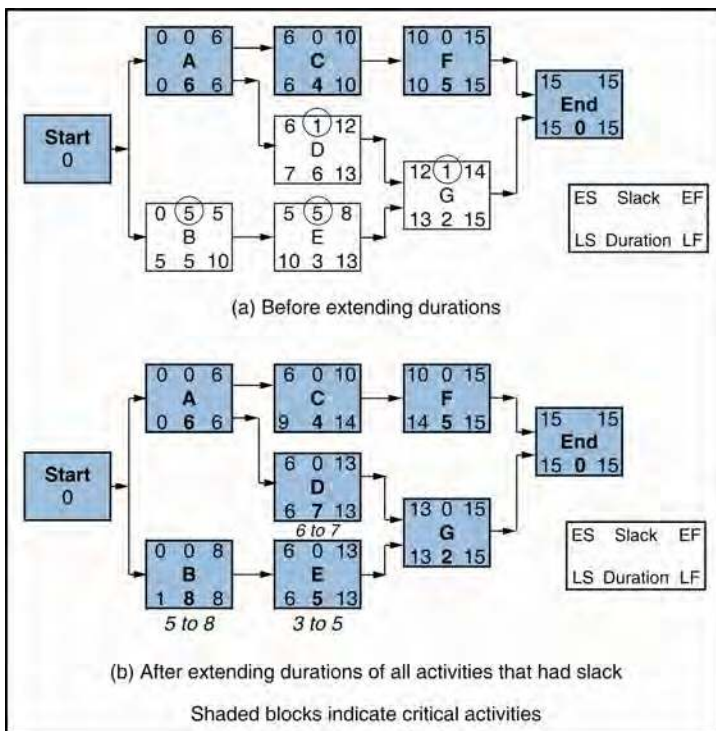


Figure 8.5
Example network
using crash times.

to a total of 5 weeks without extending the project. Three weeks could be added to Activity B (bringing it to the normal duration of 8 weeks) without lengthening the project. Also, 2 weeks could be added to E and 1 week to D, both without changing the project duration. (But remember: in extending an activity's duration, the duration can never exceed the activity's normal time.) The final activity times are shown in Figure 8.5(b). Notice, all paths are now critical.

The final project cost is computed by subtracting from the initial crash cost the savings obtained in extending B by 3 weeks, E by 2 weeks, and D by 1 week.

$$\$104\text{K} - 3(\$3\text{K}) - 2(\$2\text{K}) - 1(\$5\text{K}) = \$86\text{K}$$

In summary, to obtain the shortest project duration, first crash all activities (called **crashing the project**), then stretch the noncritical activities, starting with the one with the greatest cost slopes first, to use up available slack and obtain the greatest cost savings. An activity can be stretched, at most, to its normal duration, which is assumed to be its least-costly time (Figure 8.2).

Lowest total project cost

The previous analysis dealt only with **direct costs**—costs immediately associated with individual activities and that increase directly as resources are added to them. But beyond direct costs, the cost of conducting a project also includes **indirect costs** such as administrative and

overhead charges. (The distinction between direct and indirect cost is elaborated in the next chapter.) Usually, indirect costs are a function of, and are proportionate to, the project duration, which is to say that *indirect costs* (in contrast to direct costs) *decrease as project duration decreases*.

The mathematical function for indirect cost can be derived by estimation. As an illustration, suppose indirect costs in the previous example are approximated by the formula

$$\text{Indirect cost} = \$10K + \$3K(T_c)$$

where T_c is the expected project duration in weeks. Figure 8.6 shows this indirect cost as a function of project duration; it also shows the direct cost, obtained from Table 8.2. **Total project cost** is the sum of direct and indirect costs. Notice from the figure that by combining direct costs and indirect costs it is possible to determine the project duration that results in the lowest total project cost. In Figure 8.6, from a cost standpoint the “optimum” project duration is 20 weeks.

In addition to direct and indirect costs, some other costs that influence total project cost (and hence the optimum T_c) are **contractual incentives** such as **penalty charges** or **bonus payments**. As described in Chapter 12, a penalty is a fee imposed on the contractor for the project not meeting schedule or performance requirements, and a bonus is a reward for exceeding requirements.

Suppose in the previous example the contract specified a target completion date of Week 18, with a bonus of \$2K per week for finishing before 18 weeks and a penalty of \$1K per week for finishing after 18 weeks. Figure 8.7 shows the influence of these incentives on total project cost. Notice that even with incentives, the optimum duration (for the contractor) is 19 or 20



See Chapter 12

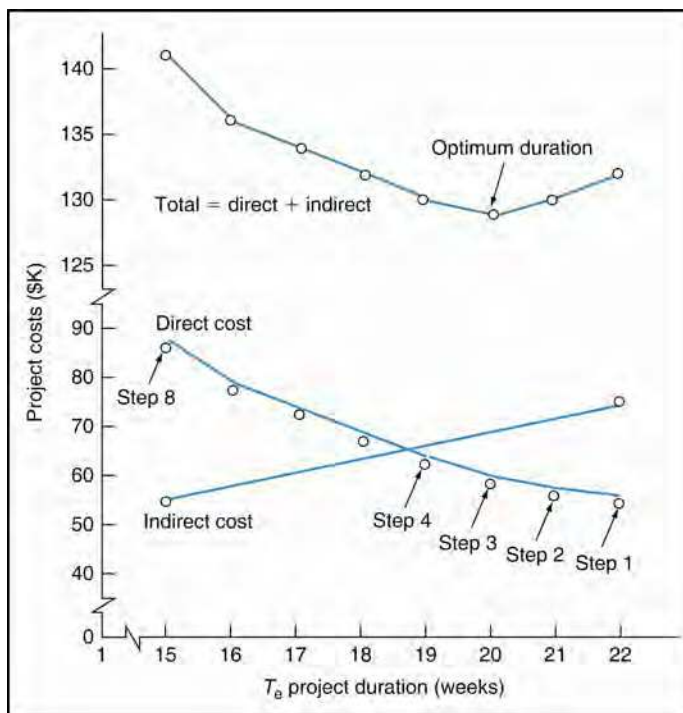


Figure 8.6
Total time-cost
tradeoff for the
project.

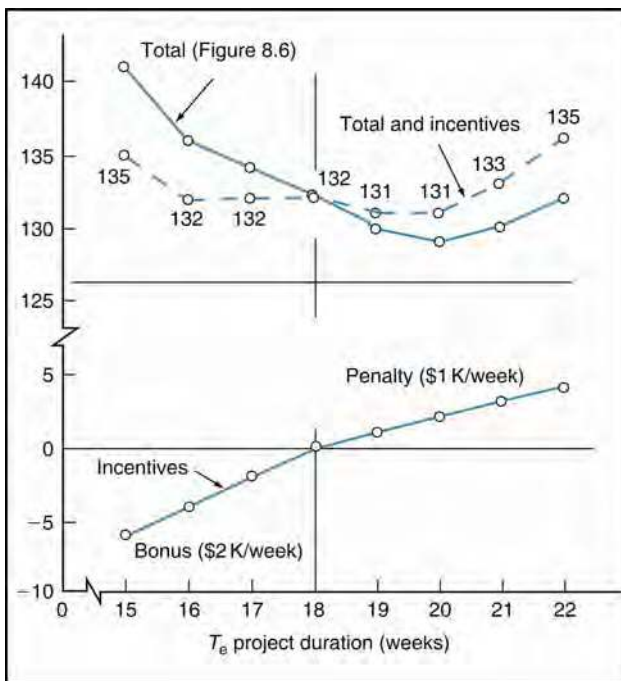


Figure 8.7
Time-cost tradeoff for the project with incentives.

weeks, not the contractual 18 weeks. This example reveals that a formal contractual incentive alone is not necessarily enough to influence performance. For the incentive to motivate the contractor it must have “teeth”; i.e., it must be big enough with respect to other project costs to influence the contractor’s performance. Had the penalty been raised to \$3K (instead of \$1K) per week for finishing after 18 weeks, the contractor’s optimum duration would have shifted to 16 weeks.

8.2 Variability of Activity Duration

Suppose you are driving in a car to somewhere, and Figure 8.8 shows the estimated time it will take you to get there. If everything goes well (no traffic or mechanical problems) you will arrive in the shortest time—the “optimistic duration.” However, “most likely” it will take you about 30 minutes. Of course, it could take longer than this—say, when traffic is congested or, worst case, you get in an accident. In the figure, note that the area below the curve to the left of the most likely duration is much less than to the right of it. This indicates that the chances of arriving later are much greater than the chances of arriving earlier.

Like your travel time, the activity durations in a project are variable. The question is, since you cannot say for sure how long each activity in a project will take, how can you possibly say when the project will be completed?

The scheduling approaches discussed in Chapter 7 and the preceding section on time-cost tradeoff ignore variability and assume that activity durations are known and constant; this is called the **deterministic** approach. In the following sections we consider what happens when the activity durations are assumed variable; this is called the **stochastic** approach.



See Chapter 7

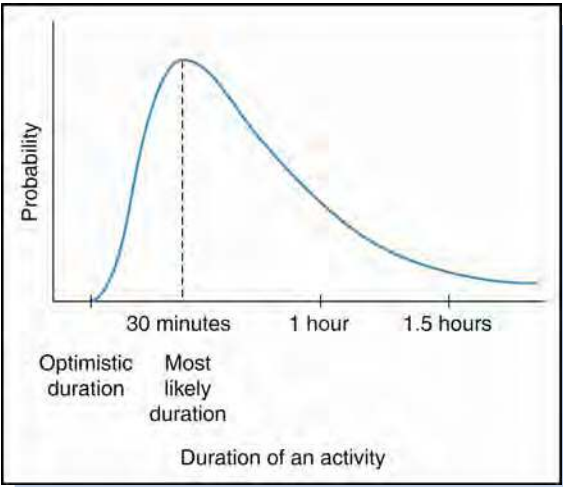


Figure 8.8
Variability of activity duration.

Variability effects on a project network

In a project, some activities will be completed earlier than expected, others later. When combined in a network, however, the earlier-than-expected activities and later-than-expected activities do not average out: in general, it is the *later-than-expected* activities that impact the project completion. This is one reason why projects tend to take longer than estimated.

For example, consider the project schedule Figure 8.9. If Activity A takes longer than planned, it would delay Activity B, which in turn would delay Activities C and D and, thus, the completion of the project. Suppose, however, that Activity A is finished *earlier* than planned. In that case will Activity B start earlier? *Not necessarily*. Resources needed for Activity B (people and equipment) will likely have other commitments, which would preclude Activity B starting early. So, Activity A finishing later delays the project, but Activity A finishing earlier does not necessarily speed it up. Consider a second example. Most project networks consist of several paths that merge into a critical path. Figure 8.10(a) illustrates a project with two critical paths, each with a 50 percent chance of finishing on time. The probability that the project will finish on time is the probability that both paths will finish on time, or 25 percent (0.5×0.5). Figure 8.10(b) shows five paths merging (which is typical of what happens near the end of project networks), each with a 50 percent probability of finishing on time. The probability of finishing the project on time is now about 3 percent $(0.5)^5$. This effect is called **merge bias** or **merge-point bias**.

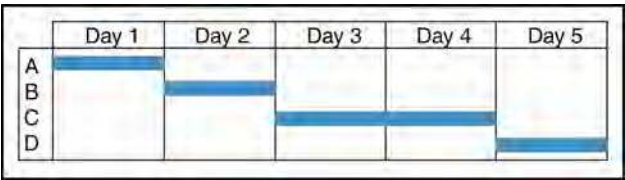


Figure 8.9
Activities delayed if Activity A is delayed.

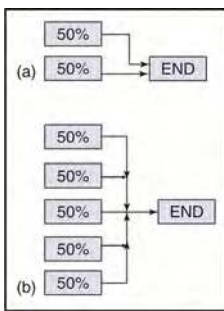


Figure 8.10

Activities delayed where paths merge. (a) Two paths merging, each with 50 percent chance of being on time; (b) Five paths merging, each with 50 percent chance of being on time.



See Chapter 7

Chapter 7 addressed the fact that the critical path is not necessarily stable but can change if noncritical activities take longer than planned and noncritical paths become critical. Thus, while delays on critical activities delay the project, long-enough delays on non-critical activities also delay the project.

Several methods have been developed to help grapple with the uncertainty about activity durations and when a project will be completed. These are addressed in the following sections, starting with PERT.

8.3 PERT

The **PERT method** originated during the US Navy's Polaris Missile System program, an example of a complex research and development program with much uncertainty about the kind of research and stages of development to be done, and how fast they can be completed. In projects like this, project definition is occurring at the same time as technological developments are unfolding and before many of the problems with technology, materials, and processes have been identified. The project activity durations are uncertain and there is great risk that the project will exceed the target completion date.

To provide greater certainty in estimating the duration of the Polaris Missile program, an operations research team was formed in 1958 with representatives from the Navy's Special Projects Office, the consulting firm of Booz, Allen, and Hamilton, and the prime contractor Lockheed Missile Systems. They devised a method called PERT (**Program Evaluation and Review Technique**) that would provide insight into the likelihood of finishing a project by a certain time.⁴ PERT is not a scheduling tool, per se, but is a method for *analyzing the project duration*.

Three time estimates

The network methods discussed in Chapter 7 determine the critical path and slack times using *best estimates* for activity durations. PERT, however, addresses uncertainty in the activity durations by using *three time estimates*—optimistic, most likely, and pessimistic. Presumably the estimates are obtained from the people most knowledgeable about difficulties likely to be encountered and the potential variability in time; usually they are expert estimators or have performed similar activities or will perform or manage the activity.

The three estimates are used to calculate the **expected time** for an activity. The range between the optimistic and pessimistic estimates is a measure of variability that permits making statistical inferences about the likelihood that a project event will happen by a particular time.

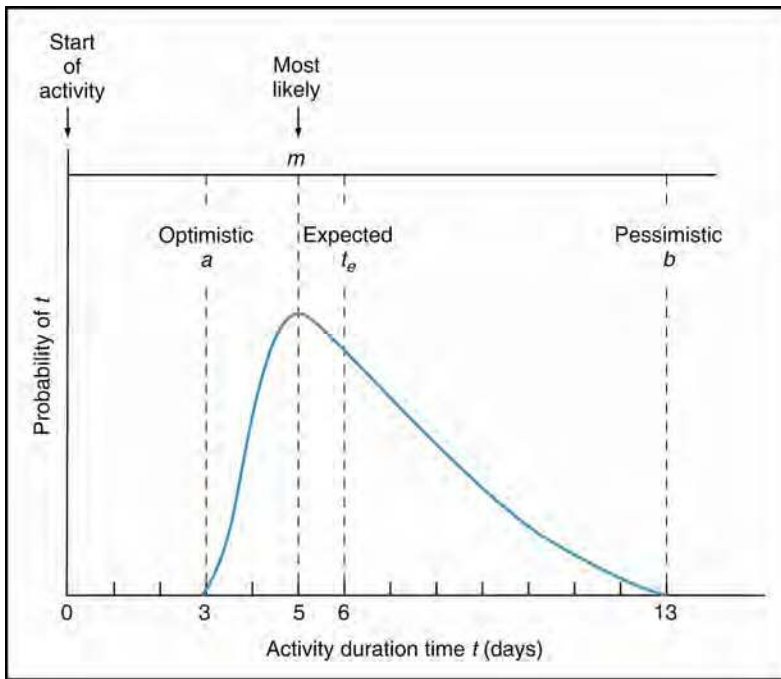


Figure 8.11

Parameters for estimating mean and standard deviation of activity duration.

Shown in Figure 8.11, the **optimistic time**, a , is the minimum time for an activity—the situation where everything goes well and there is little possibility of finishing earlier. A normal level of effort is assumed with no extra personnel. The **most likely time**, m , is the time that would occur most often if the activity were repeated. Finally, the **pessimistic time**, b , is the longest time for the activity—the situation where bad luck is encountered at every step; it includes likely problems in the work, but not highly unlikely occurrences such as natural disasters.

The three estimates are related in the form of a **Beta** probability distribution with parameters a and b as the end points and m , the most likely value. The Beta distribution is used because it is unimodal (has a single peak value) and is not necessarily symmetrical—properties that seem desirable for a distribution of activity durations. Note that whereas the distribution in Figure 8.8 had no end point on the right-hand side, the curve in Figure 8.11 precludes very unlikely events and has end point b .

Given the three time estimates and assuming a Beta distribution, the **mean** or **expected time**, t_e , and the **variance**, V , of each activity are computed with the following formulas:

$$t_e = \frac{a + 4m + b}{6}$$

$$V = \left(\frac{b - a}{6} \right)^2$$

Since $V = \sigma^2$, where σ = standard deviation,

$$\sigma = (b - a) / 6$$

The expected time, t_e , represents the duration in Figure 8.11 with a 50–50 chance that the activity will be completed earlier or later than t_e . In the figure

$$t_e = \frac{3 + 4(5) + 13}{6} = 6 \text{ days}$$

The variance, V , is a measure of variability in the activity duration:

$$V = \left(\frac{13 - 3}{6} \right)^2 = (1.67)^2 = 2.78$$

The larger the V , the less reliable t_e , and the higher the likelihood the activity will be completed much earlier or much later than t_e . This simply reflects that the farther apart a and b , the more dispersed the distribution and the greater the chance that the actual time will significantly differ from the expected time. In routine jobs, estimates of a and b are close to each other, V is small, and t_e is more likely. In unique, one-of-kind jobs, a and b are far apart.

Probability of finishing by a target completion date

The expected time, t_e , is used in the same way as the estimated activity duration was used in the deterministic networks in Chapter 7. Because statistically the expected time of a sequence of independent activities is the sum of their individual expected times, the expected duration of the project, T_e is the sum of the expected activity durations along the critical path; that is

$$T_e = \sum_{CP} t_e$$

where each t_e is the expected time of an activity on the critical path.

Since PERT is a stochastic approach, the project duration T_e is not considered a fixed point but rather an estimate subject to uncertainty owing to the uncertainties of the activity durations along the critical path. Because the project duration T_e is computed as the sum of expected (but uncertain) activity durations, it follows that T_e is also an expected but uncertain duration. The project duration can be thought of as a probability distribution with an *average* of T_e , where the probability of completing the project before T_e is 50 percent, and so is the probability of completing it after T_e .

The variation in the project duration distribution, V , is computed as the sum of the variances of the activity durations along the critical path:

$$V_p = \sum_{CP} V$$

These concepts are illustrated in the network in Figure 8.12 and Table 8.3.

Unlike the distribution of activity durations, which is Beta, the distribution of *project durations* is assumed to be the **normal distribution**—the familiar bell-shaped curve. Given this assumption, it is easy to determine the probability of meeting any specified project target completion date T_s .

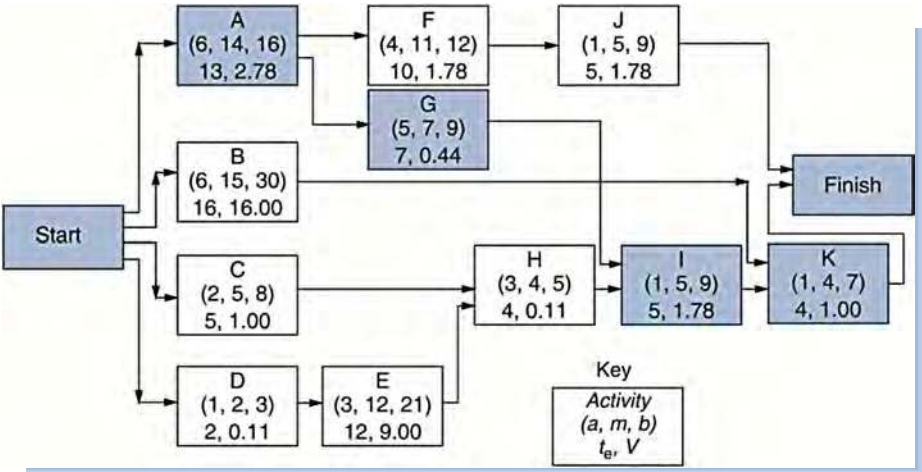


Figure 8.12
PERT network with expected activity durations and activity variances.

Table 8.3 PERT network expected activity durations and activity variances.

Path	$T_e = \sum t_e$	$V_e = \sum V$
(a)	A-F-J	28** 6.34
(b)	B-K	20 17.00
(c)	A-G-I-K	29* 6.00
(d)	C-H-I-K	18 3.89
(e)	D-E-H-I-K	27** 12.00

*Critical path
**Near-critical paths

As examples, consider two questions about the project in Figure 8.12: (1) What is the probability of completing the project in 27 days? (2) If we want to be 95 percent sure about meeting a project deadline, what deadline should we quote? To answer these questions, we invoke the assumption that the distribution of project duration is a standard normal distribution, and we begin by determining the number of standard deviations, z , that separate T_s from the expected project duration, T_e . The formula for the calculation is:

$$z = \frac{T_s - T_e}{\sqrt{V_p}}$$

To answer the first question, use $T_s = 27$ days. From the network critical path, the expected project duration, T_e , is 29 days. Therefore

$$z = \frac{27 - 29}{\sqrt{6}} = -0.82$$

Therefore, the probability of completing the project within 27 days is equal to the area under the normal curve to the left of $z = -0.82$, which, referring to Table 8.4(a), is about 21 percent.

To answer the second question (95 percent certainty of meeting deadline): from Table 8.4(b), for a probability of 0.95, the z value is 1.6. Using the z formula, then solving for T_s :

$$1.6 = \frac{T_s - 29}{\sqrt{6}}, \text{ so } T_s = 33 \text{ days}$$

In other words, it is 95 percent probable (“highly likely”) that the project will be completed within 33 days. Note that since we are working with values that are merely estimates, there is no point computing figures of great precision.

Near-critical paths

The PERT procedure has been criticized for providing overly optimistic results, a justifiable criticism since it does not account for the effect of **merge-point bias**. For example, Figure 8.12 has two paths that are **near critical** in length. The variance of these paths is large enough that either path could easily become critical by exceeding the 29 days of the current critical path. In fact, as you may wish to verify using the statistical procedure described previously, the probabilities of not completing Path (a) (A–F–J) and Path (e) (D–E–H–I–K) within 29 days are 34 percent and 29 percent, respectively. So, there is more than a slight chance that these paths will become critical. The warning is: *don’t ignore the near-critical paths*—paths that could become critical and jeopardize the project completion date. Furthermore, the 50 percent probability of completing the project within 29 days, as presumed with the normal distribution, is overly optimistic. Because all activities in the network must be completed before the project is completed, the probability of completing the project within 29 days is the same as the probability of completing all five paths within 29 days. Although the probability of completing Paths (b) and (d) within 29 days is close to 100 percent, the probabilities of completing Paths (a) and (e) within that time is $100 - 34 = 66$ percent and $100 - 29 = 71$ percent, respectively, and the probability of completing (c), the critical path, is 50 percent. So, the chance of completing all five paths within 29 days is less than 25 percent ($1.0 \times 1.0 \times 0.66 \times 0.71 \times 0.5$).

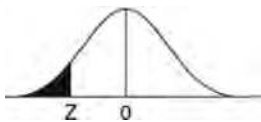
Meeting the target date

Clearly, one way to increase confidence in meeting a target date is to delay it, but when the target date cannot be delayed the alternative is to revise the project network and shorten the critical and near-critical paths, so T_c is somewhat less than T_s . Ways to do this include:⁵

1. Look for opportunities to **fast-track** (overlap) activities on the critical path, which implies scheduling an activity to start before its predecessors are completed. An alternative is to split the predecessors into sub-activities and start successors when only some of the predecessor sub-activities have been completed.
2. Employ more resources or transfer resources from non-critical activities with large slack times to critical and near-critical activities.
3. Substitute time-consuming activities with ones that are less so, or delete activities that are not absolutely necessary.
4. Crash activities on the critical path.

Table 8.4 Normal distribution function for completing a project completion by time T_s .

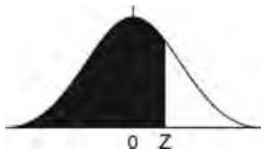
(a) Probability that project will be completed on or before T_s , when $T_s < T_e$.



Z Value	Probability
0	0.50
-0.1	0.46
-0.2	0.42
-0.3	0.38
-0.4	0.34
-0.5	0.31
-0.6	0.27
-0.7	0.24
-0.8	0.21
-0.9	0.18
-1.0	0.16
-1.1	0.14

Z Value	Probability
-1.2	0.12
-1.3	0.10
-1.4	0.08
-1.5	0.07
-1.6	0.05
-1.7	0.04
-1.8	0.04
-1.9	0.03
-2.0	0.02
-2.1	0.02
-2.2	0.01
-2.3	0.01

(b) Probability that project will be completed on or before T_s when $T_s > T_e$.



Z Value	Probability
0.0	0.50
0.1	0.54
0.2	0.58
0.3	0.61
0.4	0.66
0.5	0.69
0.6	0.72
0.7	0.76
0.8	0.79
0.9	0.82
1.0	0.83
1.1	0.86

Z Value	Probability
1.2	0.88
1.3	0.90
1.4	0.92
1.5	0.93
1.6	0.95
1.7	0.96
1.8	0.96
1.9	0.97
2.0	0.98
2.1	0.98
2.2	0.99
2.3	0.99

Each of these has drawbacks. Fast-tracking increases the risks of having to repeat activities in case of changes or mistakes. Adding resources to speed up activities increases the cost, and transferring them between activities involves changes to plans and schedules, increases administrative costs, and aggravates the managers who supply those resources. The final alternative, substituting or eliminating activities, jeopardizes project performance, especially when it equates to making “cuts” or using poorer quality materials or lower-skilled labor.

Criticisms of PERT⁶

The PERT method assumes that activity durations are independent, though often they are not. When resources are transferred from one activity to another, the durations of both are affected.

PERT also assumes that three activity estimates are better than one. Unless based upon good historical data, however, the three estimates are still guesses, which might not improve upon a single “best” guess. An advantage of the pessimistic estimate, however, is that it allows for the possibility of setbacks, which a single estimate cannot.

The accuracy of estimates often depends on experience. Whenever a database is formed, based upon experience from similar activities in previous projects, a “history” can be developed for each kind of activity that can be used to estimate the durations of future activities. In fact, reliance on good historical data for estimating times makes PERT more appropriate for somewhat “repeatable” projects and less so for the first-of-a-kind projects. Thus, the PERT method tends to be used in construction and standardized technical projects, but seldom elsewhere.

Some of PERT’s shortcomings are addressed by Monte Carlo simulation.

Monte Carlo simulation of a PERT network

Monte Carlo computer simulation is a procedure that considers the effects of near-critical paths and merge-point bias. The critical path is computed from activity durations that are randomly selected from probability distributions. The procedure is repeated thousands of times to generate a distribution of project durations. It gives an expected time and standard deviation for the project duration that is more reliable and accurate than simple PERT computations, and it also gives the probabilities that other paths will become critical.⁷

Simulation allows the use of a variety of probability distributions besides Beta, including distributions based on empirical data. As a result, the generated project durations more accurately represent the range of expected durations than the single-network PERT method.

Simulation can also avoid some limitations of PERT assumptions, such as the normal distribution of the project durations, and the duration of one activity having no influence on the duration of another. The following example from Evans and Olson illustrates simulation to assess the likelihood of project completion time.⁸

Example 8.2: Simulation to Determine Project Duration

Refer to the project activities and time estimates in Table 8.5 and the network in Figure 8.13.

The critical path is B-F-G-H-I-K-M-O-P-Q; summing t_e and V on this path gives a project duration of 147.5 days with a variance of 56.63.

Table 8.5 Activities and time estimates.

Activity	Predecessors	Minimum	Most Likely	Maximum	t_e	V
A Select steering committee	–	15	15	15	15	0
B Develop requirements list	–	40	45	60	46.67	11.11
C Develop system size estimates	–	10	14	30	16	11.11
D Determine prospective vendors	–	2	2	5	2.5	0.25
E Form evaluation team	A	5	7	9	7	0.44
F Issue request for proposal	B, C, D, E	4	5	8	5.33	0.44
G Bidders' conference	F	1	1	1	1	0
H Review submissions	G	25	30	50	32.5	17.36
I Select vendor short list	H	3	5	10	5.5	1.36
J Check vendor references	I	3	3	10	4.17	1.36
K Vendor demonstrations	I	20	30	45	30.83	17.36
L User site visit	I	3	3	5	3.33	0.11
M Select vendor	J, K, L	3	3	3	3	0
N Volume sensitive test	M	10	13	20	13.67	2.78
O Negotiate contracts	M	10	14	28	15.67	9
P Cost–benefit analysis	N, O	2	2	2	2	0
Q Obtain board of directors' approval	P	5	5	5	5	0

Source: Evans J. and Olson D. *Introduction to Simulation and Risk Management*. Upper Saddle River, NY: Prentice Hall; 1998, p. 116, with permission.

Suppose the customer wants the project to be completed within 140 days. Using the PERT method, the probability of completing the project within 140 days is found from

$$z = \frac{140 - 147.5}{\sqrt{56.65}} = -0.996$$

Referring back to Table 8.4(a), the probability is about 16 percent.

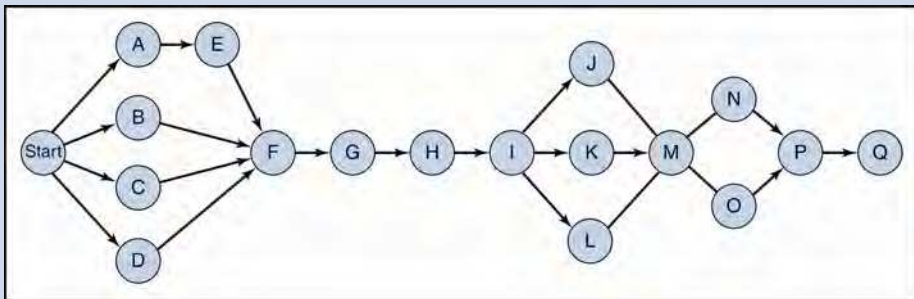


Figure 8.13
Project network.

Using the simulation program Crystal Ball™ to generate the completion times for 1,000 replications of the project yields the distribution in Figure 8.14. (Various other programs such as Risksim™, @Risk™, or Simul-8™ could also be used.)⁹

The simulated distribution has a mean of 155 days and gives a probability of completing the project in 140 days of about 6.9 percent (the sum of the probabilities to the left of 140 on Figure 8.14). It is thus unlikely that the project will be finished in less than 140 days, and only 50 percent likely that it will be completed within 155 days, which is 7.5 days longer than the PERT estimate of 147.5 days.

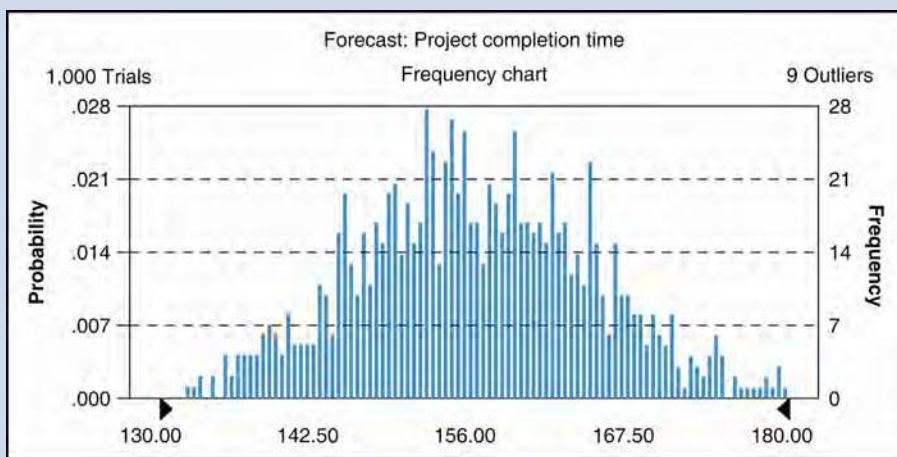


Figure 8.14
Crystal Ball simulation results for project completion times.

Simulation provides more realistic results than PERT because it compensates for noncritical paths that could become critical. But like PERT, it is merely a method for analyzing project durations, not for scheduling. It is a “better” analysis tool than PERT but, like PERT, does not eliminate the uncertainty associated with scheduling or specify what to do to reduce project risk; other tools are needed for that, as discussed in Chapter 11.



See Chapter 11

Why projects are often late

The project manager might face considerable risk when committing to a target due date as determined solely by the duration of the critical path. For example, according to Table 8.6 and Figure 8.15, adding the critical-path activity durations gives a most-likely project duration of 130 days. However, according to a Monte Carlo simulation to calculate the probability of finishing the project for those same critical-path activities (lower part of Figure 8.15), the chance of finishing the project in 130 days is only 15 percent. Further, the simulation accounted for

Table 8.6 Values a , m , and b for critical-path activities.

Activities on Critical Path	Optimistic Duration a	Most Likely Duration m	Pessimistic Duration b
A	10	12	15
B	14	15	17.5
C	18	20	22
D	12	13	14.5
E	22	24	27
F	14	15	17
G	13	14	15.5
H	16	17	19
Sum of most likely durations		130	

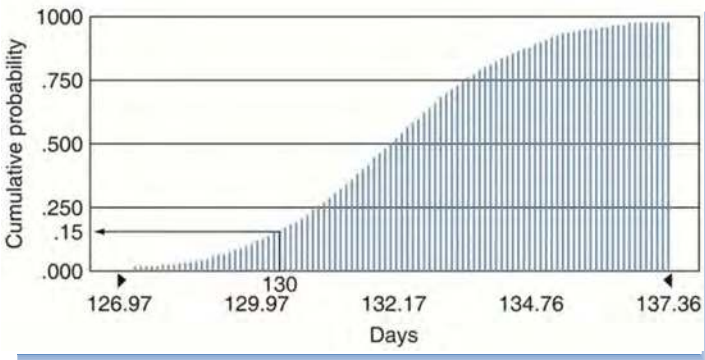


Figure 8.15

Simulation results show low probability of finishing within critical-path time. Generated by Crystal Ball software, assuming triangular distributions.

the critical path only and not for noncritical paths that might become critical—which would reduce the probability even more. While individual m (most likely) values might be considered “realistic,” the sum of the m values is not realistic at all! The project manager faces a similar risk when committing to a project cost that is derived from the sum of the most likely activity cost estimates. Many project managers estimate project duration and cost by simply adding up most likely estimates of activity durations and costs—one reason why projects overrun due dates and budgets.

Another reason for overruns is human behavior. During the feasibility or proposal (tendering/bidding) stage of the project, champions and supporters work hard to “sell” the project. Everyone is optimistic, which is necessary to gain buy in from stakeholders, but also leads to underestimating project duration and cost. The Channel Tunnel is an example. Originally it was stated that 30 million people and 100 million tons of freight would be transported through the Chunnel *per year*, a claim that proved slightly exaggerated since in the first 5 years the actual numbers were 28 million people and 12 million tons of freight. The cost, initially estimated at £7.5 billion, ultimately reached £15 billion, and the project took nearly 18 months longer to complete than originally estimated.

8.4 Allocating Resources and Multiple Project Scheduling



See Chapter 15 draw. This section addresses the matter of scheduling multiple projects that share the same resources. For example, organizations in construction, consulting, systems development, and maintenance commonly use a pool of shared skilled workers and equipment from which all projects

draw. The same happens in matrix organizations (Chapter 15) where projects share resources from the functional departments.

Multiple projects that share resources must be planned and scheduled such that in combination they do not exceed the resources available in the shared pools. Although these projects might in all other ways be considered independent, with regard to shared resources they are interdependent.



See Chapter 19 multiple projects, discussed in see Chapter 19. Each project has its own target completion date, and a project must be scheduled to finish as close to that date as possible to avoid deferred payments, penalty costs, or lost sales and revenues. Further, when projects are interdependent, delays in one project can ripple to others (the delay of a satellite development project will cause subsequent delay of a telecommunication-network project). In any case, scheduling of multiple projects requires first determining the relative priority among the projects to determine which project should get first dibs on scarce resources.

Because most organizations prefer to maintain a uniform level of personnel and other resources, the combined schedule for multiple projects ideally results in a uniform loading of those resources. In other words, the resource loading for the combined projects is ideally flat. In theory, projects are scheduled so that as resources are released from one project they are assigned to others. This minimizes costs associated with hiring, layoffs, and idle workers and facilities, and helps maintain worker morale and efficient use of resources.

When many activities or projects are ready to start and all require the same resource, the question is, to which activities or projects should the resource be allocated? When 10 activities are ready to start, the number of possible sequences in performing them is $10!$ (10 factorial), or more than 3.6 million. If n activities are ready to start and all of them require m resources, the number of possible schedules would be $(n!)^m$. Schedule optimization using

normal polynomials is usually not feasible (the problem is “NP hard”—very difficult), so heuristic methods must be used.

Heuristic methods for allocating resources

A heuristic method is a procedure based upon a simple rule. Heuristic methods for allocating resources to projects often employ *priority rules* or *dispatching rules*. While these methods do not produce optimal schedules, they do produce “good enough” schedules for most situations.

Heuristic methods start with early and late times as determined by traditional network methods, and then analyze the schedule for the required resources (i.e., the resource loading). Whenever a resource requirement exceeds the constraint, the heuristic determines which activity gets high priority and should receive the resource. The most common heuristic rules for determining scheduling priority are:

- a. **As soon as possible:** activities that can be started sooner take priority over (or scheduled ahead of) those that can be started later.
- b. **As late as possible:** activities that must be finished earlier take priority over those that can be finished later.
- c. **Most resources:** activities requiring more resources take priority over those requiring fewer resources.
- d. **Shortest task time:** activities with shorter durations take priority over those with longer durations (also referred to as **shortest activity duration**, **shortest processing time**, or **shortest operating time**).
- e. **Least slack:** activities with less slack take priority over those with more slack; critical path activities thus have highest priority (also referred to as **slack time remaining**).
- f. **First come, first served:** activities that originate earlier or require work or resources earlier take priority.
- g. **Earliest due date:** activities or projects with the earliest target completion dates take priority. Alternatively, activities with the earliest next operations take priority.

All of the priority rules are subordinate to precedence requirements, i.e., no matter the rule, the resulting schedule must not violate predecessor–successor relationships. Most project management software employ some combination of these rules (e.g. using *shortest task time*, then using *as soon as possible* as a tie breaker).

Figure 8.16 shows examples of applying the above rules a through e in assigning workers to activities and their impacts on the project schedule given a constraint of ten workers per week maximum. As the figure shows, the rules yield different results, some better than others. With the *as late as possible* rule, some activities are delayed from their early dates; the drawback of this is that it increases the risk of delaying the project. In contrast, the *least slack* rule is good since it reduces the risk of noncritical activities delaying critical ones.

The *shortest task time* rule is good when multiple projects must be executed at once, since it allows people responsible for succeeding activities to perform them sooner. Figure 8.17 shows what happens when activities are scheduled (a) *longest-task time* versus (b) *shortest-task time*. As represented by the area under the bars, the total waiting time in (b) is much less than in (a). The rule says: when you have several things to do, do the shortest ones first.

The typical scheduling goal is to complete the project by the target completion date; sometimes that is not possible, regardless of the priority rule. For example, suppose the target completion date for the project in Figure 8.16 is 9 weeks, the critical path length. Given the

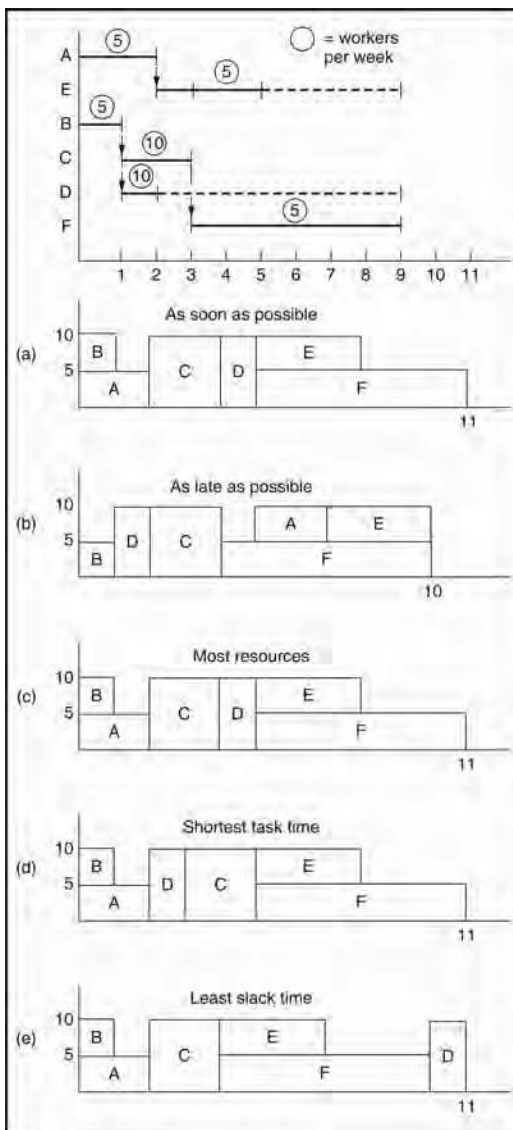


Figure 8.16
Results of different priority rules
on project schedule and completion
times.

constrained-resource level of ten workers, none of the heuristics in the example meets this target, although one of them (*as late as possible*) results in 10-week completion.

8.5 Theory of Constraints and Critical Chain Method

The **theory of constraints (TOC)** is a systems approach to improving the performance of business systems.¹⁰ A premise of TOC is that every system has a goal and that, often, only one element of the system, called the **system constraint**, precludes achieving that goal.

The application of TOC to project scheduling and control is called **critical chain project management (CCPM)** or the **critical chain method**. The role of CCPM is to reduce the

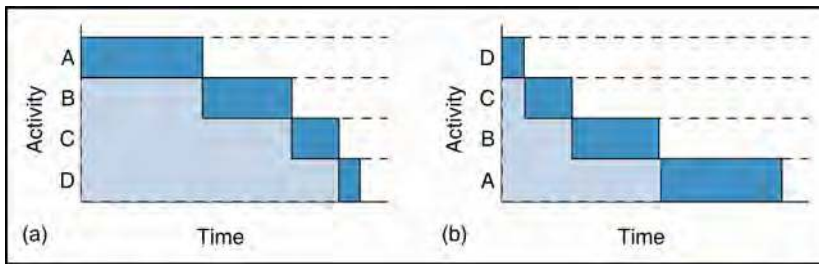


Figure 8.17

(a) Longest task time; (b) shortest task time. The shortest task time rule reduces waiting time (light shaded area).

duration of the project and provide a more predictable completion date. CCPM accounts for both the stochastic nature of activity durations and the human behavioral impact on project scheduling and execution. It can be applied to single projects or multiple concurrent projects.

Commitment to due dates

With traditional scheduling methods, people working on each project activity must commit to completing the activity by a target date (e.g., the scheduled early or late finish date) even though the activity duration is uncertain. There might be penalties for finishing late or rewards for finishing early, but out of fear of finishing late, people responsible for an activity often provide a time estimate that is pessimistic or “padded.” Although this behavior is customary, it results in inflated estimated activity durations. Since not every activity necessarily requires pessimistic durations, padding all activities results in longer-duration projects.

The CCPM approach to avoid time padding is twofold: (1) While the project manager obviously must commit to a due date for the project, people responsible for individual project activities are not required to commit to due dates. They are encouraged to work in earnest but are not held to target dates. In requesting time estimates, everyone is asked to provide a “realistic” time, meaning the time with an excellent chance of being achieved. (2) The time estimate is then cut in half to obtain an “aggressive” estimate, which eliminates any possible padding. (It should be noted, however, that whenever project members promise to provide “aggressive” (not padded) estimates, those estimates are not cut in half.) Of course, removing padding leaves the project vulnerable to delays and increases the risk of it not meeting the project due date—unless the project due date is protected with a time buffer.

Project buffer and feeding buffers

To protect against unforeseen delays, a **project time buffer** (or *time contingency*) is placed at the end of the project. The date at the end of the buffer is the date to which the project manager commits to completing the project. But you might ask, won’t adding a time buffer lengthen the project duration? The answer is no, because the aggressive time estimates used to schedule project activities are typically only half the “realistic” estimates originally provided. Dividing the realistic estimate by two results in two estimates: one, an aggressive estimate for the duration; the other, an estimate of the padding. Summing estimates of the padding for all the

activities forms the so-called *project time buffer*. Placing this buffer at the end of the project will account for delays in any of the activities.

To illustrate, consider the project shown in Table 8.7 and Figure 8.18. The critical path, P–Q–R–Z, is 32 weeks long. Now look at Figure 8.19 where the activity durations have been cut in half. The 16 weeks (4 + 6 + 4 + 2) cut from the critical path P–Q–R–Z becomes the project buffer. Note also in Figure 8.19 the presence of a **feeding buffer**, which is the number of weeks cut from noncritical path S–T, 12 weeks (2 + 10). In general, CCPM calls for a single project buffer at the end of the critical path as well as a feeding buffer located wherever a

Table 8.7 Activities for small system development project.

Activity Description (from WBS)	Activity Code	Duration (Days)	Resources
Design Subsystem A	P	8	Design Team A
Manufacture Subsystem A	Q	12	Technician
Test Subsystem A	R	8	Test team
Design Subsystem B	S	4	Design Team B
Build Subsystem B	T	20	Technician
Assemble Subsystems A and B	Z	4	Technician

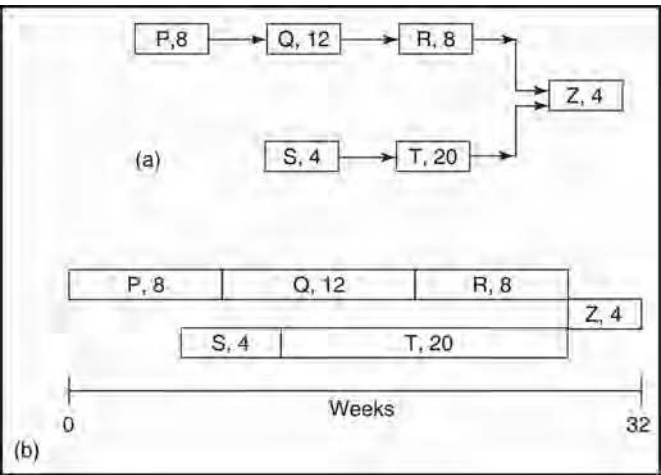


Figure 8.18
(a) Network for activities in Table 8.7;
(b) time-scaled network for the activities in Table 8.7.

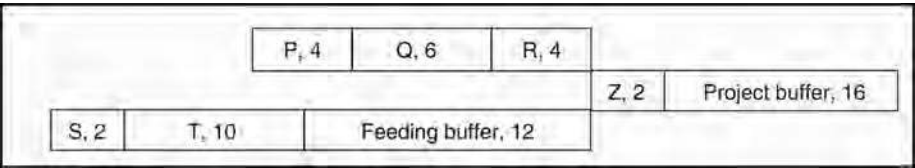


Figure 8.19
Schedule with contingency reserves (time buffers).

noncritical path feeds into the critical path. The feeding buffer protects the critical path from delays in noncritical activities.

While project buffers and feeding buffers bear a resemblance to slack, they are *not* slack. Whereas with traditional network methods activities are scheduled as early as possible and slack may be used to absorb any delays, in CCPM all activities are planned to *start as late as possible*, but with buffers. The buffers all “belong” to the project manager, and only she can allocate buffer time to them, which she does whenever an activity exceeds its planned duration.

The size of the buffers in Figure 8.19 is the same as the lengths of the paths they follow, but in practice they can be substantially less than that. There are two reasons:

- 1. The mathematical **principle of aggregation**, which, in this application, says that the uncertainty of finishing a project is much less than the sum of the uncertainties of finishing the individual activities in the project. In other words, the paddings removed from individual activities can be replaced by a project buffer that is much less than the sum of those paddings.
- 2. Parkinson’s Law states that “work expands to fill the time available.” The obverse of this is that, given less time to do the work, people tend to work faster. In other words, removing padding from an activity creates a sense of urgency.

For these reasons, the project buffer size can be substantially reduced—typically by 50 percent. As shown in Figure 8.20, where the buffers have been cut in half, the estimated project completion time is 28 weeks. (A buffer of only a third the length of the path it follows is often sufficient for setting a project commitment date.)

Thus, week 28 is the date to which the project manager commits to completing the project, while week 20—at the start of the project buffer—is the date the project team strives to meet. Consequently, while there is a chance (perhaps small) that the project will finish in 20 weeks, there is a very good chance of it being completed within 28 weeks. This compares to the critical path method, which would yield an estimated project duration of 32 weeks (plus a high likelihood of it taking longer than that).

Critical chain

Now let’s also consider resources. Figure 8.20 shows that activities Q and T both use the same “technician” resource. Assume the technician can work on only one activity at a time, in which case the schedule must be adjusted. This is shown in Figure 8.21, putting activity T before activity Q (putting Q before T is another possibility). In the revised schedule, path S–T–Q–R–Z is called the **critical chain**, defined as the *longest path that accounts for both precedence and resource dependencies*. Whenever the length of the critical chain plus the project buffer exceeds the length of the critical path, it is the critical chain, not the critical path, that determines the project duration.

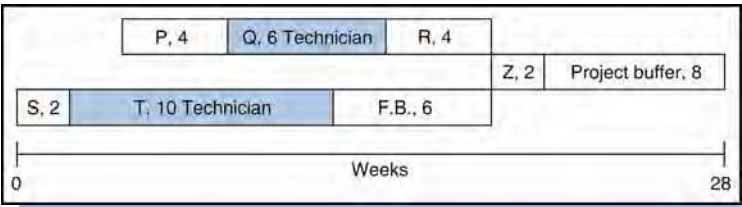


Figure 8.20
Schedule with
buffer sizes
reduced.

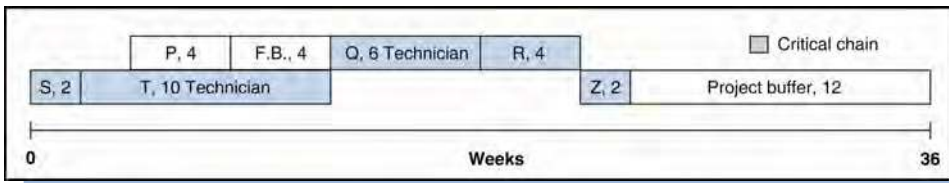


Figure 8.21

Schedule adjusted so every resource performs only one task at a time.

Traditional network scheduling is able to address resource-conflict problems by means of resource leveling, discussed in the previous chapter, but the resulting schedule will not necessarily be the same as with CCPM. Whereas the former method starts with an initial schedule and uses available slack to address resource conflicts, CCPM addresses resource requirements during the initial scheduling process and gives priority to resolving resource conflicts.

Note that the feeding buffer in Figure 8.21, F.B., is 4 weeks, not 2. This is because it follows only one activity, P, and hence the principle of aggregation mentioned above does not apply. Ultimately, however, the actual size of the buffer is at the manager's discretion.

Resource buffers: capitalizing on finishing work earlier than planned

Mentioned earlier was the fact that whenever activities finish late, their successors start late, but whenever they finish early, their successors *don't necessarily start early*. Resources that have been prescheduled often are not available to start earlier because they are busy doing something else. Consequently, activities finishing late tend to delay the project, but activities finishing early do not speed up the project (they have no effect!).

In CCPM, however, the project is able to capitalize on activities finishing early through the use of **resource buffers**. Unlike project and feeding buffers, a resource buffer does not add time to the schedule. Rather it is a *countdown signal* or warning to alert resources that an activity on the critical chain will possibly finish earlier than planned and to be *prepared to start early*. In a relay race, each runner is prepared to accept the baton whenever the previous runner arrives; likewise, resources on the critical chain are prepared to start work as soon as predecessors finish their work, regardless of the schedule. In practice, a resource buffer can take the form of a series of emails or other messages to a resource, counting down the time remaining before it must be ready to start an activity. The locations of resource buffers are illustrated in Figure 8.22. When the priority is to start work early, the resource must drop whatever else it might be doing.

Note in Figure 8.22 that resource buffers are inserted only on the critical chain (S-T-Q-R-Z) since feeding buffers shield the critical chain from uncertainty on noncritical paths. Note also there is no resource buffer between Activity T and Activity Q since the same resource (technician) does both and, obviously, needs no notification about when Activity Q will finish and Activity T must start.

Milestone buffers

Sometimes milestone dates are set at intermediate times in the project, such as at scheduled completion dates for project phases. In such cases a **milestone buffer** is inserted before each

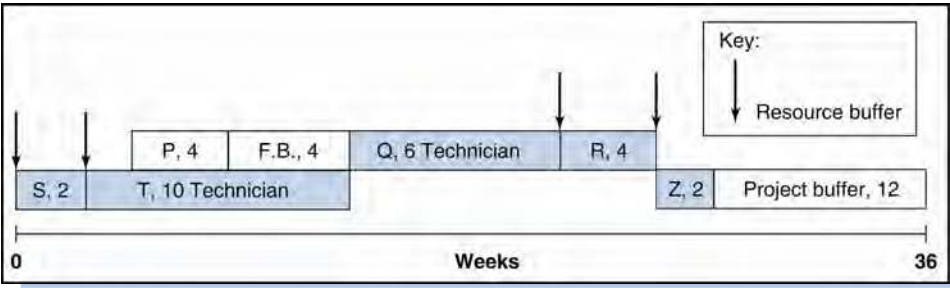


Figure 8.22
Resource buffers providing countdown on when critical activities should start.

Table 8.8 Summary of buffer types for a single project.

Buffer Type	Function of the Buffer
Project buffer	Comprised of aggregated paddings (contingency reserves) taken from activities on the critical chain; provides a schedule reserve between the earliest possible completion date and the committed, target date.
Milestone buffer	Similar to a project buffer but used when a project phase or milestone has a fixed due date.
Feeding buffer	Comprised of aggregated contingencies taken from activities on noncritical paths; prevents noncritical activities from delaying critical-chain activities.
Resource buffer	An early warning or “countdown” to prepare resources on critical activities to be ready to start work as soon as all predecessor activities have been completed.

milestone. When milestone buffers are used, the size of the project buffer is reduced; in effect, the project buffer is divided among milestone buffers. The different types of buffers used in CCPM are summarized in Table 8.8.

Sizing of buffers

CCPM relies heavily on project and feeding buffers, so making them the right size is important. As illustrated in the examples in Figures 8.19 and 8.20, commonly activity durations are cut in half, and then the resulting longest path through the network is cut in half to yield the size of the project buffer. This method, which reduces project duration by 25 percent, is referred to as the “50 percent of chain” and “cut and paste” method.

When a path consists of many activities, even a buffer of 50 percent of the path length is too large. Several methods have been proposed for determining the size of buffers, but these are beyond our scope.¹¹

Padding, multi-tasking, and procrastination

Projects take longer than necessary for many reasons. First, as already stated, people **pad** their time estimates, and the effects of this worsen as managers at all levels in the WBS add to the padding. If the person responsible for an activity pads the time by 10 percent and each person higher in the WBS also pads it by 10 percent, the padding at the project level for an n-level WBS would be $(1.1)^n$. For a five-level WBS, this yields a total contingency of 60 percent; if each pads 15 percent, the total contingency would be 101 percent.

Second, people **multi-task**. For example, suppose a contractor has three independent projects, X, Y, and Z, each of expected duration 10 weeks. The contractor is anxious to finish all of them as soon as possible so he divides each into small pieces so that, in a sense, he can work on all of them at once. But in doing so, he actually delays the completion of at least two of the projects. If he had scheduled the projects sequentially X first, Y second, and Z last, without interruption, then, as shown in Figure 8.23(a), he would finish X at Week 10, Y at Week 20, and Z at Week 30. But when he breaks up the projects into segments of, say, 5-week periods, and alternates working among them, he increases the elapsed time for each project from 10 weeks to 20 weeks. As shown in Figure 8.23(b), the result is that two of the projects are delayed: X finishes in Week 20 and Y finishes in Week 25. In general, the more activities or projects are broken up and intermixed with other projects, the greater the elapsed time to finish any of them.

Compounding the effect is that multi-tasking precludes gaining the momentum that would have occurred by focusing uninterrupted on only one task, as was illustrated in Chapter 7, Figure 7.24. Multi-tasking should thus be avoided. By focusing on just one activity (or project) at a time, each is completed sooner, successors start sooner, and everything finishes sooner.

A third reason projects take longer than necessary is **procrastination** and wasting available slack. Given a choice between two scheduled times, one early and one late, many people choose the late one; this automatically eliminates slack, puts activities on the critical path, and increases the likelihood of project delay. Also, whenever they perceive slack, people are less motivated to complete an activity early. The effect is called the “students’ syndrome,” hinting to students’ initial enthusiasm for a new course that soon wanes, not to resume until just before the final exam. The same effect occurs in production and project environments (Figure 8.24). Shortening activity durations and scheduling activities as late as possible (with buffers) reduces the tendency to procrastinate.

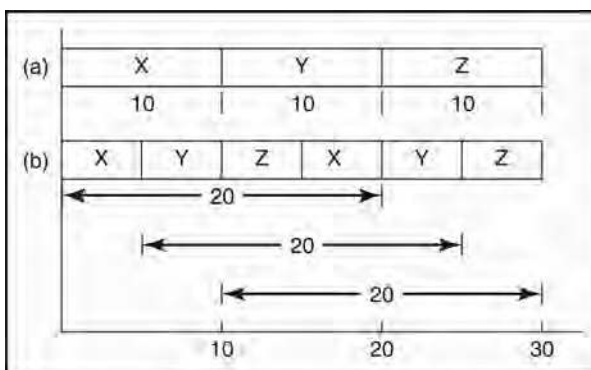


Figure 8.23
Effect of multitasking on elapsed and completion times: (a) uninterrupted case; (b) interrupted, multitasking case.



See Chapter 7

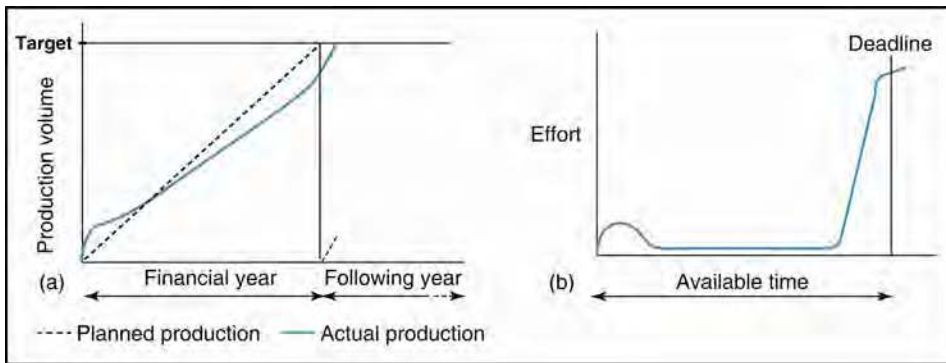


Figure 8.24

Students' syndrome (a) in a production environment and (b) in a project.

Buffer management for project control

Every activity in the project is linked to a buffer: activities on the critical chain are linked to the project buffer and all others to feeding buffers. During project execution those buffers are monitored to predict the project completion date and assess the risk of missing the due date. Every time an activity is delayed, buffer is “consumed.” A delay occurs when the actual time to complete a task exceeds the estimated, aggressive time. The amount of buffer consumed is determined each day by simply asking people working on activities the amount of work remaining and if they exceeded the estimated time. If all the delays consume less time than the buffer holds, the project will be on time; if they consume more time than the buffer holds, the project will be late. Buffer consumption also provides a clear, objective way to determine priorities. Buffers are monitored and whenever someone is required to work on more than one activity, the most-consumed buffer indicates which activity gets highest priority. Buffer management is further discussed in Chapter 13.



See Chapter 13

Challenge of CCPM: changing behavior

A belief in most project organizations is that because the project manager commits to the target completion date, everyone in the project also commits to target dates. The premise in CCPM is that only the project manager commits to a completion date, while everyone else works toward “aggressive” estimated times but doesn’t commit. While getting everyone to accept this in small projects might be relatively easy, in big projects it isn’t. Since most people are accustomed to working toward deadlines, this requires no less than cultural and behavioral changes at all levels of the organization, including top management. Not understanding the principles of CCPM, senior managers and customers will try to enforce deadlines on everyone.

Software support for CCPM

Many project management software packages include provision for CCPM, although others require add-ons for compatibility with CCPM. For example, TameFlow™, Sciforma™ and Concerto™ software fully support CCPM, but MSProject requires an add-on such as Prochain™.¹²

8.6 TOC Method for Allocating Resources to Multiple Projects¹³

As much as 90 percent (by value) of all projects are done in multi-project environments.¹⁴ The theory of constraints (TOC) recommends the following five-step procedure for scheduling projects so as to maximize the number of projects an organization can do concurrently.

Step 1: Identify the constraint

What prevents a company from doing more projects? If the company has few projects to do, the constraint might be limited market share or the size of the market. But if it has more projects than it has the capacity to do them, the constraint will be anything that reduces the rate at which projects are completed. The most common constraint is a highly loaded resource or multi-tasking.

Production environments such as manufacturing “job shops” use priority rules (discussed earlier) to select the next job to which a resource (usually a machine) should be assigned. In job shops it is easy to identify the constraint—it is the machine at which queues of work pile up. Such a resource, the *system constraint*, is called the **drum resource** because it sets the pace of everything flowing through the process. The TOC philosophy emphasizes keeping the drum resource busy, preventing starvation and blockage, and to keep work flowing. (To ensure this, drum buffers are used; their application to project management is described elsewhere.)¹⁵

Experience with implementing TOC, however, suggests that while a specific resource might be identified as the constraint in a multi-project environment, often the real constraint is something *other* than that resource. In practice, the identified constraints might be different for *planning* a set of projects than for *executing* them. Such constraints are discussed next.

In a sequence of activities, regardless of the actual constraint, an activity near the end of a project may be chosen as a substitute for the constraint. For *planning* a set of projects, a rule regarding the scheduling of this activity may be used as proxy for the constraint. For example, in one electronics company that conducts multiple, concurrent projects, all projects must pass through “final integration” just before closeout, and a specialist engineer with high workload was identified as the resource constraint. Rather than using that resource to stagger projects, however, it was decided instead to use the rule of *only three projects in final integration at a time*. In Figure 8.25, the shaded activities represent final integration in nine projects. To ensure there

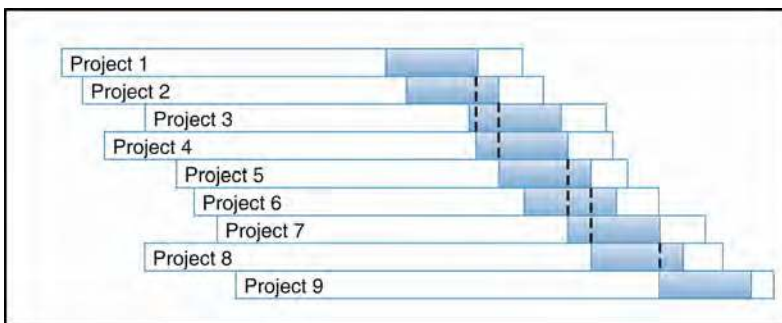


Figure 8.25

Only three projects in final integration at any point in time.

will always be just three projects in final integration: Project 4 starts final integration as soon as Project 1 completes final integration; Project 5 starts final integration as soon as Project 2 completes final integration, and so on. This staggers the work for all resources through subordination of individual project schedules to the three-project rule, as discussed in Step 3 below. In the same company, the constraint for executing projects was identified as management time available to provide project support.

Step 2: Decide how to exploit (utilize) the constraint

The rule “only three projects in final integration” enables the company to exploit the constrained final integration capacity for planning purposes. Starting with a project’s scheduled start date for final integration and working backward through the project’s duration determines the date when the project should be initiated (release date). Whenever an integration task cannot start according to the rule, or work must wait for an integration task, then the duration of integration tasks is adjusted to enable the appropriate staggering of project release.

In the event that a project’s final integration is completed early, resource buffers (discussed earlier) will ensure that subsequent projects can be started early.

Because fewer projects are being worked on, staggering them in this way reduces the workload on most resources, reduces multi-tasking across projects, improves the flow of projects (i.e., rate at which projects are completed), and ensures that commitments to customers are met.

Step 3: Subordinate everything else to decisions regarding the constraint

During scheduling, the release date of projects is based on the load on the constraint. When the company decided that it should have at most only three projects in the final integration phase, the proxy constraint was the rule “three projects in final integration.” Each new project will be slotted to start final integration at whatever time is necessary to maintain that three-project rule. The schedules of individual projects are therefore *subordinated* to the multi-project constraint—i.e., the rule regarding the number of projects in the integration phase. The project with the highest risk of missing its due date would be scheduled to start first, as indicated by its project buffer status. The utilization of all resources for the set of projects is also subordinated to the schedule, and people do not multi-task.

If the constraint for project execution is time available for management support, then all other work to be performed by managers must be subordinated to this support role.

Step 4: Elevate the constraint

The constraint is “elevated” by providing additional capacity. Elevating the constraint involves, for example, increasing the final integration capacity so as to increase the number of projects in that phase from three to four, perhaps by adding another specialist engineer. It sometimes requires costly measures such as building new facilities and hiring additional people. Steps 2 and 3 therefore ensure that existing capacity is utilized effectively before spending money acquiring additional resources.

If the constraint is management time available for project support, then the management system might be simplified to improve its effectiveness in performing support functions.

In multi-project environments, resource buffers have less utility. Upon completion of work in one project, resources simply begin work on the next project, even if they have to wait a while before starting. Actually, it is okay that they are idle between projects: for the sake of maximizing project flow, the resources should have to wait for the project, but the work should not wait for the resources.



See Chapter 13

In this way it is possible that all activities (including those on noncritical paths) can start as soon as predecessor activities are completed. All that is necessary to keep projects on track is to monitor the project buffers (illustrated in Chapter 13, Example 13.2). Tracking project buffers simplifies the tracking and control process during project execution and relieves managers' workloads. In turn, this elevates the constraint for project execution—the time managers have available to provide project support.

Step 5: Return to Step 1

Adding resources might remove the constraint, in which case a new constraint would be identified, and the process repeated. Sometimes, however, the new constraint would be too disruptive, in which case the extant constraint is allowed to remain and not be elevated.

Discussion

One company has applied the TOC method for managing multiple projects by using only three rules:¹⁶

Rule 1: During planning, stagger the release of projects.

Rule 2: Plan aggressive project durations using project buffers only one-third the length of the critical chain.

Rule 3: During execution (a) ensure that activities are executed according to priorities by monitoring and controlling project buffer status (Example 13.2) and (b) minimize buffer consumption by doing all tasks as soon as possible.

How good is TOC for planning compared to traditional priority rules? The answer is somewhat equivocal. For some simple projects the results are the same. In one exploratory study where TOC was compared with the oft-used least slack rule, TOC gave better results,¹⁷ but another study yielded poorer results.¹⁸ Although TOC is based on logic and seems to make sense, verification in practice would require empirical research from a variety of different industrial settings, which has yet to be done.

8.7 Discussion and Summary

This chapter covered project scheduling methods that address time constraints, resource constraints, uncertainty about activity and project durations, and multiple projects that share resources. The methods offer ways to accelerate projects and reduce uncertainty about completion dates, although they have lower acceptance than simpler techniques such as Gantt charts and critical path networks and are applied mainly in relatively sophisticated industries. All have limitations, yet all have merits.

- CPM enables managers to determine the least costly way to reduce project duration, or complete a project by a due date or in the shortest time.
- PERT enables managers to estimate the probability of finishing a project by a predetermined date. But the method considers only the current critical path and ignores noncritical paths that could become critical. Monte Carlo simulation overcomes this limitation by accounting for the possibility of any path becoming critical.
- CCPM aims at reducing project duration. Using time buffers it transforms a stochastic problem into a simpler deterministic one. Unlike CPM wherein noncritical activities are scheduled as early as possible, CCPM schedules them as late as possible but with buffers. With other methods, variability in activity durations can lead to changes in the critical path, but buffers in CCPM provide relative stability to the critical chain—the path connecting activities that require the same constrained resource. CCPM offers a relatively simple way to schedule projects, but it requires a shift in human behavior since only the project manager and nobody else commits to due dates. Many managers find that concept hard to swallow.
- Multi-project scheduling presents the challenges of allocating scarce resources to concurrent projects. The traditional way to allocate resources among projects (and among activities within projects) is to use priority rules. The TOC way aims to allow as many concurrent projects as possible by improving the flow of projects through the system.

All these methods are supported by commercial software, which simplify their application and eliminate computational difficulties. But as with all management methods, appropriate application of the techniques assumes a sound understanding of the principles that underlie them and management acceptance.

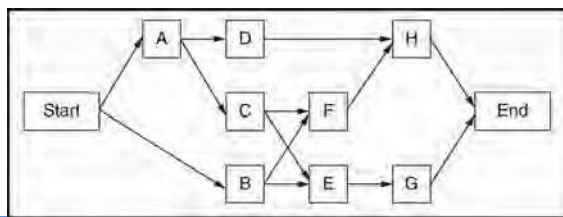
Summary List of Variables

- n** Number of activities on the critical path.
- C_n** Normal activity cost: The direct cost of completing an activity under normal work effort; usually, the lowest cost for completing an activity.
- C_c** Crash activity cost: The direct cost of completing an activity under a crash work effort; usually, the highest cost for completing an activity.
- T_n** Normal activity duration: The expected time to complete an activity under normal work effort; usually, assumed to be the longest time the work will take.
- T_c** Crash activity duration: The expected time to complete an activity under a crash work effort; the shortest possible time in which an activity can be completed.
- t_e** Expected activity duration: In PERT, the mean time to complete an activity, based on optimistic (a), most likely (m), and pessimistic (b) estimates of the activity duration.
- T_e** Expected project duration: The probability of the project finishing earlier than this time is 50 percent and the probability of it taking longer is 50 percent.
- T_s** Target completion time for project: A time specified for project completion.
- V** Variance of an activity: The variability in expected activity duration.
- V_p** Variance of the project duration: The variability in the expected project duration.



Review Questions and Problems

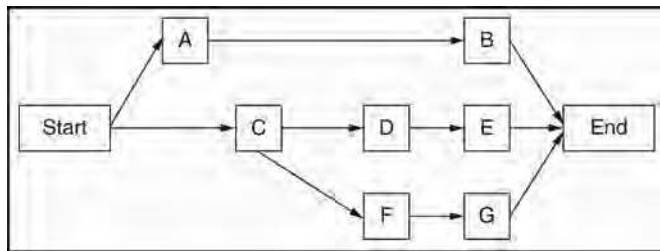
1. Define crash effort and normal effort in terms of the cost and time they represent. When should a project be crashed?
2. How do CPM and PERT differ? How are they the same?
3. What does the cost slope represent?
4. The cost slope always has a negative value. What does this indicate?
5. Time–cost tradeoff analysis deals only with direct costs. What distinguishes these costs from indirect costs? Give examples of both direct and indirect costs.
6. What are the criticisms of CPM? How and where is CPM limited in its application?
7. The project network and associated costs (T in days, C in \$1,000s) are shown below.



Activity	Normal		Crash		Cost Slope
	T_n	C_n	T_c	C_c	
A	4	210	3	280	70
B	9	400	6	640	80
C	6	500	4	600	50
D	9	540	7	600	30
E	4	500	1	1100	200
F	5	150	4	240	90
G	3	150	3	150	–
H	7	600	6	750	150

- a. Verify that the normal duration is 22 days and that the direct cost is \$3,050.
- b. What is the least costly way to reduce the project duration to 21 days? What is the project cost?
- c. What is the least costly way to reduce the duration to 20 days? What is the project cost?
- d. Now, what is the earliest the project can be completed and what is the least costly way of doing this? What is the project cost?

8. The project network and associated costs (T in days, C in \$1,000s) are shown below.



Activity	Normal		Crash		Cost Slope
	T_n	C_n	T_c	C_c	
A	6	6	3	9	
B	9	9	5	12	
C	3	4.5	2	7	
D	5	10	2	16	
E	2	2	2	2	
F	4	6	1	10	
G	8	8	5	10	

- What is the earliest the project can be completed under normal conditions? What is the direct cost?
 - What is the least costly way to reduce the project duration by 2 days? What is the project cost?
 - What is the earliest the project can be completed and what is the least costly way of doing this? What is the project cost?
9. The following table gives information on a project (T in days, C in \$1,000s):

Activity	Immediate Predecessors	Normal		Crash	
		T_n	C_n	T_c	C_c
A	–	6	10	2	38
B	–	4	12	4	12
C	–	4	18	2	36
D	A	6	20	2	40
E	B, D	3	30	2	33
F	C	10	10	6	50
G	F, E	6	20	2	100

- Draw the network diagram. Under normal conditions, what is the earliest the project can be completed? What is the direct cost? What is the critical path?
- What is the cost of the project if it is completed 1 day earlier? Two days earlier?

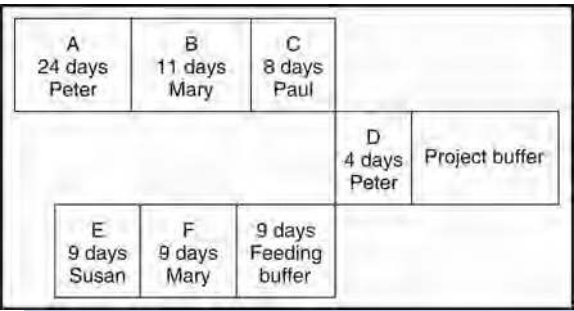
- c. What is the earliest the project can be completed? What is the lowest cost for completing it in this time?
 - d. If overhead (indirect) costs are \$20,000 per day, for what project duration are total project costs (direct + indirect) lowest?
10. Has variability in a time estimate ever caused you to be late for an appointment? Describe.
11. A procurement officer discovers that the delivery time for a specific item is never less than 5 days. Ten days is the most frequent, but in the worst-case scenario, it takes 30 days.
 - a. Calculate the expected delivery time.
 - b. Calculate the variance.
 - c. What factors would you consider when deciding the amount of time to allow for delivery of the item in the project plan?
12. The tables below show the immediate predecessors and a , m , b for each activity in the two projects. For each project compute:
 - a. t_e and V for each activity.
 - b. ES, EF, LS, and LF for each activity.
 - c. T_e and V_p for the project.

Activity	Predecessors	a	m	b
A	-	7	9	11
B	A	1	2	3
C	A	7	8	9
D	B	2	5	11
E	C	2	3	4
F	C	1	4	8
G	D, E	6	7	8
H	F, E	2	6	9

Activity	Predecessors	a	m	b
A	-	2	4	6
B	-	2	2	3
C	-	4	8	10
D	A	4	6	7
E	A, B	7	9	12
F	D, E	1	2	3
G	C	2	3	4
H	F, E	2	6	9

13. Refer to the first network in the above problem.
 - a. What is $P(T_e < 23)$?
 - b. What is $P(T_e < 32)$?
 - c. For what T_s is the probability that the project will be completed 95 percent?
14. For the network in Figure 8.12, what is the probability of completing each of the five paths within 30 days? What is the probability of completing all of them within 30 days?

- 15. How would you use buffers to ensure that you are on time for appointments? What factors would you take into account when you make a decision on the size of the buffer?
- 16. Explain in your own words how the principle of aggregation plays a role in reducing project duration.
- 17. The diagram below was drawn before it became clear that Mary would have to perform both Activity B and Activity F.



- a. With the realization that Mary must do both tasks, indicate two possible critical chains.
- b. Reschedule the work and show the position and the size of the feeding buffer.
- 18. Refer to the network in Question 19 in the Review Questions and Problems for Chapter 7. Show the critical chain on the network diagram.
- 19. Refer to Figure 8.22 where Activity T is scheduled before Activity Q to resolve the Technician resource conflict. Explain why this alternative was not selected.
- 20. Consider the data about project activities given in the table below.
 - a. Create the Gantt chart. Schedule the work in such a way that each person always has only one task to perform (do not reduce the durations of activities or insert buffers as yet).
 - b. Indicate the critical chain.
 - c. Indicate where the feeding buffers should be inserted.
 - d. What is the difference in the lengths of the critical path and the critical chain?

Activity	Predecessor(s)	Duration (Days)	Resources
A	-	2	John
B	A	3	Sue
C	-	3	Sue & John
D	C	2	Al
E	D, J	3	Sue & Al
F	E, B	2	John
G	F	2	Ann
H	-	4	Sue
J	H	2	Al

- 21. Discuss the difference between fast-tracking, concurrent engineering, and crashing.



See Chapter 7

22. Write an essay on the reasons why projects are often late.
23. Discuss the implications that subcontracted work would have on implementing CCPM.
24. Discuss the implications of resource allocation for organizations involved in multiple projects.
25. Refer to Chapter 7, Review Questions and Problems, Question 19.
 - a. Use the shortest task-time rule to solve the problem and draw a Gantt chart.
 - b. Apply the least slack rule to solve the problem and draw a Gantt chart.
 - c. Apply the first three of the five TOC steps (section 8.6) and draw a Gantt chart.
 - d. Who would be the “drum resource”?
 - e. How does the days that Ann has no work to do on this project relate to TOC Step 3?
 - f. Assume the two people in this problem also work on several other projects as well, and the policy is to use the shortest task-time rule and the relative priorities of projects to decide on which activity a resource should work. Which rule (shortest task time or highest project priority) should have preference? Discuss.
26. In a multiple-project environment the drum resource carries a certain “status” since work performed by other resources (and the needs of other resources) are subordinated to it. In one company, management placed a flag at a work center to indicate that it was the drum resource. An improvement (TOC Step 4) removed this work center as the constraint and the flag was moved elsewhere to the new constraint. People working in the original center were disappointed and protested. How do you suggest management should handle this problem?



Questions About the Study Project

1. In the project you are studying, discuss which of the following kinds of analyses were performed:
 - a. PERT
 - b. CPM/time-cost tradeoff analysis
 - c. Scheduling with resource constraints
 - d. CCPM.
2. Discuss how they were applied and show examples. Discuss those applications that were not applied but which seem especially applicable to the project.
3. How do you rate the risk of not finishing on time and what are the factors contributing to this risk?
4. Were people (other than the project manager) required to make commitments on the duration of activities? Comment on the possibility of changing this behavior.

CASE 8.1 BRIDGECON CONSTRUCTION

Bridgecon Construction Company specializes in the design and construction of steel and concrete bridges. Phase 1 of the company's project management methodology, Initiation, includes identification of project opportunities, and assessment of its risks and alignment with strategic goals. Bridgecon's marketing department identified an opportunity: a well-known bridge architect recently completed the concept design for a cable-stayed bridge intended to cross over electrified railway lines. Senior managers

felt the company could handle the project and decided to pursue it, which marked the beginning of phase 2, Estimating.

The estimating team visited the site, reviewed available resources and skills, performed a detailed risk assessment, and prepared a preliminary plan for detail design, procurement, logistics, and construction. Phase 2 included activities A and B in Table 8.9,

Table 8.9 Activities for constructing the cable-stayed bridge.

Activity	Activity Description	Initial Duration Estimate	Predecessors	Initial Cost Estimate (\$1,000)
A	Detailed site investigation and survey	2	–	17
B	Detailed planning	6	A	16
C	Detailed design	6	B	557
D	Preparation of site	4	C	47
E	Relocate services	3	C	28
F	Re-align overhead track electrification	4	C, E	650
G	Access road and ramp construction	1	D	63
H	Piling	2	G	820
J	Construct foundations and abutments	3	H	975
K	Construct temporary supports to support bridge deck during construction	2	F, G	720
L	Fabrication planning of structural steel components	2	C	13
M	Manufacture structural steel components (off-site)	2	L	1,320
N	Transport structural steel components and erect on-site	1	M	433
P	Erect pylons and fill with concrete	2	J	840
Q	Construct main span deck on pre-cast concrete beams	3	H, K, N, P	2,800
R	Cable-stay installation and lift the bridge deck off temporary supports	3	Q	875
S	Removal of temporary supports	1	R	54
T	Electrical system installation	1	S	147
U	Roadway surfacing (paving)	2	S	142
V	Finishing and ancillaries	2	T, U	76
W	Commissioning –cut-over	1	V	14
X	Formal hand-over and ceremony	1	W	9
Y	Project sign-off	1	X	1
Z	Administrative closure	1	W	4
AA	Project end (Milestone)	0	Y, Z	10,621

which was necessary to prepare the bid for building the bridge. The phase concluded with a presentation to the customer, the rail authority.

The project manager and the estimating team met with the architect and structural engineers to acquaint themselves with the design. They then met with the subcontractors who they might choose to construct the pilings and fabricate steel components. Following these meetings the “Initial Duration Estimate” and “Initial Cost Estimate” were completed, as listed in Table 8.9.

The RFP for the bridge project stated that acceptance of the plan by the rail authority was a criterion for selecting a contractor. Early on it became evident that starting with activity D and until the completion of activity S, operation of one of the railway lines under the bridge would be impaired, although an informal discussion with the rail authority indicated that that might be acceptable. During a subsequent meeting, however, the rail authority expressed concern that the impairment would last 17 weeks and requested Bridgecon to find ways to reduce that time. The estimating team suggests the following possibilities:

- The duration of activity N could be reduced from one week to half a week by using additional trucks. The additional cost would be \$33,000.
- An alternative subcontractor for piling was approached. The subcontractor said it would be able to halve the time of activity H for a cost of \$960,000.
- Two ways were identified to shorten the duration of activity D. First, additional temporary workers could be employed. This would reduce the duration to 3 weeks and increase the cost to \$147,000. Second, a team of workers highly skilled in this procedure plus their equipment could be reallocated from another project. Adding these workers to the original team would allow the work to be completed in 1 week. The manager of the other project estimated the reallocation would cause him to forfeit an incentive fee of \$150,000 for finishing his project early. The managers of the two projects agreed that, should the reallocation be made, the value of incentive fee would be booked as a cost against the cable-stayed bridge project and transferred as a bonus to the other project.
- The duration of activity F could be reduced to 3 weeks but would increase the activity's cost to \$730,000. It could be reduced to 2 weeks at a cost of \$820,000.

The duration of activity Q can be reduced to 2 weeks, but the activity cost would increase to \$2,929,000.

QUESTIONS

1. Compile a list showing the reduced periods of impairment for the rail operation and the associated additional costs.
2. Comment on the implications that crashing might have on the risk of not meeting the committed due date.

CASE 8.2 LOGON PROJECT

After signing the contract, the management at Midwest Parcel Distribution (MPD) discovers that for many reasons it would be advantageous to complete the project in 40 weeks. It is too late for MPD to “require” the contractor Iron Butterfly Company to complete it in that time, but nonetheless it discusses the possibility with Iron Butterfly’s project manager, Frank Wesley. Reviewing the network diagram (Figure 8.26), Frank checks the feasibility of this and then asks his managers and technical staff to provide three time estimates for every activity in the project; these are shown in Table 8.10.

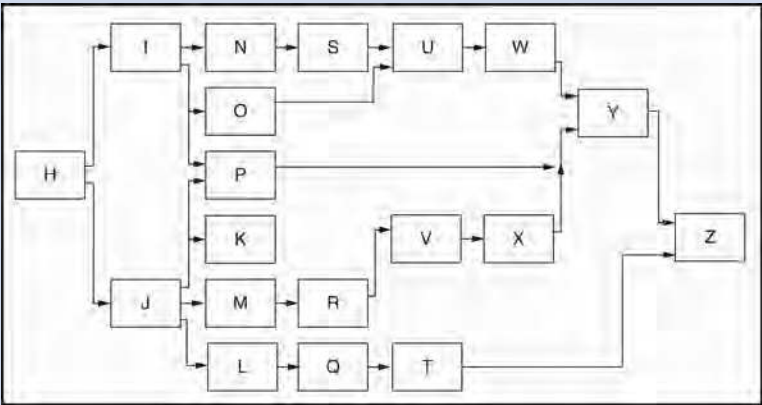


Figure 8.26
LOGON project.

Table 8.10 Time estimates for LOGON project.

Activity	Optimistic Duration (Weeks)	Most Likely Duration (Weeks)	Pessimistic Duration (Weeks)
H	10	10	10
I	8	8	16
J	1	6	6
K	4	4	4
L	2	2	2
M	2	4	5
N	4	4	10
O	5	5	5
P	5	5	5
Q	5	5	5
R	2	5	5
S	3	3	6
T	3	3	3
U	1	1	2
V	3	5	5
W	2	2	8
X	3	3	3
Y	8	8	8
Z	6	6	6

QUESTIONS

1. What is the probability of finishing the project within 40 weeks?
2. Do you foresee any significant risk of a delay that the above calculation does not take into account?
3. Determine the most likely project duration.

CASE 8.3 PAPUA PETERA VILLAGE PROJECT

The Papua Energy Company is building a village to support workers for a wind farm in Sumatra. The principal work of the project involves five work packages: build road, clear site, erect buildings, erect power generation plant, and build water purification system. Figure 8.27 is a high-level network diagram for the project.

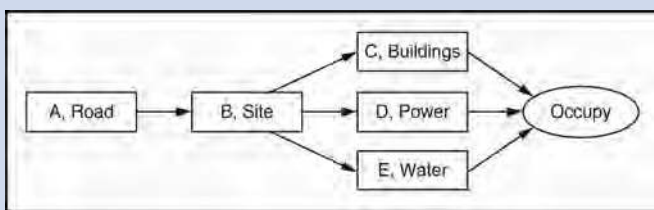


Figure 8.27
Papua Petera village project.

To explore ways to speed up the project, Papua asked its contractors to submit information about time and cost for crews working as many as three shifts a day. (Portable lighting technology would enable work to continue at night-time for second and third shifts.) The contractors responded with the following estimates, which exclude costs for materials, supplies, components, and systems that are fixed regardless of work times.

Work package A: Build road

- Road length, 10 km
- One shift is able to build 0.1 km of road
- First shift costs: labor, \$4,000/day; equipment, \$8,000/day
- Second and third shifts, cost per shift: labor, \$6,000; equipment \$9,000/day

Work package B: Clear site

- Using one shift, site can be cleared in 10 days; if two shifts, 5 days; if three shifts, 4 days
- First shift costs, labor and equipment: same as Work package A
- Second and third shifts, cost per shift: same as Work package A

Work package C: Erect buildings

Forty buildings will be constructed of three standard models, all about the same size. Each shift can construct three buildings per week. Assume 5-day work weeks.

- First shift costs: labor, \$4,000/day; equipment, \$1,000/day
- Second and third shifts, cost per shift: labor, \$6,000; equipment \$1,500/day

Work package D: Erect power generation plant, install power lines to buildings

Work package will take 10 weeks; with a second shift, it will take 5 weeks. Assume 5 days/week.

- Labor shortage does not allow a third shift
- First shift costs: labor, \$6,000/day; equipment, \$3,000/day
- Second shift, cost per shift: labor, \$9,000; equipment \$4,200/day

Work package E: Build water purification system, install water and sewer lines to buildings

Work package will take 8 weeks; with a second shift it will take 4 weeks. Assume 5 days/week.

- Labor shortage does not allow a third shift
- First shift costs: labor, \$5,000/day; equipment, \$4,000/day
- Second shift, cost per shift: labor, \$7,500; equipment \$5,500/day

The above costs are all direct costs. Additional is the indirect cost—the cost for management and administration of the overall project; this is estimated at \$3,000/day for however long the project takes.

Given this information, Papua has asked project manager Abdul Ginting to estimate alternative total project costs and project durations for two cases: (1) the lowest cost alternative and how long the project would take, and (2) the shortest project duration alternative and how much the project would cost. Abdul is preparing his analysis and recommended course of action. What will he find?

Notes

1. Goldratt AY. Institute, group email messages sent on March 17 and 18, 1999; Larry English, Habitat for Humanity, January 2007, Pretoria, South Africa; Habitat for Humanity, *The Fastest House in the World*, accessed January 2007 from <http://www.habitat.org/newsroom/1999archive/insitedoc004016.aspx?print=true>.
2. CPM was developed in 1957 by DuPont Company, Remington Rand, and Mauchly Associates for constructing a DuPont plant, and first appeared in the article by its originators: Kelley J. and Walker M. Critical path planning and scheduling. Eastern Joint Computer Conference. Boston, MA; 1959, pp. 160–173.
3. For a piece-wise approximation for nonlinear relationships, see Wiest J. and Levy F. *A Management Guide to PERT/CPM: with GERT/PDM/DCPM and other Networks*. Englewood Cliffs, NJ: Prentice-Hall; 1977, pp. 81–85. The relationship between number of workers and activity duration is nonlinear; i.e., cutting the number of workers in half will not double the time but might increase it by, say, 50–150 percent, depending on the task. See Brooks F. *The Mythical Man Month: Essay on Software Engineering*. Reading, MA: Addison-Wesley; 1995, pp. 13–36.

4. The method first appeared in the article by the originators of PERT: Malcolm D., Roseboom J., Clark C. and Fazar W. Application of a technique for research and development program evaluation. *Operations Research* 7(5); 1959: 646–670.
5. See Miller R. *Schedule, Cost, and Profit Control with PERT*. New York, NY: McGraw-Hill; 1963, p. 58; Kerzner H. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, 10th edn. New Jersey; 2009, p. 529.
6. See Krakowski M. PERT and Parkinson's Law. *Interfaces* November 5(1), 1974; and Vazsonyi A. L'Histoire de la grandeur et de la decadence de la methode PERT. *Management Science* April 16(8), 1970 (written in English). Other problems of PERT/CPM are described by Kerzner, *Project Management*, pp. 519–522; Miller, *Schedule, Cost, and Profit Control with PERT*, pp. 39–45; and Weist and Levy, *A Management Guide to PERT/CPM*, pp. 57–58, 73, 166–173. References to human behavior are in the critical chain literature referenced in this chapter.
7. See Van Slyke R. Monte Carlo methods and the PERT problem. *Operations Research* 11(5); 1963: 839–860.
8. Adapted with permission from Evans J. and Olson D. *Introduction to Simulation and Risk Analysis*. Upper Saddle River, NJ: Prentice Hall; 1998, p. 111–120.
9. For information on Crystal Ball, see <https://www.oracle.com/applications/crystalball/>; for @Risk, see https://help.palisade.com/v8_3/en/@RISK/@RISK.htm; for Simul8, see www.simul8.com. (These are examples, not product endorsements; other firms provide risk analysis software as well.)
10. Goldratt E. *What Is This Thing Called Theory of Constraints and How Should It Be Implemented?* New York: North River Press, Inc.; 1990.
11. See, for example, Newbold R. *Project Management in the Fast Lane – Applying the Theory of Constraints*. New York: St. Lucie Press; 1988; Leach L. *Critical Chain Project Management*, 2nd edn. Norwood, MA: Artech House, Inc.; 2003; Geekie A. and Steyn H. Buffer sizing for the Critical Chain Project Management method. *SA Journal of Industrial Engineering* 19(1); 2008: 73–88; Tukul I., Rom W., and Eksioğlu S.D. An investigation of buffer sizing techniques in critical chain scheduling. *European Journal of Operational Research* 172; 2006: 401–416.
12. For information on Prochain, see www.prochain.com; for Sciforma, see www.sciforma.com; for Concerto, see www.realization.com. (These are examples, not product endorsements; other firms provide CCPM software as well.)
13. Viljoen P. Goldratt Schools, Personal communication, Pretoria, SA, May 2007, May 2010, and April 2015.
14. Turner J.R. *The Handbook of Project-Based Management*. London, UK: McGraw-Hill; 1993.
15. In Goldratt E. and Cox J. *The Goal*, 4th edn. Great Barrington, MA: North River Press; 2014; also in Goldratt E. and Fox R. *The Race*. Croton-on-Hudson, NY: North River Press; 1986. The notion of using the constraint (drum) to set the pace is described.
16. Adapted from training material of Realization. See www.realization.com.
17. Dass S. and Steyn H. An exploratory assessment of project duration in multiple-project schedules where resources are allocated by the theory of constraints method. *SA Journal of Industrial Engineering* 17(1); 2006: 39–54.
18. Cohen I., Mandelbaum A., and Shtub A. Multi-project scheduling and control: A process-based comparative study of the critical chain methodology and some alternatives. *Project Management Journal* 35(2); 2004: 39–50.

Chapter 9

Project Costs and Budgeting

Cost estimates, budgets, WBSs, and schedules are interrelated. As discussed in Chapter 6, cost estimates are ideally based upon activities identified in the WBS. When the cost cannot be estimated because an activity is too complex, the activity is broken down into smaller pieces until it can. When the work is poorly-defined or uncertain, the initial estimate is based upon judgment, but is later revised as more information becomes available.

Project schedules dictate when resources are needed, but the converse is also true: constraints on resources and working capital dictate when the work can be scheduled. Imposing constraints on costs is necessary to create realistic project budgets, otherwise projects become exorbitantly expensive, or must be terminated for lack of funds. Both occurrences are relatively commonplace.

Cost estimating, budgeting, and control are sometimes thought to be the exclusive concerns of cost specialists and accountants, but in projects they should be everyone's concern. Project participants who are closest to the sources of costs—engineers, systems specialists, architects, and others—should be involved in the estimating and budgeting process. Often, however, these people are uninvolved in estimating and are unaware about how budgets work., so the accuracy of estimates is not as good as it could be.

Project managers should be involved in budgeting. They do not need to be financial wizards, but they do need to be skillful in organizing and using cost figures. The project manager oversees the cost estimating and budgeting process, often with the assistance of a staff cost accountant. On technical projects, a cost engineer (or, outside the US, a quantity surveyor) reviews the deliverables and requirements, assesses the project from cost and technical points of view, and advises the project manager.

9.1 Cost Estimates

The cost estimate can seal the project's financial fate. When project costs are overestimated in the proposal (estimate exceeds likely actual costs), the contractor risks losing the job to a lower bidding competitor. Worse is when they are underestimated (actual costs exceed estimate). A \$50,000 fixed-price bid might win the contract, but obviously the contractor will lose money if the project ends up costing \$100,000. Underestimating is often accidental—the result of

being overly optimistic, although sometimes it is intentional—the result of trying too hard to beat the competition. In a practice called **buy in**, the contractor reduces an initially realistic estimate just enough to win the contract, hoping to cut costs or renegotiate a higher price after the work is underway. The practice is risky, unethical, and, sadly, commonplace. In large capital projects the tendency is to underestimate costs so as to get the funding needed to launch the project, but then forget the estimate afterward and expect cost overruns to be funded.

But a very low bid can also signify that the contractor cut corners in the estimating process, forgot to include things, or was just sloppy. The consequences for both client and contractor can be disastrous, ranging from poor quality work to financial loss or bankruptcy.

Cost estimates are the basis for the project budget. Once the project begins, actual costs are compared to estimated, budgeted costs as one measure of the project's performance. Without good estimates and a realistic budget, it is impossible to know whether project expenditures are on track or how much the project will cost at completion.

9.2 Cost Escalation



See Chapter 6

Estimating project costs can be difficult because the estimation process, described in Chapter 6, begins during project conception and before much is known about the project. The less well-defined the project, the more difficult it is to estimate the costs and the greater the chances they will substantially differ from actual costs. As a rule, the estimate will be too low and the project will suffer a cost overrun. The amount by which actual costs grow to exceed initial estimates is referred to as **cost escalation**. Usually, the larger and more complex the project, the greater the potential for escalation. Some escalation is normal and up to 20 percent is common. The costs of cutting-edge technology and research projects frequently escalate upwards of several hundred percent. The Anglo-French Concorde supersonic airliner exceeded the original estimate by a factor of five, NASA spacecraft sometimes by a factor of four to five, and nuclear power plants often exceeded estimates by a factor of two or three.

The plot in Figure 9.1 shows percent cost overrun for 111 transportation-related projects spanning 80 years.¹ The study reporting these findings looked at 246 other projects and got a similar picture. Clearly, overruns have been and remain common. How does that happen?

Lack of information and uncertainty

Much of the information necessary for accurate estimates is simply not available when cost estimates are first needed. At NASA, for example, lack of well-defined spacecraft design and unclear definition of experiments is the principal reason for cost overruns. Not until later, during the definition phase, when designs are finalized and work activities are well-defined can material and labor costs be accurately estimated. In most development projects, many of the activities are unpredictable, of uncertain duration, or must be repeated. Despite the difficulty, management must strive for the clearest scope of work, project objectives, and requirements, because without them, it is nearly impossible to obtain accurate cost estimates.

After the project is underway, possible product design changes, developmental barriers, strikes, and inflation will cause labor and material costs to increase. Whenever that happens, the project cost estimate should be updated to serve as a valid baseline to track project costs against.

Lack of information causes uncertainty. To allow for uncertainty in the estimate, an amount called a **contingency fund** or **budget reserve** is added to the original estimate.² This is

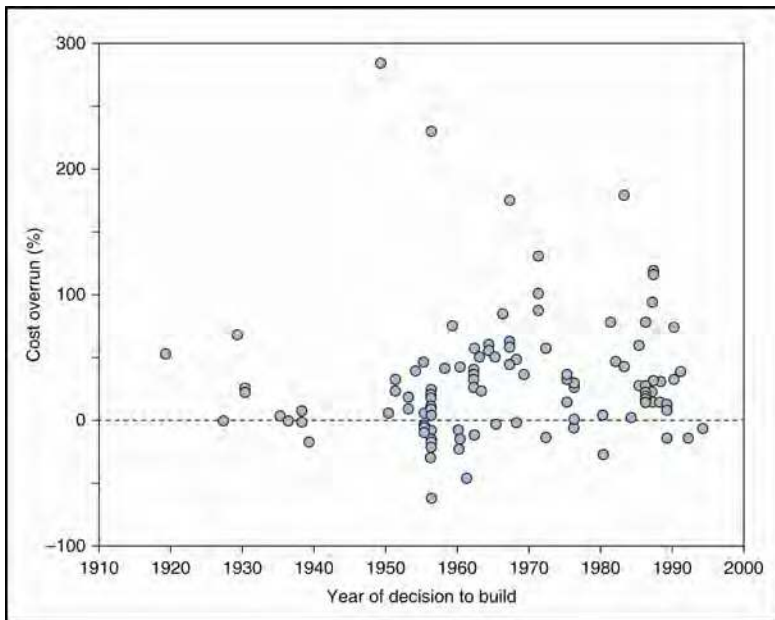


Figure 9.1

Project cost overrun versus year of project.

Source: Flyvbjerg B., Bruzelius N., & Rothengatter W. *Megaprojects and Risk: An Anatomy of Ambition*. Cambridge: Cambridge University Press, 2003, p.17. Reprinted with permission.

the budget equivalent of the **schedule reserve** or **time buffer** mentioned in previous chapters. The contingency amount is proportional to the uncertainty: the greater the uncertainty, the larger the contingency. The project manager (and sometimes the project sponsor or steering committee) controls allocation of the contingency fund. The fund is intended to offset small overruns arising from estimating errors, omissions, minor design changes and schedule slip-pages. Each time the cost estimate is updated, so is the contingency fund. The fund is not a “slush” fund, and should be cut from the project budget when no longer needed as intended; otherwise, project costs tend to rise to expend whatever remains in the fund. Contingencies are discussed in Chapter 6.



See Chapter 6

Changes in requirements or designs

Cost escalation also occurs due to **scope creep**—discretionary, nonessential changes made to system requirements. These changes come from a *change of mind*, not from oversights or mistakes in the estimate. The routine tendency is for users and contractors alike to want to upscale systems—to make “improvements” throughout the project life cycle. Such changes are especially common in the absence of thorough planning or strict control procedures.

Contracts occasionally include a **change clause** that allows the customer to make certain changes to contract requirements—sometimes for additional payment, sometimes not. The clause allows the customer flexibility to incorporate requirements not envisioned at the time of the original contract agreement. It can be exercised at any time, and the contractor is obligated



See Chapter 13

to comply. But any change, no matter how small, can cause escalation; it usually involves some combination of redesigning the product, reorganizing work, acquiring new or different resources, altering plans, or undoing or scrapping earlier work. The further along the project, the more costly the change. When accumulated, even small changes have a substantial effect on schedules and costs. A formal **change control** procedure is one way to reduce changes and contain escalation. This is described in Chapter 13.

Economic and social factors

Even with good initial estimates and few changes, cost escalation occurs because of forces beyond anyone's influence. Labor strikes, legal action by interest groups, trade embargoes, and materials shortages, all which serve to stifle progress and increase costs, cannot be precisely anticipated or factored into plans and budgets. Whenever work is suspended or interrupted, administrative and overhead costs continue to mount, interest and leasing costs on borrowed capital and equipment continue to accrue, and the date when payback begins and profit is earned is delayed. Anticipating such problems and incorporating their impacts into the contingency fund is difficult.

One economic factor that influences cost escalation is inflation. The contractor might want to offset this factor by inflating the price of the project, although not be able to because of competitors' actions or federal restrictions on price increases. Some protection may be gained by including inflation protection clauses in the contract that allow wage and material cost increases to be appended to the contract price,³ but the protection may be limited. Inflation is not one-dimensional; it varies with the labor, materials, and equipment employed, and by geographical region and country. Subcontractors, suppliers, and clients use different contracts with different inflation clauses that might be advantageous or disadvantageous to other parties in the project.

Inflation also causes cash flow difficulties. Even when a contract includes an inflation clause, payment for inflation-related costs must be tied to the publication of inflation indices, which always lags behind inflation. Contractors pay immediately for the effects of inflation but are not reimbursed for those effects until later.

Cost estimates are typically based upon prices at the time of estimation. Thus, whenever inflation rates become known the estimates should be adjusted so as to provide a valid baseline from which to later identify cost increases and take corrective action. In long-term projects, future wage rates should be forecasted, starting with the estimated current wage costs and then applying inflation rates over the project's duration.



See Chapter 20

In international projects, costs escalate due to changes in exchange rates. When the costs are incurred in one currency but paid for in another, changes in the exchange rate will cause the relative values of costs and payments to change, possibly resulting in a cost escalation. This topic is discussed in Chapter 20.

Inefficiency, poor communication, and lack of control



See Chapter 13

Cost escalation also results from work inefficiency, poor management and supervision, poor communication, and weak control. In large projects especially, these lead to conflicts, misunderstandings, duplication of effort, and mistakes. This is one source of escalation where management has substantial influence. Careful work planning, monitoring of activities, and tight control, particularly as discussed in Chapter 13, help contain cost escalation.

Ego involvement of the estimator

Cost escalation also results from the way people estimate. Many people are overly optimistic and habitually underestimate the time and cost, especially for jobs with which they are inexperienced. Have you ever estimated the time it would take for you to paint a room or tile a floor? How long did it *really* take? Sometimes, of course, the opposite happens: worried about overrunning their estimate, they “pad” it. The more the estimator’s ego is involved in the job, the less reliable the estimate.

Companies avoid this problem by employing professional cost estimators, who are not the same people as those who will do the work. Remember the earlier contention about involving project participants in planning the project? *Experienced workers* are usually good at estimating time and materials, but less so at estimating costs. Therefore, the doers (those who do the work) should estimate the time and resources they will need, but the professionals should estimate the cost.

People sometimes confuse an estimate with a goal; they think an estimate is what a job *should* take (or what they have been told it should take), not an unbiased prediction of what it will take. A cost estimate should *never* be based on a target or goal; thus, estimators must be positioned organizationally so they will not be coerced into providing the numbers that someone wants.⁴

Project contract

Different kinds of project contracts motivate or discourage escalation. Two of these in particular are relevant, fixed-price and cost-plus. A fixed-price agreement motivates the contractor to control accrued costs and, ideally, keep them below the contracted price. In contrast, a strictly cost-plus contract offers little motivation to control costs. In fact, when the profit is computed as a percentage of costs (rare these days), the contractor is actually motivated to “allow” costs to escalate. Other forms of contract agreements such as incentive contracts permit cost increases, but provide financial motivation to the contractor to minimize escalation. The relative pros and cons of different contract types are discussed in Chapter 12.



See Chapter 12

Inaccurate information and assumptions

Estimates are always based on available information and assumptions, and these should always be cross-checked. Are the customer’s and contractor’s assumptions about the costs correct? Does everyone agree about the work, materials, and other factors covered in the estimate? Are the cost rates for labor, material, equipment, and services current? Is information about available facilities, equipment, systems, and services to be provided by the customer or other stakeholders accurate? Every cost estimate should explicitly identify all the cost factors used to produce it.

Bias and ambition⁵

It is human nature for project champions to be optimistic. In fact, without champions most big projects would never start and everyone might be worse off. That optimism, however, can lead to overestimating the benefits and underestimating the costs. Promoters of big projects, like

Bandra-Worley in Example 9.1, know that if a project is important enough, sufficient funding will materialize, no matter the size of the overrun.

Example 9.1: Escalation of the Bandra-Worli Sea Link Project

Headlines from local news media referring to the Bandra-Worli Sea Link (BWSL) roadway and cable-stayed bridge in Mumbai—India's equivalent to San Francisco's Golden Gate Bridge:

January 1999 Government Clears Worli-Bandra Cable Bridge
February 2001 Worli-Bandra Sea-link Enters Crucial Stage
October 2002 Bandra-Worli Sea Link Toll To Be Costlier
October 2003 Bandra-Worli Sea Link May Hit a Dead End
January 2004 Bandra-Worli Sea Link Project Under Threat
July 2005 Sea Link In Trouble over Extension
May 2006 Bandra-Worli Sea Link To Be Ready By 2008

The 8-km bridge and its approaches bend 200 m into the Arabian Sea to connect downtown Mumbai with its western suburbs, and reduce travel time by half to about 30 minutes. Although architecturally stunning and a boon to local drivers, it is a good example of large-scale project woes.

The project was approved in early 1999 following 7 years of study, and was supposed to start in May, cost 650 crores (US \$120M), and finish mid-2001. But work did not begin until December, and the estimated completion date had already slipped to mid-2002. Then came the monsoons in 2000 and 2001, which brought the project to a near halt. In late 2001 the project's prime consultant, Sverdrup, was dropped for failure to provide a "competent project engineer." The replacement, Dar Consultants, modified the bridge design and added 2.8 km to its length and split the eight-lane main bridge into two four-lane roadways. By January 2002 the target completion date had slipped to March 2004. In October 2002 came the announcement that the project costs had increased by 50 crores (US \$9.2M), and due to a "paucity of funds" work had to be slowed and the completion date slipped to September 2004. A year later, monsoons and rough seas again halted work, further delaying the completion date to 2005. Meantime, complaints grew from fishermen concerned about the link's interference with their boats, and from environmentalists about its harm to marine ecology. In 2003 rains again stalled the project. The project's primary contractor, Hindustan Construction, requested an additional 300 crores to cover delays and design changes, but the government balked and offered to pay only 120 crores. The controversy stalled the project for another year, though eventually funds materialized and the project resumed. In June 2005 the completion date was reset to September 2006 and the project cost to 1,306 crores. In May 2006 the completion date was again pushed back to April 2008, but not until June 2009 was the bridge dedicated. In March 2010, all lanes opened to traffic. Estimated final cost: 1,600 crores (US \$336M).

As illustrated, schedule delays and cost escalation are inextricably connected. The 11-year history of the BWSL project saw a 9-year schedule slip and 180 percent cost increase. Contributing factors included unknowns (weather), changes in scope and requirements (bridge

and roadway design), social factors (livelihood and environmental impact concerns), economics (growing land values and interest), and management (dismissal of a major contractor).

Among the worst-offending projects in terms of overruns are **megaprojects**—projects costing at least \$1 billion. Beyond the reasons cited here, they suffer overruns from poor front-end planning, politics, and conflicting stakeholder requirements—topics covered in Chapter 20.



See Chapter 20

9.3 Elements of Estimates and Budgets

Both budgets and cost estimates state the cost of doing something; the difference is that the estimate comes first and forms the basis for the budget. A cost estimate might have to be refined many times, but once approved it becomes the budget and the amount to which the organization and work units then commit for performing the work. It is the agreed upon amount for what the work should cost and the **baseline** against which actual expenditures will be compared.

Project budgets and traditional fiscal operating budgets are similar; the difference is that the former covers the entire life of a project, the latter only a year at a time.

Estimates and budgets share the following elements:

- Direct labor expense,
- Direct non-labor expense,
- Overhead expense,
- General and administrative expense,
- Profit and total billing.

Direct labor expense⁶

Direct labor expense is the labor charge for the project. For each activity or work package, an estimate is made of the number of people or hours or days needed for each labor grade. This gives the distribution of labor hours or days required per labor grade. The labor hours for the various grades are then multiplied by their respective wage rates. The work package budget in Figure 9.2 shows the wage rates for three labor grades and the associated labor hours as distributed (time-phased) over a 6-month period. For example, the direct labor cost for an assistant in months 2–5 is $\$20 \times 100 = \$2,000$ per month.

When the wage rate is expected to change over the course of the work, a weighted average wage rate is used. In Figure 9.2, suppose the rate for an assistant is expected to rise from \$20 to \$25 in months 3, 4, and 5. Instead of \$8,000 for the assistant, the labor cost would be $100(\$20) + 100(\$25) + 100(\$25) + 100(\$25) = \$9,500$. The average wage rate for the assistant would thus be $\$9,500/400 \text{ hours} = \$23.75/\text{hour}$.

Direct non-labor expense

Direct non-labor expense is the total expense of non-labor charges applied directly to the activity. It includes subcontractors and consultants (where expenses are determined by contract), travel, telephone, computer time, material costs, purchased parts, and freight. This expense is represented in Figure 9.2 by the line “other direct cost.” Material costs and

Project <u>CASTLE</u>		Date <u>April 1, 1592</u>							
Department <u>Excavating</u>		Work package <u>Moat</u>							
Charge	Rate	Months*						Totals	
		1	2	3	4	5	6	Hours	Cost
Direct labor									
Professional	\$35/hour	50				50		100	3,500
Associate	\$30/hour		100	100	100	100		400	8,000
Assistant	\$20/hour								
Direct labor cost		1,750	2,000	2,000	2,000	3,750			11,500
Labor overhead	75%	1,312	1,500	1,500	1,500	2,813			8,625
Other direct cost*			100						100
Total direct cost		3,062	3,600	3,500	3,500	6,563			20,225
General/administrative	10%	306	360	350	350	657			2,023
Total costs		3,368	3,960	3,850	3,850	7,220			22,248
Profit	15%								
Billing total									

+Should extend for as many months as required by the project.
 *Should be itemized to include costs for materials, freight, subcontracts, travel, and all other nonlabor direct costs.

Figure 9.2
Typical 6-month budget for a work package.

freight charges include allotments for waste and spoilage and should reflect anticipated price increases, which sometimes appear as separate line items called **direct materials** and **overhead on materials**, respectively. Consultants and time on special equipment may appear as **support**.

Direct non-labor expenses also include necessities for installation and operation, such as instruction and maintenance manuals, engineering and programming documentation, drawings, and spare parts. Note that these are costs applied only for a specific project or work package and do not include general or overhead costs of doing business unless those costs are tied solely to the specific project.

On smaller projects the direct non-labor expenses are individually estimated for each work package. In larger projects, a simple percentage rate is applied to cover travel and freight costs. For example, 5 percent of direct labor cost might be included as travel expense and 5 percent of material costs as freight. These percentages are estimated in the same fashion as overhead rates, discussed next.

Overhead, general, and administrative expenses

Direct expenses for labor and materials are easily charged to a specific work package, but other expenses cannot so easily be charged to specific work packages or even to specific projects. These expenses, termed **overhead** or **non-direct expenses**, are the costs of doing business. They include whatever is necessary to house and support the labor, including building rents, utilities, clerical assistance, insurance, and equipment. Overhead is usually computed as a percentage of the direct labor cost. Frequently, the rate is 100 percent but ranges from as low as

25 percent for companies that do most of their work in the field to over 250 percent for those with expensive facilities, laboratories, and equipment.

The overhead rate is computed by estimating the annual business overhead expense, then dividing it by the projected total direct labor cost for the year. Suppose the total overhead for next year is projected to be \$180,000. If total anticipated direct labor charges are \$150,000, then the overhead rate is $180,000/150,000 = 1.20$. Thus, for every \$1.00 charged to direct labor on a project, \$1.20 is charged to overhead.

Although this is the traditional accounting method for deriving the overhead rate, for projects it results in an arbitrary allocation of costs, which is counterproductive for project cost control because most overhead cost sources are not tied to any particular project. More appropriate for projects is to divide overhead costs into two categories: **direct overhead** for costs that can be allocated in a logical manner; and **indirect overhead** for costs that cannot. Direct overhead costs can be traced to the support of a particular project or work package; such costs are allocated only among the specific projects or activities for which they apply. For example, the overhead cost for a department working on four projects is apportioned among the four projects based on the percentage of labor time it devotes to each; none of the department's overhead cost is allocated to projects that it is not working on.

The other kind of overhead, indirect, includes general expenses for the corporation. Usually referred to as **general and administrative expense**, or **G&A**, it includes taxes, financing, penalty and warranty costs, accounting and legal support, proposal preparation, marketing and promotion, salaries and expenses of top management, and employee benefits. Some of these expenses apply to no projects in particular, and others apply to only to certain projects. For example, corporate-level overhead (e.g., salary of the president, which applies to no particular project) would be allocated across all projects; project management overhead (e.g., salary of the project manager) would be allocated only among the projects she manages; departmental overhead (e.g. marketing and legal departments) would be allocated only to segments of specific projects to which they contributed. Often, G&A overhead is charged on a time basis, so the longer the project duration, the greater the G&A expense for the project.

In practice, the actual manner in which indirect costs are apportioned among projects varies. Table 9.1 for the SETI Company shows three methods for distributing indirect costs between two projects, MARS and PLUTO.⁷ Notice that although company-wide expenses remain the same, the cost of each project differs depending on the method of allocating indirect costs.

Customers want to know the allocation method used by their contractors, and contractors should know the allocation method used by their subcontractors. For example, method I in Table 9.1 is good for the customers when the project is direct labor (DL) intensive, but bad when it is direct non-labor (DNL) intensive. Method III is the opposite and gives a lower cost when the project is relatively non-labor intensive (i.e., labor costs are low but material and parts costs are high). This can be seen by comparing MARS (somewhat non-labor intensive) to PLUTO (labor intensive): for MARS, method I yields a project cost 162, but method II yields a project cost of only 146.25.

Overhead costs appear in projects in different ways. Any overhead expense that can be traced to specific work packages should be allocated to them directly. In Figure 9.2, overhead expenses that can be tied directly to the project appear on the line "General/administrative." Remaining overhead expenses that cannot be traced to specific work packages are assigned to a special "overhead" work package. This can be a single overhead work package for the entire project, or a series of packages each tied to an individual project stage or phase.

Table 9.1 Examples of indirect cost apportionment.

SETI Company: Company-Wide (Indirect Costs)			
Overhead (rent, utilities, clerical, machinery)		OH	120
General (upper management, staff, benefits, etc.)		G&A	<u>40</u>
		Indirect Total	160
Project Costs	MARS Project	PLUTO Project	Total
Direct labor (DL)	50	100	150
Direct nonlabor (DNL)	<u>40</u>	<u>10</u>	<u>50</u>
	90	110	200
		Direct Total	<u>200</u>
		Direct and Indirect Total	360
Three methods for apportioning indirect costs:			
I. Total indirect proportionate to total direct costs			
	MARS Project	PLUTO Project	Total
DL and DNL	90	110	200
OH and G&A	<u>72</u>	<u>88</u>	<u>160</u>
	162	198	360
II. OH proportionate to direct labor only; G&A proportionate to all direct costs			
	MARS	PLUTO	Total
DL	50	100	150
OH on DL	40	80	120
DNL	40	10	50
G&A on (DL and DNL)	<u>18</u>	<u>22</u>	<u>40</u>
	148	212	360
III. OH proportionate to direct labor only; G&A proportionate to DL and OH and DNL			
	MARS	PLUTO	Total
DL and OH and DNL	130	190	320
G&A	<u>16.25</u>	<u>23.75</u>	<u>40</u>
	146.25	213.75	360

Profit and total billing



See Chapter 12

Profit is the amount left over after expenses have been subtracted from the contractual price. It can also be an agreed-to fixed fee or a percentage of total expenses (15 percent in Figure 9.2), determined, in part, by the kind of contract, as discussed in Chapter 12. **Total billing** is the sum of total expenses and profit. Total billing and profit are included in estimates for the overall project, groups of work packages, and subcontracted work, but usually do not appear on budgets for lower-level work elements. All these numbers reflect budgeted costs, *not actual costs*. A goal of project management is to execute the project as close to budgeted costs as possible, though almost always actual costs and budgeted costs will differ. The sometimes-worrisome question is: by how much?

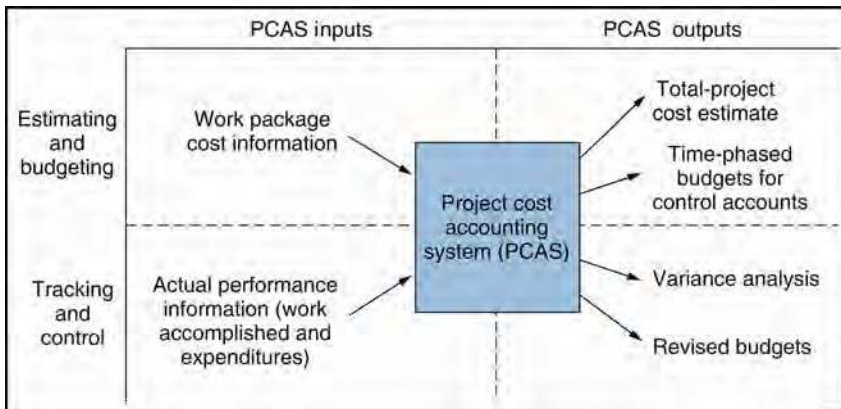


Figure 9.3
Elements of a project cost accounting system.

9.4 Project Cost Accounting System

A project might consist of hundreds or thousands of elements—workers, materials, and facilities, every one of which must be estimated, budgeted, and controlled. To expedite the process, reduce confusion, and improve accuracy, it helps to use a system to compute the estimates, create, store and update the budgets, and track the actual expenses. Such a system, called a **project cost accounting system (PCAS)**, is initially set up by the project manager, project accountant, or project management office (PMO). The main focus of the PCAS is project costs, although it also assists with tracking and controlling schedules and work progress. When the PCAS is combined with functions for project planning, control, and reporting, the whole system is referred to as the **project management information system (PMIS)**. The PMIS is discussed in Chapter 6.



See Chapter 6

The PCAS is used throughout the project life cycle. During project conception and definition, it accumulates work package costs estimates and produces the total project cost estimate. These estimates become the basis upon which project and work package budgets will be created.

During project execution, the PCAS accumulates, credits, and reports project and work package actual expenditures. It creates time-phased budgets (example, Figure 9.2), which help managers monitor costs and verify that the work has been completed and charged. The system also enables budget revisions. The functions of the PCAS are summarized in Figure 9.3.

Example 9.2: PMIS for Estimating Labor Requirements and Costs⁸

Sigma Associates is a large architectural/engineering firm that developed its own PCAS, a module within the PMIS that it uses for estimating, planning, and scheduling.

At Sigma, the project manager begins each project by creating a WBS to identify the main work packages. Using a menu in the PMIS, she reviews the history of similar work



See Chapter 7

packages from previous projects and the kind and amount of labor they required. By entering factors related to project size, construction costs, and type of client, she can estimate the labor requirements for every activity in the project. Using the PMIS, the project manager combines these labor estimates with requirements for existing projects to produce a labor resource loading forecast; this enables her to determine whether sufficient labor is available. If it is not, the project manager uses the system to review options such as modifying the schedule, using overtime, or leveling resources (discussed in Chapter 7).

The project manager gives the labor requirements estimate to the project comptroller who, through the PCAS, applies existing or projected hourly rates to every activity. The comptroller then adds in employee benefits and labor overhead to produce an estimate for direct labor cost.

With information from the company general ledger, the comptroller computes the overhead rate, which he uses to charge the project for its share of company-wide expenses. He then uses the PCAS to roll up all of the estimated expenses and create an estimated project budget. Last, the comptroller analyzes the estimated budget along with the project plan for profitability. If the budget and plan show a reasonable profit and the comptroller and project manager both agree to the budget, the project is accepted. If not, a different plan is sought that maintains the same quality standards but is more profitable.

Time-phased budgets

The project manager needs an easy way to monitor and control the project—to know where and when expenses are accruing, how well the project is progressing, and where problems are developing. One way is with a **time-phased budget** that consolidates the budgetary and schedule information to show the distribution of budgeted costs according to the project schedule. Figure 9.2 is an example; it shows the distribution of costs in a work package over months 1 through 6. The PCAS generates reports like this for each work package, thus enabling a project manager during project execution to compare budgeted costs for the work package with actual expenditures on a month-by-month basis.

In projects where a substantial portion of the costs originate from subcontracts and purchased items or services, a special time-phased budget is prepared for *procured* goods, work, and services. In large projects this budget is controlled by a materials or procurement manager.

9.5 Budgeting Using Control (or Cost) Accounts

Budgeting and cost monitoring in small projects can be done using a single budget for the project as a whole. This budget, perhaps similar to the one in Figure 9.2, is used to compare actual costs with budgeted costs throughout the project.

On larger projects, however, a single, project-wide budget is too insensitive, making it difficult to quickly locate the sources of cost overruns once the project is underway and expenses accrue. The better way is to subdivide the project budget into smaller budgets called **control accounts** (or **cost accounts**), each representing a work package on the WBS. Large projects have tens of control accounts; very large projects have hundreds.

The control account is the basic project tracking element in the PCAS. The accounts are set up in a hierarchy, similar or identical to the WBS. The lowest level account usually corresponds

to a work package, although when the number of work packages is very large, one account might represent several work packages combined. Like work packages, each account might include:

- A work description
- A time schedule
- Who is responsible
- Material, labor, and equipment required
- A time-phased budget.

Control accounts are also established for project costs not readily attributable to any specific work package. For example, separate control accounts are established to cover expenses for materials or equipment for use by anyone or in any work package, or for work such as administration, supervision, or inspection that applies across the entire project. These accounts are usually set up for the duration of the project or period-by-period as needed or as funds are appropriated.

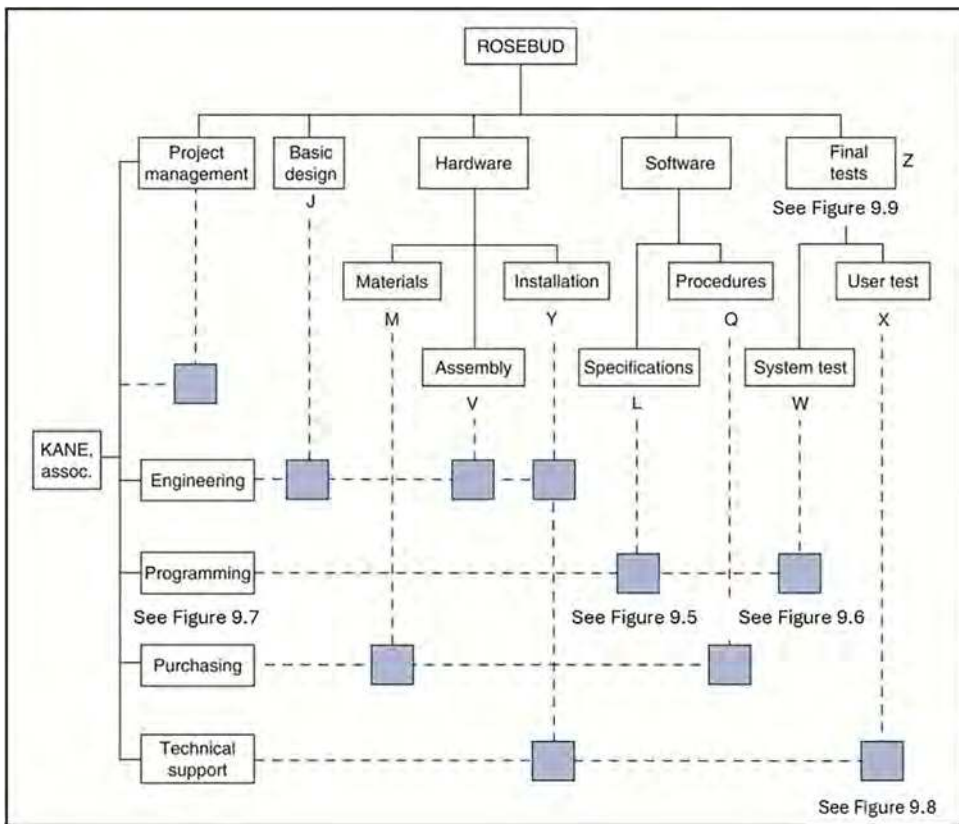
With a PCAS and control-account structure, it is easy to monitor cost performance for each work package, group of work packages, and the project as a whole. For example, consider the ROSEBUD project and its contractor, KANE & Associates. Figure 9.4 shows the ROSEBUD project WBS and the KANE organization chart. The shaded boxes represent locations of control accounts; notice that each account represents all or part of a work package for which a functional area is responsible. Example: the Programming Department is responsible for work packages L and W; Engineering and Tech Support share responsibility for work package Y. (In the latter case, good practice is for one of the departments—Engineering or Tech Support—to take primary responsibility for the work package and to delegate the remaining responsibility to the other.)

The WBS for ROSEBUD consists of nine work packages performed by four functional departments, plus an additional work package for project management. During the estimating phase each department submits a cost estimate for the work packages in its part of the project. Upon approval, with additions for overhead and G&A, each department's estimate becomes a budget. In Figure 9.4, the ten shaded boxes represent departments/work packages for which initial cost estimates were made and, subsequently, budgets and control accounts were established. Figures 9.5 and 9.6 show, respectively, the time-phased budget portions of the control accounts for work packages L and W.

9.6 Cost Summaries⁹

With the control account structure, Figure 9.4, high-level summary accounts can be developed by consolidating control accounts for the WBS and organizational hierarchies. Such consolidation is useful for monitoring the performance of individual departments and segments of the project. Consolidating the accounts in Figure 9.4 horizontally results in a control account for each functional department. Figure 9.7 illustrates: it is the time-phased budget for the programming department, which is the sum of the budgets for work packages L and W (Figures 9.5 and 9.6). Consolidating control accounts vertically through the WBS results in control accounts for individual work packages, clusters of work packages, or the project as a whole. Figure 9.9 is the time-phased budget part of the account for the final tests portion of the project—the sum of the budgets for work packages W and X (Figures 9.6 and 9.8).

The PCAS and control-account structure permit costs to be summarized in a variety of ways: Figure 9.10 shows budgeted amounts aggregated vertically and horizontally, and Table 9.2 summarizes the budgeted cost elements for the five departments and nine work

**Figure 9.4**

Integration of WBS and organization structure showing control accounts. (See Figures 9.5 through 9.9 for details.)



See Chapter 13

packages in the ROSEBUD project. Thus, through the PCAS and cost account structure, it is easy for a project manager to identify cost deviations from budget at the project level or the company level, and to readily trace such deviations to the work packages and departments responsible for them. Chapter 13 will describe this more fully.

9.7 Cost Schedules and Forecasts¹⁰

Questions arise during project planning about how expenditures will vary throughout the project, which periods will have the heaviest cash requirements, and how expenditures will compare to income. To answer these and other such questions, it helps to anticipate the expected, estimated “pattern of expenditures” throughout the project. Following are examples.

Cost analysis with early and late start times

A simplifying assumption used in cost estimating is that the costs in each work package are distributed uniformly. For example, a 2-week, \$22,000 work package is assumed to have

Project <u>ROSEBUD</u>		Date _____							
Department <u>Programming</u>		Work package <u>L - Software specifications</u>							
Charge	Rate	Months*						Totals	
		1	2	3	4	5	6	Hours	Cost
Direct labor									
Professional	\$35/hour		130					130	4,550
Associate	\$30/hour		50	100				150	4,500
Assistant	\$20/hour			100				100	2,000
Direct labor cost			6,050	5,000					11,050
Labor overhead	75%		4,538	3,750					8,288
Other direct cost*									0
Total direct cost			10,588	8,750					19,338
General/administrative	10%		1,059	875					1,934
Total costs			11,647	9,625					21,272

*Should extend for as many months as required by the project.

*Should be itemized to include costs for materials, freight, subcontracts, travel, and all other nonlabor direct costs.

Figure 9.5

Budget for programming department for work package L.

Project <u>ROSEBUD</u>		Date _____							
Department <u>Programming</u>		Work package <u>W—System test</u>							
Charge	Rate	Months*						Totals	
		1	2	3	4	5	6	Hours	Cost
Direct labor									
Professional	\$35/hour						20	20	700
Associate	\$30/hour						50	50	1,500
Assistant	\$20/hour								0
Direct labor cost							2,200		2,200
Labor overhead	75%						1,650		1,650
Other direct cost*							0		0
Total direct cost							3,850		3,850
General/administrative	10%						385		385
Total costs							4,235		4,235

*Should extend for as many months as required by the project.

*Should be itemized to include costs for materials, freight, subcontracts, travel, and all other nonlabor direct costs.

Figure 9.6

Budget for programming department for work package W.

Project <u>ROSEBUD</u>		Date _____							
Department <u>Programming</u>		Work package <u>ALL</u>							
Charge	Rate	Months ⁺						Totals	
		1	2	3	4	5	6	Hours	Cost
Direct labor									
Professional	\$35/hour		130				20	150	5,250
Associate	\$30/hour		50	100			50	200	6,000
Assistant	\$20/hour			100				100	2,000
Direct labor cost			6,050	5,000			2,200		13,250
Labor overhead	75%		4,538	3,750			1,650		9,938
Other direct cost*									0
Total direct cost			10,588	8,750			3,850		23,188
General/administrative	10%		1,059	875			385		2,319
Total costs			11,647	9,625			4,235		25,507

⁺ Should extend for as many months as required by the project.
^{*} Should be itemized to include costs for materials, freight, subcontracts, travel, and all other nonlabor direct costs.

Figure 9.7
Budget for programming department.

Project <u>ROSEBUD</u>		Date _____							
Department <u>Technical support</u>		Work package <u>X—User test</u>							
Charge	Rate	Months ⁺						Totals	
		1	2	3	4	5	6	Hours	Cost
Direct labor									
Professional	\$35/hour						10	10	350
Associate	\$30/hour						40	40	1,200
Assistant	\$20/hour								
Direct labor cost							1,550		1,550
Labor overhead	75%						1,163		1,163
Other direct cost*						1,200	2,107		3,307
Total direct cost						1,200	4,820		6,020
General/administrative	10%					120	482		602
Total costs						1,320	5,302		6,622

⁺ Should extend for as many months as required by the project.
^{*} Should be itemized to include costs for materials, freight, subcontracts, travel, and all other nonlabor direct costs.

Figure 9.8
Budget for work package X.

Project <u>ROSEBUD</u>		Date _____							
Department <u>Technical support; Programming</u>		Work package <u>(W + X) Final Tests</u>							
Charge	Rate	Months [*]						Totals	
		1	2	3	4	5	6	Hours	Cost
Direct labor									
Professional	\$35/hour						30		1,050
Associate	\$30/hour						90	30	2,700
Assistant	\$20/hour							90	0
Direct labor cost							3,750		3,750
Labor overhead	75%						2,813		2,813
Other direct cost*						1,200	2,107		3,307
Total direct cost						1,200	8,670		9,870
General/administrative	10%					120	867		987
Total costs						1,320	9,537		10,857

^{*} Should extend for as many months as required by the project.
^{*} Should be itemized to include costs for materials, freight, subcontracts, travel, and all other nonlabor direct costs.

Figure 9.9
Budget for final tests, work packages W and X combined.

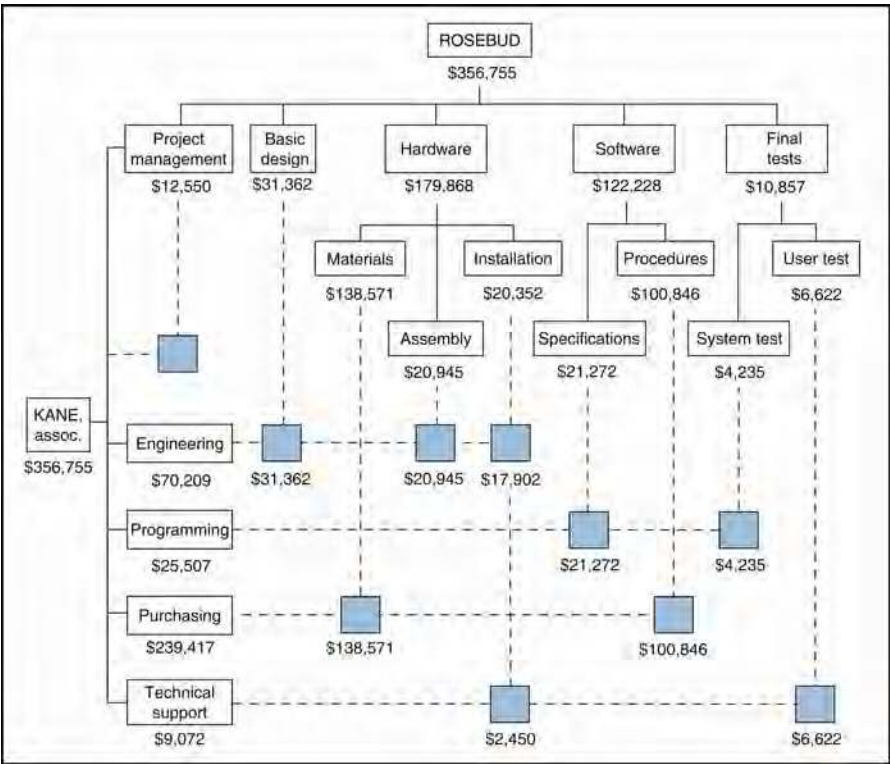


Figure 9.10
Aggregation of control account information by project and organization.

Table 9.2 Cost summary for ROSEBUD project.*

	Labor(\$)				Overhead(\$)				Materials	General and Administrative	Total Cost
	Engineering	Programming	Purchasing	Technical Support	Engineering	Programming	Purchasing	Technical Support			
Total project	22,800	13,250	2,230	2,850	22,800	9,938	1,673	2,138	235,236	31,290	356,755
Project management											12,550
Activity J	7,200				7,200				14,111	2,851	31,362
Activity L*		11,050				8,288				1,934	21,272
Activity M			1,100				825		124,050	12,596	138,571
Activity Q			1,300				818		89,700	9,168	100,846
Activity V	8,200				8,200				2,641	1,904	20,945
Activity Y	7,400			1,300	7,400			975	1,427	1,850	20,352
Activity W		2,200				1,650				385	4,235
Activity X				1,550				1,163		602	6,622

*Refer to Figure 9.5 to see, for example, how costs in this row were developed.

expenditures of \$11,000 per week. With this assumption, it is easy to create a project **cost schedule** that shows the cost *each week* of the entire project.

As an example, look at Table 9.3, where the weekly direct cost for each activity is the total cost divided by the time. Now look at Figure 9.11—the time-based network for the LOGON project based upon early start times. Using this network, the project cost *each week* can be computed by adding the costs for all activities scheduled in the week. (Note, this is the same procedure as described in Chapter 7 for determining the resource loading.) According to Figure 9.11, during the first 10 weeks only Activity H is scheduled, so the project weekly cost will be \$10K. Over the next 6 weeks Activities I and J are scheduled, so the weekly cost is their sum, \$16K + \$8K = \$24K. Further along, in weeks 17 and 18, four work packages—I, K, L, and M—are scheduled, so the weekly cost is their sum, \$8K + \$4K + \$18K + \$21K = \$51K. These weekly expenses, shown in the third column in Table 9.4, represent the forecasted project cost schedule. The fourth column, cumulative expense, is the forecasted total project cost for any given week. Cumulative and weekly expenses are graphed in Figure 9.12; the cumulative part is called an **S-curve**.



See Chapter 7

Using the same procedure, the forecasted project cost schedule based on *late* start times can be prepared. Figure 9.13 is the time-based network for LOGON using late start times. Table 9.4, the last two columns, are the late-start weekly and cumulative costs.

Given the early and late cost figures in Table 9.4 the effects of delaying activities on project costs can be analyzed. By comparing costs based on early start times with those from late start times, shown in Figure 9.14, the influence of schedule changes on project costs is readily apparent.

Table 9.3 Activities, time, cost, and labor requirements (result of work breakdown analysis).

Activity	Time (Weeks)	Total Cost (\$K)	Weekly Direct Cost (\$K)	Weekly Labor Requirements (Workers)
H	10	100	10	5
I	8	64	8	4
J	6	96	16	8
K	4	16	4	2
L	2	36	18	6
M	4	84	21	3
N	4	80	20	2
O	5	50	10	5
P	5	60	12	6
Q	5	80	16	2
R	5	0	0	0
S	3	0	0	0
T	3	0	0	0
U	1	14	14	9
V	5	80	16	14
W	2	24	12	6
X	3	36	12	6
Y	8	104	13	14
Z	6	66	11	5
Total Direct Cost—\$990K				

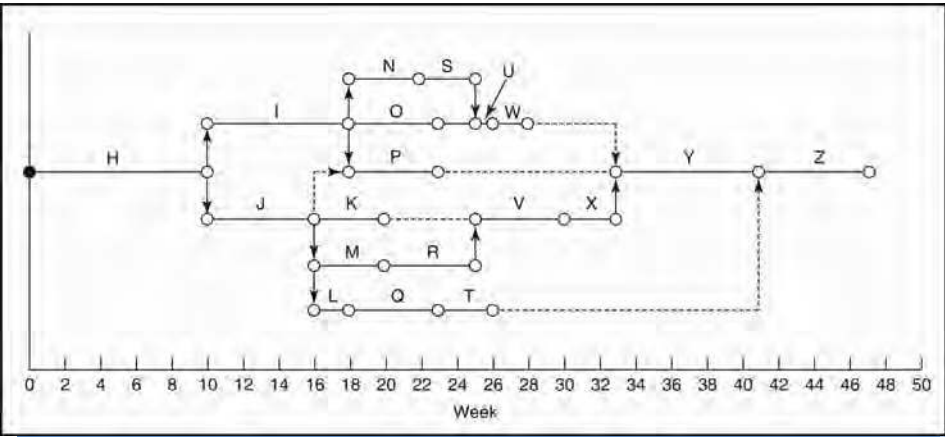


Figure 9.11
Time-based network for the LOGON project using early start times.

Table 9.4 LOGON project weekly expense using early start times in \$1,000.

Week	Early Start Activities During Week	Early Start Weekly Expense	Early Start Cumulative Expense	Late Start Activities During Week	Late Start Weekly Expense	Late Start Cumulative Expense
1	H	10	10	H	10	10
2	H	10	20	H	10	20
3	H	10	30	H	10	30
4	H	10	40	H	10	40
5	H	10	50	H	10	50
6	H	10	60	H	10	60
7	H	10	70	H	10	70
8	H	10	80	H	10	80
9	H	10	90	H	10	90
10	H	10	100	H	10	100
11	I, J	24	124	J	16	116
12	I, J	24	148	J	16	132
13	I, J	24	172	J	16	148
14	I, J	24	196	J	16	164
15	I, J	24	220	I, J	24	188
16	I, J	24	244	I, J	24	212
17	I, K, L, M	51	295	I, M	29	241
18	I, K, L, M	51	346	I, M	29	270
19	K, M, N, O, P, Q	83	429	I, M	29	299
20	K, M, N, O, P, Q	83	512	I, M	29	328
21	N, O, P, Q	58	570	I, R	8	336
22	N, O, P, Q	58	628	K, I, R	12	348
23	O, P, Q	38	666	K, R	4	352
24	-	0	666	K, R, N	24	376

Week	Early Start Activities During Week	Early Start Weekly Expense	Early Start Cumulative Expense	Late Start Activities During Week	Late Start Weekly Expense	Late Start Cumulative Expense
25	–	0	666	K, R, N	24	400
26	U, V	30	696	N, O, V	46	446
27	V, W	28	724	N, O, V	46	492
28	V, W	28	752	S, O, V	26	518
29	V	16	768	S, O, P, V	38	556
30	V	16	784	S, O, P, V	38	594
31	X	12	796	U, P, X	38	632
32	X	12	808	W, P, X, L	54	686
33	X	12	820	W, P, X, L	54	740
34	Y	13	833	Y, Q	29	769
35	Y	13	846	Y, Q	29	798
36	Y	13	859	Y, Q	29	827
37	Y	13	872	Y, Q	29	856
38	Y	13	885	Y, Q	29	885
39	Y	13	898	Y, T	13	898
40	Y	13	911	Y, T	13	911
41	Y	13	924	Y, T	13	924
42	Z	11	935	Z	11	935
43	Z	11	946	Z	11	946
44	Z	11	957	Z	11	957
45	Z	11	968	Z	11	968
46	Z	11	979	Z	11	979
47	Z	11	990	Z	11	990

The shaded region between the two S-curves in the top part of the figure—the **feasible budget region** based on the early and late cumulative expenses—shows the range of budgets permissible by changes in the project schedule. The lower part of the figure shows the impact on weekly costs of delaying activities.

When funding restrictions constrain project expenditures, cost schedules reveal places where the schedule must be changed. For example, Figure 9.14 shows a peak weekly expense of \$82,000 in Weeks 18 and 19. If a weekly budget ceiling of \$60,000 per week were imposed on the project, then late start times would be preferred because they result in a more “level” cost profile and peak expense of only \$54,000 in weeks 32–33. The methods for leveling and smoothing resources discussed in Chapter 7 are applicable to leveling project costs, where costs are treated as just another resource.



Effect of late start times on project net worth

Owing to the time value of money, the net present worth of work done farther in the future is lower than the same work if done earlier. Thus, delaying work in a lengthy project can provide savings in the present worth of project costs. For example, suppose the expected duration of the LOGON project is 47 months (rather than 47 weeks as assumed previously). If the annual

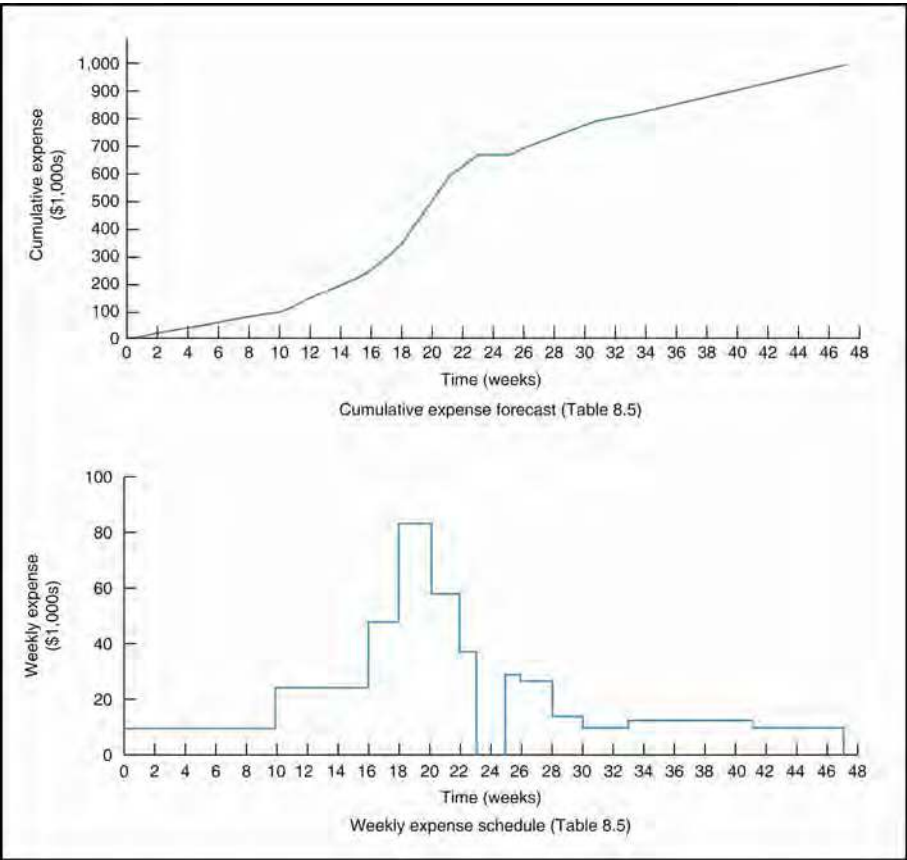


Figure 9.12
Cumulative expenses (top) and planned weekly expenses for the LOGON project.

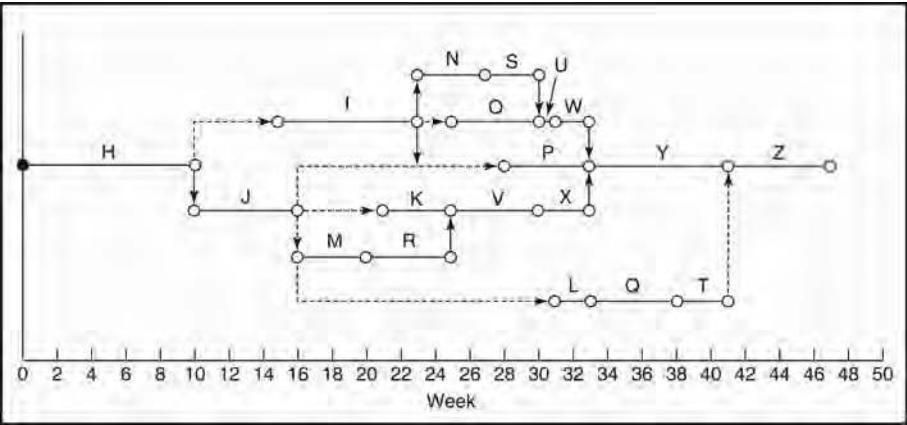


Figure 9.13
Time-based network for the LOGON project using late start times.

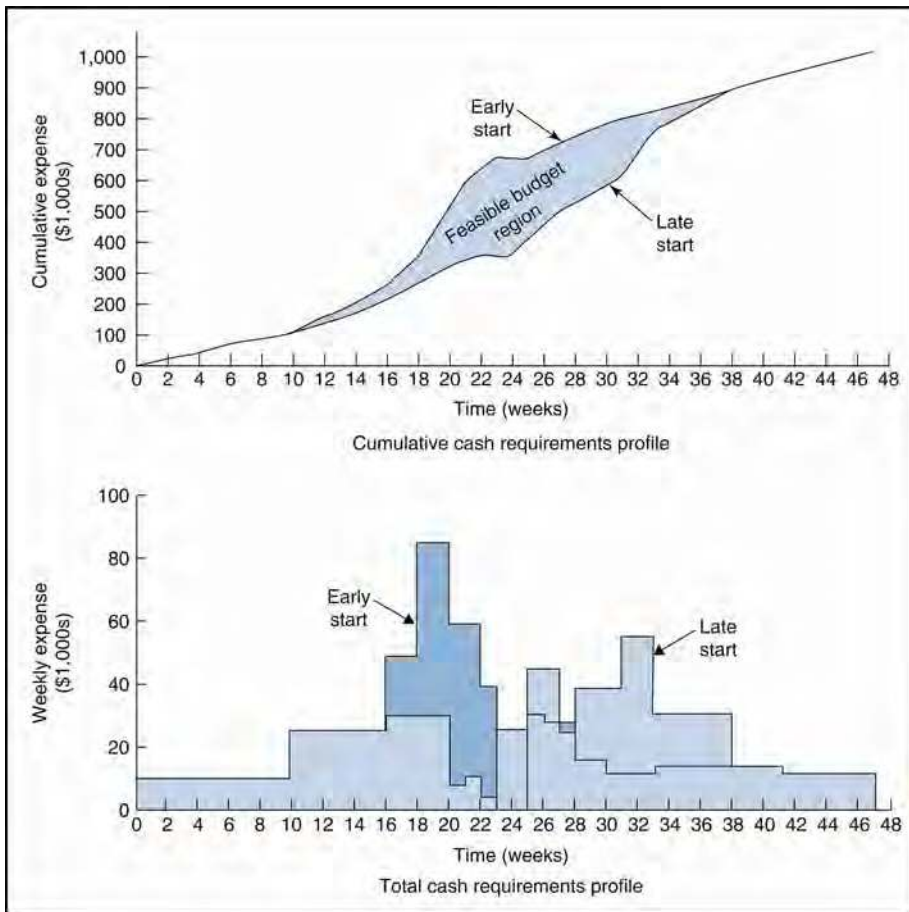


Figure 9.14
Comparison of expenses, early versus late start times.

interest rate is 24 percent, compounded at 2 percent per month, the present worth for the project would be \$649,276. This is computed by using the monthly expenses in Table 9.4, column 3 (again, assuming the “weekly” shown to be monthly instead) and discounting them back to time zero. Now, when the late start times and the corresponding numbers in column 6 are used instead, the present worth is only \$605,915—a savings of \$43,361.

Does this mean that activities should be delayed until their late start dates? Not necessarily. Remember, delaying activities consumes slack time and leaves less time to deal with problems that could delay the project. Thus, whether or not to delay work should depend also on the *certainty of the work*. Activities that are unlikely to encounter problems can be started later to take advantage of the time value of money. As discussed in Chapter 11 on project risk, activities that are less familiar, such as research and development work, should be started earlier to allow time (and retain slack that might be needed) for unanticipated problems and delays. Also, whether or not to delay work should depend on the schedule of customer payments. If payments are tied to project milestones and impact cash flow, then work tied to those milestones should not be delayed.



Cash flow

A problem that project managers often face is maintaining positive cash flow, i.e., assuring that the cumulative cash inflows (payments received) exceed the cumulative cash outflows (payments made). Ideally, differences between cash in and cash out throughout the project will be small.¹¹ The project manager must do a juggling act, balancing income from the customer (such as milestone payments) or other funding sources with expense payments for labor, subcontractors, materials, and equipment. To help maintain this balance the manager can, for example, take advantage of the time lag between when materials and equipment are acquired, and when payments for them must be made.

Figure 9.15 shows an example of forecasted cash flow. The S-curve for all contractually agreed-to sources of income over the life of the project is compared to the S-curve for all foreseeable expenditures, direct and indirect, plus any penalty costs or scheduled payments in case the project is completed late. The deficit between forecasted income and estimated expenditures represents the amount of **working capital** needed to meet payments for expenditures. Based upon this forecast, a **funding plan** should be created to ensure sufficient working capital will be available throughout the project.

As mentioned, customer payments are sometimes made at milestones tied to completion of deliverables or project phases. Such payments help the contractor to cover its costs. However, with these payments comes a risk to the customer. If the project encounters problems, an unscrupulous contractor who has already received several payments could walk away from the project and leave the customer in a fix. One way the customer can keep “hold” on the contractor is to withhold a significant portion of the agreed upon payment, called **retention money**, until all work is satisfactorily completed. Another way is to withhold a portion of the final payment, called a **performance guarantee**, for a period following handover of the end-item and until all defects discovered by the customer have been rectified.

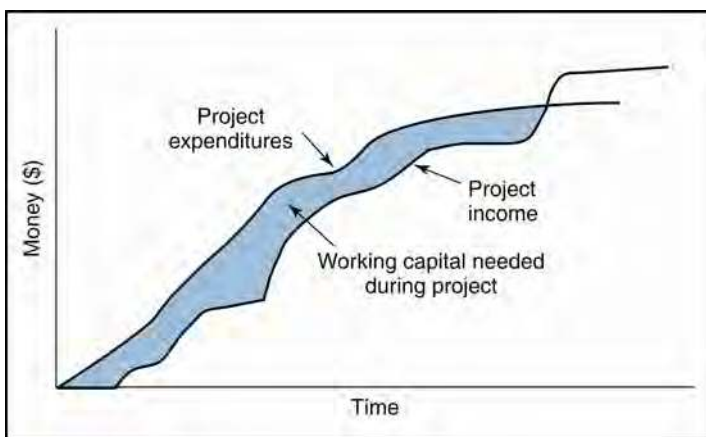


Figure 9.15
Balancing project income with expenditures.

9.8 Life Cycle Costs

Life cycle costs (LCC) represent all the costs of a system, facility, or product throughout its life, cradle-to-grave. The concept originated in military procurement when managers realized that costs to develop and implement a system represent but the tip of the cost iceberg and that the costs to operate (e.g. fuel consumption) and maintain (e.g. parts replacement) the system are usually far greater. Whereas the emphasis in this chapter has been on **project costs**, i.e., costs incurred during the project phases of definition and execution, LCC include costs *after the project* and for the remainder of the system life cycle—the **operation phase** and eventual disposal of the end-item (and, occasionally, the conception phase too).

Anticipating the LCC is necessary because it influences decisions. For example, suppose three contractors submit proposals to build a plant, and each proposal contains not only the plant's construction cost but also its expected operating costs. If the bids are similar in terms of plant features and construction costs, the proposal with the lowest plant operating costs will likely win.

The LCC similarly affect decisions regarding development projects, and the economic analysis in feasibility studies (Chapter 3) should consider *all costs* for acquisition, operation, maintenance, and disposal of the systems to be developed. For example, most US aerospace manufacturers in the 1970s were hesitant to develop a supersonic commercial aircraft because of cost and environmental impact concerns. Costs to develop and produce the aircraft were projected to be very high, as were costs for operation and maintenance. Among the concerns were whether enough people would pay the very high ticket prices necessary for airlines to make a profit, and whether enough airlines would purchase the aircraft for the manufacturers to make a profit. Ultimately, many felt the answer was no on both counts. In the US, Congress cancelled subsidies to the companies developing the aircraft, and the program was dissolved. Meantime, the Europeans decided differently and went ahead to develop and manufacture the Concorde, only 14 of which went into service. Although Concorde flew for nearly 27 years (the last one was retired in 2003), the LCC were never recouped. No one made a profit, and had the governments of Great Britain and France not provided subsidies, the airlines and manufacturers would all have lost money.

Key design decisions affecting the operation and maintenance of a system are made early in the project life cycle—during conception and definition. A product with a high development cost and purchase price becomes appealing only if it can be designed such that it will have a relatively low operating cost. For example, an electric vehicle might be higher priced than gasoline-powered vehicles, but customers pay the premium knowing that over the vehicle's life they will recoup the excess price through fuel savings and less pollution. Estimating LCC involves making assumptions about the technology, the market, and product demand, and relies on historical costs of similar systems and projects, but it is a sensible way to assess projects, especially when there is a choice between alternative designs or proposals and the assumptions are credible.

The LCC should also account for the time necessary to develop, build, and install the end-item, i.e., the time *before* the facility or system becomes operational or the product is “launched” to market. Time is important because it determines how soon the end-item will start generating revenues, gaining market share, and accruing profits or other benefits. The higher costs of speeding up the project are compared to the benefits gained from an earlier completion or product launch. Similarly, the **cost of disposal** at the end of the life cycle is also estimated; for facilities such as mines and nuclear power plants that will require an elaborate formal shut-down and rehabilitation process after their useful lives, this cost can be substantial.

Analysis of LCC is also necessary for setting targets on development and operating costs and making design tradeoff decisions to achieve those targets. Following is an example.

Example 9.3: Life Cycle Costs for an Operational Fleet of Spaceships

This illustration extends previous SpaceShipOne examples, but the numbers are hypothetical (and certainly too low).

Based on experience from SpaceShipOne, a larger spaceship and mothership are to be designed. The new spaceship will carry a pilot plus four paying passengers, go as high as 120 km, and be capable of 20 flights per year over an operational life of five years. The cost of developing and producing four of these spaceships and two motherships is estimated at \$80 million. Meantime, a survey indicates that the number of people worldwide willing to pay the \$190,000 ticket price to fly on these spaceships is at least 1,000 per year.

A “spaceline” that will use and maintain the spaceships is being created for a startup cost of \$10 million. Operational costs for the spaceline consist of two parts: annual costs for ground operations (reservations, personnel, ground facilities, etc.)—\$2 million/year; and per-flight costs for flight operations (fuel, parts, repairs, etc., for the spaceship and the mothership)—\$0.4 million/flight. These costs are assumed constant for every year and flight, although realistically they would vary up or down depending on inflation, the learning curve, efficiencies, and economies of scale as more spaceships are added to the fleet. Annual revenues are assumed constant too, though they will likely grow as additional spaceships become operational. Given these costs and ignoring other factors (e.g., time value of money), what is the LCC for the venture?

Assumptions

4 spaceships @ 20 flights/year each = 80 flights/year (320 passengers/year, which lies well within the estimated annual demand); 5 years of operation.

Costs

Development and manufacturing: \$80 million
 Spaceline startup: \$10 million
 Ground operations: \$2 million/year
 Flight operations: \$0.4 million/flight
 Ticket price: \$190,000 (marketing slogan: “Now YOU can go to space for under \$200,000!”)

LCC model

LCC (\$ million) = Development and production cost + Startup cost + Operating cost (5 years)
 $= \$80 + \$10 + \{[5 \text{ yr} \times \$2] + [5 \text{ yr} \times 80 \text{ flights} \times \$0.4]\} = \$260 \text{ million}$
 Total revenues (\$ million) = $[5 \text{ yr} \times 80 \text{ flights} \times 4 \text{ passengers} \times \$0.190] = \$304 \text{ million}.$

Bottom line: assuming the assumptions are correct, revenues will exceed costs by \$44 million.

All the numbers are estimates, but some are more certain than others. For example, based upon experience with SpaceShipOne, the development cost might be fairly certain, but due to lack of long-term operational experience the per-flight cost is fairly uncertain.

Startup and ground operations costs, if analogous to airline operations (a big *if*), might be somewhat certain, although passenger demand might be fairly uncertain.

The LCC plays an important role in system design and development. The LCC can be modeled, and a sensitivity analysis of the model performed to see what happens if costs increase or decrease to show best case, most likely, and worst-case scenarios. The model can be used to determine by how much and in what combination the costs must vary for the enterprise to become lucrative (or disastrous).

The LCC model can also be used to set cost targets. If the decision is made to proceed with the \$80 million development and production cost, then the project must be planned, budgeted, and controlled so as to stay close to that amount. If the per-flight cost is set at \$0.4 million, the project must strive to develop vehicles that will cost no more than that to operate. This will affect innumerable design decisions pertaining to many details. Early on, the design analysis must consider major alternatives (e.g., to carry five or six passengers, not four) and the expected costs, revenues, and benefits for each.

The best and only truly comprehensive approach to estimating and analyzing LCC is with a team comprised of people that represents all phases of the system development cycle—a cross-functional team of designers, builders, suppliers, and users, i.e., a **concurrent engineering** team as discussed in Chapter 15.



See Chapter 15

Impact of early decisions on life cycle costs

The importance of carefully defining requirements and the end-item system and preparing the project plan—in other words, devoting careful attention to decisions made in Phases A and B of the project is illustrated in Figure 9.16, which shows the percent of life cycle costs committed to versus stages of the project. For example, the figure shows that as much as 80 to 90 percent of a product's LCC is determined by decisions made in the project's concept and design stages (coinciding with Phases A and B), which is well before the product will be produced and used. This means that whatever the total product LCC, 80–90 percent of it is based upon choices made in those two stages of the project.¹² Unless those decisions correctly account for what will happen later during production and operation, the result will be a protracted systems development effort, delayed launch of the product, and higher than necessary production and operating costs.

9.9 Summary

Cost estimation and budgeting are part of the project planning process. Cost estimation logically follows work breakdown and precedes project budgeting. Accurate cost estimates are necessary to establish realistic budgets and to provide standards against which to measure actual costs; they are thus crucial to the financial success of the project.

Costs in projects have a tendency to escalate over time. Defining clear requirements and work tasks, employing skilled estimators, being realistic in estimating, and anticipating escalation causes such as inflation all help to minimize escalation.

The approved estimates become budgets after contingency reserves have been added. The project budget is subdivided into smaller budgets called control accounts. Control accounts are

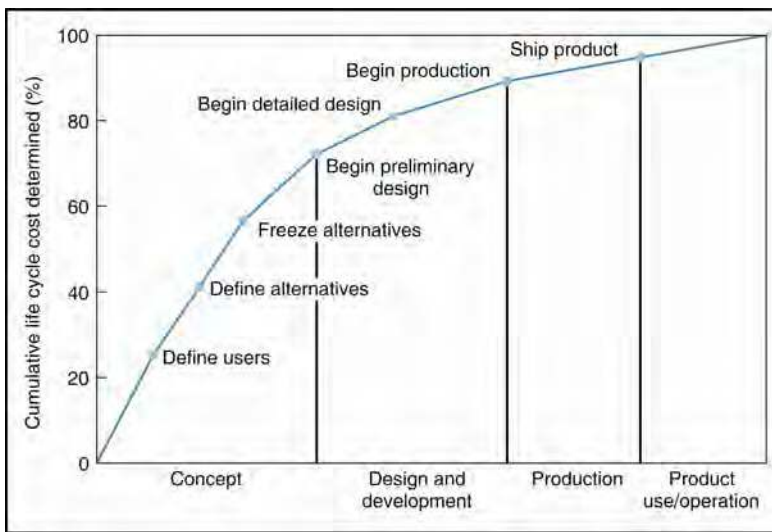


Figure 9.16
Percent of cumulative life cycle cost as determined by stage of the systems development life cycle.

derived from the WBS and project organization hierarchies and are the budget equivalent to work packages. In large projects a project cost accounting system (PCAS) is useful for aggregating estimates and maintaining a system of control accounts for budgeting and control.

Cost schedules are derived from time-phased budgets and show the pattern of costs and expenditures throughout the project. They are used to identify cash and working capital requirements for labor, materials, and equipment.

Forecasted project expenditures and other cash outflows are compared to schedule payment receipts and income sources to predict cash flow throughout the project. Ideally expenditures and income are balanced so that the contractor can maintain a positive cash flow. The forecasts are used to prepare a plan that guarantees adequate funding support for the project.

Life cycle costs (LCC) represent all costs of a system throughout its lifetime, cradle-to-grave. Whereas project management tends to emphasize costs incurred during the project, LCC includes costs after the project and for the remainder of the system's life. Anticipating the LCC impacts key decisions about the system made early in the project; in theory, a system with a high development cost is worthwhile only if it can be designed and built to have a low operating cost over its lifetime.



Review Questions and Problems

1. Define cost escalation. What are major sources of cost escalation?
2. Describe what the term "phased (rolling wave) project planning" means.
3. How do changes in requirements cause cost escalation?
4. How does the type of contractual agreement influence the potential for cost escalation?
5. What is the relationship between phases of the project life cycle and cost escalation?
6. What are life cycle costs and how are they different from project costs?

7. Describe the PCAS. What is its purpose and how is it used in project planning?
8. What is a time-phased budget? What is the difference between a budget and a cost estimate?
9. Distinguish recurring costs from nonrecurring costs.
10. What are six cost elements shared by most estimates and budgets?
11. How are direct labor expenses determined?
12. What expenses are included under direct non-labor?
13. How is the overhead rate determined?
14. What is a control (cost) account and what kinds of information does it contain? How does a control account fit into the structure of PCAS?
15. How are control accounts aggregated horizontally and vertically? Why are they aggregated like this?
16. How are time-based forecasts prepared and how are they used?
17. What are the reasons for investigating the influence of schedules on project costs? What is the feasible budget region?
18. What might happen if top management submitted a bid for a project without consulting the business unit or department to be involved in the project?
19. Refer to Chapter 6, Case 6.1, the Barrage Construction Company. The project manager Sean Shawn employed the analogy with adjustment method to estimate the cost of constructing a three-car garage. Specifically, he started with the cost of an average two-car garage, \$43,000, and increased it by 50 percent to \$64,500. Comment on the likely accuracy of the three-car garage estimate. Suggest a different approach that might yield a more accurate cost estimate, and then use this approach and made-up time and cost figures to compute the estimate. Argue why your estimate is better than Sean's. See Chapter 6, Figure 6.18, for Sean's WBS.
20. The example in Table 9.1 shows three possible ways of apportioning total direct costs. Using the same example, suppose the direct non-labor (DNL) cost and G&A are broken down as follows:

	Direct Non-Labor			
	MARS	PLUTO		G&A
Materials	30	5	Freight	8
Other	<u>10</u>	<u>5</u>	Other	<u>32</u>
	40	10		40

Assuming all remaining costs shown in Table 9.1 are unchanged, compute the project costs for MARS and PLUTO using the following apportioning rules:

- a. Overhead (OH) is proportionate to direct labor (DL).
 - b. Freight G&A is proportionate to materials.
 - c. Other G&A is proportionate to DL, OH, DNL, and freight.
21. Chapter 8 discussed the impact of crashing activities and the relationship of schedules to cost. The CPM method assumes that as activity duration is decreased, the direct cost increases owing to the increases in direct labor rates from overtime. Overhead rates also may vary, although the overhead rate is often lower for overtime work. For example, the overhead rate may be 100 percent for regular time but only 20 percent for overtime. In both cases, the overhead rate is associated with the wage rate being used.

Suppose that in the MARS project in Table 9.1, 1,000 direct hours of labor are required at \$50 per hour, and the associated overhead rate is 100 percent for regular time. Now suppose the overhead rate is 10 percent and overtime wage rate is time-and-a-half. Compare the project cost if it were done entirely on regular time with the cost if it were done entirely on overtime. Which is less expensive?

22. Use the table below and the network in Figure 9.17 to answer questions about the ARGOT project:

Activity	Time (Wks)	Weekly Cost (\$K)	Total (\$K)
A	4	3	12
B	6	4	24
C	3	5	15
D	4	5	20
E	8	3	24
F	3	4	12
G	2	2	4
			111

- Compute the ESs and LSs for the project. Assume T_s is the same as the earliest project completion date.
- Construct a time-based network for the project such as Figure 9.11 (use early start times).
- Construct two diagrams similar to those in Figure 9.12 showing the weekly and cumulative project expenses.

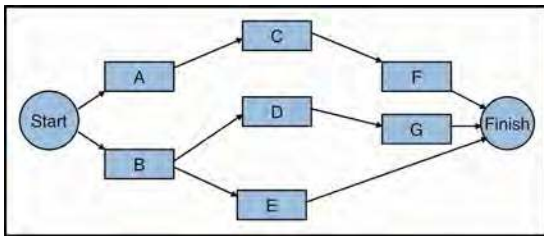


Figure 9.17
ARGOT project network.

- Using the data in problem 22, repeat steps b and c using late start times. Then identify the feasible budget region using the cumulative curves.
- Explain retention money and performance guarantee.



Questions About the Study Project

- What, if any, were the principle causes of cost escalation in the project?
- Was a life cycle cost analysis performed? If so, who did it, when, and using what methods? How did the analysis affect the design, development, and production of the project deliverables or main end-item?
- What kind of project cost accounting system (PCAS) was used? Was it manual or computerized? Describe the system and its inputs and outputs. Who maintained the system? How was it used during the project?

4. Describe the process of creating the project budget. Show a sample budget (or portion thereof).
5. How were management and supervisory costs handled in the budget?
6. Was the project budget broken down into control accounts? If so,
 - a. How were they related to the work packages and WBS?
 - b. How were they tied into the PCAS?
7. What kinds of cost summaries were prepared? Who were they sent to? How were they used? Show some examples.
8. Did the PCAS produce time-phased cost schedules and forecasts? Show some examples. How were they used by the project manager?
9. Were life cycle costs a consideration in the project? Explain.

CASE 9.1 LIFE CYCLE COSTS FOR FLEET OF TOURIST SPACESHIPS

At the time of writing, Burt Rutan and Sir Richard Branson had teamed up to form The Spaceship Company, which will develop and manufacture the commercial spacecraft (SpaceShipTwo, or SS2), launch aircraft (WhiteKnightTwo, or WK2), and support systems. Branson's "spaceline," Virgin Galactic, will handle the operations for space tourist flights. Their hope is to eventually reduce by half the proposed initial ticket price of \$190,000.

No information has been released about development and operating costs for the spaceline and equipment, so the figures used in this case are guesses. Refer to Example 9.3 for *hypothetical* life cycle costs for the spaceline and spaceship fleet, but assume the following changes to the numbers:

- Five spaceships, seven passengers per spaceship.
- Development and manufacturing costs, \$120 million.
- Flight operations cost: \$0.5 million/flight.
- Ticket price: \$190,000 for passengers on the first 100 flights, then \$150,000 for passengers on the next 100, and \$100,000 for passengers on flights thereafter.

QUESTIONS

1. Assuming all other numbers from Example 9.3 are the same, what is the "bottom line" profit of the venture for 5 years of operation?
2. If the profit goal is \$70 million,
 - a. What is the maximum development and production cost for the fleet?
 - b. What is the maximum per-flight operational cost; (note: assume \$120 million development/production cost)?
3. Brainstorm. What are some ways that the development cost might be reduced? What are some possible design decisions for the spacecraft and mothership that would reduce the per flight operational cost? Next, research articles and news releases about SS2 and WK2 to see what the developers, Scaled Composites and The Spaceship Company, have been doing to contain costs.

CASE 9.2 ESTIMATED COSTS FOR THE CHANNEL TUNNEL PROJECT¹³

Before construction began on the English Channel Tunnel project, the banks underwriting the project hired consulting engineers to review cost estimates prepared by the contractors. The consultants concluded that the tunneling estimates were 20 percent too high. Their analysis was based on comparisons of costs from recent European tunnel projects, including 50 German railroad tunnels ranging in length from 400 m to 11 km, to the Channel Tunnel, which would be 49 km in length. The costs of the tunnels ranged from £55 to £140 per cum (cubic meter) of open tunnel; the cost of the Channel Tunnel was estimated at £181 per cum on the British side of the channel and £203 on the French side (the difference owing to more difficult conditions on the French side). The Channel Tunnel is actually three interconnected tunnels—one for trains going in each direction and a smaller service tunnel in between them. Note, however, that the cost estimates are per cubic meter of tunnel, so, *presumably*, differences in tunnel lengths and diameters are not major factors.

QUESTION

Why might the estimates for the Channel Tunnel be so much higher per cum than the costs for the analogy projects? Discuss possible, logical adjustments to the analogy tunnel project costs to arrive at a cost estimate for the Channel Tunnel.

CASE 9.3 FIONA'S ESTIMATE FOR THE GORGY PROJECT

Fiona McDonald is preparing the cost estimate to accompany Highwire Systems' bid proposal for the Gorgy project. Her ballpark guess is that the project should take about 2 years and cost \$3 million, however, to help prepare the estimate she creates a WBS (Figure 9.18).

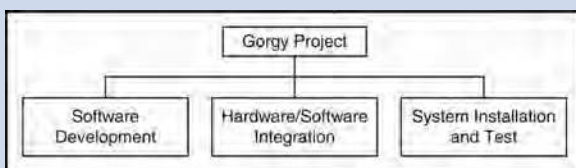


Figure 9.18
Gorgy project.

She estimates the costs for the three work packages as follows:

Development	\$2 million
Integration	\$1 million
Installation/Test	\$1.5 million
Project total	\$4.5 million

Although the total estimate is 50 percent more than her ballpark guess, she believes it is probably more accurate because it was developed “bottom up.”

She gives the estimate to Shireen Ghopthal, Highwire Systems' manager of contracts, who asks her, "Fiona, how did you arrive at the individual costs for the \$4.5 million?" Fiona explains, "The development cost, \$2 million, that was simple: I used the number provided in the RFP for what the development portion of the project *should* cost, which the customer derived from experience in working with developers in similar projects. Besides, the number seemed ample to me, and since it was the only cost figure provided in the RFP, I considered it as sort of a mandate for the maximum development cost. As far as the integration cost, well, I looked at the customer's hardware and software we'd be working with as listed in the RFP, and I compared it with the other projects we'd done with similar equipment and systems and what those projects cost. Finally, for installation and test, I reviewed six projects I'd completed in the last few years and their costs. Costs for installation and test ranged from \$0.6 to \$2 million, with \$1.3 million average. So, \$1.5 seemed reasonable."

Shireen replies, "Well, I ordinarily don't question your work. But are you sure you've covered everything in the project in the work breakdown? Do the three work packages include *everything*? And don't we usually do a site visit to inventory the customer's equipment and systems that we'll have to work with? Do you trust the RFP? Do they really know what they have? And looking at the project, it'll take maybe 2 to 3 years. It'll be a big project. Are you sure you and your staff will be able to manage and coordinate everything for that cost?"

Fiona responds, "Everything is covered. As far as the site visit is concerned, the proposal is due next week and we don't have time. We'll have to go with what they say in the RFP. As for installation, based on my experience the average installation/test cost was \$1.3 million. I picked \$1.5 million to be safe and cover any overages in development cost."

Then Shireen repeats, "And what about the coordination and integration effort?" to which Fiona replies, "Yes, that will probably be huge, but I'm pretty certain that if we get the contract, Highwire will let me hire maybe ten additional experienced analysts and engineers for my management staff. As you know we've run over on our last several projects, and I've been arguing all along I just need more people to help coordinate and keep things under control."

QUESTION

Shireen suspects that Fiona's cost estimation approach is too simplistic and leaves ample room for error. List at least four inadequacies in her approach and places where the estimates can go wrong.

CASE 9.4 MELBOURNE CONSTRUCTION COMPANY, D

Bill Asher, the estimator for Melbourne Construction Company, is estimating the days required to install the wall footings in the foundation of a hotel building. As is common for many of the activities in construction projects, he will develop the estimate using labor productivity standards.

Architectural drawings for the hotel indicate that the square foot contact area (SFCA) for the formwork footings is 13,340 sq. ft. (1,239 m²). Installation of footings is

considered “standard,” so Bill refers to a manual of labor hour standards to estimate the total labor hours required for the task. The manual indicates that the standard for the footings specified for the hotel is 0.066 labor hours per SFCA.¹⁴

QUESTIONS

1. Given the SCFA standard and the estimated SCFA for the footings, what is the estimated labor hours to install the footings?
2. The company intends to assign ten workers to install the footings. Assuming an 8-hour workday, what is Bill’s estimate for the number of days needed to complete this task? (Note: An assumption here is that for each additional worker assigned to a task, the task duration decreases proportionately. This is an important assumption since in many projects the task durations are not proportionate to the number of workers, and in fact adding workers does not always shorten task times and sometimes even increases them.)

Notes

1. Flyvbjerg B., Bruzelius N. and Rothengatter W. *Megaprojects and Risk: An Anatomy of Ambition*. Cambridge: Cambridge University Press; 2003, p. 16.
2. See Archibald R.D. *Managing High-Technology Programs and Projects*. New York, NY: John Wiley & Sons; 1976, pp. 167–168.
3. Harrison F. *Advanced Project Management*, Hants, England: Gower; 1981, pp. 172–173, gives an example of an escalation clause.
4. Politically, how independent should the estimators be? So independent, says DeMarco, that the project manager has “no communication with the estimator about how happy or unhappy anyone is about the estimate.” DeMarco T. *Controlling Software Projects*. New York, NY: Yourdon Press; 1982, p. 19.
5. Flyvbjerg, Bruzelius and Rothengatter, *Megaprojects and Risk: An Anatomy of Ambition*.
6. See Flyvbjerg, Bruzelius and Rothengatter, *Megaprojects and Risk: An Anatomy of Ambition*, Chapter 14, for discussion of labor costing in projects.
7. This example is derived from a similar one in Rosenau M. *Successful Project Management*. Belmont, CA: Lifetime Learning; 1981, pp. 89–91.
8. This example is derived from Wilson T. and Stone D. Project management for an architectural firm. *Management Accounting* October; 1980: 25–46.
9. The kinds of cost summaries used often depend on the kind available in the software, though many software packages permit customizing of reports.
10. Wiest J. and Levy F. *A Management Guide to PERT/CPM*, 2nd edn. Upper Saddle River, NJ: Prentice Hall; 1977, pp. 90–94.
11. Harrison, *Advanced Project Management*, p. 185, notes that balancing cash in foreign contracts is difficult because “In many cases, the profits from [currency dealings] can exceed the profits from the project; [if funds are] not managed effectively, the losses from foreign currency commitments can bring about large losses on a project and lead to bankruptcy.”
12. Smith P. and Reinertsen D. *Developing Products in Half the Time*. New York: Van Nostrand Reinhold; 1991, pp. 224–225.
13. Fetherston D. *The Chunnel*. New York, NY: Times Books; 1997, pp. 141–142.
14. *RS Means Labor Rates for the Construction Industry*, 41st edn. Norwell, MA: RS Means; 2013.

Chapter 10

Project Quality Management

Besides meeting the budget and schedule, project success depends on meeting performance requirements. Performance requirements are generally based on project stakeholders' needs and expectations about the functioning and performance of the product—the project end-item. A "high-quality" project is one that meets performance requirements, and, thus, satisfies the needs and expectations of key stakeholders and causes no harm elsewhere.

10.1 The Concept of Quality

In the 1950s quality was viewed as the process of inspecting products that had already been produced, and separating the good ones from the bad. But in the current business environment, so the thinking goes, you have to *prevent* defects and failures rather than inspect for them i.e., you cannot right a wrong by inspection. You have to build processes to ensure things are done right the first time, every time, and a culture where everybody is quality focused.

But in the competitive spirit, project teams often seek ways to accelerate schedules and cut costs, even though this sometimes results in rework, mistakes, greater workload for the project team, and a "quality meltdown." They become preoccupied with costs and schedules, even though "the bitterness of poor quality lives long after the sweetness of cheap price and timely delivery has been forgotten."¹

An example is the space shuttle Challenger. On January 28, 1986, defective seals allowed flames to breach a joint in a rocket motor and ignite the main fuel tank shortly after launch, causing an explosion and the deaths of the seven astronauts on board. While engineers had previously warned about the risk of this happening, the launch proceeded according to schedule because of a promise to politicians; for the sake of meeting a schedule, quality was compromised.

The London Tower Bridge, Figure 10.1, offers a contrast.² It opened in 1894, four years late and costing nearly twice the estimated £585,000, but well over a century later, it has withstood the test of time. Originally designed and built to enable pedestrians and horse-drawn vehicles to cross the Thames River, it now carries 10,000 vehicles per day and is a major tourist attraction. It has survived floods, pollution, and war—problems its original designers never



Figure 10.1

London Tower Bridge.

Source: MEDITERRANEAN/iStock/Getty Images ®

considered. In terms of time and cost the project was a failure; in terms of quality, it has been a raving success.

What is quality?

Quality is meeting specifications or requirements—but it means more than that. While meeting project specifications will ordinarily prevent a customer from taking a contractor to court, it alone cannot ensure the customer will be pleased, or even satisfied, with the end result and that the contractor will receive gratitude or win repeat business.

Ideally a project looks *beyond* specifications and to fulfilling customer expectations—including those not articulated; it aims at delighting the customer.

Fitness for purpose

The term **quality** implies that a product or deliverable is **fit** for the **intended purpose** and fulfils a wide range of criteria such as functional performance, safety, reliability, ease of handling, maintainability, logistical support, and no harmful environmental impacts. Beyond fitness, however, quality also accounts for a product's **value for money** and whether it is priced right for the intended purpose. Optimizing only one aspect of a product—fitness for purpose, value for money, or strategic benefit to the organization—will not necessarily result in an optimal product. Project managers sometimes wrongly assume that customer needs, expectations, and requirements are readily evident or will require little effort to research and specify. The project

manager must verify that the product will meet the customer's requirements and seek to balance the multiple aspects of a product and define its specifications to reflect that balance.

Absence of defects

Quality also implies an absence of defects, which is why people often associate the terms *quality* and **defect**. A defect is a **nonconformity**—a problem or mistake—something other than what the customer had expected. One way to achieve quality is to identify and correct as many nonconformities as possible—and to identify them as soon as possible. In general, the longer a nonconformity persists before it is discovered, the more costly it is to remedy it. It might be relatively easy and inexpensive to fix a defect in a component part, but it is usually more expensive to fix it after the component has been put into an assembly, and even more expensive after the assembly has been imbedded inside a system. A defect is most expensive when it causes a product or system to malfunction or fail while in use by the customer.

But “absence of defects” requires qualification, and the presumption that zero defects equates to high quality is not always true. A quality project is one that satisfies multiple requirements, and devoting too much attention to any particular one, such as eliminating *all* defects, may detract from fulfilling other more important requirements.³ For example, in most projects the requirements relate to time, cost, and performance. When the schedule must be maintained, trying to remove *all* defects can prove exceptionally costly. The customer might prefer to keep to the budget and schedule rather than eliminate all defects. Of course, in some cases it is necessary to eliminate every defect.⁴ Even the most minor defect in a critical component of an air traffic control system or artificial human heart can result in injury or loss of life. The point is, it depends on the customer and the deliverable. Often the customer prefers a deliverable to be completed on time, at lower cost, and with a *few* minor defects than completed late, at higher cost, but with no defects.

Good enough quality

In removing defects, emphasis is on those that would prevent the system from meeting its most important requirements. This is the concept of **good enough quality**—the default criteria when priorities on performance requirements, time, and cost preclude meeting all the requirements and force the project team into meeting only the most important ones. Says Bach, creating systems “of the best possible quality is a very, very expensive proposition, [though] clients may not even notice the difference between the best possible quality and pretty good quality.”⁵ The customer, of course, must be able to judge what is “good enough” and prevent unnecessary “gold plating.” To do that, the customer must be constantly updated about project progress, problems, costs, and schedules.

In the ideal case, everyone on the project team contributes to quality; each:

1. Knows what is expected of her.
2. Is able and willing to meet those expectations.
3. Knows the extent to which she meets the expectations.
4. Has the ability and authority to take necessary corrective actions.

Such conditions require quality-focused leadership, training, and motivation efforts. Once everyone starts contributing, however, attention to quality should become automatic and require little influence from the project manager.

What quality is *not*

Quality implies that the product is fit for the purpose. But fit for purpose does not necessarily relate to the product's expense, reliability, or features, all of which refer to the product's **grade**. In other words, *quality* and *grade* are not the same. For example, coal mines produce different grades of coal. The highest grade is used in steelmaking while lower grades are used in chemical products and power plants. Even though coal for a power plant is lower grade than that for steelmaking, it is the most appropriate—hence best-quality—coal for the purpose, and it would be inappropriate and uneconomical to use higher-grade coal in power plants. Of course, companies that mine the coal should strive for *high-quality processes* to deliver all grades of coal to meet the specifications of all their customers, including price and delivery specifications.

Quality movements and progress

The “quality revolution” started in the 1950s in Japan, in part under the influence of Dr. W. Edwards Deming, an American management consultant. He proposed a quality philosophy that included **continuous improvement**, skills training, leadership at all levels, elimination of dependency on inspections, reliance on single-source rather than many-source suppliers, and use of statistical techniques. Since then, a number of other quality movements have come and gone. The most lasting and popular movement since the 1980s is **total quality management (TQM)**, whose practices mostly originated in Japan and from Deming's philosophy. TQM is a set of techniques and a mindset for their use, an ambitious approach to improving the total effectiveness and competitiveness of an organization. The key elements of TQM are identifying the mission, goals, and objectives of the organization, acting in ways consistent with those goals and objectives, and focusing on customer satisfaction. TQM involves the total organization, including teams of frontline workers and visible support from top management. Quality problems are systematically identified and resolved to continuously improve processes. In projects, this purpose is served by project reviews and closeout sessions, discussed later.

Complementing TQM is the management philosophy of **lean production (LP)**, which also originated in Japan at Toyota. LP gives recognition to the fact that quality problems typically originate from “broken processes,” and it provides methods and tools to analyze processes and expose and eliminate sources of waste in processes. It includes relatively easy-to-implement methods to improve quality, and reduce costs and lead times.⁶ The most difficult aspect of implementing LP is developing a culture wherein employees everywhere have the authority and skills to continuously improve their processes—an unusual concept for many organizations. Principles of LP have been successfully adopted around the world. In some industries (e.g., autos and electronics) virtually all the big players have adopted LP practices. In project environments, LP methods are being applied to product development and construction, some of which are described in Chapter 14.



See Chapter 14

Another quality movement called **Six Sigma** originated in the 1980s at Motorola and was later popularized by General Electric. The term “Six Sigma” refers to the fact that in a normal distribution, 99.99966 percent of the population falls within -6σ to $+6\sigma$ of the mean, where “ σ ” is the standard deviation. If the quality of a process is controlled to the 6σ standard, there would be less than 3.4 parts per million defects in the process—near perfection!

But the Six Sigma movement goes beyond statistics and is a philosophy for reducing process variability. It includes two five-step processes, one for improving existing processes and the other for designing new processes and products, both aimed at 6σ quality levels. The first process, called **DMAIC** for Define, Measure, Analyze, Improve, and Control, involves the steps

of defining the best actions to improve a process, implementing those actions, tracking the results, and reducing defects so that fewer outcomes fail to meet specifications. The second process is **DMADV**—Define, Measure, Analyze, Design, and Verify. The Six Sigma approach in projects translates into defining clear deliverables that relate to the mission of the organization and are approved by management. In some companies the DMAIC process is the project methodology and defines the stages of the project.

Project quality management

Project quality means **quality of the project end-item**, deliverable, or product, as well as **quality of the processes** employed during the project. Quality of the end-item or product starts with clearly defined system requirements or specifications that are agreed upon by both contractor and customer. If the contractor feels the customer has provided requirements that are unrealistic, he should review them with the customer and alter them so the desired end result can be achieved realistically. The agreed-upon specifications should reflect the customer's expectations for the product's fitness for its intended purpose and any negotiated compromises. Comprehensive specifications for the deliverable should be included in the project scope statement, hence the scope statement is an important input for project quality planning.

Project quality management refers to the management processes and techniques used to reduce the risk that products or deliverables will not meet requirements. The following sections discuss these processes and related techniques and concepts.

Costs of quality

Since quality is related to value for the money spent, quality planning should consider the costs and benefits of quality activities. Money spent on quality assurance and control should be justified in terms of the expected savings or benefits from fewer or no nonconformities. A **cost–benefit analysis** is performed to evaluate the proposed quality activities.

The **costs of quality** are classified as costs to *avoid failure*, called costs of conformance, and costs that result from *failure*—costs of nonconformance:

1. **Cost of conformance**
 - a. Prevention: costs of training, design reviews, and activities aimed at preventing errors; includes cost of quality planning.
 - b. Appraisal and control: costs of evaluating products and processes, including product reviews, audits, tests, and inspections.
2. **Cost of nonconformance**
 - a. Internal failure: costs associated with nonconformities discovered by the producer; includes costs for scrap, rework, and retest.
 - b. External failure: costs incurred because of product failures after delivery to the customer; includes costs for replacements, warranty repairs, liability, lost sales, and damaged reputation.

The saying “prevention is better than cure” is appropriate here because the cost of failures can be much greater than conformance costs to prevent them. While the overall costs of quality can be as little as 2 percent of earnings for a company with a good quality management system, it can exceed 20 percent for a company with a poor quality management system.⁷ It therefore

makes sense to invest in a good system, i.e., to spend more on design reviews, audits, training, modeling, and testing so as to spend less on internal and external failures.

For projects, the costs of prevention, appraisal, control, and internal failure are incurred during the project; costs associated with external failure come *after* the project is completed. The costs of conformance are among the many aspects the project manager must justify to management and the customer and include in the project plan and budget.

10.2 Project Quality Management Processes

Project quality management has three processes: quality planning, quality assurance, and quality control (Figure 10.2). **Quality planning** guides future quality activities: it sets the requirements and standards to be met and the actions necessary to meet them. **Quality assurance** performs the planned quality activities and ensures the project utilizes processes necessary to meet quality standards and end-item requirements. **Quality control** ensures that quality assurance activities are performed according to quality plans and that requirements and standards are being met. Think of quality control as the “medicine” to eliminate existing

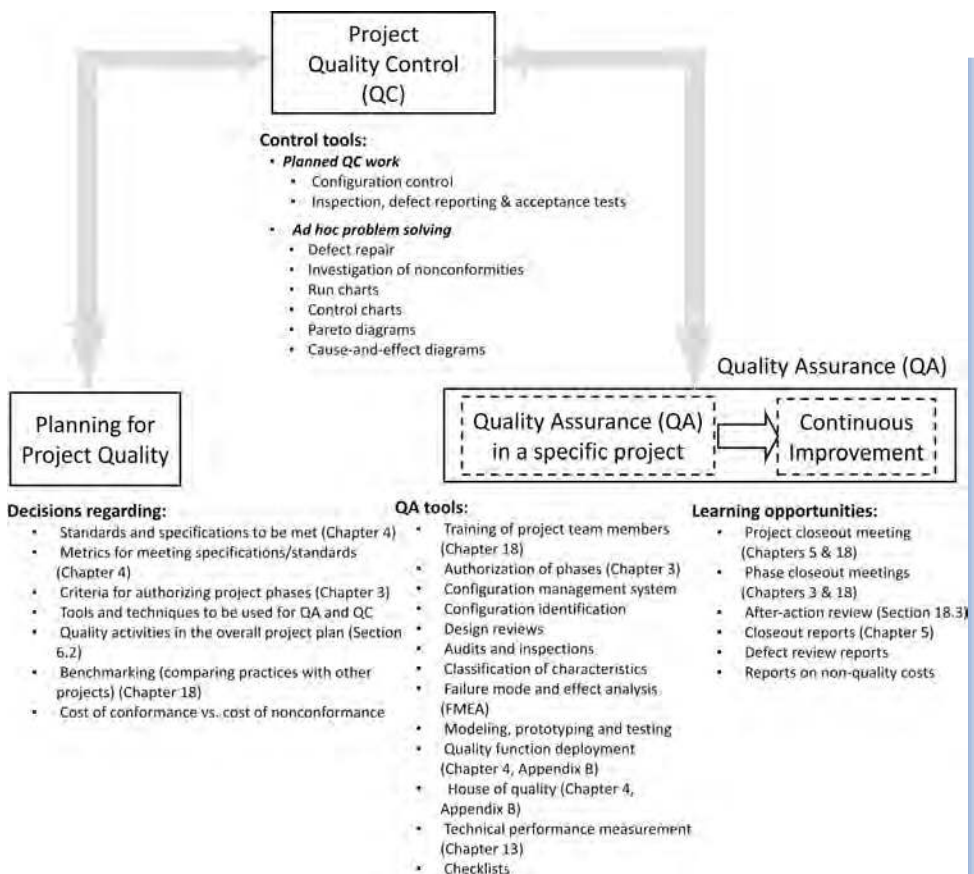


Figure 10.2
The project quality management process.

nonconformities, and quality planning and assurance as the “healthy lifestyle” to prevent nonconformities in the first place.

As shown in Figure 10.2, project quality control provides the links to ensure that quality assurance happens in accordance with the quality plan. Quality assurance aims to ensure appropriate quality standards for a project, and to take advantage of learning opportunities from completed projects to improve future projects. Also shown in Figure 10.2 are tools and techniques for quality planning, quality assurance, and quality control, as well as references to other chapters that provide more information about them.

Quality planning

Quality planning should provide confidence that everything necessary to ensure quality has been taken care of. It has two aspects: (1) establishing quality management procedures and policies for the entire organization and (2) establishing a quality management plan as a part of the execution plan discussed in Chapter 6. The purpose of quality planning is to identify the *project quality requirements* (as implied by specific standards and specifications) and to predetermine *how quality will be managed* so the project and its product will comply with these requirements.

Responsibility for establishing organization-wide policies and procedures to improve project quality typically falls on functional managers, especially the quality manager. Projects often employ quality standards already existing such as the **ISO 9001** standard (a quality management system).⁸ For design and development projects, ISO 9001 prescribes that an organization should set procedures for (a) the design and development stages; (b) the necessary reviews, verifications, and validations appropriate to each of the stages; and (c) the responsibilities and authorities for the stages.



See Chapter 6

Project quality management plan

The project **quality management plan** (or quality plan) is a key component of the project execution plan and indicates how relevant policies and procedures will be implemented in the project.

A central tenet of what is called **quality by design** is that quality can be planned and that many problems can be prevented by the way it is planned. Creating a quality plan therefore is important to the successful execution of projects.

Identifying, scheduling, budgeting, and assigning responsibility for quality assurance and control activities is done following the same principles and methods as for other project activities, discussed in later chapters. The plan addresses the quality management approach of the project, the stakeholders involved, and how the project would respond to any changes in customer needs. It typically references relevant organizational policies and procedures (e.g., configuration management system and classification of characteristics procedures—both discussed later) and how they would be implemented in the project.

If not already covered in the project management methodology, the plan should indicate how each of the project phases will be initiated and authorized, and how phases and the entire project will be closed out.

The plan should address all the elements shown in Figure 10.2, including how the project team will ensure that the quality requirements as stated in the specifications and standards will be met. This can include the following:

- Any models to be produced and tested, and associated test specifications, procedures, and reports.

- Inspections, equipment required for inspections, calibration of equipment, and required reports.
- Final acceptance tests, including when they should take place and test specifications, procedures and reports.
- Any required design reviews, and for each its purpose, people involved, and outputs required.
- Audits.
- Checklists.
- Techniques that will be used, e.g., FMEA or control charts.



See Chapter 6

The plan can also indicate how non-conformances, customer complaints, and corrective actions would be handled. It should specify the person primarily responsible for each task and the roles and responsibilities of others involved. The responsibility matrix discussed in Chapter 6 can be used for this purpose.

Quality assurance

Project quality assurance relates to the execution of the project quality management plan and aims to reduce the risks of not meeting desired features or performance requirements of deliverables.

As shown in Figure 10.2, quality assurance covers the following:

1. Activities done in a *specific project* to ensure that requirements are being met and the project is being executed according to the quality plan.
2. Activities that contribute to the *continuous improvement of current and future projects* and to the project management maturity of the organization.

Quality assurance should provide confidence that everything necessary is being done to ensure the appropriate quality of project deliverables.

Continuous improvement and project post-completion reviews

Continuous improvement is the foundation to progress: without it, humankind would not have moved beyond the Stone Age. Project organizations strive to continually improve their technical operations and managerial processes, in part, by conducting a formal **post-completion review** for every project. The review happens upon completion of the project or, better, upon completion of each phase of the project. Its purpose is to understand what happened and to learn lessons that can be applied to other projects and avoid repeating mistakes.

The project manager's responsibility during reviews is to facilitate candid and constructive discussion about what happened—what worked and what did not, and to make sure that every participant is heard. The discussion is formally documented, and a **lessons learned list** created. This process, though essential for continuous improvement, is often neglected because people lose interest as the project winds down or as they become busy on new, upcoming projects. As a result, organizations repeat mistakes, reinvent the wheel, and do not learn from their experiences.⁹ Post-completion reviews are covered more in Chapter 5; they play an important role in **knowledge management**, discussed in Chapter 18.



See Chapters 5
and 18

Quality control

Quality control is the ongoing process of monitoring and assessing work and taking corrective action to achieve the planned quality outcomes, i.e., to meet functional and nonfunctional requirements as prescribed in standards and specifications. It also verifies that quality assurance activities are being performed as specified in the quality plan. Quality control is an aspect of **project control**—the topic of Chapter 13 but covered here for continuity.



See Chapter 13

Quality control can be contrasted to **scope validation**: whereas scope validation refers to the acceptability of project deliverables by the customer, quality control refers to conformance to specifications as set by the contractor. Scope validation refers to confirming the *acceptability* of specifications and standards; quality control refers to verifying adherence to previously set specifications and standards.

The quality control process includes inspections to verify that deliverables are meeting specifications, plus tests to confirm compliance with acceptance criteria before handing the product to the customer. In the event that a minor feature does not meet specification, the contractor might request a waiver or deviation. A **waiver** applies to an unplanned condition discovered only after the item is produced. It authorizes a temporary nonconformity, such as a scratch discovered on the paint of a hardware item. A **deviation** is also a temporary departure from specification, but it is discovered *before* production. For example, if a specified material is temporarily unavailable, the contractor can apply for a deviation to allow use of an alternative material. A third form of variation from specification is a **modification**, which is a permanent change to the specification. Control activities as illustrated in Figure 10.2 include both planned quality control activities and ad hoc problem solving. Planned activities include, for example, inspections on a construction site, tests on a product component, or audits to ensure a supplier is using correct materials. Ad hoc problem solving refers to handling problems and risks as they emerge. Techniques for analysis and problem solving are discussed later.

Quality control cannot happen in isolation; it must be integrated with scope control, cost control, progress control, and risk control. Thus, in the same way that the quality plan should be integrated with other aspects of the project plan, quality control should be integrated with the other aspects of project control.

Quality of procured items

Quality requirements for *off-the-shelf* items **procured** from suppliers are set by industry standards, in which case the main criterion for choosing a supplier is price. To buy a batch of standard items such as bolts, the procurement officer obtains price quotes from multiple suppliers and picks the lowest. When the batch arrives, an inspector checks the bolts to determine if they are acceptable. But to procure a system or item that must be *newly developed*, there likely is no industry standard. In that case the purchaser must collaborate with the supplier and assist in planning for the quality assurance and control to assure the item meets specifications.

Of course, even for procurement of standard items, far better than selecting the supplier with the lowest price is selecting one that has proven capability and willingness to meet the contractor's requirements, and then seeking to develop a mutually beneficial long-term relationship with the supplier. The two parties work together as partners and share responsibility for each other's success. Establishing this kind of relationship is not always easy, especially when the supplier is much larger than the contractor or does not value the relationship or consider the contracted work a priority.

Contractors often invest heavily to make sure they can procure subsystems and components of the appropriate quality. A contractor often has a special vendor quality section within its procurement division to manage quality assurance of all its procured items—including their development and manufacture or construction. The purpose of the quality section is to assist in selecting suppliers, monitor suppliers' processes to ensure quality, and perform inspections and acceptance tests of purchased items. Other responsibilities are described next.

Example 10.1: Companies Working Together for Quality Assurance and Control

Company A develops and manufactures mining vehicles. It is working on a new vehicle and must choose a supplier to develop, manufacture, and support a transmission for the vehicle. The company's vendor quality section and procurement staff review proposals from supplier candidates and select Company B to provide the transmissions for the vehicles. Company A's engineering division develops a functional specification for the transmission that includes performance characteristics, maintenance requirements, interfaces with other parts of the vehicle, and test requirements. Its vendor quality section then works with Company B's engineers to ensure they will be using appropriate processes for cost-effective compliance with the specification, and that they will test all transmissions according to Company A's functional specification for compliance to performance before shipment.

10.3 Techniques for Quality Assurance in System Development

This section further explains the items in Figure 10.2. In phased project planning, authorization of a phase implies that plans for the phase have satisfied pre-specified criteria, including that the plans contain sufficient measures for quality assurance. System developers employ a variety of such measures, as discussed next.

Configuration management⁹

During design and development of a system, vast amounts of information are generated for use in the design process and later in manufacturing (or construction), maintenance, and support. The information can include hundreds or thousands of documents (specifications, schematics, drawings, etc.), each likely to be modified in some way during the project. Keeping track of all the changes and knowing which version is the most current for every item can be difficult. Thus, any project aimed at delivering a physical product needs a system or process to keep up with and manage all the information; such is the purpose of **configuration management**.

Configuration management includes the policies and procedures for monitoring and tracking design information and changes and ensuring that everyone involved with the project (and, later, the end-item's operation) has the most current information. Those policies and procedures should be included in the **configuration management plan**, which is part of the quality plan. As with all procedures, the best configuration management system is whatever permits the desired level of control and is the simplest to implement. The two main aspects of configuration management are configuration identification and configuration control.

Configuration identification

Configuration identification is an inherent part of systems design and involves defining a system's overall structure and its subsystems and components. Mentioned in Chapter 4, any subsystem, component, or part that is to be tracked and controlled as an individual entity throughout a system's life cycle is identified as a **configuration item (CI)**. A CI can be a piece of hardware, a manual, a parts list, a software package, or even a service. Any part of the system that is procured is also treated as a CI. Every physical and functional characteristic that defines and is important for controlling the CI is identified and documented. Ultimately, every functional and physical element of the end-item system should be associated in some way with a CI, either as a CI in its own right or as a component within a subsystem that has been identified as a CI. Ideally, each CI is small enough to be designed, built, and tested by a small team.



See Chapter 4

The master copies (electronic or paper) of the configuration documents for every CI are retained in a single, secure location (the “configuration center”) and managed by someone not involved in the functions of design, construction, manufacture, or maintenance. (Documentation about design premises, assumptions, and calculations are not considered configuration documents and are retained elsewhere by the design authority.)

Any modifications, waivers, or deviations to a CI are recorded so that all CI documents reflect the **as-built** status of the system. In the case of a deliverable such as a building, ship, or other one-of-a-kind system that becomes operational, the as-built specification will later be used during its operation and maintenance. Where multiple units are produced (e.g., cars, airplanes, appliances) and where modifications and improvements are introduced over time, the specific configuration for each individually produced unit must be known, which requires that each specific CI in the product must be traceable to its specific as-built specifications. This is necessary so that, for example, the correct spare parts, training, and operating manuals can be supplied, and problems can be traced and analyzed in the event of accidents, customer complaints, or claims regarding product liability. This concept of **traceability** was introduced in Chapter 4; in CI, traceability refers to maintaining a comprehensive historical paper trail of an item through its entire life cycle. This is illustrated in the following example.



See Chapter 4

Example 10.2: Traceability and the Apollo Spacecraft¹⁰

To establish the reliability of an item, either many units of the item are tested until one fails or the required reliability is designed-in through methods of engineering analysis. Regardless, to assure reliability everything about the item must be known—its manufacturing process, the composition of its parts and materials, and even the sources of those materials. For the Apollo space mission, the goal of achieving mission success was set at 99 percent and crew survivability at 99.9 percent. To meet such high-level goals, every CI (subsystem, component, part, etc.) as it moved through the design and manufacturing process was accompanied with a package of documents that established its genealogy and pedigree. The saying went, “If you ordered a piece of plywood, you wanted to know from which tree it came.” Half-inch bolts for the Apollo spacecraft involved an 11-step manufacturing process with certification tests at each step. Every bolt was subjected to rigorous testing, as were the steel rods from which they were made, the billets from which the rods were rolled, and the ingots from which the billets were forged. Everything about the processes and tests for the bolts was documented, including

the source of the iron for the bolts—Minnesota—and even the *mine* and the *mine shaft*. Such extreme tracking and control is necessary to ensure high reliability and enable problem diagnosis in case things go wrong. But it comes with a price though, which is why bolts available for 59 cents at the hardware store cost \$8 or \$9 apiece on rockets and spacecraft.

Configuration control

The topic of **configuration control**, the second aspect of configuration management, relates more to quality control than quality assurance, but we cover it here for the sake of continuity. The design of a system is normally specified by means of several documents such as performance specifications, drawings, manuals, and testing procedures that are generated during the design process. As the design evolves these documents are subject to change, so a scheme is needed to manage and keep track of the changes. Such is the purpose of **configuration control**.

Any organization or individual may request a modification, waiver, or deviation, in which case configuration control requires the following:

1. The proposed change and its motivation are documented. A standard document exists for this purpose; it is called a **change proposal**, **change request**, **change order**, **engineering change proposal** (ECP), or **variation order**.
2. The impact of the proposed change on system performance, safety, and the environment is evaluated, as is its impact on other hardware items, software, manuals, and methods of manufacturing or construction and maintenance.
3. The change is assessed for feasibility, which includes estimating the resources needed to implement the change and the impact of the change on schedules.
4. The change is approved or rejected by a **configuration board** (CB) or **configuration control board** (CCB). The board usually includes the chief designer and representatives from manufacturing or construction, maintenance, and other important stakeholders, and often is chaired by the project manager or program manager.
5. Upon approval of the proposed change, the work to implement the change is planned. The plan includes actions with regard to the disposition of anything that might be affected by the change, including spare parts, equipment and processes for manufacturing or construction, manuals and other documentation.
6. The implemented change is monitored and controlled to ensure it complies with the approved change proposal.



See Chapter 13 **change control**, both discussed in Chapter 13.

Design reviews

The project manager must ensure that the proposed design is acceptable in all respects, which is the purpose of **design reviews**—to ensure that the users' requirements and assumptions have been correctly identified and that the proposed design is able to meet the requirements in an appropriate way. Design reviews (not to be confused with general **project reviews**, described in Chapter 13) provide confirmation of design assumptions (e.g., load conditions)



See Chapter 13

and other information used in the design process and for design calculations. Ideally, they ensure that all important life-cycle aspects of the end-item have been addressed and pose no unacceptable risks. In particular, reviews check the designs for:

- Omissions or errors
- Compliance to regulations, codes, specifications, and standards
- Cost of ownership
- Safety and product liability
- Reliability
- Availability
- Ability to be constructed or manufactured (manufacturability)
- Shelf life
- Operability
- Maintainability
- Intellectual property rights
- Ergonomics.

The reviews involve representatives from all disciplines, functions, and users who will be connected to the deliverable throughout its life cycle, and, often, outside designers and subject matter experts (the concept of **concurrent engineering**, discussed in Chapters 4 and 15.) For example, in addition to the designers, the design review for a chemical plant, mine, or factory would include representatives from:



See Chapters 4
and 15

- The organization that will operate the facility.
- The technical support area that will be maintaining the facility.
- The construction company and manufacturers of plant equipment.
- The marketing, procurement, legal services, and quality areas that will occupy, make use of, or have to deal with the consequences of the facility.

Design reviews conducted early in the conceptual phase involve representatives from only a few functions, but as the project progresses, representatives from more functions are involved. For the design of a simple part or component, a single review upon completion of design but before manufacture might be sufficient; for the design of a complex system, several reviews would be necessary at successive stages of the project.

Formal reviews

Formal design reviews are planned events, usually chaired by the project manager or someone else not directly involved in the design of the end-item. Projects aimed at developing and delivering a product commonly have four reviews:

1. **Preliminary design review:** review of the functional design to determine if the concept meets the basic operational requirements.
2. **Critical design review:** review of the details of the hardware and software designs to ensure they conform to the preliminary design specifications.
3. **Functional readiness review:** (for mass-produced products only) evaluation of tests performed on early-produced items to check the efficacy of the manufacturing process.
4. **Product readiness review:** comparison of manufactured products to specifications to ensure that design control documents will result in products that meet requirements.

Formal reviews serve other purposes too: minimize risk, identify uncertainties, assure technical integrity, and assess alternative engineering approaches. Unlike peer reviews, formal reviews are overseen and conducted by a group of outsiders who use information accumulated by the project team. These outsiders are technical experts who are familiar with the end-item and workings of the project and project organization but are not formally associated with the project organization or its contractors. Since a formal review may last for several days and require considerable preparation and scrutiny of results, the tasks and time necessary to prepare and conduct the review and obtain approvals should be incorporated in the project schedule. The schedule should also allow time for possible corrective actions resulting from the review.

Since one prerequisite for each design review is thorough design documentation, common practice is to convene a **pre-review meeting** during which the design team provides the review team with an overview of the proposed design; documentation explaining the design premises, philosophy, assumptions, and calculations; and specifications and drawings. The review team is then allowed time (typically 14 days) to evaluate the design and prepare for the formal review meeting. The review team sometimes uses a checklist to ensure that everything important is covered. The Internet has become an effective medium for conducting design reviews through groupware that allows review team participants to conduct reviews online.¹¹

Informal design reviews

Although formal reviews are essential, the project manager should encourage informal design reviews, which are informal discussions among designers and between designers, manufacturers and other stakeholders. Good suggestions can originate anywhere, but it is up to the designer to decide whether to use them. Draft designs, reports, and other deliverables should be shared regularly (and, ideally, voluntarily) among peer designers and others for informal review. This includes, for example, an engineer simply asking a colleague to check his calculations or to comment on a sketch of a proposed design.

In a healthy quality culture, teams use **brainstorming** to evaluate and edit not just designs, but other aspects of project delivery. The principle behind brainstorming is to freely generate as many ideas as possible, and to withhold any form of evaluation or criticism until after numerous ideas have been generated. Only later are the ideas assessed and the good ones sep-



See Chapter 11 arated from poor ones. This is discussed more in Chapter 11.

Example 10.3: External and Internal Reviews in the Mars Pathfinder Project¹²

All major NASA projects require formal reviews by outside review “boards.” These reviews are crucial since a project’s termination or continuation can depend on the board’s findings. For the Mars Pathfinder project [see Example 13.3, Chapter 13] the review board comprised 25 consultants and seasoned managers from NASA and Jet Propulsion Laboratory (JPL, the site responsible for most of the Pathfinder design work), none of whom was associated with the project.

Preparation for a formal review can take an enormous amount of time, and managers for the Pathfinder project estimated that preparation for one review, the critical design review, would require about 6 weeks of dedicated attention. This would divert time from the

actual management of the project, which, paradoxically, could increase the likelihood of the project falling behind schedule and failing the review. To prepare for the formal review, project manager Brian Muirhead ordered an internal review.

In contrast to formal reviews, internal peer reviews address a narrow range of topics and require only a few days' preparation. The value of these reviews lies in making sure that everyone understands the decisions being made, nothing is overlooked, and the project is kept on track. Over 100 internal reviews were conducted during Pathfinder's 3-year development phase.

The internal review in preparation for the critical review revealed many problems—including lack of progress in defining system interfaces, rapid growth in the weight of the Mars lander, and a shortage of good engineers—and did little to inspire confidence about the project's ability to pass scrutiny in the critical design review.

The verdict from a critical design review is an all-or-nothing decision: the project either passes or fails. Failure initiates a cancellation review that can result in project termination. A project such as Pathfinder could be canceled if it overran the budget by as little as 15 percent. Beyond determining the future of a project, formal design reviews serve another purpose: to give the project a kick in the pants. Preparation for each review forces the project team to make decisions about unresolved issues. Formal reviews may be held three or four times during the project.

The critical design review board for Pathfinder was not happy with many aspects of the project but they did not recommend project cancellation. They approved the project but instructed Pathfinder's managers to be more critical of designs, focus less on performance and more on cost, and stop obsessing over business innovations. These recommendations later proved useful and helped make Pathfinder one of the most successful projects in the history of space exploration.

There is always more than one means to an end, and no designer, regardless of competency, can be expected to think of all of them. It is human nature for people to feel less than enthusiastic about others' ideas and to resist suggested changes to their own, but mature designers appreciate the design review process for its networking experience, innovative ideas gained from others, and reduced risks. Less mature designers, however, tend to feel insulted or intimidated by the process. The design review process seeks to achieve "appropriate quality," a balanced compromise between insiders' and outsiders' ideas, and to refrain from faultfinding or perfecting minor details.

Project quality audits

A **quality audit** verifies that quality management processes comply with prescribed processes such as system engineering, configuration management systems, and safety procedures. It is also used to verify that technical processes for critical aspects of work, such as welding, conform to prescribed procedures. Any senior stakeholder such as the customer, program manager, or executive can request an audit. Like formal design reviews, audits are relatively formal and normally involve multifunctional teams; unlike design reviews where innovative ideas can originate, audits focus strictly on verifying that processes are being implemented as required. The auditor can be an internal staff member or an external party, whoever is deemed credible, fair, honest, and unbiased.

Audit preparation begins with the auditor and the stakeholder who requested the audit agreeing on the audit's scope and schedule, and the audit team's responsibilities. The auditor prepares for the audit by compiling checklists and sometimes attending a briefing session to learn about the project. A typical thorough audit investigation will take one or two weeks. Within a few days after the audit, the auditor prepares a report that describes any nonconformities found, the importance of the nonconformities, the circumstances under which they were found and the causes (if known or determinable), and suggestions for corrective action. While the focus is on uncovering nonconformities, the report might also note any commendable findings.

Example 10.4: Unsafe Scaffolding Audited

The safety officer of a construction company requested a safety audit of scaffolding at a work site. An external consultant was hired to perform the audit and a US Department of Labor standard was used as the requirement. The audit report, produced 10 days after the audit started, indicated that all except one of the processes followed in the design and construction of the scaffolding met requirements. The scaffolding however failed the audit because no written evidence could be produced about the footing of the scaffolding to prove that a registered/licensed engineer had found it sound, rigid, and capable of carrying the maximum intended load without settling or displacement. Work on the site was stopped pending an engineering investigation on the footing. Executive management subsequently requested engineering reports on scaffold footing from all other sites.

Classification of characteristics

A project end-item or product is *specified* or *defined* in terms of a number of **characteristics** or attributes, including functional, geometric, chemical, or physical properties. Characteristics—often specified quantitatively—usually include tolerances of acceptability. In a complex system numerous characteristics are defined on drawings and other documents. The Pareto principle (discussed in more detail later) states that the large majority of problems are caused by relatively few sources. Therefore, the cost-effective way to address quality assurance is to attend to the few characteristics that result in the most quality problems or failures. This is not to say that other characteristics are ignored but that limited resources for inspection and testing should first be directed at those items classified as most crucial or problematic.

Characteristics are typically classified into four categories: *critical*, *major*, *minor*, or *incidental*. The **critical** classification is reserved for characteristics where a nonconformance would pose safety risks or lead to system failure. Quality plans often specify that items with critical characteristics be subjected to 100 percent inspection. The **major** classification is for characteristics where nonconformance would cause the loss of a major function of the deliverable. The **minor** classification is for characteristics where nonconformance would lead to small impairment of function or inconvenience with manufacturability or serviceability. Nonconformance of **incidental** characteristics would have minimal effect.

The classifications are assigned by the designers of each system in collaboration with designers of the next higher-level system and interfacing systems, and staff from manufacturing or construction. Together they analyze design characteristics regarding safety and other requirements and classify them using a set of ground rules.

Classification of characteristics should not be confused with the classification of defects. In a welded structure, for example, the specified characteristics might include the “absence of cracks or impurities” in the welded metal. A crack (defect that could cause catastrophic failure) would be classified as “critical” whereas a small amount of impurity in the weld (defect that would not affect the structural integrity) would be classified as “minor.”

Characteristics classifications are sometimes listed in a separate document, although it is more practical to show them directly on drawings and other specifications using symbols such as “C” for critical, “Ma” for major, “Mi” for minor, and so on. Absence of a symbol normally indicates the lowest priority. Only a very small percentage of characteristics should be classified as critical. A large percentage classified as critical could be a sign of poor design: when everything is critical, nothing is critical!

Characteristics classification serves as a basis for decisions regarding modifications, waivers, and deviations at all levels of a system. For example, the characteristics classification for a higher-level system provides guidance to designers of the system’s lower-level subsystems and components. Classifying the braking performance of an automobile as critical (e.g., an automobile traveling 25 miles per hour should be able to stop within 40 feet on dry pavement) tells the braking system’s designers that the brake’s components should also be classified as critical. **Failure mode and effect analysis (FMEA)** sometimes plays a role in the classification process.

Failure Mode and Effect Analysis (FMEA)

A system can fail as the result of a variety of conditions such as the short-circuiting, cracking, collapsing, or melting of its components, or inadequate, missing, or incorrect steps and procedures in its design, production, or operation. FMEA is a technique to determine the conditions (called modes) under which a system might fail, and what effects the identified failures would have on the system’s performance, safety, and environment.

The FMEA procedure is normally used during the early stages of system development and involves the following steps:

1. List the *relevant components* (or items/functions) of the system.
2. Identify all the *possible ways* that the component or system might fail (*failure modes*). This is best done by a team brainstorming the failure modes. For each mode, the causes and conditions under which they likely occur are also listed.
3. Assign a probability (P) of occurrence to each failure mode.
4. Describe and assess the *probable effects* (or impacts) of each failure mode on the performance and safety of the system, and on the environment.
5. Assess the *severity* or *seriousness* (S) of the effects.
6. Compute the *criticality* of each identified failure mode. Criticality is a function of both the probability of failure and the severity of the effects.
7. Prepare a plan to circumvent the failure mode, mitigate the effects of failure, or respond in case the failure occurs. When conformance to a specific characteristic is necessary to prevent failure, the characteristic is classified as critical.

In step 6, a **risk priority number (RPN)** can be used to determine the criticality of each identified failure mode by rating the probability (P), severity (S), and difficulty of detecting the failure (D) of each, say, 1 to 5, and then computing the *product of the three values*, $C = P \times S \times D$.

Although a failure by itself might not be critical, combined with other failures it could be very serious. The Chernobyl disaster is an example where a chain of errors (each alone not very serious) led to catastrophic failure—the meltdown of a nuclear reactor. Thus, FMEA must



See Chapter 11

consider combinations of failure modes as well as individual failure modes. Besides use in design and engineering analysis, FMEA can also be used to identify issues affecting project costs and schedules and as a tool in project risk management, described in Chapter 11.

Modeling and prototypes

Designers use various kinds of models—full-scale physical mockups, scale models, mathematical models, computer simulation models, breadboards, and full-scale prototypes—to learn how a final product, system, or subsystem will look and perform. Models and prototypes are also used in marketing to enable customers to “envision” the product or system. A full-scale wooden or plastic mockup of the cab for a truck or the cockpit of an airplane, for example, helps the producer sell the product and obtain suggestions or criticisms about it.

In product development projects, models help reduce the risk of failure to meet technical requirements. Table 10.1 shows the kinds of models built and tested in the project phases and

Table 10.1 Phases for development of products.

Project Phase	Model Built and Tested	Objectives Relating to the Elimination of Risks	Risks Eliminated
Concept	Exploratory development models (XDM) or breadboard models; such models could be built for the entire system or for specific high-risk subsystems	Proof that the new concept would be feasible	The risk that the concept would not be feasible
Verification	Advanced development models (ADM)	Proof that the product would perform according to specifications and interface well with other systems (form, fit and function)	The risk that the performance of the system and its interfaces with other systems would not be acceptable
Development	Engineering development models (EDM) manufactured from the intended final materials	Proof of reliability, availability, and maintainability	The risk of poor operational availability
Ramp-up	Pre-production models (PPM)	Proof that the product could be manufactured reliably in the production facility and could be deployed effectively	The risk of unforeseen problems in manufacturing

Table 10.2 Phases for development of processing plants.

Project Phase	Objective
Laboratory experiments	To prove the basic concept
Pilot plant	To learn how the process works when scaled up To provide inputs for the design of the final plant
Demonstration plant	To provide a full-scale plant that demonstrates to potential customers the economic feasibility as well as operational aspects

the kinds of risks they eliminate. Projects for the development of large processing plants often use pilot or demonstration facilities in a similar fashion (Table 10.2). Models for such projects usually start out using laboratory equipment but grow in complexity and capacity to enable a pilot operation that leads to a demonstration plant that functions as a commercial unit.

The kind of model used depends on the information needed versus the expense of creating and using the model. For a small product comprising only a few components, building and testing a full-scale model that closely resembles the final product is usually cost-effective; for a large, complex system, usually it is not, and computer simulation models and physical mockups are better.

Example 10.5: Modeling the Form and Fit of Boeing 777 Components¹³

One of the most pervasive problems in the development of large aircraft is aligning vast numbers of parts and components so that no interference or gaps between them happen during assembly. In the mid-1980s Boeing invested in three-dimensional CAD/CAM (computer-aided design/computer-aided manufacture) technology that would enable designers to see components as solid images and simulate their assembly into subsystems and systems on computer screens. By 1989 Boeing had concluded that “digital preassembling” of an airplane could significantly reduce the time and cost of rework that usually accompanies introducing a new airplane into the market. In 1990 Boeing began involving customers, design engineers, tool makers, manufacturing representatives, and suppliers in the concurrent engineering design process of its 777-twinjet program (see Example 4.5, Chapter 4). The physical geometry of the plane’s components was determined using CAD/CAM technology instead of physical mockups, which are time-consuming and expensive to build. This reduced changes and rework in the 777 program to 50 percent less than earlier programs.



See Chapter 4

Since the development of the Boeing 777 in the 1980s, modeling methods such as “digital twin” and “BIM,” discussed in Chapter 6, and 3-D printing have gained much in popularity.



See Chapter 6

System inspection and testing

A variety of inspections and tests are performed to ensure that components and the end-item system meet requirements. Often, the tests are performed using models and prototypes, especially in the development of new products and systems.

Tests falls into three categories: (1) tests conducted by the *designer* to ensure the system design meets user requirements and (2) tests conducted by the *producer* or *builder* to ensure the system meets system requirements, and (3) tests conducted by the *customer* and others to ensure the system meets user requirements and other contractual agreements.¹⁴

The first category of tests is aimed at validating the design. If these tests should reveal inadequate performance because of faulty or poor design, then the design stage must be repeated; if they reveal problems because of faulty requirements, then the requirements definition stage also must be repeated. Since repeating steps is costly and time-consuming, the tests should be devised to catch problems as early as possible. Of course, even if the design is perfect, if the builders do not understand the design or are not able to conform to it, the system will be inadequate; hence, the second category of tests to verify that the builders are correctly conforming to the design and that materials and workmanship meet specifications. The final category of tests consists of trials, reviews, and audits conducted by the customer to ensure that user requirements have been met and that test documentation is complete and accurate. These tests, conducted by user operators of the system, may expose design or building deficiencies that designers and builders overlooked.

Testing should follow the sequence of components first, subsystems next, then the whole system last; this will minimize the need to redesign an entire system because of faulty components. Each part is tested to ensure it functions individually; components formed from the parts are tested to ensure each component works; subsystems formed from the components are tested to ensure each subsystem performs; finally, the full system comprised of all the subsystems is tested to ensure it meets the user's performance requirements. Components or subsystems that employ innovative and leading-edge technologies pose the highest quality risks and need additional quality assurance.

Example 10.6: Boeing 787 Batteries¹⁵

In the 2010s, lithium-ion batteries used in aerospace applications were developing rapidly. For example, initially the batteries contained lithium cobalt oxide (LiCoO₂), but later they contained lithium iron phosphate (LiFePO₄); this caused much confusion and consternation among users. Boeing had been installing lithium-ion batteries in its 787 commercial airliner, and in their first year of service, 2011, at least four of the aircraft experienced battery-related problems. Later they experienced more problems and in January 2013 the Japanese airlines ANA and JAL suspended flights and grounded all Boeing 787s, the FAA (Federal Aviation Administration) ordered all US-based airlines to ground their 787s, and Chile and India followed suit.

The cost of nonconformance because of these batteries was significant: repairs on 50 grounded 787s were estimated to cost \$465,000 per plane. Among the airlines that claimed losses and negotiated compensation from Boeing was ANA, which claimed it was losing \$686,300 per plane over a two-week period.

Tests are performed against earlier developed system objectives, systems specifications, and normal user requirements. Sometimes, they are performed in excess of specifications for normal conditions to determine the system's capacity or point of failure. In **stress tests** an increasingly severe test load is applied to the system to determine its capability to handle

heavier than probable conditions, sometimes until the system fails. **Fatigue tests** determine the system's ultimate lifespan by subjecting it to *repeated cycles* under a normal load until it fails. **Failure tests** are conducted to determine the *maximum load* or stress at which the system will fail. **Nondestructive tests** (NDT) on components include a range of nondestructive methods e.g., X-ray and ultrasound examinations.

Contracts for development projects sometimes not only specify design requirements and performance criteria, but also the types of tests to verify them. Often the criteria and conditions for the tests are specified in a **test specification** referenced in the quality plan.

Documentation inspection

Projects employ a variety of testing and inspection methods to eliminate defects from documents and code. The following illustrates one approach used in design engineering and software development projects.

Example 10.7: Team Inspection Process¹⁶

The purpose of the team inspection process is to improve quality, shorten development time, and reduce costs by avoiding defects. The development team meets in a group to review the requirements documents, design documents, and software code. Members are assigned roles during the meeting:

- **Moderator:** oversees the inspection procedure and records defects spotted in the document or code.
- **Reader:** reads the document or code, line by line.
- **Inspector(s):** person who is most knowledgeable about the document or code, has the most information, and is best able to detect errors.
- **Author:** person who created the document or code.

The author of the document or code initiates the process by scheduling an inspection meeting and providing every team member with a copy of the document or code and any supporting documentation at least two days before the meeting.

An average team can inspect 10–15 pages of text or 400 lines of code in about 2 hours. Defects are documented, and the team decides whether it should meet again after the defects have been corrected. The process is considered complete when the inspector signs off on the corrections and the material is approved.

To reduce the chances of other project teams making similar mistakes, mistakes and defects discovered are entered into a database for common reference.

10.4 Techniques for Quality Control

Quality control involves performing the tasks defined in the quality management plan and taking the necessary corrective actions to assure quality. It involves a variety of techniques, as discussed next.

Inspection and acceptance testing of the final product

Whereas testing of models and prototypes provides information for use in design and development, **acceptance testing** of the final product or other deliverables verifies that the product meets specifications. Characteristics classified as critical are always inspected, but those classified as minor or incidental are not. In automobile production, for example, the braking and steering performance of every vehicle is tested. For mass-produced products, a few units might be subjected to destructive tests (i.e., tested until they break). Products that are produced one-of-a-kind or in a small batch are inspected and subjected to nondestructive testing.

Although testing end-items produced in a *production process* does not fall under the realm of *project* quality management per se, any development project where the resulting product is to be mass-produced should include specifying the testing procedures and other quality assurance processes to be used in producing that product. Product designers who are intimately familiar with key characteristics of the product and its components are best suited to specify the ways to check the quality of the product and its components during their production. For items produced in high volume, sampling is a common way to reduce the cost of inspection: based on the results from testing a few samples, the quality of the entire batch or process can be statistically inferred. Obviously, sampling is the only choice when testing destroys the product.

Tools for quality control

In the 1980s, Kaoru Ishikawa of Tokyo University defined the “basic tools” of quality control.¹⁷ These tools aim at identifying the sources of defects and nonconformities in products and processes, but they are applicable for identifying sources of and resolving problems of all kinds, including problems associated with risks. Developed for use in a production environment (Kawasaki Steel Works), the tools discussed below are nonetheless applicable to projects.¹⁸

Run chart and control chart

A **run chart** is a graph of sequential measured observations plotted versus time to reveal possible trends or anomalies in a process. A variant of this is the **control chart**, which is a graph of sampled observations plotted versus time. The control chart has upper and lower “control limits” to monitor variation in the observations and determine statistically whether it is random or nonrandom, the latter suggesting that the process being monitored might have changed and is no longer “in control” (i.e., conforming). Control charts are widely used for tracking and controlling repetitive processes and detecting process changes. Projects that involve repetitive activities, e.g., testing batches of concrete for a structure, should use control charts. Readers working on projects that involve repetitive activities or are aimed at the delivery of repetitive operating systems should refer to books on statistical control techniques, such as *Juran’s Quality Handbook*.¹⁹

Pareto chart and histogram

Vilfredo Pareto, a nineteenth-century Italian economist, formulated “Pareto’s Law” of income distribution, which states that the income and wealth distribution in a country follows a regular pattern: 80 percent of the wealth is owned by 20 percent of the people. This principle, dubbed the “80/20 rule,” has since been found to apply in principle to a wide variety of

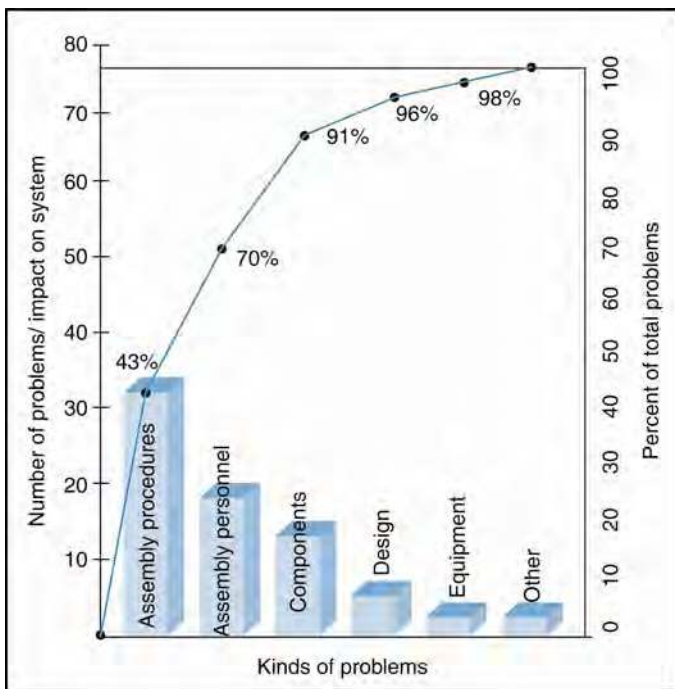


Figure 10.3
Pareto diagram.

situations, including those relating to quality. Quality consultant Dr. Joseph Juran in the late 1940s posited that the large majority of defects in anything result from relatively few causes; thus, for economic reasons it makes sense to identify these vital few causes and to direct the most effort at removing them.

Figure 10.3 is a **Pareto chart**: the bars at the bottom of the diagram show the number of problems versus the sources of the problems; the diagonal line across the figure is the cumulative effect of the problems (corresponding to scale on the right). As shown, the first problem source accounts for 43 percent of all problems; the first and second combined account for 70 percent. Thus, resolving just the first two problem sources would eliminate 70 percent of the problems.

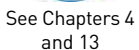
A Pareto chart is a particular kind of **histogram**, which is a chart that displays the relative frequency of events with vertical bars. The bars on a Pareto-type histogram are arranged in decreasing frequency.

Cause-and-effect diagram

Problems are often best addressed through the collective experience of project team members. Team members brainstorm ideas about the causes of a problem, and these causes are recorded on a **cause-and-effect (CE)** diagram (also called a *fishbone*, *Ishikawa* or *why-why* diagram), which is a scheme for arranging the causes for a specified effect in a logical way. Figure 10.4 shows a CE diagram to determine why a control system malfunctions. As the team generates ideas about causes, each cause is assigned to a specific branch (e.g., “Assembly procedures” is assigned to the branch “Quality of assembly”). CE diagrams and brainstorming can be used in two ways: (1) given a specified or potential outcome or problem (*effect*), to identify



11



Journal of Management Education 36(7) 809–824

being overlooked. Of course, a disadvantage of checklists is that people tend to overly rely on them and ignore things not on the list.

10.5 Summary

Project quality accounts for an end-item's compliance to specifications, fitness for the purpose, and customer expectations. It does not necessarily imply the highest grade, most product features, or even zero defects. What quality does imply is whatever is considered "best" in terms of customer expectations about the end-item's intended use.

Quality management includes the processes of quality planning, quality assurance, and quality control. Quality planning is a part of project planning and involves setting standards and specifications to be met, identifying quality-related activities in the project, and scheduling and budgeting these activities. Quality assurance is performing the planned quality activities and ensuring that the project utilizes whatever resources necessary to meet the requirements. Quality control is the ongoing process of monitoring and appraising work for quality assurance and taking corrective action. It is a part of project control and includes inspection, testing, and ad hoc problem solving.

Project quality management has benefited from the quality philosophies of TQM, lean production, and Six Sigma, all of which emphasize continuous improvement. Continuous improvement in a project environment is supported by the quality assurance process, post-project reviews, and documented lessons learned. It has also benefited from the statistical methods and basic problem-solving tools used for manufacturing and production. Beyond these, however, project quality management benefits from techniques applicable to all engineering and technical endeavors, including design reviews, configuration management, characteristics classification, FMEA, experimenting with models and prototypes, plus tools like run and control charts, Pareto diagrams, histograms, cause-and-effect diagrams, scatter diagrams, and simple checklists. Many of these tools and techniques also apply to project risk management, the subject of the next chapter.



Review Questions

1. Describe your understanding of "quality."
2. A Rolls Royce is considered a high-quality vehicle. Is this always true? Consider different users and uses.
3. How does *compliance to specification* differ from *satisfying requirements*?
4. What is the difference between *satisfying requirements* and *fitness for purpose*? Explain.
5. Explain the difference between *quality* and *grade*.
6. How does the role of the quality manager (a functional manager) regarding quality planning differ from that of the project manager?
7. The schedule or syllabus showing lectures, exams, and assignments for a college course can be considered a CI. Explain why there should be only one master copy. How does the same principle apply to an engineering drawing?
8. Indicate for each of the following whether to apply for a modification, a deviation, or a waiver:
 - a. The supplier of oil filters to an automobile manufacturer says it plans to terminate production of a filter to be used on a car that is under development.
 - b. An inspector discovered a kink in reinforcing steel. A structural engineer says that, while the steel will not comply with her drawings, the kink would have no negative effect on the strength of the steel.



See Chapter 11

- c. A damaged ship has to be repaired. The corrosion protective coating specified is not available, although a more expensive (but acceptable) coating is available.
9. Describe the differences between design reviews and audits.
10. Discuss how design reviews contribute to the approach of concurrent engineering.
11. Explain how a narrow tolerance on a manufacturing drawing differs from classifying the characteristic as critical or major.
12. Explain how classification of characteristics differs from classification of defects.
13. Discuss the relationship between project quality management and project risk management.
14. Describe how FMEA in this chapter resembles the risk management approach described in Chapter 11.
15. Perform an FMEA analysis on an electric kettle with cord and plug.
16. How do customer tests for acceptance differ from tests used to obtain design information?
17. How would you expect the bars of a Pareto diagram to change as the result of an improvement program?
18. How does the information on the x-axis of a Pareto diagram used in project control differ from the information on the x-axis of a Pareto diagram used to analyze defects in a mass production environment?
19. Describe the pros and cons of CE diagrams.



Questions About the Study Project

1. In which ways would you be able to uncover customer expectations that have not been articulated explicitly?
2. Describe the quality plan for the investigation project. If there was none, develop one. Include all aspects discussed in the Project Quality Management Plan section relevant to the specific project.
3. Discuss how the quality plan is (would be) integrated with the schedule, budget, risk management plan, and, if applicable, with the procurement plan.
4. Identify project budget items that aim to reduce the cost of external failures.
5. Draw a CE diagram and a Pareto diagram to illustrate a project management problem that you have experienced in your study project.
6. Compile a list of “lessons learned” for the project and indicate how these lessons could contribute to more successful future projects.



See Chapter 16

CASE 10.1 CEILING PANEL COLLAPSE IN THE BIG DIG PROJECT

[For more about the Big Dig project—Boston’s central artery/tunnel project, see Chapter 16, Example 16.4 and Case 16.3.]

Boston, July 11, 2006—four concrete panels weighing about three tons each fell from the ceiling of a Big Dig tunnel, crushing a woman in a car to death. The accident occurred in a 200-foot section that connects the Massachusetts Turnpike to the Ted Williams Tunnel. Said the Modern Continental Company, the contractor for that section of

the project, “We are confident that our work fully complied with the plans and specifications provided by the Central Artery Tunnel project. In addition, the work was inspected and approved by the project manager.”²⁰

The panels, installed in 1999, are held with metal trays secured to the tunnel ceiling with epoxy and bolts. The epoxy-bolt system is a tried-and-true method: holes are drilled into the concrete ceiling, cleaned, and filled with high-strength epoxy; a bolt is screwed into the hole and, as the epoxy cures, it bonds to the bolt. “That technique is used extensively,” said an engineering professor at the Massachusetts Institute of Technology.²¹ For work like this, he said, safety “redundancies” are added, that is, enough epoxy-and-bolt anchors are used to hold the ceiling panels even if some failed. But in the connector tunnel, he said, too few anchors were used. “They didn’t have enough to carry the load. There was no room for error.” But, he added, the evidence was preliminary and such a conclusion would be premature.

Some of the bolts in the ceiling wreckage had very little epoxy, and three of them had none. State Attorney General Thomas Reilly’s investigation is focusing on whether the epoxy failed or construction workers who installed the bolts misused or omitted the epoxy. An accident caused by improper installation or errors in mixing the epoxy, he said, would implicate the tunnel’s design and designers. (Epoxy requires on-site mixing before use.) He added that some documents reflected “substantial dispute” among engineers over the anchor system’s adequacy to support the weight of the ceiling panels.

Seven years before the accident, safety officer John Keaveney wrote a memo to one of his superiors at contractor Modern Continental Construction Co. saying he could not “comprehend how this structure can withhold the test of time.”²² He said his superiors at Modern Continental and representatives from Big Dig project manager Bechtel/Parsons Brinckerhoff (B/PB) assured him the system had been tested and proven. Keaveney told the *Boston Globe* he began to worry about the ceiling panels after a third-grade class toured the Big Dig in 1999. While showing the class some concrete ceiling panels and pointing to the ceiling bolts, a girl asked, “Will those things hold up the concrete?” “I said, ‘Yes, they will hold,’ but then I thought about it.”

Some have argued that the investigation should look at the tunnel’s design: why were the concrete panels so heavy, weighing 2½ to 3 tons apiece? Why were they there at all? And why did the failure of one steel hanger send six to ten of the panels crashing down? Eyewitness reports indicate the accident began with a loud snap as a steel hanger gave way, which set off a chain reaction that caused other hangers holding up a 40-foot steel bar to fail and send 12 tons of concrete smashing below. Were the bars under-designed for the weight?

Investigators are also looking at whether the wrong epoxy may have been used.²³ Invoices from 1999 show that at least one case of quick-drying epoxy was used to secure ceiling bolts rather than the standard epoxy specified by the designers; this epoxy holds 25 percent less weight than standard epoxy.

Additional issues raised during the investigation include the following:²⁴

- Design changes, which resulted in using heavier concrete ceiling panels in the connector tunnel than in the Ted Williams Tunnel.
- Lack of steel supports in sections of the connector tunnel ceiling to which bolts holding the concrete panels could have been connected.
- Possible tunnel damage caused by blast vibrations from nearby construction of an office tower.

- Use of diamond-tipped drill bits, instead of carbide bits, in drilling holes for the bolts (epoxy may not hold as well in smoother holes drilled with diamond bits).
- Impact of cold weather during installation of the epoxy-bolt system.

B/PB, the project management contractor, said in a statement “Determining the causes of this specific failure will require a thorough forensic analysis of design, methods, materials, procedures, and documentation.” As investigators scrutinize the history of the \$14 billion project, criticism is reviving that Massachusetts lacked adequate supervision of private contractors. B/PB was involved in both the design and construction efforts—an arrangement some say may have compromised oversight. There was no one checking the checkers. Wrote one blogger, “I wouldn’t want to be the registered engineer whose signature is on the design. It will be his fault if the materials and workmanship are found not to be up to specifications. But who knows if it is his fault. This is a huge mess and the whole bunch of them, engineers, managers, inspectors, and testers, should be investigated.”²⁵

QUESTIONS

1. With 20–20 hindsight, draw a cause-and-effect diagram to illustrate possible causes and effects. Include possible causes mentioned in the case and other possible failure modes and causes you can think of. How would the diagram (developed after the accident) be of value during litigation?
2. List the characteristics that should have been classified as critical.
3. Propose guidelines for a process to ensure that the epoxy would provide sufficient bonding to the concrete ceiling.
4. Explain the role that configuration management should have played in preventing the accident.
5. What role could modeling/prototyping, laboratory tests, checklists, and training have played?
6. Explain how someone within B/PB would be accountable regardless of the findings of a forensic investigation. Would B/PB be off the hook if a subcontractor were found guilty?
7. What would the implications have been if the engineer who signed off on a specific design was an engineer-in-training instead of a registered engineer?
8. Comment on the relationship between project quality management and project risk management. How could risk management have prevented the accident? How does project quality management relate to project cost management?
9. Comment on the contribution that inspection and audits could have played.

CASE 10.2 FIFA 2010 WORLD CUP SOUTH AFRICA™²⁶

Ten South African cities were selected for hosting the FIFA 2010 World Cup soccer games. In some cities existing soccer stadiums had to be upgraded while in others new stadiums had to be built at a cost of approximately R17B (approximately US \$2.4B). The centerpiece stadium for the games was the newly constructed Cape Town Stadium shown in Figure 10.5. The requirements for the once-off FIFA games typically far



Figure 10.5
Cape Town Stadium used for 2010 FIFA World Cup.

Source: GroblerduPreez /iStock/Getty Images®.

exceed what would be required by stadium owners after the games ended. For example, for each stadium FIFA required provision for 2,000 journalists for the final game, whereas an ordinary international match would draw only about 200; normally a stadium would require about ten broadcasting positions, but FIFA required 150. It therefore made sense to design facilities for normal use after the games, and to meet the temporary FIFA requirements by adding temporary items called “the Overlay.” The Overlay, which would be removed after the event, included extra commentary positions, press desks, security equipment, hospitality and other tents, as well as numerous additional cables and other equipment. It was obviously easier to design accommodations for the Overlay in new “greenfield” stadiums than in existing stadiums that had to be upgraded.

The major stakeholders in the design and construction of the stadiums are listed in Table 10.3. These stakeholders had to interface with each other and with additional parties such as national security services, police, local transport organizations, and owners of land and buildings, including schools.

The FIFA publication *Football Stadiums Handbook* provides guidelines for planning and executing FIFA events and is updated after each World Cup event. Members of the local organizing committee (LOC) visited Europe several times to learn from the 2006 FIFA World Cup event held in Germany and the Euro 2008 event held in Austria and Switzerland. One LOC member commented that items on the “wish list” for the 2006 World Cup in Germany had become the norm for the 2010 World Cup.

The stadiums were constructed by companies appointed by the host cities, while contractors for the Overlay were appointed by the LOC. Some subcontracts for the Overlay were controlled by the Overlay contractor while others, such as security,

Table 10.3 Main stakeholders in FIFA 2010 and their roles.

Stakeholders	Roles
FIFA (International Federation of Association Football) (French: <i>Fédération Internationale de Football Association</i>)	Main customer
Host cities and their planners	Provide infrastructure including match venues, training venues, and roads
Stadium owners (in some cases the sporting bodies owned the stadiums, but most were owned by the host cities)	Customers with requirements regarding their properties
SA Government (Treasury and Department of Sport)	Financial guarantees
Task team appointed by the South African Government	Monitor and control finance on behalf of Government
South African Football Association (SAFA)	Arrange the World Cup on behalf of FIFA
Local organizing committee (LOC)	Arrange World Cup on behalf of SAFA; design, construct and finance the Overlay.
LOC technical team (reporting to LOC Executive Committee and Board)	Inform host cities about FIFA and Government requirements and assist with interpretation of requirements. Combine the technical guides from: • TV host broadcaster • Hospitality rights holder • Media rights holders • FIFA marketing and security • LOC constituent groups. Prepare a Technical Guide to assist the host city planners on the requirements. Monitor and report to the LOC Executive Committee and Board regarding: • Quality • Progress • Finance • FIFA Compliance.
Stadium designers and construction companies	Design and construction of stadiums
Host city professionals	Design and construction of the precinct (surrounds) and the access roads
Overlay contractors (designers and suppliers), appointed by the LOC	Specifications and supply of Overlay items

electric power, backup electricity, water supply and wastewater drainage, were controlled by other parties. While the Overlay contractors reported to the LOC, the host cities authorized their contractors to take over spaces to construct the Overlay. The different parties—the LOC and their Overlay contractors, the host cities and their stadium contractors—worked in the same spaces and at the same time, but with different responsibilities and reporting structures. This proved a challenge to coordinate work and caused some conflict.

When a stadium was nearly completed, all relevant stakeholders were required to attend an on-site inspection and agree to the sign-off. Several such events were properly recorded by minutes and photographic recordings.

Progress reviews and audits to ensure that all stadiums and other spaces were FIFA compliant were mainly performed by the LOC technical team. FIFA, LOC, and government constituent groups also regularly met with members of the host cities to assess and assist them with FIFA compliance. These meetings were chaired by FIFA, though one member of the technical team later remarked: “This was a mistake—LOC should have taken control.” In between meetings, the relevant stakeholders would assemble in Johannesburg and take “virtual tours” of the sites; the host cities would present their progress through multimedia means, which included a satellite link with FIFA Headquarters in Zurich, Switzerland. This process ensured that the host cities and their technical teams were fully aware of the requirements, and it afforded them the opportunity to discuss any concerns they had with the customers.

QUESTIONS

1. Given two sets of requirements, one for the FIFA games and the other for after the games, what would be an appropriate way to define “quality”?
2. List the quality management activities mentioned in the case.
3.
 - a. Comment on the reporting structures and responsibility for audits and reviews.
 - b. Who should have provided quality guarantees?
 - c. What planning processes and techniques would have been helpful regarding the roles of the various stakeholders?
4. Comment on the problem of people from different organizations working in the same space at the same time.

CASE 10.3 AIRBAG ADVERSITY

Insufficient quality management during product development, launch, and production can subsequently lead to costly measures to rectify problems. This is especially true when safety-critical items are involved. The 2015 global recall of a large variety of automobile makes, which affected millions of owners, is just such a case. The massive recall was related to potentially defective airbags used by auto manufacturers following reports that the airbags had inflators that could explode and expel metal and plastic shards into vehicle occupants. The airbags were supplied by Takata Corporation of Japan, the second largest global supplier of airbags and seatbelts. A *New York Times* report found

at least 139 reported injuries across all automakers; in Honda vehicles alone, at least two deaths and 30 injuries were reported.

When the fault was first announced in 2013 only six makes were involved, but by 2015 a large range of makes and 34 million vehicles in the USA alone were potentially affected. It was said that Takata was ramping up to produce replacements at the rate of 10 million per year.

Initially it was thought that propellant chemicals were mishandled and improperly stored during assembly, which might cause the metal inflators to explode due to extreme internal pressure. Later, humid weather was thought to have played a role as well. Takata cited other possible contributors, including rust, bad welds, and in at least one case chewing gum dropped into an inflator. Documents showed that in 2002 Takata's plant in Mexico allowed a defect rate that was "six to eight times above" the acceptable limit, or roughly 60 to 80 defective parts for every 1 million airbag inflators shipped.

QUESTIONS

1. Explain the role of classification of characteristics in reducing costs of ensuring quality.
2. Discuss why any pressure to rush or cut costs on the development of safety critical items should be resisted. List specific costs incurred by recall programs to withdraw vehicles from use; include costs for (a) loss of automaker or Takata reputation and future sales, (b) litigation between automakers and Takata and (c) lawsuits resulting from people killed and injured.
3. Discuss specific procedures and steps in the design and manufacturing processes for safety-critical components and systems, and how they should differ from procedures and quality management steps for noncritical items.
4. What specific techniques and procedures would you recommend for the design and manufacture of such safety-critical items?
5. Specifications often state that a certain incident or event should not occur more than, say, once in 1 million or once in 10 million times. Explain why tests to ensure such a requirement are difficult and costly.

Notes

1. Carruthers M. *Principles of Management for Quality Projects*. London: International Thompson Press; 1999.
2. Carruthers, *Principles of Management for Quality Projects*.
3. Yourdan E. *Rise and Resurrection of the American Programmer*. Upper Saddle River, NY: Yourdan Press/Prentice Hall; 1998, pp. 157–181.
4. Crosby P. *Quality Is Free*. McGraw-Hill; 1979.
5. Bach J. The challenge of 'good enough' software. *American Programmer* 8(10); 1995: 2–11.
6. Nicholas J. *Lean Production for Competitive Advantage: A Comprehensive Guide to Lean Methodologies and Management Practices*, 2nd edn. Boca Raton, FL: CRC/Productivity Press; 2018.
7. Crosby, *Quality Is Free*.
8. International Systems Organization *ISO 9001, Quality Management Systems—Requirements*. Geneva; 2015.
9. Kransdorff A. The role of the post-project analysis. *The Learning Organization* 3(1); 1996: 11–15.
9. The ISO/CD 10007 standard offers guidelines on configuration management systems: International Standards Organization, *ISO/CD 10007 Quality Management Systems—Guidelines for Configuration Management*. Geneva; 2017.

10. Gray M. *Angle of Attack: Harrison Storms and the Race to the Moon*. New York: W.W. Norton; 1997, pp. 170–171.
11. East E., Kirby J., and Perez G. Improved design review through web collaboration. *Journal of Management in Engineering*. April; 2004: 51–55.
12. From Muirhead B. and Simon W. *High Velocity Leadership: The Mars Pathfinder Approach to Faster, Better, Cheaper*. New York: Harper Business; 1999, pp. 86–89, 178–179.
13. <https://www.nesfircroft.com/resources/blog/the-boeing-777-and-how-computer-aided-design-changed-the-face-of-air-travel/> (accessed June 17, 2024).
14. Where multiple units of a system are produced, the first and second groups of tests are performed on a prototype. See Chapter 4 for prototyping. When the system is single copy, the second group of tests occurs during construction, and the design is verified afterward.
15. Kerzner, H. *Project Management Case Studies*, 5th edn. New York: Wiley; 2017.
16. Information for this application provided by Elisa Denney and Jennifer Brown.
17. Ishikawa K. *What Is Quality Control?* Englewood Cliffs, NY: Prentice Hall; 1982.
18. Bamford D. and Greatbanks R. The use of quality management tools and techniques: A study of application in everyday situations. *International Journal of Quality and Reliability Management* 22(4); 2005.
19. Defeo J. (Ed.). *Juran's Quality Handbook*, 7th edn. New York: McGraw-Hill Education; 2017.
20. Belluck P. and Zezima K. Accident in Boston's Big Dig kills woman in car. *New York Times* July 12; 2006.
21. Bradley M. Bolt failure at Big Dig: an anomaly? *The Christian Science Monitor* July 21; 2006.
22. Murphy S. Memo warned of ceiling collapse: safety officer feared deaths in '99, now agonizes over tragedy. *Boston Globe*, July 26; 2006.
23. Allen S. and Murphy S. Big Dig job may have used wrong epoxy. *Boston Globe* May 3; 2007.
24. Drake B. Investigators probe Boston Tunnel design. CENews.com September 1; 2006, www.cenews.com/article.asp?id=1108; accessed May 15, 2007.
25. Waters P. Physics forums, www.physicsforums.com/showthread.php?t=126374, russ_waters July 17; 2006, 9.30 pm; accessed June 20, 2024.
26. Personal communication, Eugene van Vuuren. Member of LOC Technical Team. Nov. 2010.

Chapter 11

Project Risk Management

Every project is risky, meaning project outcomes will not necessarily turn out as planned. The project could significantly overrun cost or schedule targets, or the end-item fall short of requirements. Project outcomes are the result of many factors, including some that are unpredictable and uncontrollable. Risk level is associated with the certainty that outcomes will be as expected. High-certainty outcomes have low risk; low-certainty outcomes have high risk. Certainty derives from knowledge and experience gained in prior projects as well as management's ability to mitigate anticipated risks and respond to newly emerging ones.¹

11.1 Risk Concepts

Risk is a function of a project's uniqueness and the project team's experience. When activities are routine or have been performed many times before, managers can anticipate the risks and manipulate the system design and project plan to achieve the desired outcomes. But when the work is unique or the team inexperienced, the outcomes are less certain, which makes it difficult to anticipate problems or know how to deal with them. Even routine projects can be risky due to factors that newly arise or are beyond anyone's control.

The notion of project risk involves two concepts:

1. The **likelihood** that some unplanned event will occur.
2. The **impact** of the event if it does occur.

Risk is a joint function of the two:

$$\text{Risk} = f(\text{likelihood, impact})$$

A project will ordinarily be considered “risky” whenever at least one—the likelihood or the impact—is large. For example, it will be considered risky when the potential impact is anything *negative*—human injury or big financial loss, even if the likelihood is small. Risk can also mean unplanned **opportunities**, such as potential greater rewards, savings,



Figure 11.1
Risk management process and elements.

or benefits. Typically, however, the bigger worry is about **threats**—risks with negative consequences.

Most managers are accustomed to dealing with facts, figures, and hard numbers, so they find the concept of risk hard to deal with. Faced with uncertainty, they prefer to ignore potential problems, though, of course, that doesn't make the problems go away.

Although risk cannot be eliminated, it can be reduced and plans readied in case things go wrong; this is the purpose of the risk management process, shown in Figure 11.1.

11.2 Risk Identification

You can only manage things you are aware of. Thus, risk management begins with identifying the risks and predicting their consequences.

The risk of bad outcomes in projects is sometimes referred to as the **risk of failure**, which implies that a project might fall short of schedule, budget, or technical performance goals by a significant margin.

Among ways to identify project risks, one is to proceed according to project chronology—that is, to look at the phases and stages in the life cycle (feasibility, contract negotiation, system concept, definition, etc.) and identify the risks in each. Each phase presents unique hurdles and problems that could halt the project immediately or lead to later failure (as noted in Chapter 10, Table 10.1). In product development projects, the risk of failure is highest in the early stage of preliminary design and diminishes thereafter. One of the purposes of formal design reviews, discussed in Chapter 10, is to reveal risks related to the end-item design very early in the project. Some risks remain throughout, such as potential loss of funding or management commitment.

Risk can also be identified by type of work or technical function, such as engineering risks associated with product reliability and maintainability, or production risks associated with the manufacturability of a product or the availability of raw materials.

Risk identification starts in the conception phase and focuses on those risk factors that would hinder the project or leave it destined to fail. Factors that contribute to high risk include:

- Using a unique or unusual approach.
- Attempting to both develop a new system and advance technology at the same time.



See Chapter 10

- Developing and testing new equipment, systems, or procedures.
- Operating in an unpredictable or unstable environment.
- Uncertainty about the desired solution, outcome, or requirements.

High-risk factors must be studied and well understood before the project can be approved and funds committed. Risks identified in the conception phase are sometimes broadly defined and subjectively assessed, though they might also be analyzed using the methods discussed later. When multiple, competing projects are under consideration, an assessment is performed to decide which of them, based upon tradeoffs of the relative risks, benefits, and available funding, is best.² Comparing and selecting projects based upon criteria, including risk, is discussed in Chapter 19.



See Chapter 19

Risk sources

Any uncertain factor that can influence the outcome of a project is a **risk source**, **hazard**, or **threat**. Identifying risk sources requires learning as much as possible about potential things known to go wrong and their outcome, as well as trying to identify things not already known—the “unknown unknowns.”

Risk sources in projects can be classified as internal and external.

Internal sources

These are risk sources that originate inside the project and over which project managers and stakeholders have some degree of control. They fall into three categories: market risk, assumptions risk, and technical risk.

Market risk is the risk of not fulfilling the needs of the market or the requirements of particular customers. Sources of market risk include:

- Failure to adequately define the market or customer needs and requirements.
- Failure to identify changing needs and requirements.
- Failure to identify products newly introduced by competitors.

Market risk stems from the developer misreading the market environment. It can be reduced by working closely with customers or market representatives; thoroughly defining needs and requirements early in the project; closely monitoring trends and developments among markets, customers, and competitors; and updating requirements as necessary throughout the project.

Assumptions risk is risk associated with the numerous implicit or explicit assumptions made in feasibility studies and project plans during project conception and definition. Faulty, inaccurate, or invalid assumptions put the project in jeopardy of failing to meet time, cost, or technical requirements, or causing unanticipated and harmful side effects.

Technical risk is the risk of encountering technical problems in project work or with the end-item. (Sometimes these risks are listed in special categories—*schedule risks* being those that would cause delays, *cost risks* being those that would escalate costs, and so on.) Technical risk is high in projects that involve new and untried technical applications, but low in projects that involve familiar activities and technologies.

One approach to expressing technical risk is to rate the project's primary processes or end-item as high, medium, or low according to the following features:³

- **Maturity.** How experienced or knowledgeable is the project team in the project technology? An end-item or process that takes advantage of existing experience and knowledge (off-the-shelf) is less risky than one that is innovative, untried, or cutting edge.
- **Complexity.** How many steps, elements, or components are in the product or process, and how tightly are they interrelated? *Ceteris paribus*, an end-item or process with numerous, interrelated steps or components is riskier than one with fewer steps and simpler relationships. Complexity tends to correlate with quality and dependency—higher complexity increases interdependencies and quality-related issues.
- **Quality.** How producible, reliable, and testable is the end-item or process? In general, an end-item or process that has been produced and is reliable and testable is less risky than one that has yet to be produced or has unknown reliability or testability.
- **Concurrency or dependency.** To what extent do multiple, dependent activities in the project overlap? Activities performed in sequence with no overlap are less risky than activities that overlap (i.e., the discrete-staged approach is less risky than fast-tracking).

A sub-category of technical risk is health, safety, and environmental risk, which includes hazards to project workers, the surrounding society, and the ecology because of the project. This risk stems from short-term hazards due to working conditions, procedures, and materials used in the project, and long-term hazards from the functioning, operation, or mere existence of the project end-items.

External sources

These are risk sources that originate from outside the project and over which project managers have limited or no ability to control. They include:

- Government regulations
- Competitors' actions
- Interest rates and exchange rates
- Senior management or customer decisions regarding project priorities, staffing, or budgets
- Customer needs and behavior
- Supplier/subcontractor relations and business failures
- Local physical environment (weather, terrain, infrastructure)
- Labor availability (strikes and walkouts)
- Material or labor resources (shortages)
- Customer or subcontractor control over project work and resources
- Local culture
- Adverse impacts of climate change
- Project stakeholders

The last, project stakeholders, includes anyone affected by or interested in the project, many of whom can influence project outcomes—both positively and negatively, which is the topic of Chapter 16. And all these risk sources tend to exacerbate for international projects and megaprojects, the topic of Chapter 20.



See
Chapters 16,
and 20

Identification techniques

Risk sources (hereafter just called “risks”) for the project and its end-items are identified in many ways; the main ones are project analogy, checklists, WBS analysis, process flowcharts, project networks, brainstorming, affinity diagrams, cause–effect diagrams, and the Delphi technique.

Project analogy

Project analogy involves scrutinizing the records, post-completion summary reports, and project team members’ recollections of completed analogous projects to identify risks in future projects. The more complete, accurate, and well-catalogued the documentation of past projects and the better people’s memories, the more useful these are for identifying risks. Beyond just investigating past projects, the method requires identifying ones that are similar in *significant* ways to the project for which risks should be assessed. **Knowledge management** methods, described in Chapter 18, promote learning from past projects to help anticipate risks in future ones.



See Chapter 18

Checklists

Documentation from prior projects is used to create **checklists** of risk sources. A checklist is initially based upon the experiences from past projects and is updated as new experience is gained from recent projects. Risk checklists can pertain to the entire project or to specific phases, work packages, or tasks within it.

To illustrate, the checklist in Table 11.1 shows the risks and risk severity associated with three categories of risk sources: (1) status of implementation plan, (2) number of module interfaces,

Table 11.1 Risk checklist.

Risk Sources	Risk Severity
Status of implementation plan	
1. No plan required	None
2. Standard plan, existing, complete	Low
3. Plan being prepared	Medium
4. Plan not started	High
Number of interfaces between modules	
1. Less than 5	None
2. 5–10	Low
3. 11–20	Medium
4. More than 20	High
Percent of system components requiring tests	
1. 0–1	None
2. 2–10	Low
3. 11–30	Medium
4. Over 30	High

and (3) percentage of components requiring testing. Suppose, for example, an upcoming project will use a standard plan, have eight module interfaces, and test 20 percent of the system components. These project risk sources will thus be rated as low, low, and medium, respectively.

As experience grows with completed projects, the checklists are expanded and updated. The more experience a manager or company gains with projects, the more they learn about the risks, and the more comprehensive the checklists become. While a checklist cannot guarantee that all significant risk sources in a project will be identified, it does help ensure that the important known ones won't be overlooked.

A disadvantage of risk checklists is that people tend to look only at the risks listed and not consider any not listed. The last item on every checklist should be "Now, list all the possible important things not already on this list and check them too!"

Work breakdown structure (WBS)

Risks can be identified using the WBS. Each work package is scrutinized for potential technical hurdles or problems with work tasks, customers, suppliers, equipment, or resource availability. Each is assessed for internal risks in terms of, e.g., complexity, maturity, quality, and concurrency, and for external risks, e.g., relying on a subcontractor to manage the work package. The risk of every work package is rated as, for example, high, medium, or low.

Process flowchart

Project risks can also be identified from process flowcharts that illustrate the steps, procedures, and workflow between tasks and activities in a project or work package. Examining the flowchart enables potential trouble spots and risky areas to be pinpointed.

FMEA and HAZOP

The **Failure Mode and Effects Analysis (FMEA)** method described in Chapter 10 can be used to identify conditions leading to system failure and, thus, subjecting the project, people, and environment to risk. A related method called **HAZOP—Hazard and Operability Study**—is a rigorous investigation to assess what happens when the system starts up, shuts down, or encounters problems. The method focuses on the system design and possible errors, omissions, or inherent hazards. Both FMEA and HAZOP are widely used in technical projects—HAZOP most commonly in process industries and infrastructure projects.



See Chapter 10

Project networks and convergence points

Similarly, risks can be identified through scrutiny of the precedence relationships and concurrent or sequential scheduling of activities in project networks, the subject of Chapters 7 and 8. Risk tends to increase at merge points in the network where work done by different teams comes together and must be integrated; sometimes only then do problems become evident, such as subsystems produced by different teams not matching up or functioning correctly. The risk of project delay from this so-called **merge-point bias** is discussed in Chapter 8.



See Chapter 7
and 8

Brainstorming, affinity diagrams, and cause-and-effect diagrams

Risks can be identified from the collective experiences of project team members who participate in a **brainstorming** session to share opinions about possible risk sources. One way to record their opinions is on sticky notes, and then to cluster the notes by common theme or relationship; the result is an **affinity diagram**. Similar to an affinity diagram is a **cause-and-effect diagram (CE)**: it clusters opinions into groups, then goes a step further to show how the opinions about causes relate to a common outcome. Figure 11.2 is an example: it shows opinions about plausible causes for delaying the project. Brainstorming with CE is used in two ways: (1) starting with an outcome (effect), to identify the potential causes (sources); (2) starting with possible risk sources (causes), to identify the possible outcomes (effects). Figure 11.2 illustrates the first use.



See Chapter 10

The risks shown in the diagram in Figure 11.2 are grouped into five categories (software, hardware, and so on), the result of clustering the opinions about risk from brainstorming. Each category is subdivided into more fundamental sources of risk; for example, the category “staff” includes the risk of “staff shortage,” which could be caused by “inability to hire and train more staff.” CE and related analysis techniques are further discussed in Chapter 10.

To encourage original thinking and a comprehensive list of possible risks, risks should never be assessed as they are being generated during brainstorming. Any mention that a risk is “unrealistic” or “impossible” might lead to some important risks being discarded. Hence, no risks should be assessed until after the risk list has been compiled.

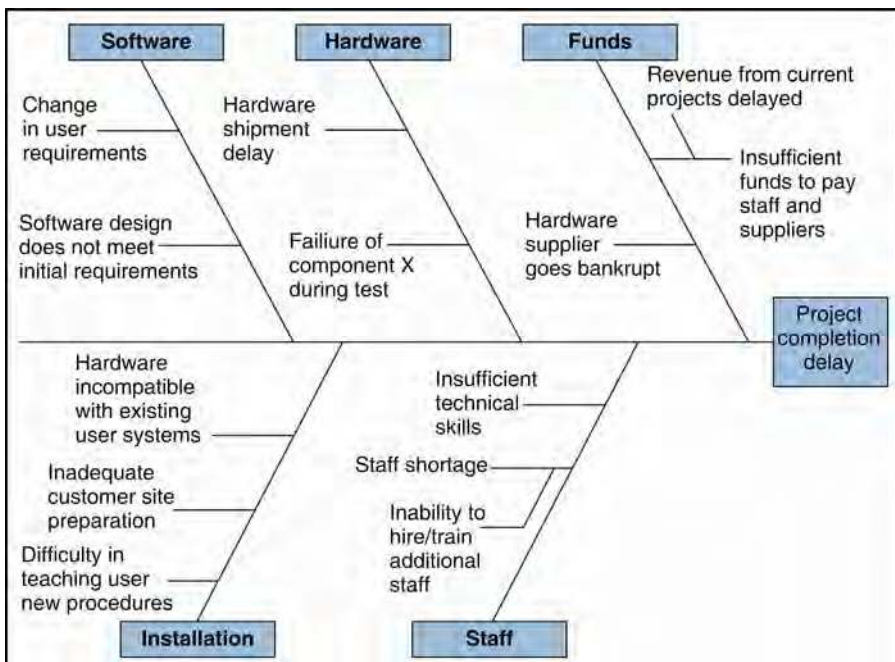


Figure 11.2
Cause-and-effect diagram.

Delphi technique

Delphi is a technique for combining the judgments of several people about risks to identify the most important ones. It starts with each person creating a list of the risks, and then everyone sharing their lists and explaining why the risks are important. They then can revise their lists and give reasons for any additions or deletions, and then share their revised lists. The process repeats until the risks on everyone's list converges. The process, also used in estimating, is described more fully in Chapter 6.



See Chapter 6

Risk symptoms and triggers

As the sources and outcomes of each risk are identified, so are its **symptoms**, which are visible indicators or warning signs that the risk is materializing; these serve as a **trigger** to initiate counteractions or contingencies to mitigate or combat the risk. For example, for the risk “failure to meet technical requirements,” a symptom might be “failure of component X during test”; should that symptom be observed, it would trigger the action “move to design plan B.”

11.3 Risk Assessment

Risks are ubiquitous, but only the notable or significant ones require attention. If a risk and its consequences are significant, ways must be found to avoid or reduce the risk to an acceptable level. What is considered “acceptable” depends on the **risk tolerance** of project stakeholders. Often, managers with experience avoid risks (are risk averse) because they understand the risks and their consequences, whereas managers with less experience take risks (are risk tolerant) because they are ignorant of the consequences of the risks.

What is considered “significant” depends on the risk likelihood, the risk impact, and the risk consequence.

Risk likelihood

Risk likelihood is the probability that a risk factor will actually materialize.⁴ It can be expressed as a numerical value between 1.0 (certain to happen) and 0 (impossible) or as a qualitative, ordinal, rating such as high, medium, or low. (Interestingly, if a risk has probability of 1.0—meaning it is *certain* to happen—it is considered not a risk but an **issue** to be dealt with, as discussed in Chapter 13). Numerical values and qualitative ratings are sometimes used interchangeably. Table 11.2 shows an example: when, for instance, someone says, “the likelihood of this risk is low,” that means the probability of it happening, according to the table, is 20 percent or less.

Table 11.2 is an illustration only, and the association between qualitative ratings and numerical values is subjective and depends on the experience of the project team and the risk tolerance of stakeholders. For example, Table 11.2 might have been created for a project with high economic stakes, in which case “high risk” equates to a numerical likelihood of greater than 50 percent. In a project with low economic stakes, “high risk” might equate to a numerical likelihood of 75 percent or more. People often have difficulty agreeing on the numerical likelihood value for a given qualitative rating and vice versa, even given the same information or experience; this is described later in Example 11.2.



See Chapter 13

Table 11.2 Risk likelihood: Qualitative ratings for quantitative values.

Qualitative	Numerical
Low	0–0.20
Medium	0.21–0.50
High	0.51–1.00

Table 11.3 Likelihoods for different sources of failure.

Likelihood	M_H	M_S	C_H	C_S	D
0.1 (low)	Existing	Existing	Simple design	Simple design	Independent
0.3 (minor)	Minor redesign	Minor redesign	Minor complexity	Minor complexity	Schedule dependent on existing system
0.5 (moderate)	Major change feasible	Major change feasible	Moderate complexity	Moderate complexity	Performance dependent on existing system
0.7 (significant)	Complex design; existing technology	New, but like existing software	Significant complexity	Significant complexity	Schedule dependent on new system
0.9 (high)	State of the art; little research done	State of the art; never done	Extreme complexity	Extreme complexity	Performance dependent on new system

Notes: M_S , failure likelihood due to software immaturity; C_S , failure likelihood due to software complexity; M_H , failure likelihood due to hardware immaturity; C_H , failure likelihood due to hardware complexity; D , failure likelihood due to dependency on external factors.

Note: “failure” refers to not meeting technical goals.

Adapted from Roetzheim W. *Structured Computer Project Management*. Upper Saddle River, NJ: Prentice Hall; 1988, pp. 23–26.

Table 11.3 is a checklist for five potential sources of failure in computer systems projects and associated numerical likelihoods.⁵ Looking at the M_S column, the likelihood of failure for existing software is low, but for state-of-the-art software it is high. (Again, these likelihood values are illustrative and would be tailored to each project depending on the experience and opinion of stakeholders). A likelihood estimate based on the opinions of several individuals (assuming all have relevant experience) is usually more valid than one based on only a few.

When a project has multiple, independent risk sources (as is common) they can be combined into a single **composite likelihood factor**, or **CLF**. Using the sources in Table 11.3, the CLF can be computed as a weighted average,

$$CLF = (W1)M_H + (W2)C_s + (W3)M_s + (W4)C_s + (W5)D \quad (11.1)$$

where $W1$, $W2$, $W3$, $W4$, and $W5$ each have values 0 through 1.0 and sum to 1.0. This is illustrated in Example 11.1

Example 11.1: Computation of CLF

The ROSEBUD project involves development of hardware and software with characteristics as follows: the hardware is existing and of minor complexity; the software will be developed as a minor redesign of current software and is of moderate complexity; the performance of the overall system will depend on how well it can be integrated into another, larger system. Thus, from Table 11.3, $M_H = 0.1$, $C_H = 0.3$, $M_s = 0.5$, $C_s = 0.3$, and $D = 0.5$. If all sources are rated equally at 0.2, then

$$CLF = (0.2) 0.1 + (0.2) 0.3 + (0.2) 0.5 + (0.2) 0.3 + (0.2) 0.5 = 0.34$$

The application of this CLF will be discussed shortly.

Note that the computation of equation 11.1 assumes that the risk sources are *independent*. If they are not—if, for example, failure due to software complexity depends on failure due to hardware complexity—then the two likelihoods cannot be summed. The sources would have to be subjectively combined into one source (“failure due to a combination of software and hardware complexity”) and a single likelihood value assigned based on judgment.

One way to show the interdependency of risk factors is with an **influence diagram**. An example is Figure 11.3.⁶ To construct the diagram, start with a list of previously identified risks (e.g., from Figure 11.2), then for each risk ask whether it is influenced by, or has influence on, any of the other risks. If so, draw them as in Figure 11.3, using arrows to show related risks and the direction of influence (e.g., S.1 influences S.2 and I.2). To minimize confusion, keep the number of risks on the diagram to about 15 or fewer. Risks with the most connections are the most important; in Figure 11.3, these would be risks I.2, S.1, and S.2; each is influenced by other risks, which increases the likelihood of failure.

Risk likelihood is also affected by the future: *ceteris paribus*, activities planned further in the future are riskier (have greater likelihood of failure) than those closer at hand.⁷ This is because activities farther in the future have greater chances of being influenced by unknowns. The greater the number of risks and unknowns for phases later on is one reason for phased (rolling wave) project planning, discussed in Chapter 4. As a project enters the execution phase and moves toward completion, the unknowns diminish and so does likelihood of failure. But there is a tradeoff: as the project progresses and the risks diminish, the stakes in the project—the amount of human and financial capital sunk into it—increase. This means that losses from failure incurred later in the project exceed losses incurred earlier.



See Chapter 4

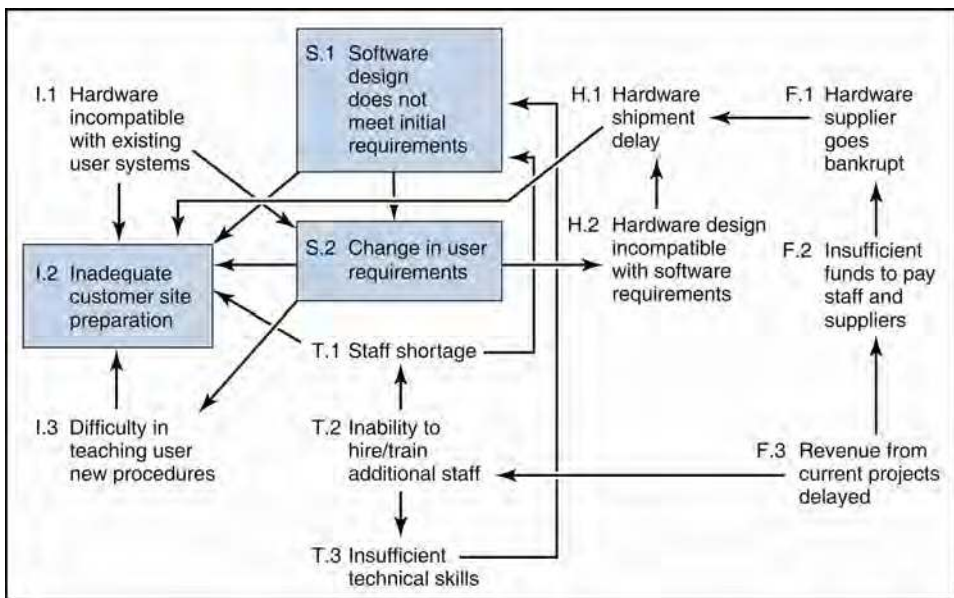


Figure 11.3
Influence diagram.

Risk impact

What would happen if a risk hazard materializes? The result would be a **risk impact**. A poorly marked highway intersection is a risk hazard; it poses the risk impact of a collision and personal injury. Risk impact in projects can be specified in terms of time, cost, performance, pollution, and so on. For example, the impact of insufficient resources might be failure to meet the target date.

Risk impact can be expressed as a qualitative rating such as high, medium, or low based upon a manager's judgment about the impact. For example, a risk leading to a schedule delay of 1 month might be considered "medium impact," whereas a 3-month delay would be deemed "high impact."

Risk impact also can be expressed as a numerical measure between 0 and 1.0, where 0 is "not serious" and 1.0 is "catastrophic." Again, the rating is subjective and depends upon judgment. Table 11.4, for example, represents judgments about the impacts associated with various technical, cost, and schedule situations, and suggested impact value ratings associated with each of them.⁸ The assigned risk impact values are subjective—even when derived from empirical data.

Table 11.4 Impact values for different technical, cost, and time situations.

Impact Value	Technical Impact (TI)	Cost Impact (CI)	Schedule Impact (SI)
0.1 (low)	Minimal impact	No cost increase; within budget	Negligible schedule slip; compensated by slack time
0.3 (minor)	Small performance reduction	<10% increase	Minor (< 1 month)
0.5 (moderate)	Moderate performance reduction	10–25% increase	Moderate (1–3 months)
0.7 (significant)	Significant performance reduction	25–50% increase	Significant (>3 months)
0.9 (high)	Technical goals not achievable	>50% increase	Large (unacceptable)

Adapted from Roetzheim W. *Structured Computer Project Management*. Upper Saddle River, NJ: Prentice Hall; 1988, 23–26.

Example 11.2: Estimating Risk Likelihood and Risk Impact in New Technologies

Risk assessment in innovative technologies is, well, difficult. The risk of a serious problem can stem from a chain of events [e.g., a machine malfunctions but a sensor does not detect it or an operator takes the wrong action], and to assign a probability to the risk requires identifying all the events in the chain, estimating the probability of each, and combining the probabilities. Managers and designers can try to think of every event, but they can never be sure they haven't missed some. When a project involves new technologies, the estimates are largely guesses. For example, an early MIT report stated that the likelihood of a nuclear reactor core meltdown is one every 17,000 years. The report said a meltdown in a particular plant would occur only after *many hundreds of years* of operation, yet less than 5 years after the report a reactor at Three Mile Island suffered a partial meltdown and released radioactivity into the atmosphere.⁹

The space shuttle is another case: NASA originally put the risk of a catastrophic accident at 1 in 100,000, but after the Challenger disaster revised it to 1 in 200. With the additional loss of Columbia (the second loss in 113 missions) the actual risk became 1 in 56. The shuttles originally were design-rated for 100 missions, yet Columbia broke up during its 26th.¹⁰ Few data points (five operational shuttles and 113 missions over 20 years) in combination with incredible complexity made it impossible to accurately predict the risks for the shuttle system, yet for many projects the data available for estimating probabilities is even sparser.

Estimating impacts is equally difficult, and experts from different fields given identical facts often reach different conclusions. In one survey that rated the hazards of nuclear waste using a 17-point scale, biologists rated it 10.1, geologists 8.3, and physicists 7.3.¹¹ Risk assessment depends on culture and training and is never completely rational; because of this, it should be based upon the opinions of many experts representing a range of disciplines.

Just as the likelihoods for multiple risks can be combined, so can the impacts from multiple risk sources. A **composite impact factor, CIF**, can be computed using a weighted average,

$$\text{CIF} = (W1)\text{TI} + (W2)\text{CI} + (W3)\text{SI} \quad (11.2)$$

where W1, W2, and W3 have values 0 through 1.0, and together sum to 1.0. CIF will thus range from 0.0 to 1.0, where 0 means “no impact” and 1.0 means “the most severe impact.” Example 11.3 illustrates.

Example 11.3: Computation of CIF

A particular failure to meet certain technical goals is expected to have minimal impact on technical performance and be corrected within 2 months and at a 20 percent cost increase. Therefore, from Table 11.4:

$$\text{TI} = 0.1, \text{SI} = 0.5, \text{CI} = 0.5$$

Suppose the most important criterion is technical performance, followed by schedule, then cost, and the weights assigned to the criteria are 0.5, 0.3, and 0.2, respectively. Therefore, from equation (11.2):

$$\text{CIF} = (0.5)(0.1) + (0.3)(0.5) + (0.2)(0.5) = 0.22$$

Again, note that the computation of equation 11.2 assumes that the risk impacts are independent. If they are not, the equation does not apply and the single value impacts must be treated jointly, an example being “the impact of both a 20 percent increase in cost and a 3-month schedule slip is rated as 0.6.” Application of this CIF is discussed in the next section.

Another way to express risk impact is in terms of what it would take to *recover* from, or compensate for, an undesirable impact. For example, suppose that use of an innovative technology risks not meeting performance requirements. The plan is to test the technology, but then if the tests reveal inferior performance, to abandon it and instead use a proven approach. The risk impact would be the impact of switching technologies in terms of schedule delay and additional cost, e.g., 4 months and \$300,000.

Risk impact should be assessed for the entire project and articulated with the assumption that no response or preventive measures are taken. In the above instance, \$300,000 is the anticipated expense under the assumption that nothing will be done to avoid or prevent the failure of the new technology. This assessed impact will be used as a measure to evaluate the effectiveness of alternative ways to reduce or prevent risk hazards, as discussed later.¹²

Risk consequence

Earlier, the notion of risk was defined as being a function of risk likelihood and risk impact; the combined consideration of both likelihood and impact is the **risk consequence** or **risk exposure**.

Mathematically, the most common way to express risk consequence is,

$$\text{Risk consequence} = (\text{Likelihood}) \times (\text{Impact}) \quad (11.3)$$

Using the previously computed likelihood (CLF) of 0.34 (Example 11.1) and impact (CIF) of 0.22 (Example 11.3), the **risk consequence rating, RCR**, is

$$\text{RCR} = (\text{CLF}) \times (\text{CIF}) = (0.34) \times (0.22) = 0.078$$

RCR ranges in value from 0 to 1.0, and a very small RCR such as 0.078 might be judged as “inconsequential.” Assessing values of RCR as being high, medium, or low is subjective, and the principal use of RCR is to compare and prioritize risks—to separate those that can be ignored (small RCR, low consequence) from those that must be heeded (large RCR, high consequence).

Risk consequence can be expressed in other ways too. For example, suppose the likelihood associated with a risk is 0.40, and, should the risk materialize, its estimated impact would be to delay the project by 4 months and increase the cost by \$300,000. The risk consequences for time and cost are thus

$$\text{Risk consequence time (RT)} = (4 \text{ months})(0.40) = 1.6 \text{ months} = 6.4 \text{ weeks}$$

$$\text{Risk consequence cost (RC)} = (\$300,000)(0.40) = \$12,000$$

These are “expected value” risk consequences, or what the average outcomes would be if the situation were repeated many times. The concept of expected value is further discussed in the Appendix to this chapter.

A disadvantage of using expected value is that it assumes people are “risk neutral,” which they are not. For example, you might be willing to play a game with a 50 percent chance of losing \$10 (i.e., $\text{RC} = \$5$), but would you still play it with a 10^{-6} percent chance of losing \$5,000,000 ($\text{RC} = \$5$ also)?

The magnitude of the consequences—whether high, medium, or low—as a function of the specified likelihood and impact values can be determined by plotting the values on a diagram such as Figure 11.4. Just as the likelihood and impact values are subjective, so is the positioning of the isobars demarcating regions of high-, medium-, and low-risk consequence. Interesting to note is that this method is analogous to those used to assess projects, discussed in Chapter 19; a quick comparison of Figure 11.4 and Figure 19.5 reveals the similarity. The method is also like Failure Mode and Effect Analysis (FMEA), discussed earlier. Both methods identify and analyze the consequences of risk, although FMEA is aimed specifically at risks in technical systems.



See Chapter 19

PERT

The PERT and Monte-Carlo simulation methods discussed in Chapter 8 can be used to account for risk in project scheduling and to estimate additional time needed to compensate for risks in meeting project deadlines.

The PERT method accounts for risk by using three time estimates for each project activity: a , m , and b (optimistic, most likely, and pessimistic times, respectively). Greater risk in an activity is reflected by a greater spread between a and b , and, more so, between m and b . For an



See Chapter 8

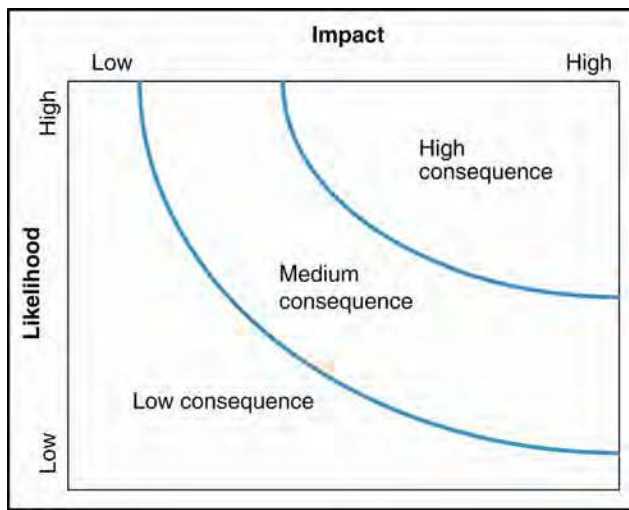


Figure 11.4

Risk consequence as a function likelihood and impact.

activity with no perceived risk, a , m , and b would be identical; any risk hazards identified are accounted for by raising the values of b and m or by moving b farther from m .

With PERT, recall it is the expected time, not m , that is the basis for scheduled times, where the expected time is the mean of the Beta distribution,

$$t_e = \frac{a + 4m + b}{6}$$

Thus, for a particular activity with given optimistic and most-likely values (a and m), using a larger value of b will result in a larger value of t_e . This logically allows more time to complete the activity and compensate for risks. In addition, however, the larger value of b also results in a larger time variance for the activity because

$$V = \left(\frac{b - a}{6} \right)^2$$

This larger V will result in a larger variance for the project completion time, which would spur the cautious project manager to add a time buffer (or schedule reserve) to the project schedule.

Risk priority

Based upon the computed risk consequences, the risk sources can be listed on a **risk register** or risk log, where those with medium-to-high consequences are carefully scrutinized by the project team, managers, and subcontractors who will prepare the appropriate responses to them. Table 11.5 is an example risk register showing rank-ordered risk sources and mitigation responses.

Table 11.5 Risk registry.

Risk ID Number	Risk Source or Condition (Highest Consequence Ranked First)	Functional Area Impacted	Risk Impact {1–5}	Risk Likelihood	Consequence Rating	Effect on Project if Risk Source Materializes	Action to Mitigate or Eliminate Risk
19	Creative robotics software does not meet customer requirements.	Application Development	5	75%	375	Schedule delays. Cost of hiring a replacement vendor.	Test drone software at IBC site; incorporate lessons learned from Godzilla and Mothra projects. Apply K-P standards to all CRC tasks.
6	Unknown site factors encountered during installation of storage racks and computer systems.	Site Operations	4	90%	360	Scheduling delays and possible system reengineering. Increased system/project costs.	Use tracking mechanism to identify issues as they occur. Be ready to shift schedule and resources as necessary.
29	Customer is unable to move operations on time to alternate site during installation.	Customer Relations, Site Operations	5	70%	350	Entire installation delayed.	Assist customer in developing plan/schedule to move to alternate site. Rehearse move one week in advance.
12	Robotic drones perform poorly due to multi-user code employed on a single-user code system.	Application Development	4	75%	300	Increased retrieval/ placement times. Inability to meet contract requirement.	Perform multi-user code testing at IBC site. Ridgeway team on-call for tech support.
4	Hyper-Drive system won't meet installation schedule, which requires uploading SKU data to the new system.	Application Development	4	75%	300	Difficulty in handling some SKUs. Could require a change in upload strategy.	Utilize lessons learned from Godzilla project for Hyper-Drive. Have second Hyper-Drive available on standby.
32	Excessive manual changes during post-allocation process.	Site Operations	4	70%	280	System startup delay.	After initial system test, determine extent of excess work and readjust resources.
3	Conversion of Wildnight to DBA and redesigned application.	Application Development	4	65%	260	If not handled properly, potential error in bucket placement.	Extensive quality assurance process. Dedicate proper staffing. Training early in project.

One drawback with using expected value consequences to prioritize risks is that very low likelihood risks might be ignored even when they have severe or catastrophic impact. Suppose, for example, the impact of a project failure is 1,000 fatalities. If the risk likelihood is infinitesimal, then the expected consequence will be very small (tiny likelihood of many fatalities) and, hence, the risk relegated to a low priority.¹³

In a complex system with many relationships where joint failures in some of them would lead to system failure, it is common to ignore such failures in the hope they will not occur. Usually, the likelihood of joint failure is very low. Very low, however, is different from impossible, and a failure with terrible impact should never be ignored regardless of how small the expected value. For example, the chemical plant accident at Bhopal, India, has been attributed to over 30 separate causes, their joint probability being so small as to be beyond comprehension. Yet they all did happen, causing an accident that resulted in between 1,800 to 10,000 deaths and 100,000 to 200,000 injuries.¹⁴ Similarly, the nuclear accident at Chernobyl was the result of six errors in human action, any one of which, if absent, would have precluded the accident. But despite the minuscule likelihood, all six did happen, resulting in an accident that immediately caused several dozen deaths, several hundred hospitalizations, and 135,000 evacuations, plus later an estimated 5,000 to 24,000 deaths from cancer in the former Soviet Union and many countries throughout Europe and Asia.¹⁵ The lesson: any risk with a severe impact should never be ignored, no matter how small the likelihood.

Among risks threatening projects everywhere are those associated with climate change. Although both the likelihood and impacts of such risks are difficult to assess, most scientists and planners agree that for many projects the likelihoods, impacts, and, hence, consequences of climate change risks will only get larger.

Example 11.4 Assessing Vulnerability to Climate Change

Among the risks imposed by anticipated adverse impacts of climate change (CC) are rise in sea level, frequency and ferocity of storms, average sea and atmospheric temperatures and urban heat islands, droughts, flooding, coastal erosion, wildfires, and landslides. Such impacts threaten the viability of projects and the life cycles of the end-item systems they create. Researchers and planners are grappling with ways to address these impacts. One example is the European Commission's guidelines for managing projects.¹⁶ The guidelines call for (1) assessment of a project's vulnerability and risk due to hazards posed by CC, and (2) adapting the project to increase its "resilience" to the hazards of CC.

Not every project is affected by CC. When there is reason to suspect a project or the life cycle of its end-product might be affected, the project's **vulnerability** to CC hazards should be assessed, where

$$\text{Vulnerability} = \text{Sensitivity} \times \text{Exposure}$$

Sensitivity means "how sensitive to CC hazards is this specific *type of project*?" irrespective of project location. For example, *any* tunneling, subway, or water-spanning bridge project is threatened by floods, regardless of project location. The sensitivity analysis should address the various themes or aspects of the project, such as on-site assets, resources needed, outputs (products and services), and transport links.

For example, Table 11.6 represents a river-spanning bridge project.¹⁷ Since, in general, aspects of such projects (e.g., elevation of the bridge and roads leading to it) depend on water levels, the CC hazard of flooding might be considered a serious threat. CC hazards such as temperature rise and drought would be considered less serious threats.

Table 11.6 Sensitivity of features of river-spanning project to climate hazards.

Sensitivity Table (example)	Climate Variables and Hazards			
	Flood	Heat	...	Drought
On-site assets, ...	High	Medium	...	Low
Transport links	Medium	Low	...	Medium
Highest score (Worst case)	High	Medium	...	Medium

Exposure means “how protected from CC hazards is the *location* of this particular project?” irrespective of the project type. The less the protection, the greater the exposure to CC hazards. For example, *any* project located by a river or in a coastal low-lying plain might be threatened by floods, but less threatened by temperature rise or drought (e.g., coastal cities like London, Miami, Tokyo, Mumbai, Guangzhou, Dhaka, Jakarta, Lagos).

Exposure has two parts: exposure in the *current climate* and exposure in the *future climate* (as predicted by, say, climate models). In many cases, exposure to CC hazards can be expected to worsen over time. Table 11.7 reflects worsening threats to low-lying coastal areas from the hazards of floods, heat rise, and drought.

Table 11.7 Exposure of low-lying coastal area location to climate hazards.

Exposure Table (example)	Climate Variables and Hazards			
	Flood	Heat	...	Drought
Current climate	Medium	Low	...	Low
Future climate	High	Low	...	Medium
Highest score (Worst case), current + future	High	Low	.	Medium

Summarizing the threats in Tables 11.6 and 11.7:

- Flood: highest sensitivity score is High; highest exposure score is High.
- Heat: highest sensitivity score is Medium; highest exposure score is Low.
- Drought: highest sensitivity score is Medium; highest exposure score is Medium.

Vulnerability to specific CC hazards, which is a function of both sensitivity and exposure, can be expressed in a table that combines the two. For example, Table 11.8 shows the vulnerability to CC threats for a river-spanning bridge project located in a low-lying coastal area: the project has high vulnerability to floods, and medium vulnerability to heat rise and drought.

It is important to note that with climate change many things can be expected to change. For example, a project assessed today to have low exposure to a hazard might in

Table 11.8 Vulnerability of project to climate hazards.

Vulnerability Table (example)	Exposure (current + future climate)		
	Low	Medium	Drought
Sensitivity	Low		
	Medium	Heat	Drought
Highest score (Worst case)	High		Flood

20 years have medium exposure and in 100 years high exposure. Therefore, with appropriate forethought, project planners might act today for a current low-risk situation that prevents it from becoming high risk in the future.

The vulnerability assessment raises awareness about the potential impacts of climate change on a project. In response to situations rated as high or medium vulnerability, project planners would instigate a more thorough risk assessment and adopt measures to reduce the project's vulnerability to or increase its resilience against CC threats.

You might have noticed that the risk assessment methods described rely mostly on numbers that are, well, guesses, often with little way of knowing their accuracy. Hence, some argue, that the assessment methods are invalid, and that a better way to assess risk is to start with a *thorough* collection and analysis of data, that is, for each risk to collect data surrounding *everything* about it—where and when it happened, the environment, risk sources, and the outcomes, and decide for the current project whether the risk will or will not materialize. If the former, prepare for the risk; if the latter, ignore it. The argument is valid, although it centers on (1) gathering a lot of data about the risk from previous situations, (2) knowing a lot about the current situation, and (3) comparing the two to predict if the risk will occur in the current situation. AI software trained to identify risks (see Chapter 6) is good at making predictions, although the sticking point is that it also relies on a lot of data, and data that is relevant and usefully formatted. In reality such data is not always available, nor is there time to gather and analyze it. There is no simple way around this. Ultimately, however, a manager must decide whether a risk will or will not occur; and then plan accordingly, as described next.



See Chapter 6

11.4 Risk Response Planning

Risk response planning addresses the matter of how to deal with risk. In general, the ways of dealing with a risk are to transfer, avoid, reduce, escalate, accept, or contingency plan for it.

Transfer risk

Risk can be transferred between customers, contractors, and other parties using insurance, contracts, and contractual incentives.

1. Insurance

The customer or contractor purchases insurance to protect against a wide range of risks, including those associated with:

- Property damage or personal injury suffered from the project.
- Damage to materials while in transit or in storage.
- Breakdown or damage to equipment.
- Theft of equipment and materials.
- Sickness or injury of workers, managers, and staff.
- Fluctuations in exchange rates on imported items, or “forward cover.”

2. Subcontract work

Risk can arise from uncertainty about how to approach a problem or situation. One way to handle such risk is to hire contractors that specialize in specific problems or situations. For example, to minimize the financial risk associated with the capital cost of tooling and equipment for production of a large, complex system, a manufacturer might subcontract the production of the system’s major components to suppliers familiar with them. This relieves the manufacturer of the financial risk associated with the production tooling and equipment. But as mentioned, transferring one kind of risk often means inheriting another. For example, in subcontracting work for the components, the manufacturer now must then rely on outsiders, which increases the risks associated with quality control and scheduling. But such risks often can be reduced through the contract agreement and careful management of the subcontractors.

3. Choice of contract

Risk and contracts are linked since risk can be transferred or allocated to others by using the appropriate kind of contract. This is addressed in Chapter 12, but briefly it works like this: in a **fixed-price** contract, the contractor assumes most of the risk of a cost overrun; in a **cost-plus** contract, the customer assumes most of it. When the statement of work is clear and involves little uncertainty, a contractor will readily accept a fixed price contract because the work is known and unlikely to change. When, however, the scope of the work is unclear and the potential for change is great, the contractor prefers a cost-plus contract, which will cover all expenses incurred in the event of changes. Sometimes the two parties negotiate to reach an agreement that neither finds too risky.

But not all risks can be transferred. Even with a fixed-price contract where ostensibly the contractor assumes the risk of overruns, the customer will nonetheless incur damages and hardship should the project fall behind schedule, or the contractor declare bankruptcy. The project still must be completed, and someone must pay for it. To avoid losses, a contractor might feel pressured to cut corners, which increases the customer’s risk of receiving a sub-par quality end-item.



See Chapter 12

4. Risk responsibility

Risk may be transferred, but never is it completely “offloaded.” Usually, a warranty or guarantee in the contract specifies the time or place at which the risk is transferred from one party to another. For instance, when an item is procured and shipped from abroad, the risk of damage usually remains with the seller while the item is being shipped, but as soon as it is hoisted over the rail of the ship, the risk is transferred to the buyer.

A party willing to accept high responsibility for risk in a project will usually demand a high level of authority over the project. For example, a customer willing to accept the risk of inferior

quality or cost overrun will almost certainly insist on a large measure of control over aspects of the project that influence quality and cost. Parties bearing high-risk will also usually insist on compensation to cover the risks. In cost-plus contracts, for example, the contractor's risk is covered by compensation for all expenses, but the customer's risk is covered by his management oversight of the contractor to prevent expense abuses.

Avoid risk

Risk can be avoided by such measures as increasing supervision, eliminating risky activities, minimizing system complexity, altering end-item quality requirements, changing contractors, and incorporating redundancies. But attempts to avoid risk often entail the addition of innumerable management controls and monitoring systems, which tend to increase system complexity and, perversely, introduce new sources of risk. Risk avoidance measures can also diminish payoff opportunities. Important to keep in mind is that not all risks can be avoided, especially in complex and leading-edge projects. Research and new product development projects are inherently risky but offer potential for huge benefits later. Because the size of the risk is often proportionate to the potential payoff, rather than avoiding risk it is better to try to reduce risk to an acceptable level.

Reduce or mitigate risk

Among the ways to reduce technical risk (its likelihood, impact, or both) are to:¹⁸

- Employ the best technical team.
- Base decisions on models and simulations of key technical parameters.
- Use mature, computer-aided system engineering tools.
- Use parallel development on high-risk tasks.
- Provide the technical team with incentives for success.
- Hire outside specialists for critical review and assessment of work.
- Perform extensive tests and evaluations.
- Minimize system complexity.
- Use design margins.

The latter two points deserve further explanation. In general, risk and uncertainty increase with system complexity: the more elements in a system and the more they are interconnected, the more likely an element or interconnection will go wrong. Thus, minimizing complexity through reorganizing and modifying elements in product design and project tasks reduces risk. For example, **decoupling of activities** and subsystems, i.e., making them *independent* of one another, prevents a failure in one activity or subsystem from spreading to others.

Incorporating **design margins** into design goals is another way to reduce risk associated with meeting technical requirements.¹⁹ A design margin is a quantified value that serves as a safety buffer. In general, a design margin is incorporated into a requirement by setting the target design value to be stiffer or more rigorous than the design requirement. In particular:

$$\text{Target value} = \text{Requirement} + \text{Design margin}$$

By striving to meet a target value that is stiffer than the requirement, the risk of not meeting the requirement is reduced.

Example 11.5: Design Margin Application for the Spaceship

The weight requirement for the spaceship navigation system is 90 lbs maximum. To allow for the difficulty of reaching the requirement (and the risk of not meeting it), the design margin is set at 10 percent or 9 lbs. Thus, the *target weight* for the navigation system becomes 81 lbs.

A design margin is also applied to each subsystem or component within the system. If the navigation system is entirely composed of three major subsystems, A, B, and C, then the three together must weigh 81 pounds. Suppose C is an OTS item with a weight of 1 lb that is fixed and cannot be reduced. But A and B are being newly developed, and their design goals have been set at 50 lbs for A and 30 lbs for B. Suppose a 12 percent design margin is imposed on both subsystems; thus, the *target weights* for A and B are $50 (1.0 - 0.12) = 44$ lbs, and $30 (1.0 - 0.12) = 26.4$ lbs, respectively.

Design margins provide managers and engineers a way to address problems in an evolving design. Should the target value for one subsystem prove impossible to meet, then portions of the margins from other subsystems or the overall system can be reallocated to the subsystem. Suppose subsystem B cannot possibly be designed to meet its 26.4 lb target, but subsystem A *can* be designed to meet *its* target; thus the target for B can be increased by as much as 3.6 lbs (its margin value) to 30 lbs; if that value also proves impossible to meet, the target can be increased by another 6 lbs (subsystem A's original margin value) to 36 lbs. Even if that value cannot be met, the target can be increased again by as much as another 9 lbs (the margin value for the entire system) to 45 lbs. Even with these incremental additions to B's initial target value, the overall system would still meet the 90 lbs weight requirement.

While design margins help reduce the risk of not meeting requirements, they also encourage designers to exceed requirements—e.g., to design systems that weigh less than required. Thus, the margins must be carefully set to reduce the risks while not increasing the costs.

Schedule and cost risk

Design margins focus on risks associated with meeting technical requirements. Among ways to reduce risks associated with meeting *schedules* are:²⁰

- Create a master project schedule and strive to adhere to it.
- Schedule the riskiest tasks as early as possible to allow time for failure recovery.
- Maintain close focus on critical and near-critical activities.
- Put the best workers on time-critical tasks.
- Provide incentives for overtime work.
- Shift high-risk activities in the project network from series to parallel.
- Organize the project early, and staff it adequately.
- Use project and feeding buffers (contingency reserves) in the schedule, as discussed in Chapter 8.



See Chapter 8

To reduce the risk associated with meeting *budget* or cost targets:²¹

- Identify and monitor the key cost drivers.
- Use low-cost means to review and assess design alternatives.
- Verify system design and performance through modeling and assessment.
- Maximize usage of proven technology and commercial off-the-shelf equipment.
- Provide contingency reserves in project budgets.
- Perform early breadboarding, prototyping, and testing on risky components, as discussed in Chapter 10.



See Chapter 10

The last way is especially powerful. **Breadboards** and **prototypes**, i.e., test mockups and models, enable ideas to be tested experimentally so designs can be corrected early in the project.²² This reduces the need for later design changes, which can be costly. The following example illustrates other ways to reduce risk.

Example 11.6: Managing Schedule and Cost Risk at Vancouver Airport²³

The expansion project at Vancouver International Airport involved constructing a new international terminal building (ITB) and a parallel runway. The schedule for the \$355 million project called for full operation of the ITB less than 3.5 years after the project was approved and opening the new runway 5 months after that. The project team identified the following as major risks in meeting the tight budget and schedule constraints:

1. *Risk in structural steel delivery and erection.* Long procurement lead times from steel mills and difficulties in scheduling design, fabrication, and erection make big-steel projects risky. Recognizing this, the project team awarded the structural steel contract very early in the project to allow ample time to design, procure, fabricate, and erect the 10,000 tons of steel required for the ITB. As a result, the ITB was completed on time.
2. *Material handling risk.* Millions of cubic meters (cum) of earth had to be moved, and over 4 million cum of sand were required for concrete runways and taxiways. The project team developed an advance plan to enable coordinated movement of earth from one locale to another and used local sand in the concrete. This saved substantial time and money, enabling the runway to be completed a year ahead of schedule.
3. *Environmental risk.* Excavations and transport of earth and sand by barges threatened the ecology of the Fraser River estuary. These risks were mitigated by advance planning and constantly identifying and handling problems as they arose through cooperative efforts of all stakeholders.
4. *Functionality risk.* Because modern technologies pose risk, the project team adopted a policy of using only proven (OTS) components and technology whenever possible. Consequently, all ITB systems were installed with few problems and were operational according to schedule.

One additional way to reduce the risk of not meeting budgets, schedules, and technical performance is to do whatever necessary to achieve the requirements, *but nothing more*.²⁴ The project team might be aware of many things that could be done beyond the stated requirements, however in most cases these will consume additional resources and add time and cost. Unless the customer approves the added time and cost, all extras should be avoided.

Escalate

When a risk materializes that is beyond the scope, capability, or authority of the project team, contractor, or sponsor to handle, it is “kicked up” to the next authority level for a decision or resolution. Escalate risk and transfer risk are similar strategies, at least initially, since what to do about the risk will be decided by someone else—usually someone at a higher management level who has the authority to grant additional resources, if that is what is needed. The risk might then be sent to another party more capable of handling it, or kicked back to the party that first reported it, but with instructions and additional resources to handle it.

Contingency planning

Contingency planning implies anticipating risks that might arise and then preparing an alternative course of action to cope with them. The initial project plan is followed, and throughout execution the risks are closely monitored. Should a risk materialize as indicated by trigger symptoms, the contingency plan is adopted. The contingency plan can be a post-hoc remedial action to compensate for the risk impact, an action undertaken in parallel with the original plan, or a preventive action initiated by trigger symptoms to mitigate the risk impact. Multiple contingency plans can be developed based upon “what-if” scenarios for multiple risks.

Accept risk (do nothing)

Not all impacts are severe. If the cost of avoiding, reducing, or transferring the risk is estimated to exceed the benefits, then “do nothing” might be the best alternative. In Figure 11.4, the do-nothing strategy would be chosen for risks falling in the “low consequence” region (except when the impact is potentially catastrophic, which is off the chart). Besides, sometimes nothing can be done to avoid, reduce, or transfer a risk, in which case the risk must be accepted, regardless of the consequence.

Responding to a risk sometimes creates a new, *secondary* risk (see Example 12.1, next chapter). When preparing a risk response plan, the project management team should monitor for these before implementing the plan.

11.5 Risk Monitoring and Response

Identified risks are listed in a **risk log** or **risk register** and rank ordered, greatest risk consequence first. For risks with dire consequences, mitigation plans are prepared, and strategies adopted (transfer, reduce, avoid, or contingency); for those with few or no consequences, nothing is done (accept).

Thereafter, the project should be continuously monitored for symptoms of previously identified risks and for newly emerging risks (not previously identified). Known risks may

take a long time before they emerge and start to produce problems. Should a symptom reach the trigger point, a decision is made as to the course of action, which might be to institute a prepared plan or to convene a meeting to find a solution. Sometimes the response is to do nothing; however, everything should be a conscious choice, not an oversight, and be tracked afterward to ensure it was the right choice.

All risks deemed critical or important are tracked throughout the project or the phases to which they apply; to guarantee this, someone should be *assigned responsibility* to track and monitor the symptoms of every important risk.

Altogether, the risk log, mitigation strategies, monitoring methods, people responsible, contingency plans, and schedule and budget reserves constitute the project risk management plan. The plan is continuously updated to account for changes in risk status (old risks avoided, downgraded, or upgraded; existing risks reassessed; new risks added). The project manager (and sometimes other managers and the customer) is alerted about emerging problems; ideally, the project culture embodies candor and honesty, and people readily notify the project manager whenever they detect a known risk materializing or a new one emerging.

11.6 Project Management /s Risk Management

Risk management supplements and is a part of other project management practices such as requirements definition, scheduling, budgeting, change control, and performance tracking and control. Managers use all these practices to identify and assess risks so they can proactively reduce risks or plan for the consequences. If, for example, a project must finish in 9 months but is estimated to take 11, management can take a multitude of steps to increase its likelihood of finishing in 9.

Ideally, risk identification, assessment, and response planning is treated as a formal aspect of project planning, and the resulting risk management plan is integrated into the execution plan alongside the schedule, budget, quality plan, communication plan, and so on. During project execution, risk tracking is incorporated as a measure within the tracking and control process. Ideally, many project team members and other stakeholders are involved in risk identification, response planning, and risk tracking. Whenever someone detects a risk, she contacts the project manager or risk officer (described later); then, depending on the severity of the risk, messages (pop-ups) are sent to others in the project to inform them of the risk.

Of course, not all projects *need* comprehensive risk management. On small projects, a competent, well-paid, and motivated staff can usually overcome difficulties associated with the risks and, if not, the consequences are generally small anyway. In larger projects, however, where the stakes and risks of failure are high, risk management is important. Such projects require awareness and respect for all the significant risks—safety, legal, social, political, technical, and financial.

Risk management principles

Every project for which non-trivial risks are known or suspected should have a **risk management plan**. The plan should specify for a particular project the procedures for identifying and assessing risks, person(s) involved in the risk management process and their responsibilities, methods for assessing and prioritizing risks, guidelines for risk mitigation and contingency planning, and methods for tracking and reporting risks and addressing emergent, unforeseen risks. The plan should address general principles for managing risks, including the following:²⁵

- Appoint a **risk officer** to the project, someone whose principal responsibility is the project's risk management. This should not be the same person as the project manager, and it should not be a can-do person but, instead, a *devil's advocate*, identifying all the reasons why something might not work—even when everyone else believes it will.
- Create a **risk (or threat) profile** for each risk source; this includes the risk likelihood, cost and schedule impact, and contingencies to be invoked. The profile should also specify the earliest visible trigger symptoms that would signify that a risk is materializing. In general, high-risk sources should have lots of eyes watching, and contingency plans should be updated to reflect project progress and emerging risks. Figure 11.5 illustrates a risk profile template, which would include a summary of everything known about a risk. This document would be retained in a library or database, and updated as necessary until the risk is believed to no longer exist and is "closed out."
- Combine all the risk profiles into a **risk log** or **risk register**, as mentioned earlier.
- Include a **risk reserve** in the budget and schedule—a buffer of money, time, and other resources to deal with risks should they materialize. The reserve is used at the project manager's discretion to cover risks not specified by each risk's profile. It may include the RT or RC values (see the Appendix to the chapter) or other amounts. It is usually not associated with a contingency plan, and its use might be constrained to particular applications or areas of risk. The project manager keeps the amounts held in the reserves strictly confidential (else the project will tend to expend whatever the amount is), although others should know a reserve is available (otherwise they will create their own secret reserves).
- Establish **communication channels** (sometimes anonymous) within the project team to ensure any unwelcome news gets to the project manager quickly, risks are continually monitored, and risk status is continuously assessed, and communicated.
- Specify **procedures** to ensure **accurate** and **comprehensive documentation** of proposals, project plans, change requests, progress reports, and the post completion summary report. In general, the better the documentation of past projects, the more information available for planning for future, similar projects and identifying possible risks.
- Periodically convene a **risk review meeting** to assess the status of existing risks, identify new risks, and update the risk register.

Expect the unexpected

Having identified myriad risk hazards and consequences and prepared all kinds of specialized controls and safeguards, people can be led to believe that everything that could go wrong has been anticipated and accounted for; thus, when something still goes wrong, it catches them *completely off guard*. Although it is true that risk planning can cover many or most risks, it can never cover all of them. Thus, risk planning should be tempered with the concept of "non-planning" or Napoleon's approach, which is to *expect that something surely will go wrong* and to be ready to find ways to deal with it as it emerges. This is as important to coping with risk as is extensive planning and believing that all risks have been covered.²⁶

Risk Profile and Management Plan			
Risk Number	Last Update	Originator	Risk Category
Project	Phase	Department	WBS Number
Likelihood	Impact	Consequence	Priority
Risk Assessment			
Risk Description			
Risk Sources			
Risk Assessment			
Strategy:		Risk Plan	
☐ Accept ☐ Avoid ☐ Contingency ☐ Reduce ☐ Reserves ☐ Transfer		1. 2. 3. 4. 5. 6. 7.	
Risk Tracking			
Member Responsible		Risk Officer	
Measures/Symptoms		Comments	
Trigger Event		Comments	
Sign-offs			
Cost Engineer	System Engineer	Quality Manager	Project Manager
Date:	Date:	Date:	Date:

Figure 11.5
Document for the risk profile and management plan of an identified risk.

Example 11.7: Managing Risks as They Arise—Development of the F117 Stealth Fighter ²⁷

An example of how to manage risk in R&D projects is the F117 Stealth Fighter program, aimed at developing a revolutionary new “low observable” (difficult to detect with radar) aircraft capable of high-precision attacks on enemy targets. The F117 involved high levels of risk because many lessons had to be learned during the program and significant challenges had to be overcome. But managers *expected* that challenges would occur throughout the program, from early design and testing, through evaluation and final deployment. To handle the risks, numerous decisions were made on the spot between program managers at Lockheed-Martin (contractor) and the US Air Force (customer). The program was set up for rapid deployment of resources to solve problems *as they arose*. Customer and contractor managers worked closely to minimize bureaucratic delays. Schedules were optimistic and based on assumptions that everything would work, but everyone throughout the management chain *expected the risks* and the challenges to overcome, so problems never came as a surprise or threatened program support—a good example of *managing* risk as opposed to *avoiding* risk.

Risk management caveats

For all the good it can provide, risk management itself *creates* risks. Most every philosophy, procedure, or prescription has caveats, and that is true of risk management as well. Misunderstanding or misapplication of risk management concepts can stymie a project by fooling people into thinking they have nothing to worry about, which can leave them worse prepared for dealing with emerging problems they didn’t anticipate.

Having created a risk management plan, managers might be emboldened to take risks they otherwise might not take. Much of the input to risk analysis is subjective; it addresses what might happen—not what will happen. Data analysis and planning gives people a sense of having power over events, even when the events are chancy. Underestimating the risk likelihood or impact can make consequences seem insignificant, leading some people to venture into dangerous territory that common sense would disallow. For example, the security of seat belts and air bags encourages some drivers to take risks such as driving too close behind the next car or accelerating through yellow lights. The result is an *increased* likelihood of an accident.

Repeated experience and good documentation are vital ways to identify risks, but they cannot guarantee that all important risks will be identified. Same and similar outcomes that have occurred repeatedly in past projects eventually deplete people’s capacity to imagine anything else happening. As a result, some risks become unthinkable. Even sophisticated computer models are worthless when it comes to dealing with the unthinkable because a computer model cannot analyze situations that have not occurred and are beyond human imagination. Experience provides only a sample of possibilities, not the entire population.

Managing risk does not mean eliminating it, although some managers don’t know that. The prime symptom of “trying to eliminate risk” is micromanagement: excessive controls and documentation requirements, and trivial demands for the authorization of everything. Projects inherently entail uncertainty and risk. Micromanagement is seldom appropriate and for some projects it can be ruinous, particularly when the projects involve the new, untried, and

untested. When management tries to eliminate risk, it stifles innovation and, say Aronstein and Piccirillo, “forces a company into a plodding, brute force approach to technology, which can be far more costly in the long run than a more adventurous approach where some programs fail but others make significant leaps forward.”²⁸ The appropriate risk management strategy for most projects is to try to accommodate and mitigate risk, not to avoid or eliminate it.

11.7 Summary

Project risk management involves identifying risks, assessing them, and planning appropriate responses. Identifying project risks starts in the project conception phase. Project risks stem from many sources such as failure to define and satisfy customer needs and market requirements; technical problems arising in the work; weather, labor, and supplier problems; competitors’ actions; and changes imposed by outsiders. Such risk threats are identified using a variety of methods that draw from experience with past projects and scrutiny of future projects.

Of innumerable risks in projects, only the important ones need to be addressed. Importance depends on the likelihood, impact, and overall consequence of the risk. Likelihood is the probability a risk will occur, impact is the risk’s effect, and risk consequence is a combination of the two. In general, measures of risk consequence are used to decide which risks should receive attention and which can be ignored. As a precaution, however, every risk with severe impact should be carefully considered, even when the likelihood is small.

Risk response planning addresses ways the identified risks will be handled. Some risks can be transferred or escalated to other parties or spread among many stakeholders or sub-contractors. Some can be avoided; some can be eliminated. Sometimes, however, high risk is associated with high benefits, and trying to eliminate the risk can also reduce the payoff. Thus, better than trying to eliminate risk is to try to reduce it to a manageable level. For areas of high risk, alternative contingency plans should be developed.

The principles for risk management include creating a risk management plan that specifies the risks, their symptoms and backup plans, a risk officer responsible for identifying and tracking them, and budget and schedule reserves. The plan must specify the ways to monitor risks and emerging problems, and to communicate them to the project manager. Proper documentation from past projects furnishes lessons learned and forewarns managers about potential risks in upcoming projects. No amount of preparation can anticipate all risks; thus, managers should expect the unexpected and be ready to deal with risks as they arise.

The following Appendix discusses common analytical methods for assessing risk consequences and deciding between alternative risk responses. Similar methods are employed in project selection—the topic of Chapter 19.



See Chapter 19

APPENDIX: RISK ANALYSIS METHODS

Four common risk analysis methods are expected value, decision trees, payoff tables, and simulation.

Expected value

Selection of the appropriate risk response sometimes depends on the risk consequences expressed in terms of the **expected value** of costs or schedules.

An expected value is the average outcome of an occurrence repeated numerous times. For risk assessment, expected value represents the average outcome of a project if it were repeated many times. Mathematically it is the weighted average of all the possible outcomes, where the weights are the likelihoods of the possible outcomes, that is

$$\text{Expected value} = \sum[(\text{Outcomes}) \times (\text{Likelihoods})]$$

The consequence of risk on project duration, called the **risk time, RT**, is the expected value of the estimated time to correct for risk, computed as

$$\text{RT} = (\text{Corrective time}) \times (\text{Likelihood}) \quad (11.6)$$

The consequence of risk on project cost, called the **risk cost, RC**, is the expected value of the estimated cost to correct for the risk, computed as

$$\text{RC} = (\text{Corrective cost}) \times (\text{Likelihood}) \quad (11.7)$$

For example, suppose the baseline time estimate (BTE) for project completion is 26 weeks and the baseline cost estimate (BCE) is \$71,000. Assume that the risk likelihood for the project is 0.3, and, if it materializes, would delay the project by 5 weeks and cost \$10,000 more. Because the probability of the risk materializing is 0.3, the probability of it not materializing is 0.7. If the risk does not materialize, no corrective measures will be necessary, and the corrective time and cost will be zero. Hence

$$\text{RT} = (5)(0.3) + (0)(0.7) = 1.5 \text{ weeks,}$$

and

$$\text{RC} = (\$10,000)(0.3) + (0)(0.7) = \$3,000$$

These figures, RT, and RC, are the **schedule reserve** and **budget reserve** (project contingency), respectively, mentioned in Chapters 8 and 9. Accounting for the risk time, the **expected project completion time, ET**, is

$$\text{ET} = \text{BTE} + \text{RT} = 26 + 1.5 = 27.5 \text{ weeks}$$

Accounting for the risk cost, the **expected project completion cost, EC**, is

$$\text{EC} = \text{BCE} + \text{RC} = 71,000 + 3,000 = \$74,000$$

When the corrective time and cost cannot be estimated, then ET and EC are computed as

$$\text{ET} = \text{BTE}(1 + \text{likelihood}) = 2.6(1.3) = 33.8 \text{ weeks} \quad (11.8)$$

$$\text{EC} = \text{BCE}(1 + \text{likelihood}) = \$71,000(1.3) = \$92,300 \quad (11.9)$$



See Chapter 8
and 9

These examples account for risk factors affecting the entire project. Another way to determine risk consequence is to first disaggregate the project into work packages or phases and then to estimate for each work package or phase the risk likelihood and corrective time and cost. These individual estimates are then aggregated to determine ET and EC for the entire project. This approach tends to give more credible RT and RC estimates than do equations (11.6) through (11.9) because risks pinpointed to individual tasks or phases can be more accurately assessed, and the necessary corrective actions and associated time and costs more easily identified.

Say a project has eight work packages; Table 11.9 lists cost information and EC for each, where EC is computed as

$$EC = BCE + [(Corrective\ cost) \times (Likelihood)]$$

As shown in Table 11.9, the EC for the project is \$84,850.

Now, for the same eight work package project, Table 11.10 gives time information, where ET is computed as

$$ET = BTE + [(Corrective\ time) \times (Likelihood)]$$

Table 11.9 EC computation.

WBS Element	BCE (\$)	Corrective Cost (\$)	Likelihood	EC (\$)
J	10,000	2,000	.2	10,400
M	8,000	1,000	.3	8,300
V	16,000	4,000	.1	16,400
Y	10,000	6,000	.2	11,200
L	8,000	2,000	.3	8,600
Q	9,000	2,000	.1	9,200
W	5,000	1,000	.3	5,450
X	5,000	1,500	.3	5,750
Total		71,000		84,850

Table 11.10 ET computation.

WBS Element	BTE	Corrective Time	Likelihood	ET
J	6	1	.2	6.2
M	4	1	.3	4.3
V	6	2	.1	6.2
Y	8	3	.2	8.6
L	2	1	.3	2.3
Q	8	1	.1	8.1
W	1	1	.3	1.3
X	1	1	.3	1.3

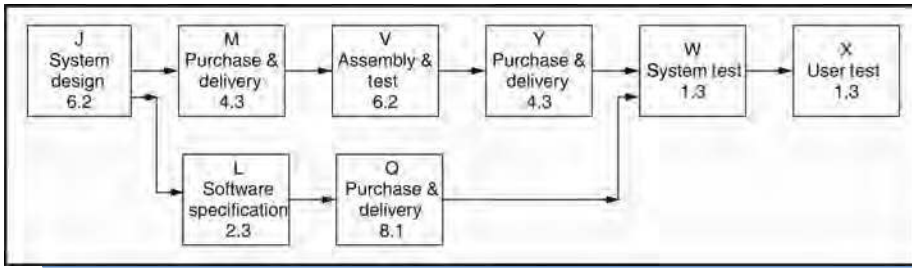


Figure 11.6
Project network; activity durations accounting for risk time.

Suppose the project network is as shown in Figure 11.6. Ignoring the risk time, the critical path would be J–M–V–Y–W–X and the project BTE 26 weeks. Accounting for risk consequences, the critical path be the same but the duration (the project ET) would increase to 27.9 weeks.²⁹

Although activities on critical and near-critical paths should be monitored, in general, every activity with high-risk consequences (high likelihood and/or high impact) should be monitored, even when not on the critical path.

Increasing the project schedule and budget to account for the expected risk time or risk cost is no guarantee of adequate protection against risk. The expected risk time and cost are equivalent to long-run averages, which result from repeating something many times; this is questionable in projects, since seldom are project activities identically repeated.

Decision trees³⁰

A decision tree is a diagram wherein the tree “branches” represent different chance outcomes. It is used to assess which risk response among alternatives yields the best-expected outcome.

One application of decision trees is to weigh the cost of project failure against the benefit of project success. Assume a project has a BCE of \$200,000, a failure likelihood of 0.25, and, if successful, will yield a net profit of \$1,000,000.

The expected value concept can be used to compute the average value of the project. Assuming the project could be repeated many times, then it would lose \$200,000 (BCE) 25 percent of the time and generate \$1,000,000 profit the other 75 percent. Thus, the expected outcome would be

$$\text{Expected outcome} = (-\$200,000)(0.25) + (\$1,000,000)(0.75) = \$700,000.$$

This suggests that although there is potential to net \$1,000,000, it is more reasonable to use \$700,000 for the BCE. It also suggests that all project costs plus actions taken to reduce or eliminate the failure risk should not exceed \$700,000.

Another application of decision trees is in deciding between alternative risk responses. Suppose a project has a BCE of \$10 million, risk failure likelihood of 0.6, and a risk impact of \$5 million. Two strategies are being considered to reduce the risk likelihood (but not the risk impact):

Strategy 1 will cost \$2 million and will reduce the likelihood of failure to 0.1.

Strategy 2 will cost \$1 million and will reduce the likelihood of failure to 0.4.

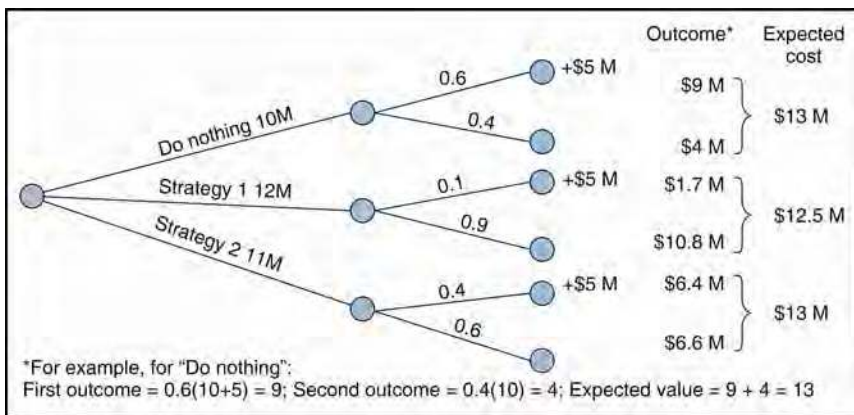


Figure 11.7
Decision tree.



The decision tree and resultant expected project costs are shown in Figure 11.7. The analysis suggests Strategy 1 should be adopted because it has the lowest expected cost.

Another application of decision tree analysis is the *expected commercial value* method used in See Chapter 19 project selection, discussed in Chapter 19.

Uncertainty and payoff tables

When there is no prior experience or historical data upon which to estimate the likelihood, then the expected-value risk consequence cannot be computed, and other criteria must be used to assess courses of action in the face of risk. This situation is referred to as **uncertainty**, which implies no information is available about the probabilities of what might occur. To determine the best strategy under uncertainty, begin by identifying feasible alternative paths the project could take in response to factors over which management has no control. These different paths are called **states of nature**. Consider different possible strategies or actions, and then the likely outcome for each state of nature. The outcomes for different combinations of strategies and states of nature are represented in a **payoff table**.

For example, suppose the success of a project to develop Product X depends on market demand, which is known to be a function of performance features of the product. The development effort can be directed in any of three directions, called Strategies A, B, and C, which will provide the product with three different performance features. Assume also that a competing firm is developing a product that will have performance features similar to those under Strategy A. When the product development effort ends, one of three future states of nature will exist: N1—no competing product enters the market for at least 6 months; N2—the competing product enters the market within 6 months of Product X; N3—the competing product is introduced before Product X. Suppose the likely profits in millions of dollars for the different combinations of strategies and states of nature are computed (the payoff table in Table 11.11).

The question: Which strategy should be adopted? The answer: It all depends! If project sponsors are *optimistic*, they will choose the strategy that maximizes the potential payoff. The maximum potential payoff in the table is \$90 million, which happens for Strategy C and state

Table 11.11 Payoff table.

Strategy	States of Nature		
	N1 (\$M)	N2 (\$M)	N3 (\$M)
A	60	30	-20
B	60	50	60
C	90	70	40

of nature N1. In general, the strategy choice that has the potential to yield the largest payoff is called the **maximax** decision criterion.

Now, if project sponsors are *pessimistic*, they will instead be interested in minimizing their potential losses, in which case they will adopt the strategy that gives the best outcome under the worst possible conditions. For the three strategies A, B, and C, the worst-case payoff scenarios are -\$20 million, \$50 million, and \$40 million, respectively. The best (least bad) of the three is \$50 million, or Strategy B. In general, the strategy that gives the best outcome out of several worst-case scenarios is called the **maximin** decision criterion.

Any choice of strategy other than the best will cause the decision maker to experience an opportunity loss or **regret**. This way of thinking suggests another criterion for choosing between strategies, the **minimax** decision criteria, which is the strategy that minimizes the regret of not having chosen the best strategy. Regret for a given state of nature is the difference in the outcomes between the best strategy and any other strategy. This is illustrated in a **regret table**, shown in Table 11.12. For example, given the payoffs in Table 11.11, for state of nature (N1) the highest payoff is \$90 million. Had Strategy C, the optimal strategy, been selected, the regret would have been zero, but had strategies A or B been selected instead, the regrets would have been \$30 million each (the difference between their outcomes, \$60 million, and the optimum, \$90 million). The regret amounts for states of nature N_2 and N_3 are determined in similar fashion.

To understand how to minimize regret, first look in the regret table at the largest regret for each strategy. The largest regrets are \$80 million, \$30 million, and \$20 million for strategies A, B, and C, respectively. Next, to minimize regret, pick the smallest of these, \$20 million, which is for Strategy C. Thus, Strategy C is the best choice to minimize regret.

Another strategy selection approach is to assume that every state of nature has the same likelihood of occurring. This is called the **maximum expected payoff** decision criterion. Referring to the payoff table, Table 11.11, assume the likelihood of each state of nature is one-third, thus, the expected payoff for Strategy A given outcomes from the payoff table is

$$1/3(60) + 1/3(30) + 1/3(-20) = 23.33, \text{ or } \$23.33 \text{ million}$$

Table 11.12 Regret table.

Strategy	States of Nature		
	N1 (\$M)	N2 (\$M)	N3 (\$M)
A	30	40	80
B	30	20	0
C	0	0	20

The expected payoffs for strategies B and C, computed similarly, are \$56.66 million and \$66.66 million, respectively. Thus, Strategy C would be chosen as giving the maximum expected payoff. Notice in the previous examples that three of the four selection criteria pointed to Strategy C. This might convince decision makers that Strategy C is most appropriate.

Simulation



See Chapter 8

Application of simulation to project scheduling, illustrated in Chapter 8, gives a probability distribution for possible outcomes, which can be used to determine the probability (or likelihood) of a particular outcome such as completion cost or time. In turn, this can be used to establish an appropriate target budget or completion date, or to prepare contingency plans. For instance, although the critical path in Chapter 8, example 8.2, indicated the project would be completed in 147 days, the simulated completion time distribution in Figure 8.14 indicated that it would be 155 days, *on average*. Thus, at the *earliest*, the target completion should be set at 155 days, although the likelihood of *not meeting* that date would be 50 percent. Using the simulated probability distribution, a target completion date can be set such that the likelihood of not meeting it is more acceptable. Alternatively, given a pre-specified project target completion date, simulation can be used to estimate the likelihood of not meeting it and, hence, to decide whether to prepare contingency plans or change the project requirements, activities, or network.



Review Questions and Problems

1. Should risks that have low likelihood be ignored? Explain.
2. How does a person's risk tolerance affect whether he rates a risk high, medium, or low?
3. What is meant by risk of failure?
4. What factors make a project high-risk?
5. Discuss the difference between internal risk and external risk. List sources of risk in each of these categories.
6. Describe each of the following sources of technical risk: maturity, complexity, quality, and concurrency or dependency.
7. Briefly describe the following risk identification techniques: analogy, checklists, WBS analysis, process flowcharts, and brainstorming.
8. Describe a cause-and-effect diagram. Pick a problem (effect) of your own choice and use a cause-and-effect diagram to illustrate it.
9. A project requires developing a system with state-of-the-art hardware and software, both complex, and where system performance depends on another, external system that is being developed concurrently. Based on Table 11.3, and assuming all risk factors are independent and equally weighted, what is the CLF for the project?
10. What is an influence diagram? How is it used to identify and analyze risk sources and to assign priorities to those sources?
11. Tables 11.3 and 11.4 are for illustration purposes. Discuss the general applicability of these tables to rating risks in projects. Would you use these tables to assess the risk likelihood and impact in a project of your choice? Why or why not?
12. Are equations (11.1), (11.2), and (11.3) ways for rating the overall likelihood, impact, and consequences of risk? Discuss pros and cons of using these equations.

13. Briefly discuss each of the following ways to handle risk: transfer risk, avoid risk, reduce risk, contingency plan, escalate, and accept risk.
14. Think of a project you are familiar with and problems it encountered. List some ways the problems could have been avoided and explain each of them.
15. What is a design margin? How does its application reduce risk?
16. One requirement of a power-generating system states that it must provide 500 kwh minimum output. The system has three power-generating subsystems, X, Y, and Z. Constraints on physical size indicate that the output capacity of the overall system will be split among the three subsystems in the approximate ratio of 5:3:2. A 3 percent design margin is applied to the system and the subsystems. Note, because the power requirement is stated as *minimum* output, the target output should be 3 percent *above* the requirement.
 - a. What is the target requirement output for the overall system?
 - b. What are the target requirement outputs for each of the subsystems? (Remember, subsystem margins are *in addition* to the system margin.)
 - c. Suppose that, at best, Subsystem X can be designed to meet only 47 percent of the power output requirement for the overall system. Assuming Subsystems Y and Z can be designed to meet their design targets, can the overall system's output requirement also be met?
17. List and review the principles of risk management.
18. How does risk planning serve to increase risk-taking behavior?
19. Risk management includes being prepared for the unexpected. Explain.
20. Can risk be eliminated from projects? Should management try to eliminate it?
21. How and where are risk time and risk cost considerations used in project planning?
22. Where would the maximax, maximin, and minimax regret criteria be used during the project life cycle to manage project risk?
23. Figure 11.8 below is the network for the Largesse Hydro Project:

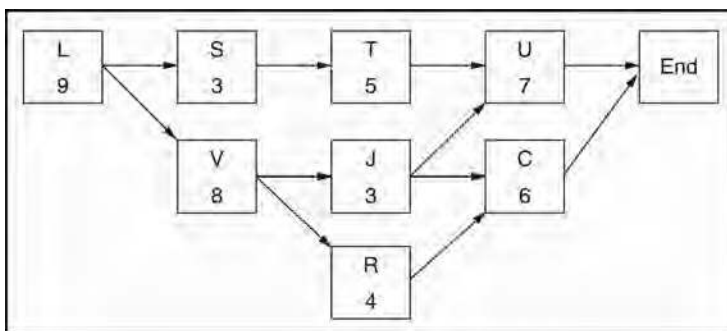


Figure 11.8
Network for Largesse Hydro project.

The table below gives the baseline cost and time estimates (BCE and BTE), the cost and time estimates to correct for failure, and the likelihood of failure for each work package.

WBS Element	BCE	BTE (wk)	Corrective		
			Cost	Time	Likelihood
L	\$20,000	9	\$4,000	2	.2
V	\$16,000	8	\$4,000	2	.3
T	\$32,000	5	\$8,000	2	.1
U	\$20,000	7	\$12,000	3	.2
S	\$16,000	3	\$4,000	1	.3
J	\$18,000	3	\$4,000	1	.1
R	\$10,000	4	\$4,000	3	.3
C	\$15,000	6	\$5,000	2	.3

- a. Determine the risk time and risk cost for all the WBS elements of the project.
 - b. Consider the risk times on noncritical paths. Which activities and paths should be watched carefully as posing the highest risks?
 - c. What is the project expected cost (EC) and expected time (ET)?
24. The geographical location of the Largesse Hydro project threatens it with weather-associated delays and costs. The bad weather likelihood is estimated at 0.30 with a potential impact of delaying work by 10 weeks and increasing the cost by \$20,000.
 - a. Ignoring the time and cost risks in Problem 23, what are the expected project completion time and completion cost considering the weather risk?
 - b. What are the estimated expected project completion time and cost considering the weather risk and the risks listed in Problem 23?
25. Softside Systems has a \$100,000 fixed price contract for the installation of a new application system. The project is expected to take 5 weeks and cost \$50,000. Experience with similar projects suggests a 0.30 likelihood that the project will encounter problems that could delay it by as much as 3 weeks and increase the cost by \$30,000. By increasing the project staff by 20 percent for an additional cost of \$10,000 the likelihood of problems would be reduced to 0.10, and the delay and cost to 1 week and \$8,000, respectively. Use a decision tree to decide whether Softside should increase the size of the project staff.
26. Corecast Contractors has been requested by a municipality to submit a proposal bid for a parking garage contract. In the past the cost of preparing bids was about 2 percent of the cost of the job. Corecast project manager Bradford Pitts is considering three possible bids: cost plus 10 percent, cost plus 20 percent, and cost plus 30 percent. Of course, increasing the “plus percent” will increase the project price and decrease the likelihood of winning the job. Bradford estimates the likelihood of winning the job as follows:

	Bid price	P(win)	P(lose)
P1	$C + 0.1C = 1.1C$	0.6	0.4
P2	$C + 0.2C = 1.2C$	0.4	0.6
P3	$C + 0.3C = 1.3C$	0.2	0.8

In all cases, the profit (if the bid is won) will be the bid price minus the proposal preparation cost, or $0.02C$; the loss (if bid is not won) will be the proposal

preparation cost. Prepare a decision tree for the three options. If Bradford uses the maximum expected profit as the criterion, which bid proposal would he select?

27. Iron Butterfly, Inc. submits proposals in response to RFPs and faces three possible outcomes: N1, Iron Butterfly gets a full contract; N2, it gets a partial contract (job is shared with other contractors); N3, it gets no contract. The company is currently assessing three RFPs, coded P1, P2, and P3. For P3 the customer will pay a fixed amount for proposal preparation; for P1 and P2 Iron Butterfly must absorb the proposal-preparation costs, which are expected to be high. Based upon project revenues and proposal-preparation costs, the expected profits (\$ thousands) are as shown:

	N1	N2	N3
P1	500	200	-300
P2	300	100	-100
P3	100	50	25

To which RFPs would Iron Butterfly respond using the three decision criteria?

28. Frank Wesley, project manager for the LOGON project, is concerned about the development time for the robotic transporter. Although the subcontractor, Creative Robotics, has promised a delivery time of 6 weeks, Frank knows that the actual delivery time will be a function of the number of other projects Creative Robotics is working on, i.e., the workload. As incentive to speed up delivery of the transporter, Frank has three options:

S1: Do nothing.

S2: Promise Creative Robotics a future contract with Iron Butterfly.

S3: Threaten to never contract with Creative Robotics again.

He estimates the impact of these actions on delivery time would be as follows:

Delivery Times Strategy	Creative Robotics Workload		
	Slow	Average	Busy
S1	4	6	8
S2	3	4	7
S3	3	6	6

What strategy should Frank adopt based upon uncertainty criteria? Use criteria like maximax, maximin, minimax regret, and maximum expected payoff, except note that the criteria must be adapted because here the goal is to minimize the payoff (time); this contrasts with the usual case, which is to maximize the payoff.



Questions About the Study Project

1. What did managers and stakeholders believe were the major risks in the project?
2. In your own judgment, was this a risky project? Why or why not?
3. Was formal risk analysis performed? When was it done (in initiation, feasibility, etc.)?
4. Was a formal risk management plan created? Discuss the plan.

5. Was there a risk officer? Discuss her duties and role in the project.
6. How were the risks identified?
7. How were risks dealt with (through risk transfer, acceptance, avoidance, reduction, etc.)?
8. Was a risk register used? Was it ever updated? If so, how often?
9. Discuss the use of contingency plans and budget and schedule reserves to cover risks.
10. What risks materialized during the project and how were they handled?

CASE 11.1 THE SYDNEY OPERA HOUSE³¹

The Sydney Opera House (SOH) is a top tourist attraction and landmark for Sydney and all of Australia. It is a major arts center, although owing to its design, it is not necessarily the best place to hear opera. The SOH is visually spectacular and has a magnificent structure (Figure 11.9), but it was nightmare to design and build.



Figure 11.9 Sydney Opera House.

Source: hatman12 / iStock/Getty Images ®

The original concept for the SOH was a sketch submitted by Danish architect, Jorn Utzon. Judges selected it from an open competition that ended with 233 entries from 11 countries. Though happy to win, Utzon was mildly shocked. The concept that had caught the judges' attention consisted only of simple sketches, with no plans or even perspective drawings. Utzon faced the challenge of converting the sketches into a design from which a structure could be built, but he had no experience in designing and constructing

such a large building. Because there were no plans, detailed drawings, or estimates of needed materials, there was little from which to base cost estimates. No one knew how it would be built; and some experts questioned whether it could be built at all. Interestingly, because the design was so unique, some people thought it would be inexpensive to build. The initial cost estimate was \$7 million, to be paid by the government through profits from a series of state-run lotteries.

Engineers reviewing the concept noted that the roof shells were much larger and wider than any shells ever built. Further, because they stuck up so high, they would act like sails in the fierce winds blowing up the harbor. Thus, they would have to be carefully designed and constructed to prevent the building from blowing away!

Government managers worried that people scrutinizing the design might raise questions about potential problems and stall the project. They thus quickly moved ahead, which seems common in big, risky projects (**megaprojects**—see Chapter 20), and divided the work into three main contracts: foundation and building except roof, the roof, and interior and equipment.

As experts had warned, the SOH project became an engineering and financial debacle, lasting 15 years and costing \$107 million (\$100 million over the initial estimate). Hindsight is 20/20, yet from the beginning this should have been viewed as a risky project. Nonetheless, risks were downplayed or ignored, and little was done to mitigate or control them.

QUESTIONS

1. Identify the obvious risks.
2. What early actions should have been taken to reduce the risks?
3. Discuss some principles of risk management that were ignored.



See Chapter 20

CASE 11.2 INFINITY & BEYOND, INC.

Infinity & Beyond, Inc. produces high-tech fashion merchandise. The company's marketing department has identified a new product concept through discussions with customer focus groups. The department is excited about the new concept and presents it to top management, who approve it for further study. Lisa Denney, senior director of product development, is asked to create a plan and cost breakdown for the development, manufacture, and distribution of the product. Despite the enthusiasm of the marketing department, Lisa is unsure about the product's market potential and the company's ability to develop it at a reasonable cost. To her way of thinking, the market seems ill-defined, the product goals unclear, and the product and its production technology uncertain. Lisa asks her chief designer to create some product requirements and a rough design that would meet the requirements, and to propose how the product might be manufactured.

A few weeks later the designer reports back with requirements that seem to satisfy the marketing concept. She tells Lisa that because of the newness of the technology and the complexity of the product design, the company does not have the experience to develop or manufacture the product on its own. Lisa checks out several design/development firms, and asks one, Margo-Spinner Works Company, MSW, to review the product

concept. MSW assures Lisa that although the technology is new, it is well within their capability. Lisa reports this to senior management, who tells her to go ahead with the development project.

Lisa sets a fixed-price contract with MSW and gives them primary responsibility for the development effort. MSW management had argued for a cost-plus contract, but when Lisa stipulated that the agreement had to be fixed price, MSW said okay, only under the condition that it, MSW, be given complete control of the development work. Lisa feels uncomfortable with the proposition but knows no other design company is qualified to do the work, so she agrees.

QUESTIONS

1. Discuss the major sources of risk in this project.
2. What do you think about Lisa's handling of the project so far? Would you have done anything differently?
3. Discuss what Lisa and the other parties did that increased or decreased the project risks.

CASE 11.3 THE NELSON MANDELA BRIDGE³²

Newtown, South Africa, is a suburb of Johannesburg that boasts a rich cultural heritage. As part of an attempt to help rejuvenate Newtown, the Nelson Mandela Bridge was constructed to link it to important roads and centers of commerce in Johannesburg. Spanning 42 electrified railway lines, the bridge (Figure 11.10) has been acclaimed for its functionality and beauty.

The project was fraught with risks. Lack of space for the support pylons (towers) between the railway lines had dictated that the bridge would have a long span. This resulted in a structure where the bridge deck was supported by stay cables from pylons of unequal height: pylons on the northern side were 48 meters high, those on the southern side were 35 meters high.

The pylons are composite columns consisting of steel tubes that had to be filled with concrete after being hoisted into the vertical position. The decision was made to pump concrete into the tubes through a port at the bottom of each tube. This had to be done in a single operation. Although the technology for casting concrete this way was not new, the columns were the highest in South Africa and filling them would set a world record for bottom-up pumping of self-curing concrete.

The pump for the concrete was placed at ground level between the electrified railway lines, which exposed workers to the risks of working right next to continuous rail operations. The pumping method posed the risk of the stone aggregate and cement in the concrete mixture separating in the pylon tubes before the concrete solidified, which would compromise the strength of the concrete. Another risk was that the pump might fail and result in the concrete solidifying in an uncompleted pylon, rendering further pumping of concrete from the bottom impossible. Two contingencies were considered: an additional pump on standby and completing the process by pouring concrete from the top of the pylon.



Figure 11.10
Nelson Mandela Bridge, Johannesburg.

Source: THEGIFT777 / iStock/Getty Images ®

The concrete had to be transported by trucks to the site, which risked interrupting the concrete supply because of heavy traffic congestion in the city.

Despite working over a busy rail yard with trains running back and forth, no serious accidents occurred at any time in the 420,000 labor-hour project. The pump never failed, and the project finished on time. The stay cables—81,000 meters in length—were installed and the bridge deck lifted off temporary supports, all while the electrified railway lines beneath remained alive. Upon the bridge's completion, some felt the costs incurred to reduce the risks had been excessive; others felt that the risks were too high and not enough was done to reduce them.

QUESTIONS

1. How would you have identified the risks (using methods from this chapter and Chapter 9)?
2. Using the table below, discuss how the risks were addressed or how they could have been addressed. Include any additional risks you can think of.
3. State whether the risks listed in the table are internal or external.
4. Describe how you would determine the expected values of the risks listed in the table.
5. Compile a list of information that you would require to assess the risk of a pump failure.
6. What information do you think would have been available early in the project and from where would you obtain it?

7. Draw a CE diagram showing several factors that could contribute to delaying the project.
8. Describe how risks are reduced over the lifespan of a project such as this one.
9. With reference to the concerns expressed upon completion of the construction, discuss the statement: "Risks always relate to the future. There is no such thing as a ***past risk***."
10. Discuss the difference between good decisions and good luck.
How could a manager protect himself against the risk of making a decision that might later have negative implications?

Possible Risk Event	Plans to Address Risk				
	Accept	Avoid	Reduce	Transfer Escalate	Contingency (Plans or Reserves)
Failure to make an acceptable profit					
Not finishing construction by Nelson Mandela's 85th birthday					
Interference with rail activities					
Geology, requiring expensive foundations					
Concrete mixture segregating when pumped into the columns					
Pump failure while concrete is being pumped					
Interrupted supply of concrete due to trucks transporting concrete delayed in traffic					

CASE 11.4 SUNRISE BEACH DEVELOPMENT

A developer purchased a plot of land located in a prime real-estate district near the downtown of a large Florida city. The developer thinks the site, currently occupied by a long-closed factory, is ideal for two 10-to-20-story buildings to contain hotels, retailers,

and condominiums. Located by the Atlantic Ocean, the site, to be called Sunrise Beach, enjoys a large stretch of beach that, the developer speculates, would draw numerous tourists as well as affluent professionals who work in the nearby city. A quarter of the land is marshy area occupied mostly by wildlife; the developer is considering retaining half of it as a reserve and filling in the other half for a third high-rise building. To lure tourists, the developer is also considering mounting a small roller-coaster atop one of the buildings, similar to an attraction in Las Vegas. Automobile parking would be provided under the buildings to maximize street-level retail space.

The site is on a low-lying sliver of land separated from the mainland by a river that is spanned by a two-lane bridge, built 70 years earlier for factory workers. The developer plans to widen the bridge to handle increased traffic, but otherwise leave the bridge as is.

In the last five years the region was hit by three hurricanes and the city suffered storm-surge flooding. Recently, some streets flood at high tide, even on sunny days.

QUESTIONS

Consider the following threats posed by climate change: rise in sea level, frequency and ferocity of storms, average atmospheric temperatures and urban heat islands, droughts, flooding, and coastal erosion. Given characteristics of the *project* (building size and use, parking, beach, roller coaster, bridge), to which threats is the project most sensitive? Then, consider aspects of the *project location* (coastal, low-lying, southeastern US). To which threats is the project most exposed? Consider both the current and future climates and assume threats will worsen. Use tables like Tables 11.6, 11.7, and 11.8 to rate the project's sensitivity, exposure, and vulnerability to the threats.

Based on your assessment would you recommend going forward with the project as currently planned? If not, what changes would you suggest to increase the project's resilience to the threats posed by climate change?

Notes

1. Asked once to define certainty, John Von Neumann, the principle theorist of mathematical models of uncertainty, answered with an example: to design a house so it is *certain* the living room floor never gives way, "calculate the weight of a grand piano with six men huddling over it to sing, triple the weight, then design the floor to hold it. That will guarantee certainty!" Source: Bernstein, *Against the Gods*, p. 233.
2. See Argus R. and Gunderson N. *Planning, Performing, and Controlling Projects*. Upper Saddle River, NJ: Prentice Hall; 1997, pp. 22–23.
3. Adapted from Michaels J. *Technical Risk Management*. Upper Saddle River, NJ: Prentice Hall; 1996, pp. 208–250.
4. The term "likelihood" is sometimes distinguished from "probability." The latter refers to values based on frequency measures from historical data, the former to subjective estimates or gut feel. If two of three previous attempts met with success the first time, then *ceteris paribus*, the probability of success on the next try is 2/3 or 0.67. Even without numerical data, however, a person with experience can, upon reflection, come up with a similar estimate that "odds are two to one that it will succeed the first time." Although one estimate is objective and the other subjective, that does not imply one is better than the other. Objective frequency data will not necessarily give a reliable estimate because a multitude of factors can influence outcomes; a subjective estimate, in contrast, might be more reliable because humans often do a pretty good job of assimilating lots of factors.

5. Roetzheim W. *Structured Computer Project Management*. Upper Saddle River, NJ: Prentice Hall; 1988, pp. 23–26; further examples of risk factors and methods of likelihood quantification are given in Michaels, *Technical Risk Management*.
6. See Dingle J. *Project Management: Orientation for Decision Makers*. London: Arnold; 1997.
7. See Gilbreath R. *Winning at Project Management: What Works, What Fails, and Why*. New York: John Wiley & Sons; 1986.
8. Roetzheim, *Structured Computer Project Management*, pp. 23–26.
9. Pool R. *Beyond Engineering: How Society Shapes Technology*. New York: Oxford University Press; 1997, pp. 197–202.
10. Kotulak R. Key differences seen in Columbia, Challenger disasters. *Chicago Tribune* February 2; 2003: Section 1, p. 5.
11. Pool, *Beyond Engineering*, pp. 207–214.
12. Michaels, *Technical Risk Management*, p. 40.
13. Statistics make it easy to depersonalize the consequences. For example, it is less distressing to state that there is a 0.005 likelihood of someone being killed than to say that 5 people out of 1,000 will be killed.
14. Mitroff I. and Linstone H. *The Unbounded Mind*. New York: Oxford; 1993, pp. 111–135.
15. Mitroff and Linstone, *The Unbounded Mind*, pp. 111–135.
16. European Commission. “Planning for Adaptation to Climate Change: Guidelines for Municipalities.” Life Project No LIFE08 ENV/IT/000436. n.d., circa 2013. <https://base-adaptation.eu/sites/default/files/306-guidelinesversionefinale20.pdf>, accessed May 1, 2024.
17. Tables 11.6–11.8 adapted from European Commission on Climate Action, Climate change and major projects, European Union Publications Office 2016: p.7. https://climate.ec.europa.eu/system/files/2016-11/major_projects_en.pdf, accessed May 30, 2024.
18. Eisner H. *Computer-Aided Systems Engineering*. Upper Saddle River, NJ: Prentice Hall; 1988, p. 335.
19. See Grady J. *System Requirements Analysis*. New York: McGraw-Hill; 1993, pp. 106–111.
20. Eisner, *Computer-Aided Systems Engineering*, p. 336.
21. Eisner, *Computer-Aided Systems Engineering*, p. 336.
22. A breadboard is a working assembly of components. A prototype is an early working model of a complete system. Both are used to demonstrate, validate, or prove the feasibility of a design concept. Breadboards, prototypes, and modeling are discussed in Chapters 4 and 10.
23. Wakabayashi H. and Cowan B. Vancouver International Airport expansion. *PM Network* September; 1998: pp. 39–44.
24. Whitten N. Meet minimum requirements: anything more is too much. *PM Network* September; 1998: p. 19.
25. DeMarco T. *The Deadline*. New York: Dorset House; 1997, p. 83; Yourdan E. *Rise and Resurrection of the American Programmer*. Upper Saddle River, NJ: Prentice Hall; 1998, pp. 133–136.
26. Dorner D. *The Logic of Failure*. Reading, MA: Addison-Wesley; 1997, p. 163.
27. Aronstein D. and Piccirillo A. *Have Blue and the F117A: Evolution of the Stealth Fighter*. Reston, VA: American Institute of Aeronautics and Astronautics; 1997, pp. 79–80.
28. Aronstein and Piccirillo, *Have Blue and the F117A*. . . , pp. 186–190.
29. For other approaches to risk time analysis, see Michaels, *Technical Risk Management*.
30. This section and the next address the more general topic of decision analysis, a broad topic that receives only cursory coverage here. A classic book on the subject is Luce R.D. and Raiffa H. *Games and Decisions*. New York: John Wiley & Sons; 1957.
31. Adapted from Kharbanda O. and Pinto J. *What Made Gertie Gallop: Learning from Project Failures*. New York: Van Nostrand Reinhold; 1996, pp. 177–191.
32. Source: Kromhout F. Divisional Director, Bridges, BKS (Pty) Ltd, Pretoria.

Chapter 12

Project Procurement Management and Contracting

Projects require resources—materials, supplies, equipment, and skilled labor, some or much of which is purchased or contracted. Indeed, in many projects almost everything is purchased or contracted and virtually nothing is done or produced internally. **Procurement management** addresses everything associated with purchasing and contracting, although its main focus is on contracting and contracted work. It is a specialized function that can require legal and contract administration skills. In some projects, procurement management is overseen by the project manager, in others, by a specialized procurement division or department.

Contracted work involves, simply, hiring someone with the capability to do the work for you; in projects it is widespread and common. Certainly, such work is just as important to project costs, quality, and profitability as work that is not contracted (i.e., done **internally** or **in-house**). Procurement management is a major aspect of project planning and control. Early in the project, procurement planning takes place to ensure that everything to be procured is identified and acquired. Once the project is underway, procurement control takes over to assure that everything procured is steered to meeting contracted requirements. All of this happens within the context of company procurement policies, procedures, and standards.

12.1 Procurement and Procurement Management Terminology

Procurement represents activities that involve the two parties of buyer and seller. In the context of project management, the buyer is the project customer; and the seller is the contractor—aka, developer or SDO (system development organization). Substitute terms for procurement are acquisition and purchase—although strictly speaking they are not the same. **Acquisition** refers to buying an *entire complex system* that is not well-defined—including its design, development, ramp-up, and production; **purchase** refers to buying a standardized (off-the-shelf) product or part. The term “procurement” is typically reserved for buying a component or subsystem (more than a part but less than an entire system) according to the customer’s specifications. Hence, it would be appropriate to say, “acquisition of a nuclear power plant,” “procurement of an automatic shut-down safety system,” and “purchasing a

batch of standard 1-inch nails.” Hereafter, though, we will tend to ignore the distinctions and use the terms somewhat interchangeably.

Procured resources for projects fall into the following categories:¹

1. **Products**—parts, components, subsystems, systems.
2. **Materials**—bulk (cement, lumber, wire, stone, etc.) and consumable (nails, rivets, fuel).
3. **Equipment**—equipment the project team does not already own (shop tools; cranes and scaffolding; welding and testing equipment; computers and office facilities).
4. **Work**—contract labor as provided by trade workers (steel workers) and professionals (engineers, programmers, consultants).
5. **Services**— **maintenance, repair, and operation (MRO)** of equipment and systems; work of an intellectual nature such as advising, management, consultation, and development of specialized products or technology.

To simplify, the above categories are hereafter referred to as **procured Goods, Work, Services—GWS**. Procured GWS can further be categorized as off-the-shelf and custom-designed and built. **Off-the-shelf (OTS)** represents familiar, in-stock or readily available GWS that are not produced specifically for the project. So-called **custom-designed and built** GWS are not OTS; they must be developed, sometimes from scratch, just for the project. Custom GWS in projects range from simple components and tasks in a work package to major portions of the project. The latter is called **turnkey** work or a turnkey project, meaning that a contractor fully designs, builds, and installs major components, equipment, or even the complete end-item system. All the customer does is “turn the key.”

Regarding the source of procured GWS (the seller), the terms supplier and contractor are often used interchangeably, though with distinctions:

1. **Supplier:** an individual or company that produces and sells GWS to the project. In this chapter it refers to producers or providers of mostly-OTS products, materials and equipment.
2. **Contractor:** an individual or company that provides the project with skilled labor (e.g., a union steelworker or freelance engineer), consultation, or custom-designed and built equipment, components, or systems.

Another term, not used in this chapter, is **vendor**, an individual or company that sells GWS but does not produce or provide it—the go-between for a customer and a supplier/contractor.

Important to note is that procurement can also occur *within* an organization, i.e., one party in the organization does work for or provides materials to another party. In such cases there is no contract, per se, although there is a formal “understanding” between the parties about the work/materials to be provided.

Procurement process

Procurement management is a process with four phases that roughly correspond to the three phases of the project life cycle:²

1. **Define and plan the procurement.** During the conception and definition phases, necessary resources (GWS) for the project are defined, and those that can be acquired internally are distinguished from those that must be procured (the “make or buy decision”).

2. **Conduct the procurement.** In the definition phase, qualified sources for procured GWS are identified and solicited; those interested are evaluated and the best are selected for the project; contract agreements are signed with these parties.
3. **Control the procurement.** During the execution phase, the GWS provided by suppliers and contractors are monitored, assessed, and controlled.
4. **Closeout the procurement.** At the end of the project, contracts and work orders are formally closed out.

Details of this process for each project are described in the **procurement management plan**, which is part of the project execution plan. Although the process roughly corresponds to the project life cycle, in fact all aspects of the process might happen at any time in the project. Large projects will involve many suppliers and contractors, some whose work will be identified, started, and finished early in the project, others whose work is identified, started, and finished much later.

Procurement activities overlap with those in project planning (Chapter 6), quality management (Chapter 10), risk management (Chapter 11), project monitoring and control (Chapter 13), and project selection (Chapter 19). This chapter will elaborate on these activities plus topics specific to contracts and the contracting process.



See Chapters 6,
10, 11, 13,
and 19

12.2 Define and Plan the Procurement

The procurement process begins during the conception and definition phases. At this time, as the objectives, requirements, work packages, and needed resources are being defined, options about ways to meet the requirements, do work, and acquire resources are assessed. During this phase, the project and procurement managers begin asking.

- What equipment is needed, and should it be rented, leased, or purchased?
- What materials are needed and from where should they be acquired?
- What work is needed, and can it be done “in-house” by the project team or should it be done by hired contractors?

Requirements and work definition

Decisions about whether work should be done **internally** or **externally** (i.e., purchased or contracted) apply to most everything in the project and at all levels of the WBS. The role of project management is to make distinctions. Every end-item and work package will likely have some procured GWS requirements. During the WBS process, those work packages that can be done in-house are differentiated from those that must be contracted. In the latter case, the project manager recognizes that certain (sometimes significant) portions of the project (maybe entire work packages) will have to be outsourced. In the former case, work packages to be done in-house will still need GWS that must be procured. A work package such as “develop new EV battery” that involves much research and testing might be entirely outsourced to contractors, whereas a “build house” work package might be done in-house (by the contractor) and will require only the sourcing of materials—lumber, duct work, wiring, windows, and so on.

When many contractors are hired or appointed to do the project work, the customer might appoint one of them, called the **prime contractor**, to oversee the work of (and sometimes hire) the other contractors, called **subcontractors**. In effect, the prime contractor is the subcontractors’ customer. More will be said about this later.

Make or buy

The decision about whether to do work internally or externally is called **make or buy**. It applies to a variety of situations (for example, whether to lease or rent, rent or buy, make or lease, etc.). The decision involves analysis of many factors; cost is often the major one and the decision is based upon break-even analysis. Commonly, however, the decision involves many considerations beyond cost, such as: retaining as much work as possible in-house to keep employees working; company policy about how much work should be contracted; availability of qualified contractors to do the work; ability of contractors to do the work faster, better, and at lower risk.

12.3 Conduct the Procurement

Soliciting and evaluating bids

Once the decision is made about which GWS should be procured, the process moves to soliciting potential suppliers/contractors. This is referred to as the **solicitation** or **conduct-procurement process**, which implies addressing all matters related to soliciting bids and proposals from potential suppliers and contractors. It includes preparing a **solicitation package**, which includes:³

- **Solicitation** or **bid document(s)** to be used (RFP, RFQ, RFT, as discussed later). These documents contain everything a supplier or contractor (**bidder**) needs to know in order to participate, for example:
 - Description of the work (SOW) requested by the customer
 - Instructions informing bidders about how they should respond to the solicitation (e.g., proposal contents)
 - Documentation to submit with the response (e.g., project plans)
 - Schedule for completing the solicitation process
 - Procedures for addressing questions and contacts
 - Plan of payment
- List of qualified bidders
- Evaluation process and criteria to be used for evaluating proposals
- Bidder's or pre-proposal conferences
- Methods to handle changes in proposals.

In many cases, much or all of the solicitation package is sent to each potential supplier/contractor.

The parties appearing on the list of bidders depend on the number of qualified bidders available and on customer policy or preference. **Multiple sourcing** means multiple qualified bidders are available and will be invited to bid competitively. **Single sourcing** means that among several qualified sources, only one is preferred and will be invited to bid; a variation of this is **sole sourcing**, which happens when there is one (and only one) bidder qualified and available to do the work (i.e., a monopolistic, non-competitive situation). What makes a bidder "qualified" is determined by customer screenings of potential bidders, sometimes based upon previous work done for the customer or elsewhere, sometimes using information received from potential bidders in response to **RFIs (requests for information)** sent by the customer. A customer that has established a valuable prior relationship with a contractor, especially one

with unique skills or capabilities, will often approach only that contractor (single sourcing). Identifying qualified bidders is an aspect of strategic sourcing, which is the policy of seeking to optimize a company's overall base of suppliers in terms of costs, delivery, and quality.

Two main methods are used for soliciting and selecting contractors: advertising/bidding and proposal/negotiation.

Advertising/bidding

When the scope of the project is somewhat simple, requirements are well defined, and bidders are pre-qualified to do the work, the **advertising/bidding** (aka **tendering**) process is appropriate. It starts when the customer advertises for bids online or with other media using a document such as an **RFQ**, **RFB**, **RFT**, or **IFB** (i.e., **request for quotation**, **bid**, **tender**, or **invitation for bid**)—which is a request for a simple price quote. Sometimes a bidder's conference is arranged to explain to bidders the work to be done and answer questions. Each interested bidder then submits its bid, quote or tender—its price to do the work, in, usually, a *sealed bid*, i.e., a bid document enclosed in a sealed envelope. Selection of the supplier/contractor is based almost wholly on the lowest-price bid. There is no negotiation; in fact, when the bidding is competitive, negotiation is prohibited and even considered unethical.

For OTS-type GWS, advertising/bidding is the most common solicitation method. Customers soliciting bids via this process must be careful to scrutinize the basis for the bidder's price estimates and question whether the bidder can realistically do the work for the bid price. In a practice called **buy in**, a supplier intentionally sets the price low just enough to get the job. In the event that costs exceed the price, the bidder hopes the customer will cover the overrun, or that the overrun will be covered by lucrative follow-up contracts with the same customer. Buy in most often happens during the bidding process, but it can also happen during contract negotiation.

Proposal/negotiation⁴

For large and somewhat undefined systems that require design, development, research, artistic, or other intellectual work, the proposal/negotiation process is more appropriate. The customer sends an **RFP**, **request for proposal**, to a short list of qualified bidders. The RFP might be preceded by an RFI to determine which contractors are qualified. Sometimes as part of the process, a so-called *pre-proposal* or *contractor conference* is held to explain to contractors the scope of the project, what the customer expects, and contractual requirements. (The contents and preparation of RFPs and proposals is discussed in Chapter 3.)

Selection of the winning contractor is based on a combination of proposal scores and negotiation. Proposals submitted by the contractors are scored using a set of evaluation criteria that includes price, completion date, ability of proposed deliverables to meet requirements, contractor's reputation, customer-supplier work relationship, and so on. In some projects, the score determines the winning contractor; in others it reduces the number of qualifying bidders to a select few, a process called *screening*. Those proposers that pass screening are invited to enter into negotiation. Methods for scoring and screening are discussed in Chapters 3 and 19.

Sometimes the selection process is secretive. Although the evaluation criteria might be published, details of the evaluation process and criteria weights are not. The process and the members of the evaluation committee are strictly confidential, which is to preclude bidders from influencing the evaluators and the evaluation process and to assure all parties of a "level playing field." During the negotiation, the customer and individual pre-selected contractors



See Chapter 3



See Chapters 3
and 19

meet to hammer out technical details and terms of the proposal, and to reach a contractual agreement on time, schedule, and performance requirements. Throughout negotiation, the project manager pushes the merits of his proposal with the goal of obtaining the best possible agreement for his company. In countering any customer objections to the proposal, the project manager's best defense is a well-thought-out project plan that shows what can or must be done to achieve the desired results. The project manager must be intimately familiar with the technical details of the proposed system's design and related costs. Sometimes the contract will include incentive or penalty clauses as inducements to complete the project before a certain date or below a certain cost. To competently negotiate such clauses the project manager must be familiar with the project plan, especially those parts of the schedule, work, or price that are relatively "fixed" and cannot be negotiated. Among the flexible parts of the plan, the project manager needs to know the maximums and minimums (in terms of schedule, price, and performance) that her company, the contractor, is willing to accept.

To be able to negotiate from a competitive position, the project manager must try to learn as much as possible about the customer. She should determine if the customer is pressured to make a particular decision, faces an impending fiscal deadline, or historically has shown preference for one approach or contractor over others. She needs to know the customer's min-max positions regarding price, performance, delivery, etc. She also needs to know about competitor bidders—their likely approach to the problem, costs, and competitive advantages and disadvantages. She learns this from historical information, published material, or employees who once worked for the competitors. (Relying on the last source is ethically questionable and, of course, works against the contractor whenever competitors hire its employees.)

Negotiation is the last activity before a contract agreement is reached, however the process of negotiation often begins much earlier during proposal preparation, since terms in the proposal must be mutually consistent and acceptable to both customer and contractor. During negotiation, terms in the proposal related to performance, schedules, and price are reviewed and converted into a legal agreement that is acceptable to both parties. In highly competitive situations, the customer will try to play contractors against each other, seeking to raise performance specifications while shortening the schedule and decreasing the price.

Although in theory negotiation is the last formal opportunity to correct misunderstandings in the proposal and reach a final contract agreement, the fact is, some customers are *always* negotiating—informally—to get a better deal, even after the project is underway. Thus, contractors must be wary of saying and writing anything that could be construed as a promise to deliver anything beyond what the contract specifies.

In some cases, after the winning contractor has been selected, the unselected ones can request an explanation from the customer about why they lost. This is to aid them in preparing future proposals or assure them that the selection process was fair.

It should be obvious that proposal/negotiation is more time-consuming and expensive than advertising/bidding since it involves considerable effort in preparing and evaluating proposals and negotiating contract terms. Nonetheless, while unnecessary for standardized projects wherein project work and costs are fairly well known, in situations where the need, problem, requirements or outcomes are less well-defined or unknown, or the work is uncertain and the customer must rely on a contractor to provide the solution, the proposal/negotiation process is appropriate (and necessary) for soliciting and selecting bidders.

Contract award process

The solicitation or conduct-procurement process ends with the selection of (or "award" to) a contractor and signing a contractual agreement. This agreement contains an SOW, perhaps

similar to the SOW in the original proposal, modified to reflect negotiated and mutually agreed-to adjustments. This so-called contract statement of work, or CSOW (also called procurement statement of work—PSOW), specifies the contractor's expected work in terms of scope, requirements, end-results, schedules, costs, etc., as well as conditions under which deliverables or end-items will be accepted by the customer. Failure to state these conditions can lead to later disputes and project delays.

Before work can begin, however, requirements as specified in the CSOW must be translated into terminology that stakeholders throughout the project will understand. These translations will appear in the CSOWs for contracts with every supplier and subcontractor, and they must be identical interpretations of the requirements and work as specified in the CSOW for the entire project. Broken down further, these CSOWs will appear on *work requisitions* or *work orders* so everyone knows what they are expected and authorized to do.

Changes to any contractual agreements thereafter must follow formal change procedures—change requests, reviews, approvals, and, sometimes, contract renegotiation.

Collaborative contracting⁵

Traditional contracting is regulated by terms stated in the CSOW. The project's scope of work and requirements are well-defined and any changes to these parameters may affect costs and give rise to claims. Project work is divvied up, usually by the customer, and contractors are held responsible for doing their share of the work as specified in the contract. A process like this, though common, does little to foster collaboration, is inflexible and adversarial, and can be exploitative.

In contrast, the **collaborative contracting** process is based on trust between parties who share goals and aspirations to achieve optimal results. All parties—the customer and all the contractors—share the pains and gains in the project. Following contractor selection, they team up, sometimes by co-locating as a single project team, to work out details of the project. This contracting strategy aims to maximize collaboration and streamline processes to achieve project goals.

Procurement schedule

Associated with each procured GWS is a schedule for when the GWS item is needed and when procurement activities must begin. The schedule must allow enough time to conduct the solicitation process and to deliver (or design, build, deliver) and install items as needed in the project. Figure 12.1 shows the schedule for a procured GWS item. It is prepared by working backwards, starting with the date when the GWS item is needed in the project (Event 10). Such a schedule is prepared for every procured GWS item. The schedules for all items are integrated with the project schedule to assure that all procured GWS items will be available when needed in the project.

Of course, preparing these schedules requires knowing the lead times for procured activities—the time needed for suppliers and subcontractors to prepare proposals, the project team to evaluate proposals and negotiate contracts, and suppliers/contractors to fulfill work commitments (which, for special design-build items could be substantial). It is not uncommon, especially in international and megaprojects, for these times to be underestimated and, subsequently, to delay the project.

The schedule in Figure 12.1 starts at the point where GWS requirements have already been identified, which happens early in the project definition phase when project work and resources are first identified. Procured GWS activities require the same considerations of

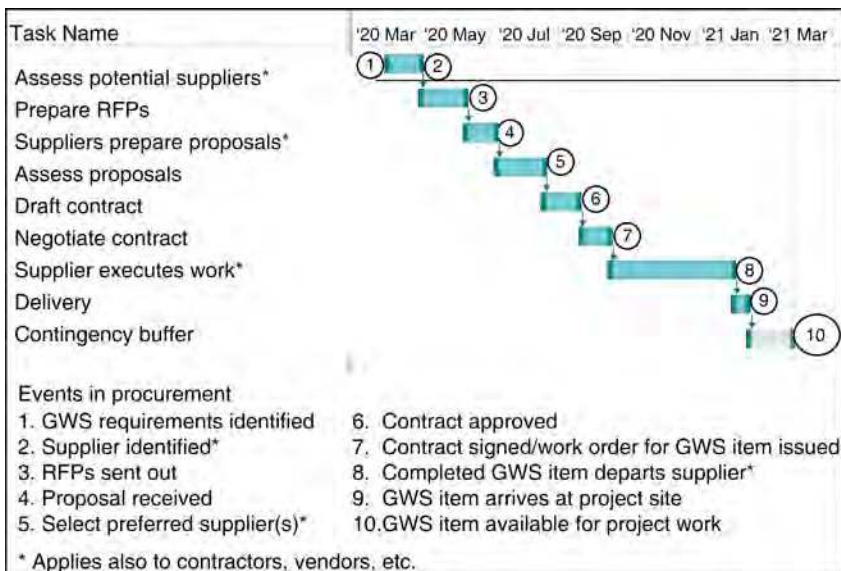


Figure 12.1

Gantt schedule of procurement activities.

responsibility, budget, quality, and risk as other aspects of the project and, hence, must be integrated with these matters in the project execution plan.

Logistics plan

Logistics relates to the transport and storage of materials. In projects that are materials-intensive, the loading, unloading, transportation, inspection, clearance and approval, and storage of materials can be major issues that must be addressed as part of the procurement management plan. For example, consider a large building project and the importance of timing the arrival of materials (steel, pipes, concrete slabs) to coincide with when they will be needed for the building. Obviously, the materials cannot arrive late because that will delay the project. But equally serious is when the materials arrive early. Where will you keep them? In congested urban areas there is no space, and if there is, materials delivered early are subject to damage, deterioration, and theft. In urban areas with narrow roadways and heavy traffic, clogged streets might constrict the number of material deliveries each day, which, *defacto*, constrains the work schedule. Whenever GWS items cannot be scheduled to arrive *just in time*, provision must be made to store and protect them, which on large projects can be costly.

12.4 Control and Closeout the Procurement

Contract administration

From the time when the contract is signed until the project is closed out, contractual agreements must be *managed*, which means (1) they will be updated with respect to ongoing changes

in the project, customer needs, and contractor capability, and (2) the work will be checked for conformance with the agreements. This process, called **contract administration**,⁶ allows the customer to retain some measure of control over contracted work. To exercise that control, however, each contract must clearly specify which areas of the project the customer is authorized to monitor and, perhaps, supervise.

In many projects, this aspect of procurement management is overseen by a **contract administrator**. The contract administrator works for the customer and oversees all matters related to and relationships with contractors; in turn, contractors might have their own contract administrators who oversee relations with their customers and their own contractors (i.e., subcontractors).

The contract administrator's role is to ensure that commitments in the contracts made by developers or contractors and the customer are met. It can include any of the following: authorizing work to begin; monitoring work with respect to budgets, schedules, and technical performance; ensuring quality and compliance to warranties (contractors' responsibilities); managing waivers and changes; checking for default or breach of contract (failure to meet commitments); resolving disputes; and closing out or terminating the project.

Contract administration is also responsible for ensuring the customer is invoiced for deliverables as specified in the contract, and that contractors and suppliers are paid for GWS provided. For simple projects, billing and payment tracking is done through the contractor's accounts receivable system; for large, complex projects, it is handled through a dedicated billing and payments tracking system.

Managing changes

Among reasons for changes in contracted work are:⁷

- Scope or other changes requested by the customer
- Impossible-to-meet requirements
- Misinterpreted contract terms
- Accelerated work
- Late or unsuitable information or resources provided by the customer
- Poor cooperation
- Adjustments to contractual compensation/payments

All requests for changes in contracted work, whether customer or contractor initiated, are assessed against conditions as stated in the contract. Requested changes that deviate from the contract are implemented only after customer/contractor approval and modification of the contract. If not properly handled, changes can lead to litigation. Procedures for change control are described in Chapter 13.



See Chapter 13

Inspection, acceptance, and closeout/termination

Contract administration is responsible for determining whether project end-items meet the requirements specified in the contract and if all, some, or none of the deliverables are acceptable. It also decides situations where the project or certain contracted work should be prematurely terminated, such as a contractor's failure to meet schedule, cost, progress, or other contract obligations. Sometimes the customer requires additional specific procedures for



See Chapter 5

testing, monitoring, and reporting, in which case these must be incorporated into the contractor's project tracking and control system.

At closeout, the product or deliverable is formally handed over to the customer. Most of the responsibilities for closeout as described in Chapter 5 fall under contract administration, but specifics vary. For example, the customer might be required to pay for work underway but not completed or accepted, or the contractor might have to repay the customer for any advances received.

The way closeout happens can depend on the conditions stated in the contract. Some contracts specify a **first handover at completion**, then a **second handover after a defect liability period** (aka a retention, guarantee, and maintenance period). At first handover, the customer must identify and report all **patent defects**, which are deviations from specifications that someone qualified could readily detect and that the contractor must rectify. Thereafter, the contractor is liable for rectifying only **latent defects**, which are defects that could not be detected through an inspection at first handover. For instance, a roof that leaks when it rains would be considered a latent defect if at the time of the first handover it wasn't raining and couldn't be detected.

The second handover affords the customer additional time to identify defects or substandard workmanship. After second handover, the contractor is no longer liable for defects, and any **retention fees** withheld by the customer to ensure contract compliance are paid to the contractor.

12.5 Contracting

Contracting environment⁸

Contracting is ubiquitous in projects. Most projects, even internal ones, involve some degree of contracting since often the customer must hire someone to perform at least some of the work. As mentioned, in some projects almost *everything* is provided by contractors. As Figure 12.2 illustrates, the project involves many organizations interlinked by contractual agreements. The customer might contract with a principal party (a prime contractor or SDO) to oversee the entire project, who in turn contracts with other parties—subcontractors (including

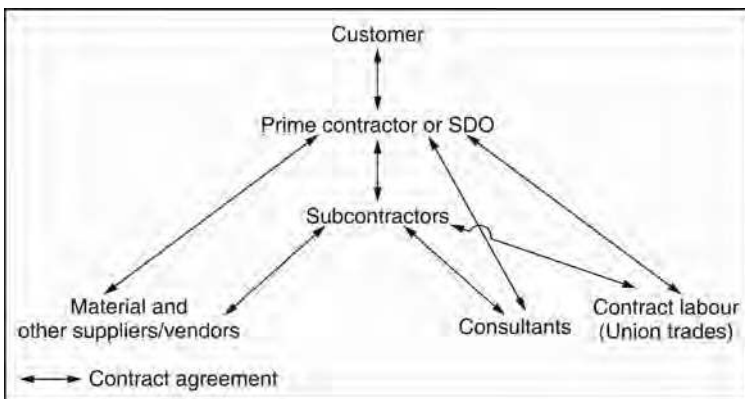


Figure 12.2
Contracting parties in a project.

consultants, material suppliers, and contract laborers)—to do portions of the project. In very large or megaprojects, multiple companies will form a **joint venture** to share the risks in managing the project and overseeing numerous subcontractors. A case in point is Boston’s Big Dig (example 16.3 in Chapter 16).



See Chapter 16

The solicitation process occurs everywhere throughout the project, addressing the question “who will do the work?” Whether it is the customer, prime contractor, or subcontractors, any party that contracts work or acquires material or services from other parties follows a similar process to identify, choose, and work with those who are most qualified.

Subcontracting⁹

Every party in Figure 12.2 must decide what portions of the project it will do itself and what it will contract others to do. For example, in a construction project the general or prime contractor typically will assemble the building structure but hire others to do specialized work such as wiring, plumbing, ventilation, and interior details.

The prime contractor’s scope of responsibilities is determined by the customer. A customer that wants the contractor to do all the work the contractor has proposed (and not give it to subcontractors) will specify that in the RFP or RFQ. Other possibilities include allowing the contractor to: freely select its own subcontractors, select subcontractors only with the customer’s approval, or select subcontractors from a customer-provided list. In the extreme case, the customer will *nominate* subcontractors that the contractor must consider hiring.

In general, the more the customer intervenes in subcontracted work, the less accountable the prime contractor is for that work. A contractor that is free to select its own subcontractors is wholly accountable for the subcontractors’ work; that accountability shifts largely to the customer when it specifies the subcontractors the contractor must hire. RFPs usually address this, stipulating that contractors’ proposals specify which project work they intend to do versus which they will give to subcontractors and the subcontractors’ names. As a rule of thumb, it is often a good idea to let the prime contractor select the subcontractors, which allows the contractor to build the best “contractor team” based on experience, and to be wholly responsible for managing the subcontractors once the project begins.

Even when a contractor is fully capable of doing all the work itself, it may choose to subcontract because it has limited capacity or believes a subcontractor could do the work for lower cost or less risk. In development projects of large-scale systems, the prime contractor will usually design the overall system and major subsystems, and perhaps produce some elements of the system itself, but will subcontract production of everything else. In projects where significant portions of work are to be subcontracted, the customer often will mandate the scope of the subcontracted work and criteria for selecting suitable contractors and suppliers.

Usually, obligations in subcontracts exist solely between a contractor and subcontractor. That means, for example, that the contractor (not the customer) is responsible for ensuring that a subcontractor performs work according to the requirements, and the contractor (not the customer) is obligated to pay for the subcontracted work. The contractor is also responsible for the quality of delivered materials, equipment, or components, and any inspections of a subcontractor’s offsite facilities. Similarly, all communication about customer changes to requirements is channeled through the contractor to the subcontractor. There are exceptions. If, for example, you, a subcontractor, are not getting paid by the contractor, you might appeal directly to the customer to pressure the contractor into paying you.

In preparing a proposal, the contractor must determine which work it can do itself and which it will contract out. Then, it must make certain there are subcontractors and suppliers who are qualified and available to do the work and provide the needed materials. In other

words, the contractor must follow the procurement management process in defining work and in soliciting and hiring contractors—ideally before it submits a proposal that includes subcontracted work.

Project manager roles in subcontracting

Large projects are often in the news—sometimes because of problems owing to cost overruns or schedule slippages. Although many factors are cited (materials shortages; labor union problems; weather; inflation), the cause is frequently poor procurement management and lack of control. Often, the project manager represents either the designer/architect or the builder. This works on small construction jobs, but not on big ones because designers and builders each represent different interests, and when things go wrong, they tend to argue their own interests and there is no one impartial to reconcile differences in the best interests of the customer or owner.

A better arrangement is when the customer appoints an *independent* project manager who will represent her interests during the entire project—including both design and construction. Figure 12.3 shows two alternative contracting structures to achieve this: **EPCM** (engineering, procurement, and construction management) and **EPC** (engineering, procurement, and construction). These kinds of agreement are most common in construction, but are used in other projects too.

The roles of the project manager differ significantly in the two kinds of agreement. In the EPCM case, the customer contracts directly with the design and building or manufacturing contractors, and the EPCM project manager merely provides *advice* to the customer about contracting with those contractors. The project manager communicates with the contractors, monitors their compliance with user requirements, design specifications, schedules, budgets, and safety standards, and reports her findings to the customer.

In the EPC case, the EPC project manager has *full authority* and accountability for all design and building aspects of a turnkey project. The customer has little involvement since the project

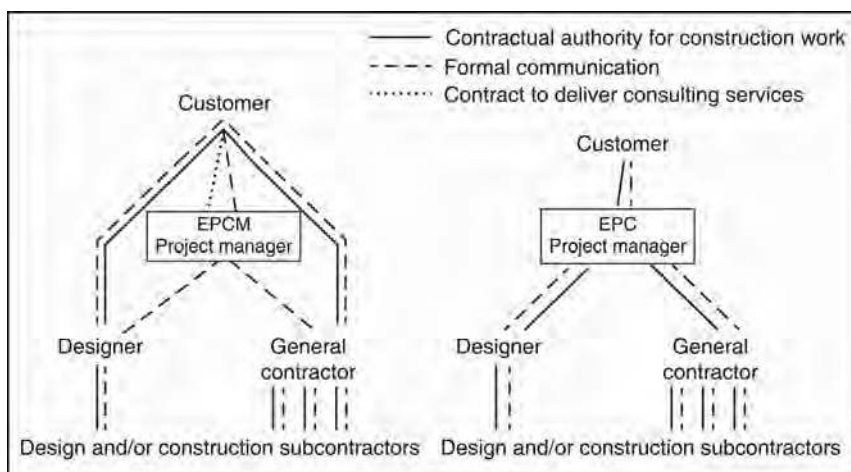


Figure 12.3

Alternative types of project manager in a construction project: EPCM and EPC.

manager, not the customer, handles the contracting for both design and construction or manufacturing.

In the EPC case, the project manager assumes most of the risks. In the EPCM case, most risks reside with the customer since the project manager plays only an advisory role.

Choosing between EPCM and EPC depends primarily on customer capabilities: managing contractors that perform design, construction or manufacturing work requires of the customer much greater technical and management capabilities in the EPCM case than in the EPC case. This includes capabilities in project planning, progress monitoring, and contract management, and a working knowledge of systems engineering, detail design, and quality management. Lacking such capabilities, the customer should appoint an independent EPC project manager or contractor to oversee all aspects of the project. The choice also depends on the project: projects that are complex with high uncertainty and risk favor EPC with cost-reimbursable contracts. In contrast, EPCM is more suitable with fixed-price contracts and simpler, lower-risk projects that the customer is capable of overseeing. Fixed-price and cost-reimbursable contract types are discussed in the next section.

Example 12.1: Project Contract in a Lawsuit

One of the authors served as an expert witness in a court case where a customer sued the project manager for doing a poor job and sought compensation. The project manager testified to the court that the customer repeatedly took actions against his recommendations. The expert witness explained to the court that, contrary to the customer's expectations, the contract was an EPCM arrangement wherein the customer, not the project manager, was responsible for overseeing the contractors. Consequently, the court ruled in favor of the project manager.

12.6 Kinds of Contracts¹⁰

A contract is an agreement between two parties wherein one party (the **seller**—project contractor) promises to provide a product or service to another (the **buyer**—project customer), typically in return for payment. The service is specified in either of two ways:

1. The seller is to provide a *deliverable* or *end-item*, and the contract is fulfilled upon its acceptance by the customer.
2. The seller is to provide a *level-of-effort* or *term of work*; i.e., to provide specific labor, skills, or facilities for a specified number of labor-hours/days or time period.

For complex work, “customized” contracts are drafted by project teams in collaboration with company lawyers. For less complex, standardized work, however, “standard” or **boiler plate** contracts are more common. An advantage of standard contracts is they use familiar terminology and, in the event of litigation, are more straightforward to interpret. These contracts have been developed by professional societies, large companies, and some countries and states.¹¹

Most contracts stipulate the following:

- Scope of work to be done or items to be produced, including support and side items such as manuals, documentation, and training. Any requirements, specifications, and standards referenced are considered as part of the contract.

- Contractor responsibilities in providing the work or items.
- Time schedule allowed.
- Customer duties regarding payments (including a schedule for milestone payments).
- Ways to handle changes to the contract and disputes.
- Ways to handle risks including warranties, penalties, or bonuses/incentives.

They also typically specify some or all of the following variables:

C_{ex}	Target (expected) cost under normal circumstances. “Cost” represents monies expended by the contractor in performing the work.
C_{ac}	Actual, reimbursable cost of the project at completion.
Fee	Amount paid to the contractor in excess of reimbursable costs .
Price	Price the customer pays for the project. The price includes reimbursable costs incurred by the contractor plus the contractor’s fee.

Within the standard contract forms are different kinds of contracts, the two most fundamental being **fixed price** and **cost-reimbursable** (or **cost-plus**). In the fixed-price contract, the price is agreed on and remains fixed as long as there are no changes to the project scope. In the cost-reimbursable contract, the contractor is reimbursed for all or some expenses incurred during the project and, consequently, the final price is unknown until the project is completed. Within these types are variations that include incentives for the contractor to meet cost, time, or performance targets. Most projects involve multiple contractors and, hence, multiple contracts and contract types. The basic forms of contracts are described next. (It must be noted that certain industries require specific contract forms beyond those described here.)

12.7 Fixed-price Contracts

Under a **fixed-price (FP)** or lump sum agreement the contractor agrees to perform all work at a fixed price. The contractor must be careful in estimating the target cost because, once agreed upon, the price will not be adjusted. If the contractor in the bidding stage estimates the target cost too low, it might win the job but make no profit or suffer a loss. Types of fixed-price agreements include firm fixed price, fixed-price incentive fee, and fixed price with economic price adjustment. These are illustrated next.

Firm fixed-price contract (FFP)

With an **FFP contract**, no matter what the project actually ends up costing (C_{ac}), the price to the customer remains the same. For example, suppose in the contract agreement is:

Cost estimate, C_{ex}	\$100,000
<u>Contractor fee</u>	<u>\$10,000</u>
Price	\$110,000

Under FFP, the price to the customer will remain \$110,000, no matter what.

When project work is straightforward and can be specified in detail, everyone prefers this kind of contract. Customers like FFP because they are less concerned about project costs. Contractors like it because customers tend to request fewer changes to the contract.

The disadvantage of an FFP contract is that it can be difficult and costly to prepare. The contractor risks underestimating the cost and losing money on the project, which might motivate it to cut corners (use cheaper quality materials, perform marginal workmanship) or extend the completion date to reduce costs. To prevent this, the customer can specify in the contract rigid end-item specifications and completion dates, and closely supervise the work.

Project sponsors usually prefer an FFP contract, thinking it transfers the risk of overruns to the contractors. Sometimes it does, but when a large project gets into trouble and the contractor cannot absorb the overrun, the sponsor will have to keep paying to finish the project. If the project is in serious-enough trouble to terminate it before being completed or bankrupting the contractor, other stakeholders might sue the customer. In short, FFP contracts do not work well in large, high-risk projects.

Also, FFP contracts can be short-sighted. *Project success* (versus *project management success*) depends on the performance of the end-item after the project is completed, yet the “fixed price” may force contractors to cut corners in the end-item, diminishing that performance.

Fixed-price incentive fee contract (FPIF)

When a contractor is unwilling to enter into a fixed-price agreement and the customer does not want a cost-reimbursable contract (discussed later), an alternative is an **incentive** contract, the purpose of which is to maximize contractor performance: good performance is rewarded with higher profits, poor performance with lower profits or losses.

In some cases, the incentive is determined by the **cost sharing ratio (CSR)**. A CSR of m/n (where $m + n = 100$) means that for any difference (savings or overrun) between target and actual cost, customer gets m percent of any saving and pays m percent of any overrun. Project price is computed as

Actual cost is under target cost: $\text{Price} = C_{ac} + (C_{ex} - C_{ac}) \times n + \text{Fee}$

Actual cost is over target cost: $\text{Price} = C_{ex} + (C_{ex} - C_{ac}) \times m + \text{Fee}$

A CSR of 80/20, for example, means that the customer and the contractor split the costs 80/20. This encourages the contractor to keep costs low because it pays 20 cents for every dollar spent above C_{ex} but earns 20 cents more for every dollar saved below C_{ex} . Such contracts appeal to both the contractor, who can earn a greater profit, and the customer, who pays a lower price. (Sometimes two CSRs are used, one for costs under target, the other for costs over target. See end-of-chapter review Question 22.)

The efficacy of incentive contracts is open to debate; some studies say they work, others say they don't or work marginally at best.¹² Incentive contracts with penalties can lead a contractor to overstate the estimated cost so as to ensure it will complete the project below that cost and earn a bonus. The customer will then believe the incentive was effective when really it only served to elevate the target cost!

Incentives also pose an issue of trust between customer and contractor. What these contracts imply is that I, the customer, “don't trust that the fee alone is enough to get you, the contractor, to hit the target (cost, price, date, performance, etc.) and that I need to bribe you to do so.”¹³

Different kinds of incentive contracts place the incentive emphasis on different targets: cost, price, schedule, end-item performance, or some combination. A **FPIF contract** places a ceiling on project price and, sometimes, profit. The contractor negotiates to perform the work for a target price based upon a target cost (C_{ex}) plus a fee, but the customer negotiates upper caps on price and profit. If the project ends up costing less than the target cost, the contractor

can earn a higher profit, but only up to the maximum. If there is a cost overrun, the contractor will have to absorb some or much of it.

Example 12.2: Fixed Price Incentive Fee Contract

Contract agreement:

Cost estimate, $C_{ex} = \$100,000$

Fee = \$10,000

Target Price = \$110,000

Maximum price = \$125,000 (fee + reimbursement); customer will pay no more than this.

Maximum Profit = \$15,000 Contractor will profit no more than this.

Cost sharing: CSR = 50/50

If $C_{ac} < \$100,000$, customer will reimburse C_{ac} plus an additional 50 percent of amount below \$100,000, as long as the additional amount does not exceed \$5,000. If $C_{ac} > \$100,000$, customer will reimburse \$100,000 plus an additional 50 percent of amount above \$100,000, but the price cannot exceed \$125,000. Again, the incentive is for the contractor to keep costs low and not exceed \$100,000. However, because the contractor cannot earn a profit of more than \$15,000, there is little incentive for the contractor to cut corners to increase profit.

Suppose C_{ac} is \$80,000 (\$20,000 under C_{ex}). The contractor gets paid \$80,000 plus the \$10,000 fee, plus an additional \$5,000 for the cost savings (50 percent of the \$20,000 savings is \$10,000, of which only \$5,000 is allowed because the maximum allowable profit is \$15,000). Total price to customer: \$95,000, a \$15,000 saving from the target price.

Suppose C_{ac} is \$200,000 (\$100,000 over C_{ex}). Fifty percent of the overrun is \$50,000; that plus the fee plus \$100,000 is \$160,000. But the specified maximum price is \$125,000, which is all the customer will pay. The contractor suffers a \$200,000 - \$125,000 = \$75,000 loss.

FPIF contracts are not true fixed-price contracts. They invite a contractor to negotiate an unrealistically high C_{ex} so that extra profit can be made through the incentive features. But unlike cost-reimbursable contracts, they provide some assurance of a maximum price and some protection against the contractor cutting corners to gain a hefty profit. FPIF contracts apply to long duration or large production projects, but not to development or other such projects where the target cost is difficult or impossible to estimate.

Fixed price with economic price adjustment (FP-EPA)¹⁴

Projects with long lead times such as construction or production have contract **escalation provisions** that protect the contractor against increases in materials, labor, or overhead costs. For example, the contract price may be tied to an inflation index and be adjusted in the advent of inflation, or it may be **re-determined** as actual costs become known. In the latter case, the initial price is negotiated with the stipulation that it will be adjusted later to reflect actual cost data. There are a variety of **FP-EPA (fixed price with economic price adjustment, also called fixed price with redetermination) contracts**: some establish a ceiling price for the contract and permit only downward adjustments, others permit upward and downward adjustments as demanded by either party; some establish one readjustment at the end of the project, others allow periodic readjustments at specified times. Each price redetermination is formalized by

modifications to the contract. FP-EPA contracts are appropriate wherever design work is difficult to specify, or the final price cannot be estimated for lack of accurate cost data.

Because the only requirement to renegotiate the price is substantiating cost data, FP-EPA contracts tend to induce dishonesty and inefficiency. After negotiating a low initial price, the contractor may produce a few items and then “discover” that the costs are much higher than expected. The contract thus becomes a cost-reimbursable kind of contract, as discussed next.

12.8 Cost-reimbursable Contracts

In complex, uncertain, or risky projects where it is difficult to estimate project costs accurately, a **cost-reimbursable** (or **cost-plus**) type contract allows the project to begin before the costs are fully determined.

Cost plus fixed fee (CPFF)

Under a **CPFF contract**, the contractor is reimbursed for all allowable costs plus an additional, fixed amount to cover profit. This contract is justified when costs cannot be accurately estimated or rise due to changes in the project scope or factors beyond anyone’s control. Regardless of the actual cost, the contractor’s fee remains the same, sometimes computed as a percentage of the initial estimated cost, C_{ex} .

Example 12.3: Cost-Plus Fixed Fee Contract

Contract agreement:

Cost estimate, $C_{ex} = \$100,000$

Fee = \$10,000 (10 percent of C_{ex})

Target Price = \$110,000

In addition to the fee, the customer will pay for all allowable costs (perhaps “all” costs, C_{ac}). Thus, if the project *were to* end up costing $C_{ac} = \$200,000$, the price to the customer is \$210,000.

In contrast to FFP contracts, CPFF agreements put the risk burden on the customer. The customer does not know the project price until the end of the project, and the contractor has little incentive to control costs or do anything beyond minimum requirements since it gets paid the same fee regardless. Among the few factors motivating the contractor to control costs and schedules is the negative effect of overruns on its reputation. Another is that the contractor cannot work on other projects while its workforce and facilities are tied up.

The contractor’s “profit” is ostensibly the fee above the cost, although, in reality, that might just be the tip of the iceberg since the contractor can profit from just about anything—materials, services, travel, etc. A contractor might specify a “fee” of \$10M but then profit another \$100M from fees tied to materials and services. The customer will learn about these added costs only through auditing the project. Contractors sometimes argue that the costs in a CPFF agreement are proprietary, although that is seldom true and the customer should check on all costs during the project and before the contractor is paid. In CPFF situations, the

customer might want to specify who is to be the project manager or to assign its own project manager to work alongside the contractor's project manager.

Despite the risks, the customer might have to resort to a CPFF contract just to attract contractors. CPFF is the contract of choice whenever the project involves high risk or costs that are difficult to estimate. Although CPFF might seem to pose little financial risk to contractors, in fact it does unless they are able to fully justify the costs. A contractor cannot just spend willy-nilly and expect to be automatically reimbursed! CPFF contracts typically permit customers to scrutinize contractor costs and financial records, and to negotiate settlements of or not reimburse any costs they find questionable or unjustified. A good illustration of a situation calling for cost-plus contracts is Example 3.8 and the development of Apollo Spacecraft.

Cost plus percent of cost (CPPC)

A variation of the CPFF contract is the **CPPC contract** wherein the customer agrees to pay all justified costs plus a percentage of those costs as the contractor's fee. In other words, the final price of the project is $C_{ac} (1 + \text{percentage fee})$, where the percentage is specified in the contract. Customers tend to dislike this kind of contract even more than CPFF because the contractor's profit increases with the costs—providing it no incentive to control those costs and essentially rewarding it for increasing costs. The customer must closely track all costs incurred and pay only for those that are “justified.” This type of contract is seldom used.

Cost plus incentive fee (CPIF)

The **CPIF contract** emphasizes hitting a *cost target*, and the contractor can earn a larger profit by reducing costs. The contract specifies the target cost, C_{ex} , and the CSR. The project price is then based on a percentage of the actual cost as set by the CSR plus the contractor's fee.

Example 12.4: Cost Plus Incentive Fee Contract

Contract agreement:

Cost estimate = Target cost = C_{ex} = \$100,000

Fee = \$10,000

Estimated Price = \$110,000

Cost sharing: CSR = 50/50

The contractor's incentive is to keep costs below \$100,000.

Suppose C_{ac} is \$80,000 (\$20,000 under C_{ex}).

Price = \$80,000 + (\$20,000) 0.50 + 10,000 = \$100,000

The customer saves \$10,000 on price and the contractor earns \$10,000 bonus. The customer must be vigilant to ensure that the incentive hasn't led the contractor to “cut corners” on work and materials.

Now suppose C_{ac} is \$200,000 (\$100,000 over C_{ex}).

Price = \$100,000 + (\$100,000) 0.50 + \$10,000 = \$160,000

The contractor is paid \$200,000 - \$160,000 = \$40,000 in the red.

Beyond cost and price, incentives can be applied to schedules and performance—to target completion dates and target performance parameters for the end-item. A **performance incentive contract** “rewards” the contractor for exceeding performance targets and “penalizes” it for underperforming; a **schedule incentive contract** stipulates rewards for completing the project early and penalties for finishing late.

Example 12.5 Schedule Incentive Contract: Hoover Dam

The project to construct Hoover Dam, one of the America’s largest and most historically significant dams, began in 1931. Before the dam could be constructed, however, engineers had to divert the Colorado River away from the site, which they did by channeling it through four, 50 ft. diameter, $\frac{3}{4}$ mile long tunnels cut through rock adjacent to the river. Eventually, some 11.5 million cubic yards of rock would be removed. So as to not delay construction of the dam, the US Government set the completion date for the tunnels at October 31, 1933. The contract agreement included a penalty of \$3,000 for each day the tunnels were completed late, and a bonus of \$3,000 for each day they were completed early.

Tunnel work started in May 1931 and was completed in November 1932 when the Colorado River was diverted through them. This was 11 months ahead of target, so the contractors received a bonus of \$990,000, (30 days x 11 months x \$3,000), a stupendous amount in depression-era 1932.

No doubt the bonus provided an incentive for contractors to beat the deadline, although, unfortunately, at the time there were few safety inspectors on site to enforce the even fewer safety regulations in place, so the contractor’s profit came at the expense of many tunnelers who lost their lives from heat prostration, carbon monoxide poisoning, and electrocution. True of most incentive contracts, the bonus/penalty system drove performance to meet one target (e.g., date) at the expense of others (e.g., safety).

Cost plus award fee (CPAF)

A **CPAF contract** is another form of cost-reimbursement incentive contract whereby the contractor is reimbursed for all justified expenses plus a fee. The fee includes a base amount that is negotiated before work begins plus an amount determined after the project, based on the customer’s judgment of the contractor’s performance (in terms of cost, schedule, or technical requirements). The base fee can be small or even zero, which means the bulk of the fee depends on the customer’s satisfaction with the work as measured by criteria specified in the contract.

Guaranteed maximum price (GMP)

With a CPFF contract the final price is unknown until the project is completed and the costs tallied: the sky is the limit. A more appealing agreement to customers is the **GMP contract**, which is a CPFF contract with a price cap. The GMP includes the contractor’s fee, which can be fixed or a percentage of costs. The customer agrees to pay actual project costs until the project price reaches the cap; for costs beyond that, the contractor is responsible.

Suppose the fee is set at \$10,000 and the GMP cap is \$110,000. If C_{ac} ends up at \$80,000, the customer pays the contractor $\$80,000 + \$10,000 = \$90,000$. If C_{ac} is \$200,000, the customer pays the GMP, \$110,000, and the contractor incurs a \$90,000 loss.

Time and materials (T&M)

Strictly speaking, the **T&M contract** is a variation of a cost-reimbursement contract since the final price is not known until after the work is completed and costs are tallied. It is called “time and materials” because the contractor is reimbursed for labor and material costs incurred in the project. It provides payment for direct labor hours at a specified per-person hourly rate that includes direct and indirect labor costs and a mark-up. The material fee includes the contractor’s material cost, handling fee and a mark-up. Sometimes a ceiling price is established that may be exceeded, depending on the agreement. Charges for private consultants and the services of electricians, carpenters, plumbers, mechanics, and so on, are usually based on T&M. The T&M contract places all cost risk with the customer and is least preferred when other contract types are available.

12.9 Contract-related Matters

Risk

One reason for the prevalence of contracting and subcontracting is **risk transfer**: by contracting the work, one party transfers the risk to another. The degree of risk transfer depends on the type of contract, as shown in Figure 12.4. For example, with an FFP agreement, the contractor assumes most of the risk of a cost overrun. Contractors find this acceptable whenever the statement of work is clear and involves little uncertainty, but unacceptable when the scope of the work is unclear, the work is likely to require many changes, and they will have to absorb cost overruns. In the latter case, they prefer a CPFF contract where all costs incurred will be covered by the customer. Between FFP and CPFF, the contractor and customer share the risks to varying degrees. In an FPIF contract the contractor accepts roughly 60 percent of the risk; the customer 40 percent. In a CPIF contract, the contractor assumes about 40 percent, the customer 60 percent.

But rarely is risk transfer as simple or complete as Figure 12.4 might suggest. Even with an FFP contract whereby the contractor assumes the risk of paying for an overrun, a large enough overrun can overwhelm and bankrupt a contractor—even a big contractor, in which case the customer must pay the overrun if the project is to be completed. Alternatively, to avoid the



Figure 12.4
Degree of risk assumed by contractor and customer for different contract types.

losses from an overrun, a contractor might feel pressured to cut corners, which increases the customer's risk of receiving a sub-par quality end-item.

Large projects use various contracts, depending on the risk associated with individual work packages or deliverables. In the Channel Tunnel project, the most uncertain part was tunneling under the English Channel; thus, that part of the work was contracted on a cost-plus basis. The electrical and mechanical works for the tunnels and terminals were perceived as low risk and were thus done on a fixed-price basis. Procurement of the rolling stock, perceived as slightly riskier, used CPPC contracts.¹⁵

The party at risk depends not only on the contract type but also on whoever estimated and set targets in the contract. A customer who estimates and sets the target cost inherits the risk of a too-low estimate: when the project cost exceeds the estimate due to the customer's estimating errors or omissions, the customer must pay the excess, even in an FFP contract. Had the price agreement been based on the contractor's estimate, the contractor would have to absorb the overrun.

Customer recourse

A standard clause found in contracting is **force majeure**, which exempts parties from fulfilling their contractual obligations for causes that could not be anticipated or are beyond their control. These include so-called acts of God (floods, earthquakes, and other natural disasters), acts of man (war, riots, etc.), and other events or occurrences that are unforeseeable and unpreventable.

When a contractor or supplier fails to meet contractual obligations for reasons other than *force majeure*, customer recourse falls into three increasingly severe categories.

1. Sue for breach of contract.
2. Claim penalties for breach of contract.
3. Demand a surety bond.

Breach of contract means, simply, failure to satisfy conditions or requirements as specified in the contract. Under breach, the customer can either cancel the contract (if the contract includes a cancellation clause) or sue for damages—where damages are determined either as harm caused by the breach itself, or whatever effort is needed to recover and fulfill the conditions of the contract. The downside of suing for breach is a sometimes expensive and lengthy litigation with no certainty of the outcome.

A **penalty clause** specifies a sum of money to be paid or other action to be taken in the event of a breach; for example, \$10,000 per calendar-day for failure to meet the target completion date. The penalty amount, so-called **liquidated damages** (liquidate means to convert to cash), is calculated to represent how much the customer stands to lose if the contract is breached or performance not delivered. It is intended to represent a fair and genuine pre-estimate of the damages negotiated before the start of the contract, and not as punishment to the contractor. The customer does not need to prove damages for the penalty to come into operation, only that the breach has occurred.

Another measure, a **surety bond**, is a contract between the customer, contractor, and a guarantor or third party—usually a bank or insurance company. Surety bonds protect the customer in a variety of ways. One, called a performance bond, guarantees the contractor's work: if the contractor is unable to complete the specified work, the guarantor may provide the contractor with needed credit or appoint another contractor to do the work.

Subcontract contracts

As discussed, the choice of contract depends on many considerations such as the risk advantage or disadvantage to the customer and the contractor. However, when the contract is between a prime contractor and its subcontractors, the type of contract will also depend on the type of agreement between the prime contractor and the customer. If the *prime contract* (between the prime contractor and the customer) is fixed-price (FP), then all subcontract agreements should also be FP, otherwise the prime contractor risks being charged more by its subcontractors than it will recoup from the customer in the FP agreement. If the prime contract is cost-reimbursable (CR) or incentive, there is more latitude in the subcontracts since any or most costs charged by subcontractors can be recovered by costs charged to the customer. Beyond simply covering its costs, however, the prime contractor can gain better overall control of project costs by setting FP or incentive agreements with its subcontractors, even if its agreement with the customer is CR.

12.10 Summary

Procurement management pertains to the acquisition and oversight of everything purchased or contracted for the project—all procured goods, work, and services (GWS). It represents the management of agreements between all parties in the project—the customer, the prime contractor, and subcontractors and suppliers. It is a four-phase process that roughly corresponds to the three phases of the project life cycle:

Define and plan procurement happens during the conception and definition phases to determine the necessary resources (GWS) for the project and distinguish those that can be fulfilled internally from those that must be procured (make vs. buy). Steps of the procurement process as well as consideration of all procured items and the delivery/receipt logistics of those items must be included in elements of the project plan—the WBS, schedule, budget, responsibility matrix, and so on.

In **conduct procurement**, which happens in the definition phase, qualified sources (suppliers, contractors) of GWS are identified and solicited via advertising or RFPs, RFQs, or RFBs, the resulting proposals or bids are evaluated, and the best sources selected and contracted with, sometimes through the process of negotiation.

During **control procurement**, which occurs throughout the execution phase, the procured GWS and relationships with suppliers and contractors are monitored and controlled to ensure conformance to contractual agreements, and to manage changes to those agreements.

Closeout procurement happens at the end of the project and focuses on the orderly review, acceptance, and handover of contract end-items, and formal closeout of all contracts.

For each project, this process is prescribed in the procurement management plan. Most projects, even internal ones, involve some degree of contracting since often the customer must hire someone to perform some of the work. Sometimes the customer hires a party to oversee the project (a prime contractor or SDO), and this party contracts with other parties—subcontractors—to do portions of the project. Relations between these parties follow the procurement management process.

Even when a contractor can do all the work itself, it may choose to subcontract because it has limited capacity or believes a subcontractor could do the work for lower cost or less risk. In large-scale development projects, the prime contractor might design the overall system and major subsystems, but then subcontract to produce the subsystems and, perhaps, even the main end-item system.

The fundamental kinds of contracts are fixed price (FP) and cost-reimbursement (CR). In the FP contract, the price is agreed on and remains fixed if there are no changes to the project scope or contract provisions. In the CR contract, the contractor is reimbursed for all or most expenses incurred during the project; hence, the final price is unknown until the project is completed. A variation of these contracts, called incentive contracts, provide incentives to the contractor to meet cost, time, or performance targets, and stipulate penalties for not meeting them. The choice of contract type depends on the project and project environment since each provides relative advantages and disadvantages to the customer and the contractor. Most projects involve multiple contractors and, hence, multiple contracts and contract types.



Review Questions

1. When does the procurement process begin?
2. Describe what procurement means. How does it differ/similarity to acquisition and purchases?
3. What types of project resources can be procured?
4. How do the steps of procurement align with the project development cycle? Describe the steps of procurement.
5. Describe five components of a solicitation package.
6. What is the process for solicitation through bidding? For what type of projects is this method used and why?
7. In bidding solicitation, why should customers be careful to scrutinize a bidder's price?
8. When does the negotiation process end?
9. How does the negotiation process work? For what type of projects is this method of solicitation and why?
10. Why is the CSOW important?
11. Why does the procurement schedule often end up grossly underestimated?
12. What is logistics? Describe the importance of logistics planning in a project.
13. Describe the role of contract administration. List three control measures it includes.
14. What happens at first handover at completion? At second handover?
15. In contracting work, does the customer relinquish all control over the project to the contractor? Explain.
16. How can a contractor be both the sender and receiver of RFPs, i.e., how can it both prepare and submit proposals, and receive and review proposals?
17. When a contractor hires a subcontractor, to whom is the subcontractor obligated—the end-user customer or the contractor?
18. What must the project manager know to be able to effectively negotiate a contract? Consider aspects of the customer, competition, and technical content of the proposal.
19. Discuss the difference between the SOW, CSOW, and work requisition or work order.
20. Describe the different kinds of contracts. What are the relative advantages and disadvantages of each to the customer and the contractor?
21. A customer holds the project manager accountable for cost overruns and a delay in completion. Why is it relevant whether the relationship between the customer and the project manager is governed by an EPC or EPCM contract?
22. Refer to the CPIF and FPIF examples problems in Examples 12.2 and 12.4.

- a. In both CPIF and FPIF cases, what is the price if $C_{ac} = \$90,000$? What is the contractor's profit?
- b. In both cases what is the price if $C_{ac} = \$160,000$? What is the contractor's profit?
- c. Sometimes two CSRs are used, a different one each for underruns and overruns. What are the answers to a. and b. if the CSR is 70/30 for underruns and 80/20 for overruns?



Questions About the Study Project

1. What are the procured GWS in the project? Were these items managed differently than in-house aspects of the project? How were they first identified and then integrated into the project plan? Did procured items pose any difficulties to the project?
2. How were contracts negotiated and who was involved in the negotiation?
3. What kinds of contracts are used in the project?
4. Did the project have a procurement manager? Does the company have a procurement team or department? If yes, how were they involved in the project and what else do they work on?
5. How was it determined what would be purchased or contracted and what would be done internally?
6. What kind of auditing occurred on procured items (quality checkpoints and testing)?
7. Were there any delays in the project due to procured GWS? Why or why not—good controls vs. poor controls? How were the logistics planned? Did logistics create any issues in terms of timing or resource availability?

CASE 12.1 CONTRACT MESS-UP AT POLANSKI DEVELOPERS

LaPage Power Company needed to upgrade the fire extinguishing system for the control room of a nuclear power plant. It selected Polanski Developers Company because that was the only contractor willing to do the work for a fixed-price contract. Polanski's \$11 million price was based on its \$9.5 million estimated cost for work and materials, and a fee of \$1.5 million. Polanski managers felt the fee was large enough to provide profit and absorb any unforeseen work difficulties. The upgrade would require interfacing with many plant safety systems, some dating back to when the plant opened in 1985 and others that had been upgraded many times since.

The interfaces with other systems would make the upgrade complex and challenging. Polanski anticipated this, and to reduce the risk of a cost overrun contracted with Moreland Systems, a company with much experience in nuclear power plants. Moreland would be responsible for almost all of the actual system design and installation. Said Billy Chester, Moreland's project manager, "You never know what you'll find in these kinds of projects." He told Polanski that Moreland would accept the job, but only on a cost-plus basis. The CPFF contract specified a target price of \$10 million using Polanski's \$9.5 million cost estimate and a fee of \$500,000. Polanski agreed.

When the project was completed—having encountered several unanticipated problems—Moreland's bill was \$14.5 million. The CPFF contract had specified periodic audits of Moreland's costs but none were done.

QUESTION

Discuss the financial consequences to Polanski, Moreland, and LaPage. What should Polanski have done that could have altered the consequences? How does the choice of contract-type depend on the risks involved?

CASE 12.2 CONSOLIDATED ENERGY COMPANY

Consolidated Energy (CE) is a public utility that generates and distributes electricity throughout five regions in the US. The company is involved in many kinds of projects such as construction of electrical generating and transmission equipment and facilities, upgrade and repair of equipment and facilities, information technology for customer service, and energy research. Much of this project work is contracted out, although about half of it is done by CE itself. The company has construction units and equipment specialists in all five regions, IT specialists in three regions, and research units in two. The research units work on projects initiated by the corporate office, but the construction, equipment upgrade and maintenance, and IT units work on projects initiated by the five regional offices. Each of these units is assigned to one or two regions; any project identified by a regional office is automatically handed to the construction, IT, or equipment unit assigned to the region.

Decisions about projects are made at regional and corporate levels: projects costing more than \$20M are handled at the corporate level, otherwise they are handled regionally. Whenever a regional office funds a project, it first decides whether the IT, equipment, or construction unit for its region can handle the job; if yes, it assigns them to the job, if no, it contracts the work to outsiders using the RFP/proposal process. A corporate project review board (PRB) makes decisions for projects that exceed \$20M. When the PRB approves a project, it awards the job to either the internal unit in the region that requested the project, or to a contractor via the RFP/proposal process.

Recently a member of the PRB had a clever idea: why not use the RFP/proposal process for all projects, including ones that might be done internally. When a regional office identifies a potential project, instead of automatically giving the project to the pre-assigned internal unit, it would send an RFP to the company's IT, construction, or equipment units in all of the regions. The unit with the best proposal would get the job, regardless of its location. Some members of the board balked at the suggestion, saying it would put units with the same expertise in competition with each other. Others argued that it did not make sense for, say, a construction unit to take on a project outside its region because moving equipment and work crews to distant project sites would increase project costs. Others countered that such arguments were pointless because competition among the units would encourage higher-quality work and reduce overall corporate costs.

QUESTION

What do you think of this idea? What are the pros and cons?

CASE 12.3 FORMAL AND INFORMAL AGREEMENTS IN CONTRACTUAL MATTERS

As he walks out of the president's office, Philip shakes his head. It was clear that the president was a bit embarrassed that she does not know the details of her company's configuration management system. She seemed even more embarrassed that she will have to cancel an agreement she made last night at a dinner meeting with her counterpart in HeavyEng. "In a way it serves her right" he mutters.

Since Philip was appointed procurement manager at TechnoVehicle, several issues relating to communication have come up, especially with the development of a vehicle that TechnoVehicle has been designing and for which HeavyEng made the prototype and is preparing to produce. The product is a state-of-the-art firefighting vehicle on order by several airports. Philip has been concerned about the constant meetings between the engineering staffs of the two companies and the numerous design modifications they have apparently led to. However, he also realizes the necessity of such meetings to ensure cost-effective manufacturing. And now the president called him in to say that she had told HeavyEng's president that TechnoVehicle would in the future supply HeavyEng with electronic copies of drawings instead of the "hard" copies they had been supplying. He had to inform her that the "hard" copies of the drawings—not the electronic ones—are under configuration control and that what she agreed to at the dinner meeting would simply not work.

After an hour of mulling around in the office, he calls the president's assistant to request a meeting; the agenda: formal and informal communication. Then he calls the engineering manager to arrange a meeting with the two of them on the same topic.

QUESTIONS

1. Why do you think Philip is upset about the agreement between the two presidents?
2. What similarity is there between the communication between the two presidents and the communication between the engineering staffs of the two companies?
3. What message should Philip convey during the two meetings he has scheduled?
4. Why is Philip's idea of face-to-face meetings with the president and the engineering manager a good one?

What is a good general rule regarding formal and informal communication for any project?

Notes

1. Joy P.K. *Total Project Management*. Delhi: Macmillan India Limited; 1998. pp. 378–380.
2. Procurement management phases as defined by PMBOK (6th edition of PMBOK combines "control procurement" and "close-out procurement" as one phase). *A Guide to Project Management Knowledge (PMBOK Guide)*, 5th edn. Newton Square, PA; 2017.
3. Adapted from Kerzner H. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, 8th edn. Hoboken, NJ: John Wiley & Sons; 2003: p. 815.
4. See Hajek V.G. *Management of Engineering Projects*, 3rd edn. New York: McGraw-Hill, 1984, Chapters 8 and 9; and Rosenau M.D. *Successful Project Management*, Belmont, CA: Lifetime Learning; 1981, pp. 34–41.

5. Walker D.H. and Rowlinson S. Project types and their procurement needs. In: *Procurement Systems*. London & New York: Routledge; 2007, pp. 56–93.
6. Management of the complete project contracting process, including what and where to contract, soliciting and assessing proposals, reaching a contract agreement, and administering the contract is also called “contract monitoring.” See Hirsch W. *The Contracts Management Deskbook*, revised ed. New York, NY: American Management Association; 1986, Chapter 6.
7. Adapted from Kerzner, *Project Management*, p. 827.
8. This section gives an overview of the important contracting issues. It is not intended to provide legal advice about contracts; for that you need an attorney or contracts specialist.
9. Hirsch, *The Contracts Management Deskbook*, pp. 290–315.
10. Complete descriptions of project contracts are provided in books on contract management. A good source is Hirsch, *The Contracts Management Deskbook*, pp. 43–75. For construction contracts, see Furst S. and Ramsey V. (eds.) *Keating on Construction Contracts*, 8th edn. London: Sweet & Maxwell; 2006.
11. Internationally, two formats popular for engineering, services, and construction work are FIDIC (Fédération Internationale Des Ingénieurs-Conseils) and NEC (New Engineering Contract). In the US, standard contract formats are used less often, and when they are the parties often make revisions. Two common formats used in construction and real-estate development are contracts created by the AIA (American Institute of Architects) and ASCE (American Society of Civil Engineers).
12. Merrow E.W. *Industrial Megaprojects: Concepts, Strategies, and Practices for Success*. Hoboken, NJ: John Wiley & Sons; 2011, pp. 286–287.
13. Merrow, *Industrial Megaprojects*, p.288.
14. Hajek, *Management of Engineering Projects*, pp. 82–83.
15. Anbari F. (ed.). *Case Studies in Project Management: The Chunnel Project*. Newton Square, PA: Project Management Institute; 2005.

Chapter 13

Project Monitoring and Control

Despite all the effort devoted to planning, scheduling, and budgeting in the definition phase, no plan is ever complete or perfect; and besides, things rarely go entirely as planned. Although it is certain that problems and deviations from the project plan will occur once project execution begins, it is not known a priori where or when. Monitoring and control is the process of tracking project performance, discovering areas where performance is lacking, uncovering extant or potential problems, overcoming obstacles, and keeping the project moving toward the target.

Planning and control are major parts of project management, but to be effective they must happen in that sequence—planning first, then control. Without good plans and requirements, there are no clear targets toward which to direct the project, and there can be no control.

13.1 Monitoring and Control Process

Project monitoring and control, which is the process of keeping the project moving in the direction as laid out by the execution plan, happens throughout the project. In simple terms, it involves observing and measuring progress, assessing progress against planned objectives and requirements, and taking corrective action. It can be compared to a home air-conditioning system, which works this way:

1. The desired room temperature is set on the thermostat.
2. The thermostat measures the actual room temperature and determines the temperature variance (actual temperature minus desired temperature).
3. If the variance is positive, the thermostat turns on the air conditioner until the room temperature coincides with the desired temperature (i.e., variance becomes zero).

Virtually every monitoring and control process follows these same steps: (1) set the performance standard or target, (2) compare actual performance to the standard, and (3) take corrective action to remove any difference.

In projects, the set *performance standards* step happens in the definition phase and early part of the execution phase. The standards are objectives, user requirements, technical specifications, budgeted costs, schedule due dates, and resource requirements.

The next step, compare *actual performance* to the standards, happens throughout the execution phase. Work performance, costs, and schedules are monitored and compared to actual expenditures, test results, work completed, and other measures.

The last step, take *corrective action*, occurs during execution whenever actual performance significantly differs from planned performance: something is done to remove the difference—either improve performance to meet the target standards, or revise the standards. In the latter case, the contractor and the customer agree to change the objectives, revise the requirements, or modify the plan.

Worth repeating is that to keep the project aligned with standards (requirements, schedules, and budgets) there must first be a plan! *The precursor to project control is project definition and planning*: without clear, complete requirements and a good plan, there can be no control.

Data for monitoring

Data collected for project monitoring must relate to project **key performance indicators, KPIs**, identified as the best metrics for assessing progress toward meeting project targets in terms of project objectives, schedules, budgets, and requirements. Importantly, the data must reflect not only measures of the cost and time expended, which are measures of *input*, but also measures of project *output*—measures to address deliverables and results as defined in project plans and requirements.

Typical data sources include material purchase invoices, worker timecards, change notices, test results, work orders, and expert opinion. The quantity and variety of data collected must be balanced: too much data will be overly costly to collect and review, too little will inadequately reflect project status and allow problems to go unchecked. Data must be analyzed and results reported frequently enough so managers can quickly spot deviations from plans and take corrective action.¹

How frequently should data be collected, assessed, and reported? A rule of thumb is to assess work progress every week. This will ensure that even work packages lasting just a few weeks will be checked at least twice. For work packages lasting several months, every 2 or 3 weeks might be adequate. The goal is to check the work often enough to enable accurate progress assessment and spot problems early, yet not so often that it becomes a burden. The frequency also depends on who is doing the work: the more competent and motivated the people doing the work, the less they need to be monitored. The kind of data to be collected and frequency of collection should be specified in the project plan.

Internal and external monitoring and control

Project monitoring and control happens internally and externally. **Internal control** refers to the contractor's procedures for monitoring work, reporting status, and acting. **External control** refers to additional measures imposed by other stakeholders, such as the customer; these include:

- Frequent reports to the customer of schedules, cost, and technical performance.
- Work inspections by the customer.
- Customer audits of the contractor's books and records.
- Strict terms in the contract on allowable project costs, pricing policies, etc.
- A customer project manager who works with the contractor project manager.

These control measures can be annoying to the contractor, especially the last, which amounts to the customer's project manager overseeing the contractor's project manager. Nonetheless, they are sometimes necessary to protect the customer's interests, especially in cost-reimbursable

projects. Ideally, customer and contractor managers work together amicably to establish compatible plans and work monitoring methods.

13.2 Project Monitoring

Project managers monitor progress using a variety of methods, metrics, and sources—both formal and informal, often as specified in advance in the project plan. Using a variety of measures and sources increases the validity of the information obtained, particularly when they all lead to the same conclusion. As mentioned, monitoring should focus on information about key performance indicators, KPIs. Among ways for collecting and conveying that information are status review meetings, observation, technology, and written reports.

Status review meetings

Status review meetings are among the most important ways to determine project status and assess performance. Meetings can be informal or formal and scheduled periodically, as needed, or at milestones. Large projects require all kinds of meetings.

Informal reviews

Informal reviews (or **peer reviews** because they are attended primarily by peers) are held frequently. Meeting attendance depends on the stage of work and any issues at hand: usually only those team members, customer reps, suppliers and managers who need to be involved participate. Before each meeting, project status, issue status, and estimated completion dates and costs are updated. Attendees with assignments are expected to give presentations.

The purpose of the reviews is to uncover problems and issues and agree on courses of action, consequently bad news and problems are *expected* and openly confronted. The project manager acts as facilitator and encourages honesty and candor. Finger pointing, passing blame, or glossing over conflict are avoided since they waste time and discourage candor and attendance.

Problems surfaced in a review are noted on the issue log (described later); one order of business in every meeting is to review all issues on the log and update their status. Always, the project manager—not a secretary or functionary—should lead the review, take notes and, where needed, write up and distribute notes. This reinforces the perception that the project manager is committed, involved, and in charge.

Ideally project reviews are convened at a central meeting place—a **project office**, or “war room,” a physical space for not only meetings but preparing, storing, and displaying project information. Gantt charts, network diagrams and cost charts are displayed on the walls for easy reference. The room provides whatever is needed to facilitate meetings: table, chairs, filing cabinets, computers, projector, teleconferencing equipment, etc.

Standup meetings

The daily **standup meeting** is perhaps the most effective form of informal review. Intended to update status, identify problems, and expedite solutions, it is short and to-the-point, and held at the worksite. Usually convened at the start of the day, the team gives a quick run-through of yesterday’s progress and today’s next steps. Problems that require more than a minute’s

reflection are deferred for a follow-up scheduled meeting. The occasional surprise attendance of a prominent person—senior manager from the contractor or customer—adds zip and keeps everyone on their toes.

Formal reviews

Formal review meetings to assess progress are scheduled weekly or at milestones or critical project stages. Two such reviews in design and development projects are the **preliminary design review** and the **critical design review**. These are convened and attended by management, outside experts, and project sponsor or customer, as described in Chapter 10. They provide independent assessments of project performance, suggestions, or instructions to improve the project, and sometimes a decision (“gate”) for the project to proceed or terminate, depending on review results.



See Chapter 10

In every project, regardless of contractual obligations, the customer should assume some responsibility as project watchdog. In one form of formal review, a **project audit**, the customer arranges for a formal assessment of project progress. The audit can be conducted anytime in the project—at milestones or upon major changes to project costs, timeline, or goals. Audits are discussed in Chapter 10.

Observation and site visits—“walking the halls”

Most project managers would never rely solely on reviews, reports, or emails to track project progress. If they cannot be always at the project site, they make a point to show up there often, unannounced and uninvited. There they try to speak to team members informally at lunch or on break. In this way they show their involvement in the project, learn what is happening, and build relations with the team. A good policy is to get at least two points of view to validate reported problems.

Just because no one reports problems does not mean everything is okay. Rather than asking people about project “status,” sometimes it is better to ask them about “life”—what is going well, what not so well, and what resources or help they need. Rather than talking to everyone, the project manager concentrates on those whose tasks have traditionally had the most problems. Signs that problems might be brewing include members not participating in meetings, avoiding discussions, or giving conflicting reports.

Technology

Project managers also monitor work using technology—websites, video- and audio-conferencing, email, and cell phone. Many project management software products take advantage of web-enabled technology whereby managers and workers display and update plans and reports on interactive websites, which is especially well-suited for situations where the project team and stakeholders are geographically dispersed. Putting information on a project website via Internet or intranet affords the benefits of immediate information availability, rapid and easy communication between workers, and information that is current and communicated in real time. More is said about this in the Appendix to Chapter 6 and in Chapter 18.



See Chapter 6
and 18

Video- and audio-conferencing can also be effective but require the appropriate technical facilities and careful scheduling so as not to waste people’s time. Once a foreign concept, most project professionals became familiar with this technology during the Covid-19 pandemic.



See Chapter 20

In long-distance, international projects, a good means for project monitoring is frequent one-on-one telephone conversations; these allow the project manager to gauge tone of voice, probe details, and get real-time feedback. Since, however, site managers and contractors might not always be truthful, the project manager additionally needs a trusted source at the site to observe work and report back progress. This is discussed in Chapter 20.

These days, project managers everywhere rely on cell phones and tablets for on-site communication. Increasingly, they adopt technologies associated with the so-called Internet of things (IoT) and Fourth Industrial Revolution (Industry 4.0), including use of “big data” with predictive and prescriptive analytics and artificial intelligence to assess and guide projects.

A rule of thumb, not only for monitoring work, but for all project communication is this: the more sensitive or important the issue, the lower the technology to communicate it. Important/sensitive issues require on-site, face-to-face meetings. Less important and less “sensitive” issues can be addressed on the telephone. Email is reserved solely for non-sensitive or minor issues.

Formal reports and documents

Managers also rely on written reports and other documentation (so-called **project artifacts**) to monitor projects. On a large project, work package leaders and supervisors send the project manager weekly or monthly reports about work completed, current and forecasted costs, and updated completion schedules. In general, information accuracy in all reports depends partly on the number of channels through which the information had to pass to get from the source to the receiver: the more channels, the lower the accuracy. Knowing this is why project managers also gather information through on-site reviews, walking around, talking to people, and making firsthand observations.²

Others besides the project manager also monitor the project and need to be kept informed about its progress. To update company management about project progress, problems, and actions taken, the project manager sends weekly or more frequent summary reports. Other stakeholders (customers, activist groups, public agencies, stockholders) also need to be kept up to date. To the customer especially, the project manager sends frequent reports about progress and the impacts of any changes in project scope, schedule, and cost. No matter who is formally charged with communicating with the customer, the project manager must personally ensure that the customer is always well-informed and be available to answer the customer’s questions and requests. Frequent, honest communication with the customer and other stakeholders builds trust and avoids “surprises.”



See Chapter 4

The kinds of review meetings and documented reports, and details about their contents, formats, schedules, and participants are specified in the **project communication plan**, described in Chapter 4. Given the volume of reports generated, many projects use a **documentation management system** (DMS) to ensure the required documents are created, conform to standards, and organized and stored for easy access. A computerized DMS might be needed to track, store, access, and update versions of digital documents.

13.3 Control Emphasis

In non-project situations, work performance is measured with *variance analysis* that compares money spent for work with money budgeted. In project situations, simple variance analysis is inadequate.

Example 13.1: Cost Variance Analysis

Consider the following weekly status report for a software development project:

Budgeted cost for period = \$12,000	Actual cost for period = \$14,000	Period variance = \$2,000
Cumulative budget to date = \$25,000	Cumulative actual cost to date = \$29,000	Cumulative variance = \$4,000

The report indicates apparent budget overruns for both period and cumulative costs, with to-date cumulative costs overrun at \$4,000. But because we do not know how much work has been completed, it is impossible to determine if the project is actually over-budget.

Suppose \$25,000 was the amount budgeted for completing 50 percent of software development. If 50 percent of the work had been completed as intended, then the project would, in fact, be over budget, and something would have to be done to reduce the \$4,000 overrun. But suppose only 30 percent of the work had been completed; in that case the project would clearly be over budget (and behind schedule too), and further cost overruns could be expected. As a third possibility, suppose 70 percent of the work had been completed—substantially more work than the 50 percent expected, which means the project is likely *not* over budget and is possibly under budget given the amount of work performed.

The point of the example is that for assessing project status, tracking cost is not enough; project scope, quality, procurement, and schedule must also be tracked and controlled.

Scope control

Projects tend to grow over time because of changes and additions to the scope, a phenomenon called **scope creep**. These changes are a reflection of alterations to the project requirements or work and are usually accompanied by increases in time and cost. The aim of scope control is to identify changes to requirements or work, either that have been requested or have occurred, ensure they are necessary or beneficial, manage their implementation, and do whatever possible to restrict or delimit them. Scope control is implemented through the **change control system** and **configuration management**, described later.

Quality control

Quality is controlled through managing the work to achieve the desired requirements and specifications, and taking preventive measures to correct, reduce, or eliminate work errors and mistakes. Discussed in Chapter 10, it starts with a **quality management plan** that specifies what is necessary to ensure quality results, including tests, inspections, and reviews. In technical projects, progress toward meeting requirements is guided using a methodology called **technical performance measurement (TePM)**, discussed later.

Procurement control

The project manager is responsible for the quality, schedule, and cost of all items procured for the project. Often, accompanied by the procurement manager, she will visit subcontractors and suppliers to inspect their facilities for ability to meet requirements. After the project is underway, she monitors progress by visiting their work sites and reviewing status updates, and does whatever is necessary to prompt or assist them when problems arise. For major outsourced items and services, she prepares contingency plans that include possible contractual provision to transfer work to other contractors/suppliers in case the original ones encounter unrecoverable problems. These contingencies are addressed in the **procurement plan** and the **project risk plan**.

Cost control



See Chapter 9

Cost control refers to tracking cost variances, containing cost changes, and preventing unauthorized or inappropriate expenditures. It identifies where and why variances have occurred at both the work-package and the project levels, sometimes with the aid of a project cost accounting system, described in Chapter 9. The project manager frequently reviews project costs, assesses work completed, and updates the project's expected completion cost and completion date.

Schedule control

Even the most carefully planned projects fall behind schedule for reasons such as scope changes, weather problems, and resources shortages. Despite these, project managers employ a variety of strategies and methods to keep the project moving toward the targeted requirements, schedule, and budget. These are discussed next.

Time buffers, fever charts, and budget reserves

A **time buffer** or **schedule reserve** is an amount of time added to the expected project duration to account for uncertainty. To implement a time buffer, extend the computed finish date by the buffer amount. If the estimated late finish date is July 31 and the time buffer is 4 weeks, the target finish date is August 31.

Time buffers are an aspect of **critical chain project management (CCPM)**, which prescribes locating buffers at the end of the project critical chain and the ends of all sub-chains feeding into the critical chain. Once a project is underway, the amount in the buffer is “consumed:” i.e., each time a task on the critical chain is delayed, it “consumes” the buffer time. The more of the buffer consumed, the more likely it will be exhausted and the target finish date will be overrun. Hence, the project should be managed to minimize time buffer consumption. CCPM is discussed in Chapter 8.



See Chapter 8

Buffer consumption is tracked with a **fever chart** that shows the percent of project buffer consumed versus the percent of critical chain completed (Figure 13.1). Early in a project, little buffer time will have been consumed; as the project progresses, the amount of buffer consumed will increase, and the plot on the graph will rise diagonally. Monitoring the chart enables the project manager to gauge whether the project will be completed early, on time, or late. A sharp upward trend, for example, indicates that the project is stalled—little progress is being made on critical chain tasks. In a healthy project the slope of the line is shallow so that much of the buffer remains unconsumed by project completion. Completing a project with

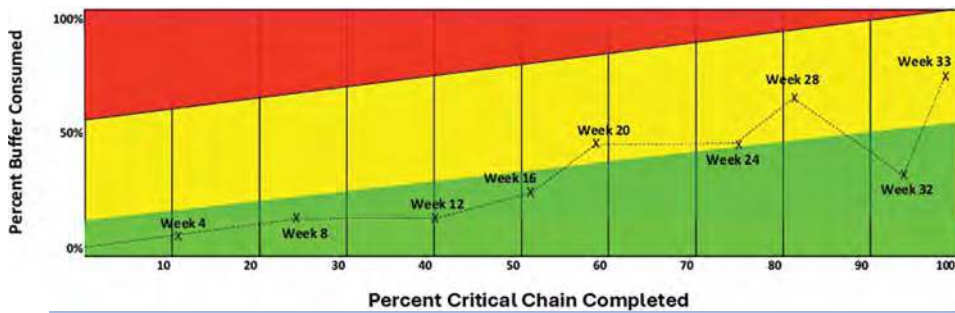


Figure 13.1

Fever chart: percent of buffer consumed versus percent of critical chain completed. The chart is divided into green, yellow, and red regions to denote project status.

some buffer remaining is equivalent to completing it *ahead* of the target finish date. Thus, the goal is to manage the project to minimize buffer consumption; the less buffer consumed, the further ahead the project.

The chart has three zones. A plot in the green zone means the project is on track. One in the yellow zone indicates potential for the project to overrun its target date; in the red zone means strong potential. The chart is updated weekly and quick action taken whenever the plot veers into the yellow or red zones. Tasks responsible for consuming the buffer are identified quickly so managers can divert more resources to them or decouple them from the critical chain.

Example 13.2: Fever Chart Computation

Points on the fever chart are computed as follows. Each week:

1. Estimate Percent Completed for each open task on the critical chain (CC). Multiply this percent by the estimated weeks needed for each task to get the Weeks Completed.
2. Sum the Weeks Completed for all open and closed (finished) tasks on the CC; this gives CC Weeks Completed. Then divide this sum by the estimated length of the entire CC to get Percent CC Completed (x-axis).
3. Compute: Weeks Buffer Consumed = Elapsed weeks to-date – CC Weeks Completed.
Compute: Percent Buffer Consumed (y-axis) = Weeks Buffer Consumed/Project buffer length

The above procedure can be modified using days. Results would be the same but more precise.

As an example, suppose the data points in the fever diagram in Figure 13.1 were derived from the project in Figure 13.2. Project status was assessed every 4 weeks (ordinarily status should be assessed every week, but 4 weeks is used here to save space). Based on

the Task Percent Completed, the Percent CC Completed and Percent Buffer Consumed are computed as shown in Table 13.1.

Per the table, last line, the project is completed in 33 weeks. The amount by which the project finished ahead of target is $36 - 33 = 3$ weeks.

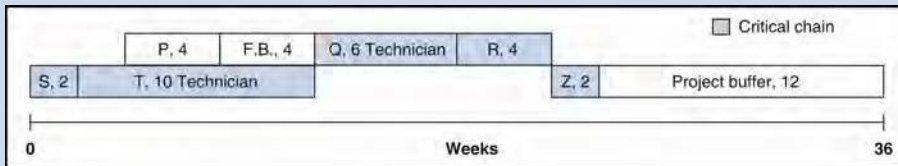


Figure 13.2

Critical chain = 24 weeks. Project buffer = 12 weeks.

Table 13.1 Computation of percent buffer consumed.

Week	Tasks & Percent Completed	Weeks Completed	Percent CC Completed	Weeks Buffer Consumed	Percent Buffer Consumed
4	S 100%; T 10%	$2 + 0.1(10)$	12.5%	$4 - 3 = 1$	8.3%
8	S 100%; T 40%	$2 + 0.4(10) = 6$	25%	$8 - 6 = 2$	16.7%
12	S 100%; T 80%	$2 + 0.8(10) = 10$	41.7%	$12 - 10 = 2$	16.7%
16	S, T 100%; Q 10%	$2 + 10 + 0.1(6) = 12.6$	52.5%	$16 - 12.6 = 3.4$	28.3%
20	S, T, 100%; Q 30%	$2 + 10 + 0.3(6) = 13.8$	57.5%	$20 - 13.8 = 6.2$	51.7%
24	S, T, Q 100%	$2 + 10 + 6 = 18$	75%	$24 - 18 = 6$	50%
28	S, T, Q 100%; R 50%	$18 + 0.5(4) = 20$	83.3%	$28 - 20 = 8$	66.7%
32	S, T, Q, R 100%; Z 50%	$18 + 4 + 0.5(2) = 23$	95.8%	$32 - 23 = 9$	41.7%
33	S, T, Q, R, Z 100%	$22 + 2 = 24$	100%	$33 - 24 = 9$	75%

The fever chart is a way to manage time buffers. The following example shows a way to manage time and cost buffers.

Example 13.3: Budget Reserve for the Mars Pathfinder Project³

The goal of NASA's Pathfinder Project was to land on Mars a skateboard-sized, self-propelled, six-wheel rover named Sojourner that would travel the terrain and send back photos and scientific data (Figure 13.3). The project's **budget reserve** (a cost buffer) was \$40 million—30 percent of the total budget (a large percentage, but common in risky technological projects), and its **time reserve** (or time buffer) was 20 weeks, about 13 percent of the project's 37-month design, build, and test schedule.

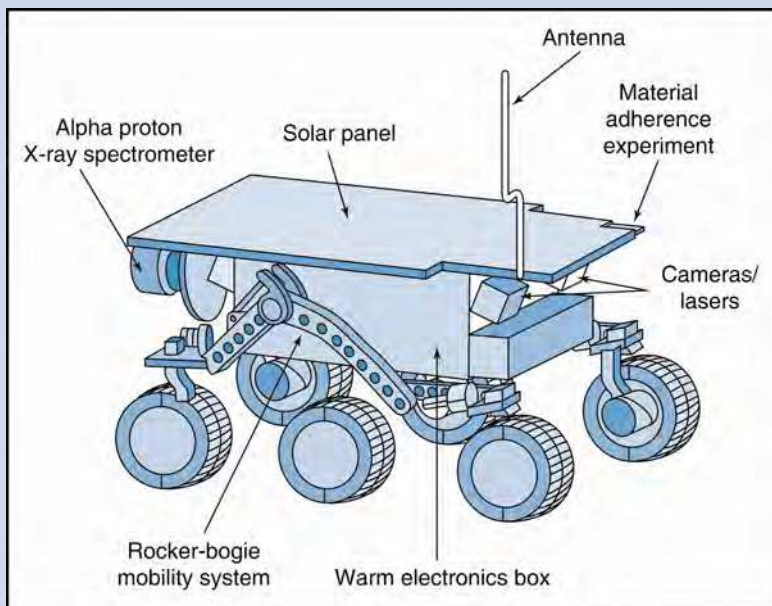


Figure 13.3
The Mars Pathfinder rover, named *Sojourner*.

Once the project was underway the question arose: How should the reserves be used? Using them too freely and too early will leave nothing remaining for later. Using them too stingily will stifle progress, increase risk, and result in leftover reserves that might have been put to good use. Management adopted the guideline to delimit the schedule reserves available for use in *each project period*. For example, *none* of it was to be used (no slippage allowed) at the start of system assembly and test. If problems arose, the guideline was to commit whatever budget reserves necessary to keep the project on schedule. (Time was a strategic issue since the launch date had to coincide with the exact relative positioning of Earth and Mars.)

The project was a success. Pathfinder landed safely and the little rover sent back thousands of pictures. The project established a new standard by designing, building, and landing a spacecraft on Mars in half the time and at *one-twentieth the cost* of previous missions.

Sojourner sent back scientific data for three months and established a pattern of success for future NASA Mars missions. As of 2024, two NASA rovers and a Chinese rover were exploring the Mars terrain.

Prepare tasks to start early

Tasks or work packages on the critical path or critical chain should be ready to start at the earliest possible time; for that to happen, teams for each task need to know the status of the task's predecessors. Thus, in time-sensitive projects like Pathfinder, every task must provide its successor tasks with **daily status notifications** stating the expected days remaining to complete and the earliest date when successor tasks should be expected to begin work. The mandate is: *as soon as* immediate predecessors of a critical task are completed, teams assigned to successor tasks will begin work, even if they must stop working on something else. In CCPM terminology, a daily status notification to enable tasks to start early is called a **resource buffer**.

Publicize consequences of delays and benefits of early finish

Everyone—team members, contractors, and suppliers—should know the consequences of a schedule overrun and the benefits of finishing early. The project contract might offer incentives for early completion, and the budget extra money for bonuses to workers who finish early.

Example 13.4: Meeting Launch Deadlines at Microsoft⁴

Microsoft meets product launch dates by utilizing visual freeze dates, internal target ship dates, and time buffers. A **visual freeze** date is a halt imposed on the product design that affects aspects of the product's visual appearance. The freeze date usually occurs at about the 40 percent mark of the schedule. Upon reaching that date, developers lock the product and thereafter allow few if any changes to features such as menus, dialog boxes, and document windows. The freeze enables the user education group to prepare training and system documentation materials (side-items) concurrent with product final debugging and testing so the materials will be ready upon product release.

Microsoft also sets **internal target ship dates** that pressure developers to decide which product features must absolutely be included and which may be forgone—else they tend to keep adding features (bloatware) and ignore the schedule. This ensures the product will contain the minimal-necessary features and still be released on time.

To account for overlooked or poorly understood tasks, difficult bugs, and changes in features, Microsoft adds time buffers to project schedules. The buffers, which can range from 20 to 50 percent of the total target schedule time, are used exclusively to cover uncertainties and non-routine tasks. The project team strives to meet the internal ship date, which is the launch target date announced to the public *minus* the time buffer.

13.4 Work Packages and Control Accounts

Earlier chapters described the role of work packages and control accounts in project planning; not coincidentally, they are also key elements of project control. Each control account consists of one or more work packages, each like a contract for a specific job with requirements, work description, budget, schedule, etc. Thus, each work package and control account is a focal point for data collection, progress evaluation, problem assessment, and corrective action. Large projects may be composed of hundreds of work packages, making it difficult to identify the sources of cost or schedule overruns. An advantage of a PCAS, described in Chapter 9, is that it can sort through all the work packages and locate the sources.



See Chapter 9

Work authorization process

One aspect of project control is **work authorization** or **start–stop control**: project work is started only upon formal authorization and is stopped upon review and acceptance. It applies to the overall project and to each work package. At the project level, the project manager can begin the project only upon authorization from the customer, program manager, or top management. The project manager then authorizes managers of contractors or departments to begin, and they in turn authorize team leaders or supervisors of work packages to begin, shown in Figure 13.4. The process is a continuation of the initiation, proposal, and authorization process described in Chapter 3 and Figure 3.9.



See Chapter 3



Figure 13.4
Project work authorization process.



See Chapters 4
and 18

The same process may apply to authorizing *phases* of a project: after each phase the customer or other stakeholders evaluate the results of the phase and review the plan and risk for the next phase; if everything is acceptable, they authorize the next phase; if not, they halt the project (the so-called “phased project planning” process discussed in Chapter 4 and “gating process” described in Chapter 18).

On large projects, authorization is subdivided into the stages of **contract release**, **project release**, and **work order release** (or work requisition). After the customer awards the contract, the contract administrator prepares a contract release document that summarizes the contractual requirements and authorizes the project manager to go-ahead. The project comptroller or accountant then prepares a project release document, which authorizes project funding.

Individual work tasks or packages begin only upon receipt of a **work order** (or “engineering order,” “shop order,” “site instruction,” or “test order,” depending on the kind of work). As the scheduled start date for a task draws near, the project manager or project office releases the authorization document to the contractor or department to begin work. For simple projects or activities, verbal authorization might suffice.

Work progress and percent complete

Upon beginning work, data is periodically collected for each work package about *actual* costs and work progress. For larger projects it is entered into the PCAS, which consolidates the data and generates performance reports for each work package and department, and the entire project.

In large projects, assessing the impact of work progress on schedules is the responsibility of the departmental manager or team leader in charge of the work. Each week, the supervisor reviews the tally of labor hours for each task on timecards. She notes tasks completed and tasks still “open,” and estimates the time still needed to complete open tasks. She also estimates the **percent complete**, which is the percentage progress for the task, and records it on a Gantt chart showing percent complete for all tasks.

Figure 13.5 is an example showing the status of the LOGON project as of Week 20. Percent complete is represented by highlighting the bar for each task. Highlighting the entire bar indicates 100 percent complete; highlighting a quarter of the bar means 25 percent complete, and so on. Work packages K, L, M, N, O, P, and Q are all underway but not yet completed (open); K, L, M, and Q are behind schedule; O is ahead.

Every week, each work package leader tallies current expenses. Labor hours as reported on timecards are converted into direct labor cost. The costs for direct labor, material and level-of-effort (testing, support, etc.) for completed and open tasks are added to the costs of work from prior periods, and the sum multiplied by the overhead percentage rate. Late charges and outstanding costs are also included. The leader documents any estimated changes to budgets or schedules for remaining work, and forwards a report to the project manager showing costs of all work completed in prior periods plus work done in the current period. This information, after validation by the project manager, is entered into the PCAS, which accumulates costs to date for all work packages and prepares a summary report. Periodically the project manager reviews these reports to reassess the project and estimate the work still needed and the cost to complete the project, as described later. These provide forecasts of the project completion date and cost at completion. When a task or work package is completed, its budget is closed to prevent additional, unauthorized billing.

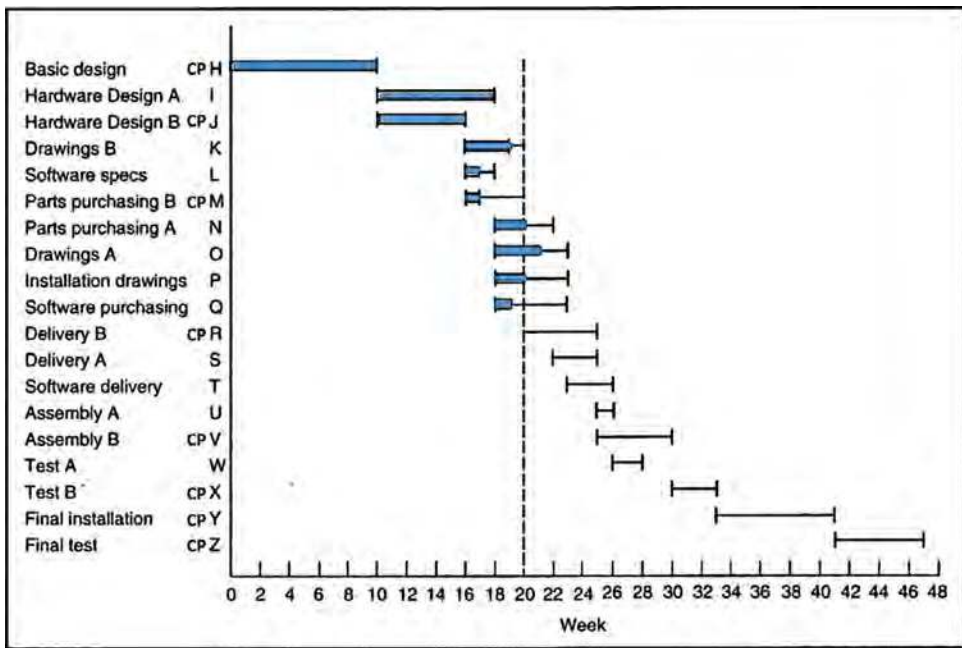


Figure 13.5

Gantt chart showing LOGON project work status as of Week 20.

Work progress metrics

How is work progress and percent complete measured? Accrued expenses and time elapsed are easy to measure but neither says much about *actual* work progress; thus, project managers must rely on other, sometimes subjective metrics. In a survey of ways to measure project performance, Thompson, an engineering management professor, identified the following.⁵

1. **Supervisors.** Managers and supervisors assess progress by direct observation, asking questions, and reviewing project reports and documentation.
2. **Milestones.** These are easily measured endpoints of tasks or transition points between tasks; examples are completion of drawings, reports, and design documents, or solutions of specific technical problems.
3. **Tests and demonstrations.** Described earlier, these can range from simple tests of system components to full-system and user acceptance tests. They are a good way to measure technical progress at intermediate stages of the project.
4. **Outside experts.** Invited experts serve on a review panel that assesses the project status by observing work in progress, talking to project personnel, and reviewing documentation.
5. **Status of design documentation.** Experienced project managers can determine when a design is nearly finished by the “completeness” of documents such as drawings, schematics, models, manuals, and test procedures.
6. **Resources requested.** A request for or change in resources may reflect progress; for example, tasks nearing completion often require special facilities, personnel, and equipment for testing and implementation.

7. **Telltale tasks.** Tasks such as concept design, requirements definition, feasibility analysis, and repeated testing typically happen early in a project; their happening later in the project signifies a lack of progress.
8. **Benchmarking** or **analogy.** Certain tasks or the entire project may be compared to similar tasks or projects as a crude way to weigh relative progress.
9. **Changes, bugs, and rework.** Because ordinarily the number of bugs, problems, change requests (discussed later), and so forth, should decrease as a project nears completion, a remaining high number may indicate lack of progress. This is discussed later under “issues tracking.”



See Chapter 14 described in Chapter 14.

13.5 Performance Analysis and Earned Value Management (EVM)

Earned value management is a quantitative approach to assessing the performance of a project or any portion of it. It involves comparing three variables: BCWS, ACWP, and BCWP. These are industry-standard acronyms; to save ink we will abbreviate them as PV, AC, and EV, respectively.

1. **PV** is the **planned value** (or *budgeted cost of the work scheduled*, **BCWS**)—the sum cost of all work and apportioned effort scheduled to be completed within a given time period as specified in the *original budget*.
2. **AC** is the **actual cost** (or *the actual cost of the work performed*, **ACWP**)—the actual expenditures so far for labor, materials, support, and administration.
3. **EV** is the **earned value** (or *budgeted cost of the work performed*, **BCWP**)—the value of the work performed so far (both fully and partially completed work packages). EV is determined by the *original budget*. Thus,
 - a. For a completed work task, EV is the same as the PV for that task.
 - b. For a partially completed work task, EV is computed as the *percent complete* for the task multiplied by the budget for the task. (Alternatively, it is computed as 50 percent of the task budget when the task is started, then 100 percent of the task budget when it is completed.)

A fourth variable, the planned **budget** for the project **at completion**, BAC, is often used as well. Application of these variables and EVM to track and assess project performance is illustrated in the following example.

Example 13.5: EV versus PV in the Parmete Company⁶

The Parmete Company has a \$200,000 fixed-cost contract to install 1,000 parking pay boxes. The contract calls for removing old pay boxes from their stands and replacing them with new ones at a cost of \$200 per box.

Parmete estimates that 25 boxes can be installed each day. At 25 per day and \$200 per box the project should finish in 40 working days with a total *PV* of \$200,000, which is the *BAC*. Also on that basis, the accumulated planned value of scheduled work (*PV*) as of *any given day* can be determined by multiplying the number of working days completed as of that day by the cost of installing 25 boxes (\$200 times 25). For example, as of Day 18,

$$PV = 18 \text{ days} \times (25 \text{ boxes}) \times (\$200) = \$90,000$$

That is to say, the project schedule and budget specify that as of the eighteenth day of work on the project, \$90,000 worth of work should have been done. Notice that PV is always associated with a specific date on the project schedule.

In contrast, the earned value (EV) for any given day represents the value of the work *actually* done in terms of the budget. In this project, EV is the number of boxes *actually* installed *to-date*, times the \$200 budgeted for each meter. Suppose, for example, that as of the eighteenth day on the project 400 boxes had been installed: thus,

$$EV = (400 \text{ boxes}) \times (\$200) = \$80,000$$

In other words, as of Day 18, \$80,000 worth of work has been performed. Now, given that \$90,000 worth of work was *supposed* to have been performed, the project is \$10,000 worth of work *behind schedule*. Notice, the \$10,000 does not represent a cost savings but rather an amount of work that should have been done but was not. It represents 50 parking boxes, or 2 days' worth of work, meaning that as of Day 18 the project is 2 days behind schedule. (The 2 days is referred to as the time variance, or TV.) Thus, EV is a translation of project cost into work progress. As of Day 18, the project has made 16 days' worth of work progress. This is represented on the graph for PV and EV in Figure 13.6.

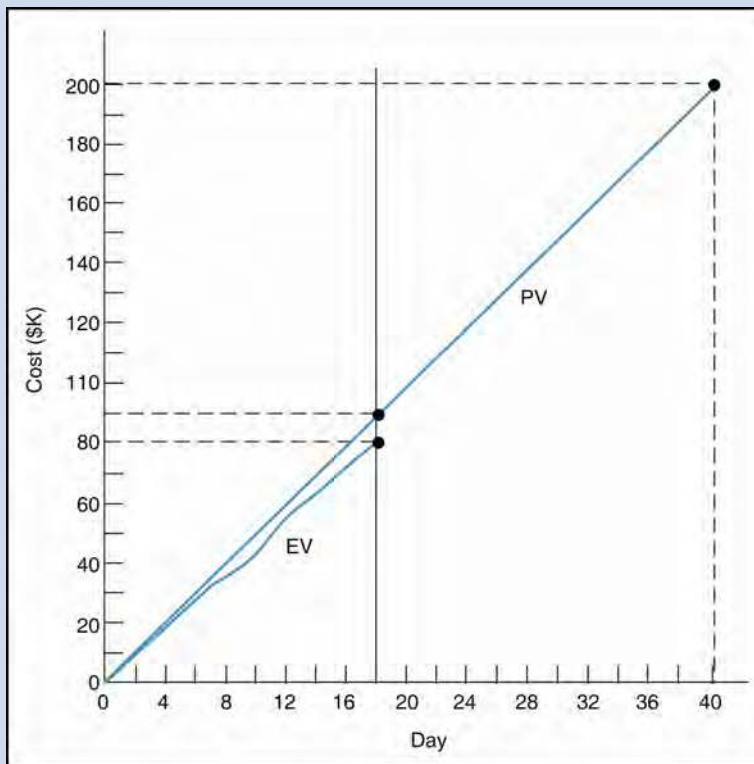


Figure 13.6
Graph of PV and EV for parking paybox project.

Besides completed tasks, *EV* should also reflect tasks started but not yet completed (open tasks). For example, suppose before quitting at the end of Day 18 the pay box installer had just enough time to remove an old box but not replace it with a new one: the work on that task was 50 percent completed. If this were meter number 401, then *EV* as of Day 18 would be the cost for the first 400 meters plus 50 percent of the cost for the 401st:

$$EV = \$80,000 + (0.50)(\$200) = \$80,100$$

Thus, the *EV* as of Day 18 is \$80,100, which represents slightly more than 16 days [$\$80,100 / (25 \times \$200) = 16.02$ days] of work completed.

The variables *PV*, *AC*, and *EV* can also be used to compute variances that reveal different aspects of a project's status. For example, assume for the LOGON Project in Week 20,

$$PV = \$512,000$$

$$AC = \$530,000$$

$$EV = \$429,000$$

Using these figures, three kinds of variances can be determined, shown in Figure 13.7.

1. **Schedule variance:** $SV = EV - PV = -\$83,000$.
2. **Time variance:** **TV.** Refer to Figure 13.7: See *EV* for Week 20, then look for which week *PV* equals this *EV* (about Week 19)]; thus, $TV = (20 - 19) = 1$ week.
3. **Cost variance:** $CV = EV - AC = -\$101,000$.

Positive *SV* suggests the project is ahead of schedule; negative *SV* that it is behind; thus, *SV* for Week 20 of $-\$83,000$ means the project is behind schedule. *TV* shows about how far behind schedule, in this case about 1 week because the *EV* of \$429,000 is roughly the value of work (*PV*) that should have been completed about one week earlier.

Negative *CV* indicates that the project is overspending; positive *CV*, that it is under-spending. Given *CV* of $-\$101,000$, LOGON is overspending.

Work package analysis and performance indexes

Figure 13.7 represents LOGON project status as of Week 20. But a complete assessment of project status should include an assessment of every work package and control account, which a PCAS can perform and provide with a figure like Figure 13.7 for each one.

Refer to Figure 13.5, which shows the status of LOGON activities as of Week 20: activities H, I, and J are completed and have been closed; activities K through Q are open and underway. Other activities have yet to begin. This gives a general overview of the status of each work package, although better measures are obtained by computing two performance indices:

1. **Schedule performance index:** $SPI = EV/PV$.
2. **Cost performance index:** $CPI = EV/AC$.

Values of *SPI* and *CPI* greater than 1.0 indicate that work is ahead of schedule and under budget, values less than 1.0 indicate behind schedule and over budget.

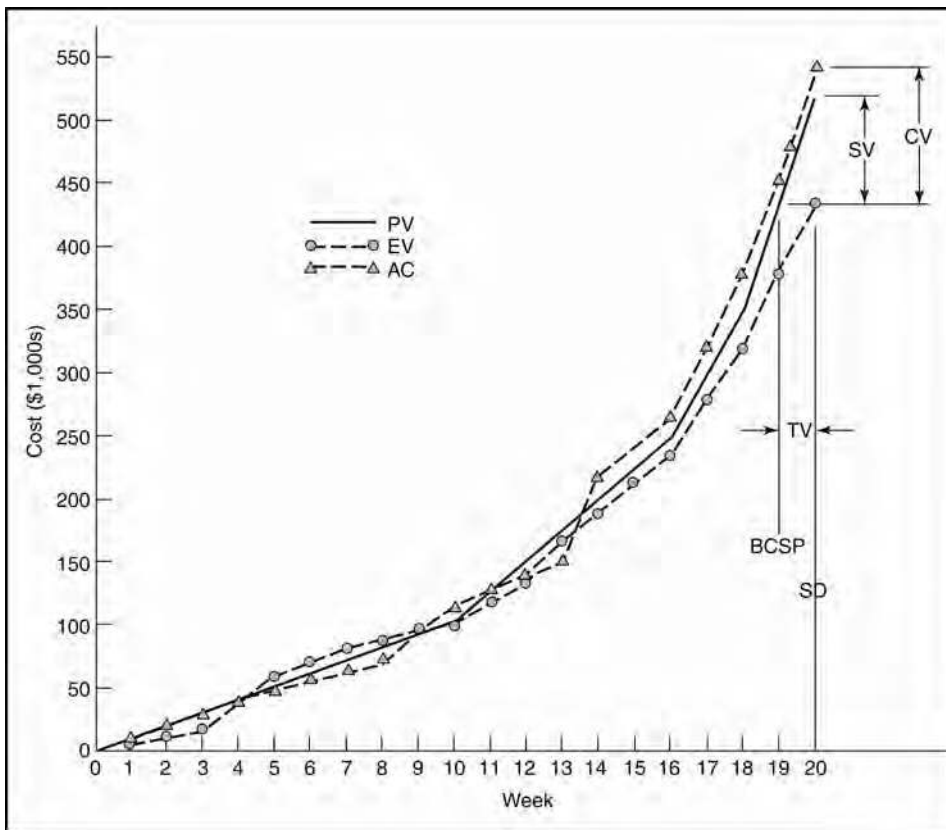


Figure 13.7
LOGON project status for PV, EV, and AC as of week 20.

Table 13.2 shows performance information for all LOGON activities as of Week 20. The CPI and SPI indices show trouble spots and their relative magnitude: L, M, and Q have fallen the most behind schedule (they have the smallest SPIs), and L and M have the greatest cost overruns relative to their sizes (they have the smallest CPIs). Overall, the project is likely “somewhat” over cost ($CPI = 0.81$) and behind schedule ($SPI = 0.84$), although maybe not, depending on whether behind-schedule activities are on the critical path.

By using project-level CPI and SPI, the project manager gets a quick “ballpark” estimate of the project’s performance to-date. Although the estimate might be somewhat inaccurate, it enables tracking of broad trends in project performance. The plot of SPI against CPI in Figure 13.8 is an example: LOGON performance started out (points 1 through 5) in the marginal and poor regions, briefly recovered, and then drifted disturbingly back to and remained in the poor region (points 11 through 20). The project manager should be searching to identify reasons why.

Variance limits

Rarely do actual and planned performance metrics coincide, so nonzero variances are more the rule than the exception. This leads to the question: What amount of variance is necessary to justify taking action?

Table 13.2 LOGON performance report week 20 cumulative to date.

Activity	PV	AC	EV	SV	CV	SPI	CPI
H*	100	100	100	0	0	1.00	1.00
I*	64	70	64	0	-6	1.00	0.91
J*	96	97	96	0	-1	1.00	0.99
K	16	12	14	-2	2	0.88	1.17
L	36	30	18	-18	-12	0.50	0.60
M	84	110	33	-51	-77	0.39	0.30
N	40	45	40	0	-5	1.00	0.89
O	20	28	24	4	-4	1.20	0.86
P	24	22	24	0	2	1.00	1.09
Q	32	16	16	-16	0	0.50	1.00
Project	512	530	429	-83	-101	0.84	0.81

*Completed

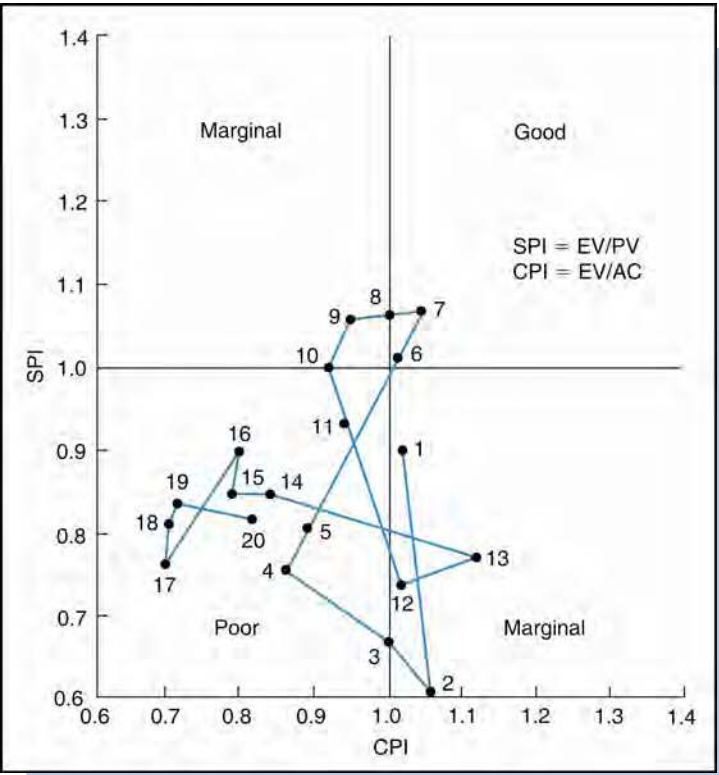


Figure 13.8
LOGON project cost/schedule performance plotted for weeks 1 through 20.

Table 13.3 Example of variance limits.

Work Package A	Variances greater than \$2,000
Work Package B	Variances greater than \$18,000
Department C	Variances greater than \$6,000
Department D	Variances greater than \$38,000
Project	Variances greater than \$55,000

Table 13.3 shows **variance limits** for different project levels—work package, department, and project. Only when a variance exceeds a limit are corrective measures considered. In some projects, the limits can vary. In research projects, the limits are initially set high but later are lowered as the project moves forward. This coincides with project risk, which typically starts out large but diminishes as the project progresses.

Use of both upper and lower variance limits on costs and schedules helps identify situations where work quality is in question. A project running ahead of schedule and under budget—an apparently desirable situation—might in fact be riddled with cutting-corners and shoddy workmanship. For technical performance, both upper and lower variance limits on technical requirements are necessary: lower limits to ensure minimal requirements are met, upper limits to discourage excessive or unnecessary development work.

Updating task/work package time estimates

Following each progress review it might be necessary to update the scheduled completion dates of individual tasks or work packages. In general,

$$\text{Forecasted finish date for a task} = \text{Start date} + \text{Time remaining}$$

where **time remaining** is determined in two ways. The first is to compute it as a function of the days worked so far and current progress, that is:

$$\text{Time remaining} = \frac{\text{Percent of task remaining}}{\text{Percent progress per day}}$$

where

$$\text{Percent progress per day} = \frac{\text{Percent of task completed so far}}{\text{Days worked on task so far}}$$

The other way is to simply accept the opinion of a reputable source (“it’ll take me another 5 days to finish the job”). This method often yields a more accurate estimate than the first because it accounts for any recent changes in the rate of work progress. See Example 13.6.

Either of these methods is okay for updating completion dates for individual tasks but not for the project since, as mentioned, the latter depends on whether delayed tasks are on the critical path. In general, EVM alone should not be used to predict the project completion date.

Example 13.6: Revising Task Completion Date

A task starts on July 10 and is planned to take 12 days (weekends included). After 5 working days (end of July 14), the leader estimates that the task is 20 percent complete. If the rate of progress stays the same, what is the forecast completion date of the task?

Five days of work represents an estimated 20 percent complete, so the work progress is 20 percent/5 = 4 percent per day. Thus, to complete the remaining 80 percent should take $0.80/0.04 = 20$ working days. The revised completion date is July 15 + 20 = August 4.

Now, assume instead that the team leader believes that the remainder of the task will proceed much faster than 4 percent per day, and that, at most, 10 more working days will be required. If the team leader's estimate is considered credible, the revised completion date would be July 25.

Estimated cost at completion

Periodically the project manager prepares a *to-complete* forecast, which is an estimate of the cost remaining to complete the project, computed as:

$$ETC \text{ (Estimated cost To Complete project)} = (BAC - EV) / CPI$$

where BAC is the budgeted total cost for the project (= total PV at target completion).

This forecast plus the actual project cost to-date yields a forecast of the project total cost at completion,

$$EAC \text{ (Estimated cost At Completion)} = ETC + AC$$

The following two examples illustrate.

Example 13.7: Forecasting ETC and EAC for ROSEBUD Project

Figure 13.9 shows the ROSEBUD project Gantt chart with percent complete and EVM metrics as of Week 13. Given this information, what is the likely additional cost for the project to be completed, and how much will the project cost at completion?

The value of the work completed so far (*EV*) is \$268,081. The total budgeted amount for the project (*BAC*) is \$344,205, hence the value of the *work remaining* is $BAC - EV = \$76,124$.

The cost performance for the project so far is:

$$CPI = EV / AC = 268,081 / 288,657 = 0.9287$$

This means the project is receiving less than 93 cents value for each dollar spent. At that rate, the estimated to-complete cost is:

$$ETC = 76,124 / 0.9287 = \$81,968.$$

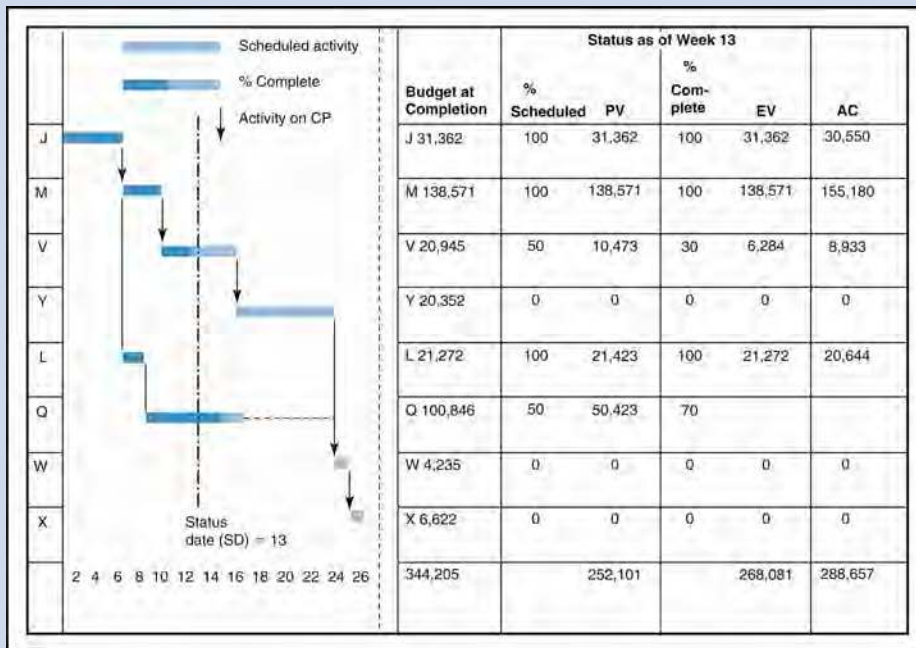


Figure 13.9
ROSEBUD project status as of Week 13.

meaning another \$81,968 will be spent. Since \$288,657 has already been spent, the project estimated at-completion cost is:

$$EAC = ETC + AC = 81,968 + 288,657 = \$370,625.$$

The variance at completion, $BAC - EAC = \$370,625 - \$344,205 = \$26,420$, represents a cost overrun of 7.7 percent.

Notice that according to *EV*, the project is slightly ahead of schedule ($EV = 268,081$) > ($PV = 252,101$); it might, however, be *behind* because Activity V is on the critical path and appears roughly 1 week behind schedule according to the Gantt chart.

Example 13.8: Forecasting ETC and EAC for LOGON Project

From discussion earlier in the chapter, for the LOGON Project at Week 20,

$$\begin{aligned} CPI &= 429,000 / 530,000 = 0.81, \text{ thus} \\ ETC &= (990,000 - 429,000) / 0.81 = \$692,000, \text{ and} \\ EAC &= 530,000 + 629,000 = \$1,222,593 \end{aligned}$$

Lacking other information, a crude estimate of the revised project completion date is shown on Figure 13.10 by extending the *EV* line, keeping it parallel to the *PV* line, until it reaches the level of *BAC*, \$990,000. The *horizontal* distance between the *PV* line and the *EV* line at *BAC* is roughly the schedule overrun for the project; on Figure 13.10 the estimated completion date is Week 50, roughly 3 weeks later than the initial target of Week 47.

As mentioned earlier, however, this revised completion date remains to be verified since any project delay will depend on whether the behind-schedule activities are on the critical path. From earlier discussion we know that Activity M is on the critical path; since it is 3 weeks behind schedule, the LOGON project is likely also 3 weeks behind.

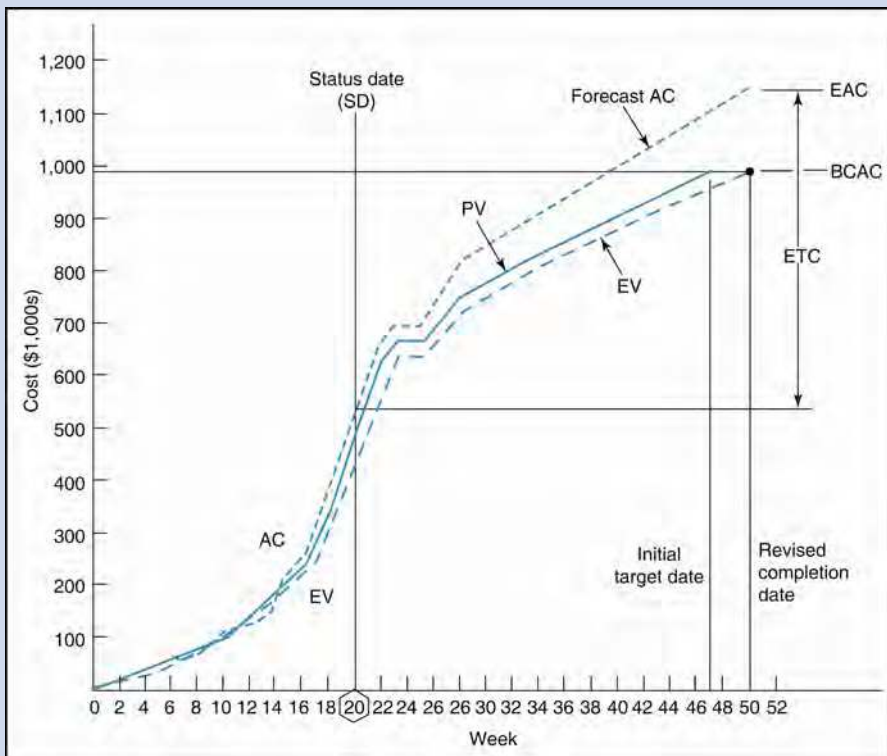


Figure 13.10
LOGON project status and completion week and cost forecasts as of Week 20.

Figure 13.10 shows another line, "Forecast AC," which is an extension of the current AC line up to the EAC level of \$1,159,630 and the revised completion date of Week 50. This gives a running estimate of how "actual" costs might rise until project completion.

It should be noted that the at-completion estimates assume that conditions and resources will neither improve nor worsen, which the project manager should question. Given the size of the current overrun (\$101,000 as of Week 20) and that the project is less than half finished, what is the likelihood of finishing all remaining work for \$692,593 without

additional overruns? (If it seems unlikely, the figure should be revised again according to best guess estimates.) Also, what is the likelihood the project will finish by the revised estimate of Week 50? As of Week 20, the *EV* is equivalent to the *PV* at Week 19, which means, in terms of budgeted cost, work remaining on the project is

$$47 \text{ weeks (target date)} - 19 \text{ weeks} = 28 \text{ weeks}$$

However, given the current $SPI = 0.84$, it seems more likely the project has $28/0.84 = 33.3$ weeks remaining. The project is now in Week 20, so the revised completion date would be $20 + 33.3 = \text{Week } 53.3$.

EVM shortcomings

Earned value metrics must be treated with caution. For example, negative *CV* (overrun) can arise because of overhead charges that originate outside the project and have no bearing on project performance. Similarly, positive *CV* (underrun) can occur because bills have yet to be paid. *EV* assumes that expenses occur at the time of work and uniformly throughout a task or project. Whenever payments occur in periods other than expenses are incurred or budgeted (as happens), the *CV* is skewed. This leads some companies to apply *EVM* to some cost factors (e.g. labor-costs for their own employees) and not others (e.g. procured items requiring advance payment). In the end, the sources of costs must be scrutinized to verify apparent overruns or underruns.

Assessing progress and forecasting completion dates using *EV* methods should be accompanied by additional information, since project delays depend on activities on the critical path. The *EV* method relies on estimates of percent complete. Accurate percent complete estimates are possible whenever work can be measured in uniform units (e.g. number of bricks or miles of asphalt laid, number of identical fixtures installed, etc.), but not when work output (e.g. drawings produced, or lines of code written) cannot be measured in uniform units (different drawings or code lines require different times to produce).⁷ In short, *EVM* metrics should be used cautiously and with information from other sources.

EVM and critical chain project management (*CCPM*) are sometimes used in combination—although the purpose of each is different. *EVM* is a cost monitoring and reporting method; as mentioned, it does not distinguish “critical” activities on the schedule. *CCPM* is a scheduling tool; it does not address costs. The methods occasionally give conflicting signs about project performance (*SPI* or *TV* in *EVM* vs. buffer consumption in *CCPM*). One way to address this is to use each for different purposes—*EVM* for cost tracking and reporting (perhaps as required by a customer), buffer consumption for resource allocation, and scheduling decisions.⁸

Technical performance measurement

Besides costs and schedules, project performance depends on meeting technical requirements. **Technical performance measurement, TePM** (not to be confused with *TPM*—traditional project management) is a method for tracking technical objectives or requirements over time.

Its purpose is to provide (1) a best estimate of current technical performance and progress to date, and (2) an estimate of technical performance at project completion. Both estimates are based upon results from models, simulations, tests, or demonstrations.⁹

To perform TePM, first specify the technical performance metrics that are *key indicators* for the end-item system—i.e., metrics tied to customer requirements and representative of major performance drivers. A large-scale system might have a dozen high-level metrics, in which case it is necessary to first define the design parameters upon which each technical measure depends and to set required values for these parameters. Examples of performance metrics include:

Availability	Capacity	Size/Space
Back-up utility	Response time	Reliability
Safety	Security	Power/thrust
Speed	Setup time	Interface compatibility
Survivability	Durability	Interoperability
Maintainability	Range	Simplicity/complexity
Flexibility	Variance	Signal-to-noise ratio
Cycle time	Cost	Trip time
Efficiency	Utilization	Idle time
Output rate	Error/defect rate	Weight

Periodically during the project, performance is calculated or measured and compared to targets. Initial metrics are based upon estimates from computation, modeling, and simulations; later metrics are derived from test and demonstration results on actual hardware and software. Estimates and actual metrics of a technical objective are plotted on a TePM chart showing progress toward achieving the objective. If actual performance for one part of the system *exceeds* the target or objective by some margin, then sometimes that margin can be traded-off against targets for other parts of the system where performance is lacking or at risk. This is illustrated next.

Example 13.9: TePM for Design Tradeoff Decisions

Based on Example 11.5 in Chapter 11, design target weights for components of a space-craft navigation system were set at 44 pounds for Subsystem A and 26.4 pounds for Subsystem B. **Design margins** were also set for the subsystems to cover the risk of not meeting these targets; margins are amounts by which target values can be missed and still achieve system requirements. The margins for A and B are 6 pounds and 13.6 pounds, respectively.

The TePM chart in Figure 13.11 shows design progress (actual versus target values) for the two subsystems; such charts are used to make design tradeoff decisions. It also illustrates the concept of **baselines**, which are target values against which progress or performance is measured. The figure shows current performance and design targets at three project milestones:



See Chapter 11

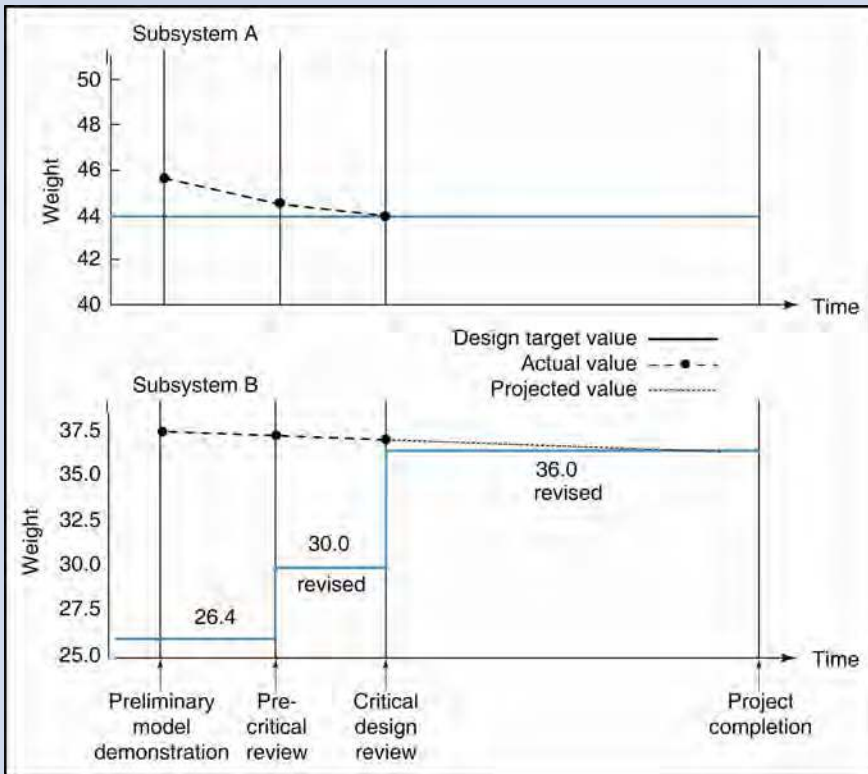


Figure 13.11
Time-phased TePM charts for subsystems A and B.

1. At the time of the **preliminary model demonstration**, the actual measured weights for both subsystems were too high compared to the baseline targets, although Subsystem A was relatively much closer to its target than Subsystem B.
2. By the time of the **pre-critical review**, Subsystem A had been improved and was close to its target weight, although Subsystem B was still far away from its target. It was clear that Subsystem A might be able to meet its target of 44 pounds but Subsystem B would *not* meet its of 26.4 pounds. The decision was made to reduce Subsystem A's unused design margin by 3.6 pounds and increase Subsystem B's target by that amount to 30 pounds.
3. As of the **critical design review**, Subsystem A had met its target, but Subsystem B still lagged behind its; further, it was anticipated that only limited additional improvement in B was possible. The decision was made to transfer the remaining 2.4 pounds from Subsystem A's unused design margin to Subsystem B's target and to use 3.6 of Subsystem B's remaining design margin, both to increase Subsystem B's design target to 36 pounds. The dotted line for Subsystem B beyond the critical design review represents improvement still necessary to achieve Subsystem B's revised target value.

13.6 Visual Management: Kanban Board and Burn Chart

The tracking and control methods described thus far are primarily for use in TPM (traditional, waterfall) projects. For APM projects, the hallmarks of which are frequent change and minimal documentation, simpler, less analytical tools are adequate. Rather than relying on formal data collection, analysis, and reporting, APM projects employ simple **visual management** tools for tracking and control. The project team can track the work it has done and has yet to do by referring to posted charts and diagrams. In this way it can gauge its performance versus goals and assess the impact of adding or removing work on the schedule. The most common tools are **kanban boards**, **burn charts**, and **cumulative flow diagrams**. These are explained more



See Chapter 14

in Chapter 14, but Example 13.10 is a preview.

Example 13.10 Burn Chart and Kanban Board

The Gonzo project team has divided the project into 20 tasks of roughly the same size and requiring about the same amount of time. As a measure of the size of each task compared to the project, it uses “story points.” The entire project is assigned a value of 100 *story points*, and each task is worth 5 points. The project is scheduled for 20 workdays, meaning that if the work were done uniformly, 5 story points worth of work would be completed each day. The dotted line on the *burn chart* in Figure 13.12 shows the *expected cumulative work progress*, totaling 100 points after 20 days.

To track progress, each day the project team checks its work and adds 5 points on the chart for each task it has completed. For example, on Day 1, say, two tasks were completed, so 10 points were added to the chart. No tasks were completed on Day 2, but two were completed on Day 3, so 10 points were added then. The next task is completed on Day 6, and 5 points more are added. The chart shows 45 points accumulated as of Day 7, but no more until Day 13 when 10 points were added. The 55 story points accumulated as of Day 13 is less than expected (per the dotted line), so the project is behind schedule. The chart in Figure 13.12 is more accurately called a **burn-up chart** because it shows *cumulative* points. Chapter 14 describes burn charts in more depth and gives an example of a **burn-down chart**.

The team on Gonzo project also uses a kanban board (aka **whiteboard**) to track progress. Whereas the burn-up chart represents tasks completed, the kanban board also shows status for tasks not completed. For example, the board in Figure 13.13 shows four status categories: Backlog (not started), Underway, Review, and Completed. Each task is represented on the board with a sticky card placed in one of the columns, depending on its status. Whenever a task changes status, it is moved to the next column. When a task is fully completed, its card is moved to the Completed column and 5 points are added to the burn-up chart. Suppose Tasks 1 and 2 were moved to the Completed column in Day 1, so 10 points were added to the burn-up chart in Figure 13.12. If no cards were added to the Completed column on Day 2, then no points would be added to the burn-up chart. If Tasks 3 and 4 will be moved to the Completed column on Day 3, 5 points for each will be added to the chart. Chapter 14 describes the use of kanban boards more fully.

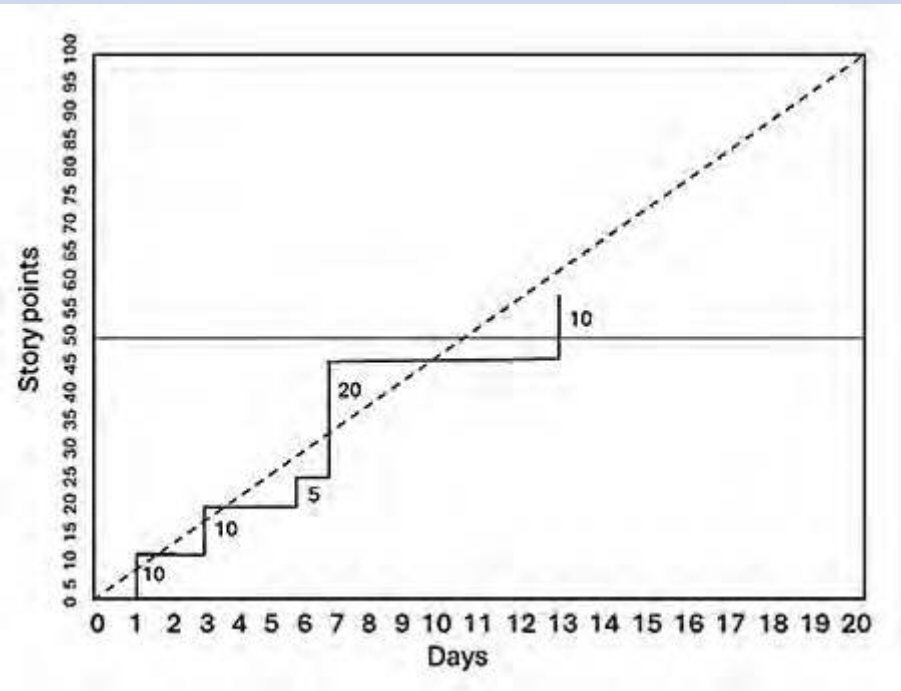


Figure 13.12
Burn-up chart showing progress as of day 13.

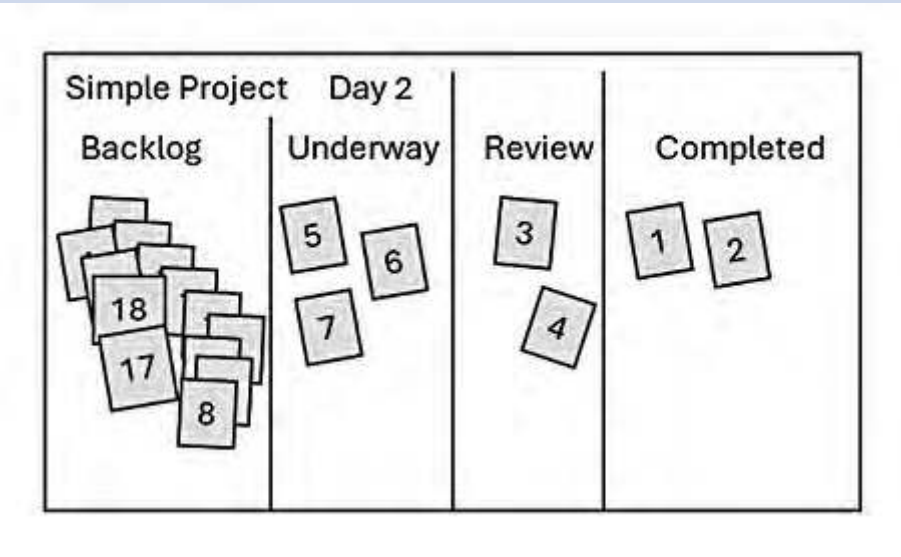


Figure 13.13
Kanban board (also known as a white board).

Kanban boards and burn charts are **information radiators**. Also called **big visible charts, BVCs**, they are created and frequently updated by the team and located at the workplace where everyone can see what the team is working on, its progress, and identify any problems or obstacles. They display whatever the team deems important and serve as a gathering place for team meetings. The concept of BVCs derives from the use of KPI boards and team huddles in lean production and Kanban project management, discussed in the next chapter.

13.7 Issue Management

An **issue**—an emergent problem, barrier, or matter in dispute—can stem from anywhere, but is something that must be resolved. Issues that originate from earlier identified risks can be resolved with the mitigation or contingency responses in the risk plan. Most issues, however, will not have been anticipated and must be dealt with on the spot. Not every detail can be addressed in the project plan or every circumstance anticipated. Issues will arise and require follow-up actions.

Issues, risks, and changes are related: an issue can result in a change request or create a new risk, and a materialized risk (a risk that has arisen or is certain to arise) or a change request can result in an issue. Each requires an “action”—i.e., a response, either a change request, issue resolution plan, or risk mitigation strategy. A project will encounter numerous issues, and for each the project manager must assess its impact on the schedule, budget, other tasks, resources, and end-item quality.

Like risks and changes, issues need to be managed. Every issue needs to be documented, prioritized, tracked, resolved, and closed out. Like controlling changes (discussed next), issue management involves the steps of identification, documentation, analysis/evaluation, communication, action, monitoring, and closure. These can be described in the **issue log** (or **issue register**) in Figure 13.14. The log, which is a record of all issues encountered, is retained throughout the project and updated whenever issues newly arise or change status.

Shown in the figure, each issue has an ID (simple alpha-numeric identifier), *issue name*/description, date raised, and originator. The issue type categorizes the issue according to function, origin, action, or impact. Some companies specify issue type as “technical,” “organizational,” “stakeholder,” or “procurement”; others use “Risk,” “Change request,” or “Emergent

Project LOGON												
Project Manager: Frank Wesley												
Last update:												
Issue ID	Issue Name/Description	Date Raised	Issue Originator	Issue Type	1-5 Priority	1-5 Severity	Impact	Due Date	Action	Assigned To	Current Status	Date Resolved

Figure 13.14
Issues log.

(problem).” *Impact*, if known, is the possible or certain consequence of the issue if not properly resolved.

Each issue is assessed for its importance to the project and/or the end-item by the project manager, project team, or others, and is assigned a *priority* (e.g. 1–5) that is based on the *severity* (1–5) of the issue and its impact. The result of the assessment is a recommended or specified action to resolve the issue with a *due date* and *assigned-to person*.

The status of all issues is monitored frequently. Weekly status meetings start with a review of the issue log and an update of each issue’s status, which might be indicated by a single letter representing, for example: **N**ot started, **A**nalysis, **X**—cancelled, action in **P**rogress, **C**ompleted/resolved, or **E**scalated. The assigned-to person is responsible for reporting on and providing substantiating proof of the status. The *date resolved* is recorded for closed-out issues. Stubborn issues are “escalated,” i.e., they are referred to a higher authority—senior management, sponsor, or customer—whoever has the authority, resources, or capability to resolve the issue.

For medium and large projects, issue management follows a process roughly analogous to risk management (substitute “issue” for “risk” in Figure 11.1) and change control (substitute “issue” for “change” in Figure 13.16, described later).

One way to gauge project progress is to monitor the status of issues. All projects encounter issues, but “healthy” projects resolve them expediently. In contrast, when issues remain unresolved, the issue backlog grows and becomes increasingly difficult to handle. Figure 13.15 shows the number of unresolved issues (status N, A, P, or E) at a particular time in the project and how long they have been on the issue log. The shallow decline in the top curve represents a “troubled” project: there are numerous unresolved and most are many weeks old. The bottom curve represents a healthy project where issues are resolved rather quickly.

13.8 Change Control

During the project, changes to the end-item system and project plan are inevitable. In an APM project, changes are expected and are readily dealt with. In a TPM project, they are generally

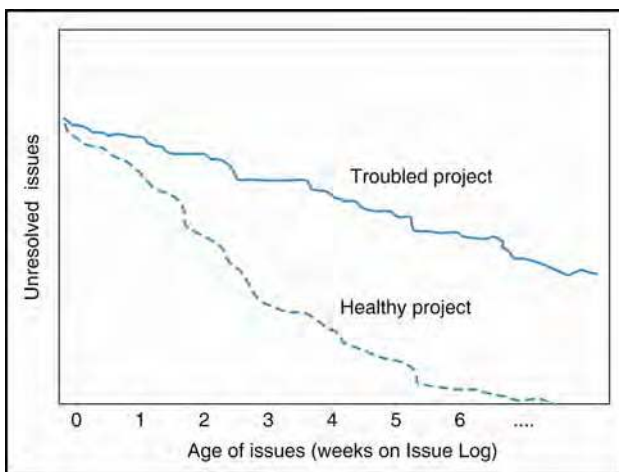


Figure 13.15
Number of unresolved issues versus age of issues.

unwelcomed and burdensome since they require modifying or redoing work, reorganizing or adding personnel, and making tradeoffs between time, cost, and performance.

Reasons for changes

Projects experience changes for many reasons:¹⁰

1. Changes in project scope and specifications. As a rule of thumb, the more uncertain the project, the more likely that scope and specifications will be altered during the project.
2. Changes in design because of errors, omissions, unknowns, afterthoughts, or revised needs. Mistakes or omissions must be corrected or accommodated, but customers often try to squeeze in unnecessary changes beyond the original scope (but, they hope, within the original price).
3. Changes mandated by changing government regulations (health, safety, labor, environment), labor contracts, suppliers, or other stakeholders in the environment.
4. Changes believed to improve the rate of return of the project.
5. Changes perceived to improve upon original requirements. People often seek to improve their work; although desirable for the customer, such improvements can expand the project beyond its original scope and requirements.

Examples of the above include: (1) after designing a space probe, increasing its payload requirement to allow for additional necessary but unanticipated hardware; (2) encountering and having to reroute a buried cable during excavation; (3) interrupting work because of municipal code violations; (4) altering the design capacity of a refinery under construction to increase the refinery's output; (5) adding more features to an already acceptable software design to enhance its perceived marketability. Note that changes 1, 4, and 5 are *discretionary*, i.e., they can be rejected, while 2 and 3 are *de facto*, i.e., they already “happened” or must be accepted.

When numerous changes are *expected or encouraged*, APM (described in the next chapter) is the preferred methodology. But when relatively few changes are expected and none are desired—as in TPM projects, then changes must be addressed through a formal change control and configuration management process.

Impact of changes



See Chapter 20

The larger and more complex the project, the greater the number of changes and the more likely that actual costs and schedules will deviate from original targets. This is especially true in big or megaprojects, as discussed in Chapter 20. Changes are a major cause of cost and schedule overruns, and poor relationships between contractors and clients. Each change has a ripple effect: in response to an emergent problem, elements of the end-item and project plan must be changed, but in chain-reaction fashion, these changes then require changes to other elements of the end-item and project.

In general, the further along the project, the more detrimental the effect of changes. Design changes made to a component during system assembly and testing often lead to rework or redesign of other components. Changes made still later in construction and installation cause even more trouble: work must be interrupted, torn down and redone, and materials scrapped. Morale is affected too: people see their work dismantled, discarded, and redone; pressure grows to get the project back on budget and schedule.

Change control process and configuration management

Any aspect of a project plan (e.g. scope statement, schedule, or budget) or a design (e.g. performance criteria) when it is first approved is referred to as the **baseline**. Thereafter, whenever changes are approved to the project plan or design baseline, these must be reflected in a new baseline—the *second baseline*, *third baseline*, and so on. These documents might all be retained and updated on the project's document management system, DMS.

To formally manage all the changes that will occur during the project and reduce their negative impacts, managers employ a formal **change control process**. The process, summarized in Figure 13.16, ensures that all changes, discretionary and de facto, to design and work tasks are documented, formally reviewed, and assessed, and accepted only as necessary. It also assures that changes will be reflected in the project plan and other project and design documents.

The change control process should:¹¹

- Continually identify changes as they occur.
- Reveal the impact of changes on project costs, duration, and other requirements and tasks.
- Accept or reject proposed changes based upon analysis of impacts.
- Communicate changes to all parties concerned.
- Specify a policy for minimizing conflicts and resolving disputes.
- Ensure that changes are implemented.
- Report monthly a summary of all changes to date and their impacts on the project.

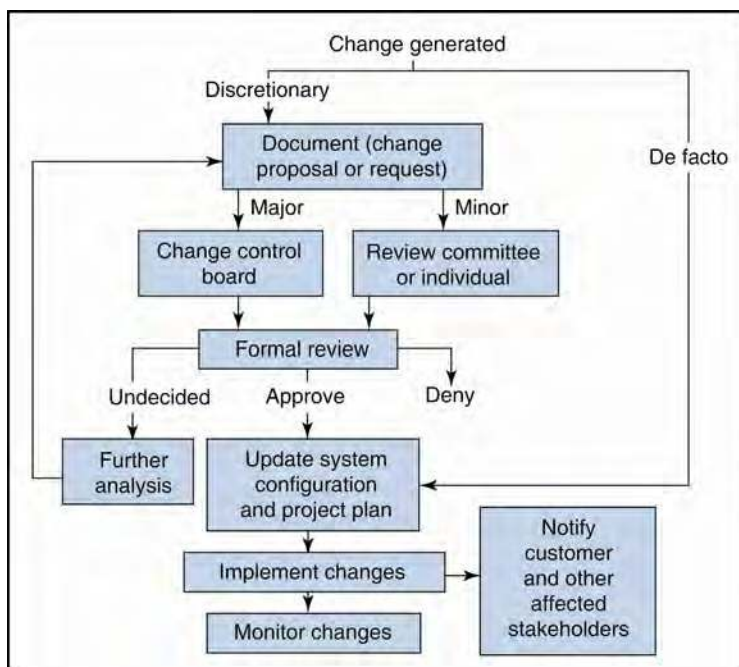


Figure 13.16
Change control process.



See Chapter 10 **management** and is discussed in Chapter 10.

The change control process is established early in the project and thereafter used to appraise the impact of proposed changes on cost estimates, schedule, and plans, and to trace any variance in current performance and original estimates to specific approved changes. Change control is part of a broader process for controlling and integrating changes into the design, development, building, and operation of the end-item system, which is called **configuration**

To control and minimize work and design changes, the change control process includes the following procedures:¹²

- Requiring that original SOWs, work orders, schedules, and budgets are unambiguous and *agreed* to by persons responsible.
- Close monitoring of work to ensure it *meets* (not *exceeds*) specifications.
- Carefully screening work for changes in work scope or cost or schedule overruns, and taking quick, corrective action.
- Requiring formal request and approval of all discretionary changes.
- Requiring similar control procedures of all contractors and suppliers for all purchase orders, test requests, and so on.
- Assessing the impact of all changes on the end-item and project and revising designs and plans to reflect the impact.
- Freezing the project against all non-essential design changes at a predefined stage; this prohibits additional design changes so the next stage (fabrication, construction, or coding and testing) can begin. The **freeze date** is set early in the project and project personnel are constantly reminded of it.

A key part of the change control process is the **change request document**, Figure 13.17, which provides information about and rationale for a proposed change. (Change requests are sometimes classified as Class I or Class II, where Class I can be approved by the contractor or the developer, but Class II requires customer approval.) Any project team member or other stakeholder can request a change by submitting a change request. Everyone, regardless of role, title, or position, must follow the same request procedure.

In large projects a committee—a **change control board**—meets weekly or bi-weekly to review change requests, assess their impacts, and decide which changes to reject and which to approve. The board consists of the project manager, functional managers, the contract administrator, and customer reps.

Any proposed or enacted change that impacts the time, cost, or the work of any task and related tasks must be documented. Everyone involved in the project has the potential to recognize or originate changes, and everyone is expected to bring them to the attention of the project manager. Changes requested and approved are listed on a **change log** or **change register** to track their progress; it includes for each change the status, persons responsible, and due dates.

13.9 Problems with Monitoring and Controlling Projects

No matter how thorough and conscientious the project manager or sophisticated the project control system, monitoring and controlling projects can be problematic for several reasons:¹³

1. The monitoring and control process focuses on only one metric, such as cost, and ignores others such as schedule and technical performance. This happens when control procedures are issued by one functional area, such as accounting, and other areas are not

IRON <i>Butterfly Corp</i>			
Change request			Page ... of ...
Title:			
Project no.	Task no.	Revision no.	Date issued
Description of change			
Reason for change			
Documentation attached			
Originated by:		Date:	
Request logged by:		Date:	
Cost implications			
Schedule implications			
Implications on performance of deliverable(s)			
Other implications (risks & issues)			
Proposed plan for implementation			
Implications evaluated by:		Date:	
Recommendation			
Recommended by:		Date:	
Documentation attached			
Approved by:		Date:	
Approved by:		Date:	

Figure 13.17
Example change request document.

involved. In general, focusing on one metric tends to encourage behavior at improving that metric while degrading other important aspects of performance, e.g. overemphasis on costs can lead to schedule delays or shoddy workmanship.

2. Project team members resent attempts to monitor and control their work (this happens especially when they were not sufficiently involved in planning the work) or do not comply with control procedures. Managers encourage this by ignoring those who don't comply with procedures.
3. Project team members do not report problems they are aware of. They may not understand the situation or might be hesitant to reveal it. Whatever information they do report may be fragmented and difficult to piece together.
4. The control system relies entirely on self-appraisal of work progress and quality, and people provide biased information.
5. Managers act indifferently about controversial issues, believing that with time problems will resolve themselves. This leads workers to believe that management doesn't care—an attitude likely to spread throughout the project team.
6. Managers overseeing several projects misrepresent charges such that overruns in one project are offset by underruns in others (or, within a project, overruns in some work packages are offset by underruns in others). The practice distorts historical data that could be used for cost estimating future projects; it is also unethical when it results in mischarging customers.

Sometimes the problem is that the wrong metrics are tracked—metrics that do not actually reveal anything useful about progress. Above all, the monitoring and control process must strive to be relevant, impersonal, objective, and uniformly applied to all aspects of the project.

13.10 Summary

Throughout the project execution phase, the monitoring and control process guides the project to keep it moving toward the objectives for scope, budget, schedule, and quality.

Project monitoring and control relies on a variety of sources and metrics for collecting and communicating information, including formal and informal reviews, observations, technology, and written project status reports. Frequent site visits, stand-up meetings, and one-to-one conversations are the best sources, although formal reviews and audits conducted by experienced outsiders provide independent assessments of project performance.

Project control means directing the project to keep it moving forward and in conformance with the project plan. This requires control over the project scope, quality, procurements, cost, and schedule. Project control begins with project goals, requirements, and a project plan. Without requirements and a plan, there can be no control.

The focal points of control are the individual work packages and control accounts. Virtually all control activities—authorization, data collection, progress evaluation, problem assessment, and corrective action—occur at the work package/control account level. The monitoring and control process begins with authorization; once authorized, work is continually tracked with reference to the project plan for conformance to scope, quality, schedules, and budgets. Key technical metrics are monitored to gauge progress toward meeting technical objectives. Performance to date is reviewed using the earned value concept.

Whenever costs and schedules move beyond pre-established limits, or new opportunities or intractable problems arise, the work must be replanned and rescheduled. Changes are inevitable, but every effort is made to minimize their impact on cost and schedule overruns. A formal change control process and configuration management system ensure that changes are documented, assessed and authorized, and communicated.

Much of this and earlier chapters describe planning and control methods commonly used in TPM (waterfall) projects. Most APM or agile projects do not need such elaborate methods and employ simpler tools. These are discussed in the next chapter.

Summary of Variables

PV = budgeted cost of work scheduled (BCWS)

AC = actual cost of work performed (ACWP)

EV = earned value = budgeted cost of work performed (BCWP)

SV = schedule variance = EV – PV

CV = cost variance = EV – AC

BAC = total budgeted cost of project

SPI = schedule performance index = EV/PV

CPI = cost performance index = EV/AC

ETC = forecast cost to complete = (BAC – EV)/CPI where BAC is the budgeted cost at completion

EAC = estimated cost of at completion = AC + ETC

TV = time variance



Review Questions and Problems

1. Why is it better to rely on a variety of information sources for project monitoring rather than just a few? Give some examples of how several sources are used to monitor projects.
2. What are the advantages of the following ways to monitor projects: (a) daily standup meetings, (b) formal reviews, (c) firsthand evaluation, (d) technology, (e) formal written reports?
3. What is the purpose of internal peer reviews? When are they held? Who participates?
4. What is a formal critical review? When are formal reviews held and what do they look at? Why do outsiders conduct them? Why would a customer or project supporter want a formal review?
5. Discuss the role of the project manager as both sender and receiver of project reports.
6. What are the three phases of the monitoring and control process?
7. Explain the differences between internal and external project control.
8. Why is scope change control an important part of the project control process?
9. Discuss quality control as applied to projects.
10. What are the principal causes of project schedule overruns? Discuss at least four practices that may be used to reduce schedule variability and keep projects on schedule.
11. Describe the work authorization process. What does a work order usually include?
12. Discuss different ways of measuring ongoing work progress.
13. If a cost or schedule variance is noticed at the project level, how is it traced to the source of the variance?
14. Discuss common causes of project scheduling overruns and management practices that may reduce these overruns.
15. Refer to Example 13.2.
 - a. Suppose in Week 28 the team discovers a procedural error that negated all work done so far on Task R. What are the revised values for percent CC completed and percent buffer consumed for Week 28? Where is the project on the fever chart?
 - b. Recompute percent CC completed and percent buffer consumed as of the following weeks for the percent tasks completed in each: 16: S, T 100%; 20: S, T 100%, Q 10%; 24: S, T, Q 40%; 28: S, T, Q 100%; 32: S, T, Q, R 50%. As of Week 32, does it appear the project can be completed by Week 36?
16. Explain PV, AC, and EV, and how they are used to determine the variances AV, SV, CV, and TV. Explain the meaning of these variances.
17. What does it signify if cost or schedule index figures are less than 1.00?
18. Explain TePM, its purpose, and how it is conducted.
19. What is an “issue”? Explain “issue management” and how it is implemented through the issues log.
20. Explain ETC and how it is related to EAC.
21. Discuss reasons why the project manager tries to resist project changes.
22. What should the change control process do? Describe procedures that minimize unnecessary changes.
23. What aspects of project control fall under contract administration?
24. What are some difficulties encountered when attempting to control a project?
25. Use the networks in Figure 13.18 to determine ES, LS, EF, and LF for all activities

(number in activity box is duration in days). Apply the buffer concept to the critical path. For Network (a) use a 3-week time buffer for the critical path, a 1-week time buffer for every path that feeds into the critical path. For Network (b) use a 4-week time buffer on the critical path, a 2-week buffer for every path that feed into the critical path.

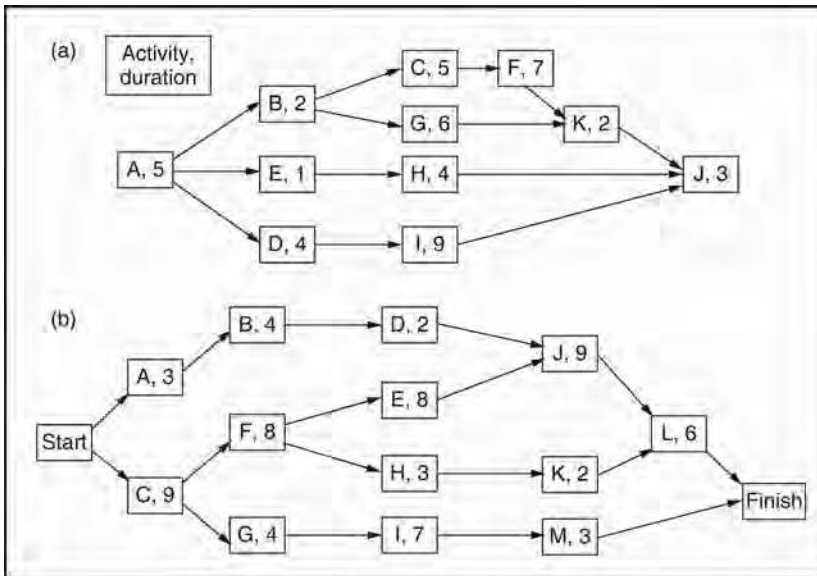


Figure 13.18
Two project networks.

26. In the LOGON project, suppose the status as of Week 22 is as follows (note usage of the longer acronyms; some project management books and software packages use these and not the shorter acronyms PV, AC, and EV).

$$\text{BCWS} = \$628,000$$

$$\text{ACWP} = \$640,000$$

$$\text{BCWP} = \$590,000$$

Answer the following questions:

- What is the earned value of the project as of Week 22?
- Compute SV and CV.
- Draw a status graph like Figure 13.10 and plot BCWS, ACWP, and BCWP. Show SV and CV. Determine TV from the graph.
- Compute SPI and CPI. Has the project performance improved or worsened since Week 20?
- Using $\text{BAC} = \$990,000$, compute ETC and EAC. How does EAC compare to the Week 20 estimate of \$1,222,593? From your status chart determine the revised completion date. How does it compare to the revised date (Week 48–49) as of Week 20?

- f. Are the results from Part (e) consistent with the results from Part (d) regarding improvement or deterioration of project performance since Week 20?
27. The budgeted cost as of April 30 for a work package is \$18,000. Suppose on April 30 the supervisor determines that only 80 percent of the scheduled work has been completed and the actual expense is \$19,000. What is the BCWP? Compute SV, CV, SPI, and CPI for the work package.
28. Using the status chart in Figure 13.19:
 - a. Estimate values for SV, CV, and TV, and use them to compute SPI and CPI for Week 30. Interpret the results.
 - b. Compute ETC and EAC. Estimate the revised completion date and sketch the lines for forecast AC and forecast EV.

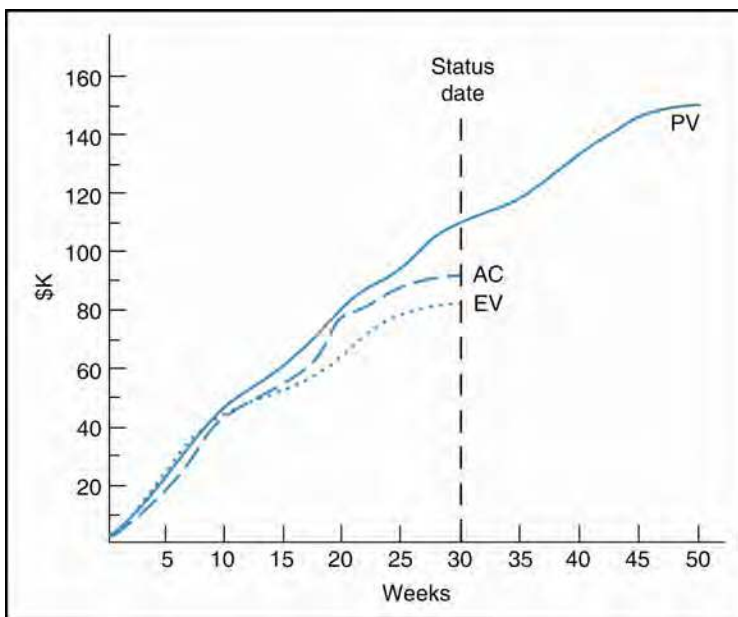


Figure 13.19
Project AC and EV status as of Week 30.

29. Assume for the following problems that work continues during weekends.
 - a. A task is planned to start on April 30 and takes 20 days to complete. The actual start date is May 3. After 4 days of work the supervisor estimates that the task is 25 percent completed. If the work rate stays the same, what is the forecast date of completion?
 - b. Task C has two immediate predecessors, tasks A and B. Task A is planned to take 5 days to complete; Task B is planned to take 10 days. The early start time for both tasks is August 1. The actual start dates for tasks A and B are August 2 and August 1, respectively. At the end of August 4, Task A is assessed to be 20 percent completed and Task B, 30 percent completed. What is the expected early start time for Task C?
30. Refer to Problem 25 and Figure 13.18.
 - a. For Network (a), suppose after 7 weeks, activities A, B, and E have been completed, D is 50 percent completed, and C is 80 percent completed. What is the revised early completion date for the project?

- b. For Network (b), suppose after 25 weeks, activities A, B, C, F, E, G, and I have been completed, and D and H are ready to begin in Week 26. What is the revised early completion date for the project?
31. For the following questions refer to Figure 13.20. As of Week 5, for the project:
- What is the planned value (PV)?
 - What is the earned value of the work completed (EV)?
 - What is the actual cost of the project (AC)?
 - What is the value of the work remaining?
 - What is the CPI?
 - What is the estimated cost to complete the project (ETC)?
 - What is the estimated cost at completion (EAC)?
 - What is the estimated cost variance at completion, and the percent overrun or underrun?
 - According to EV, is the project ahead or behind schedule?
 - According to the critical path (A-C-F-H), is the project behind or ahead of schedule?

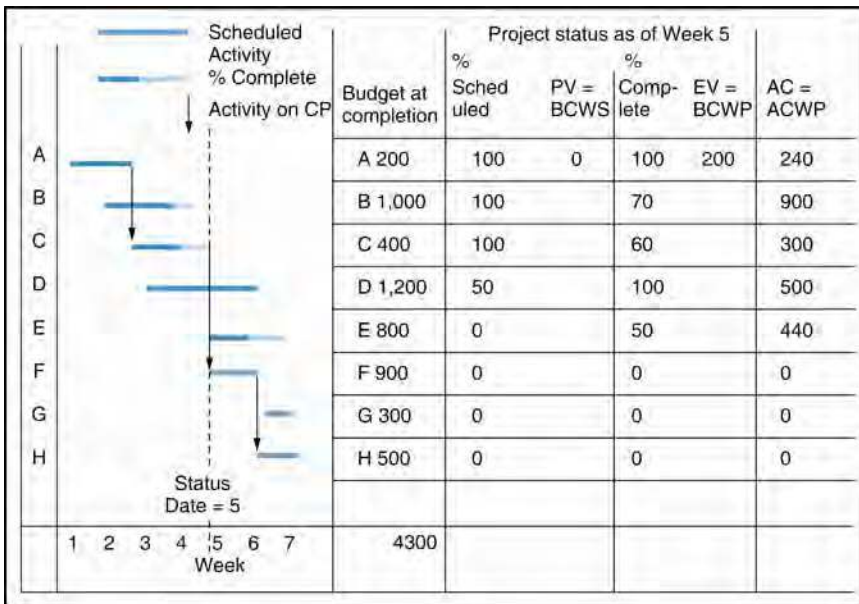


Figure 13.20

Project percent complete and PV, EV, and AC status as of Week 5.

32. Refer to the burn-up chart in Figure 13.12. Suppose on Day 14 no story points were added to the chart, but on Day 15, 20 points were added. As of Day 15, is the project progressing as expected (assuming linear progress each day)?



Questions About the Study Project

1. How often and what kinds of review meetings were held in the project? Why were they held? Who attended them?
2. When and for what reason were special reviews held?
3. How was follow-up ensured on decisions made during review meetings?
4. Was there a project meeting room? How often and in what ways was it used?
5. Describe the kinds of project reports sent to the project manager, top management and other managers, and the customer and other stakeholders. Who issued these reports?
6. What kinds of external controls, if any, were imposed by the customer and other parties on the project?
7. What kinds of internal controls were used? (For instance, work package control, cost account control, etc.) Describe.
8. Describe the project control process:
 - a. How was work authorized to begin? Give examples of work authorization orders.
 - b. How was data collected to monitor work? Explain the methods and procedures—timecards, invoices, etc.
 - c. How was the data tallied and summarized?
 - d. How was the data validated?
9. Was the concept of earned value (budgeted cost of work performed) used?
10. How was project performance monitored? What performance and variance metrics were used? Was the buffer management concept used? Explain.
11. How were problems/issues pinpointed, tracked, and acted upon?
12. Were the concepts of forecasting ETC and EAC used? If so, by whom? How often?
13. Were variance limits established for project cost and performance? What were they? How were they applied?
14. When cost, schedule, or performance problems occurred, what action did the project manager take? Give examples of problems and what the project manager did.
15. What changes to the end-item or project goal occurred during the project? Describe the change control process used. How were changes to the plan or system reviewed, authorized, and communicated? Show examples of change control documents.

CASE 13.1 CYBERSONIC PROJECT

Miles Wilder, project manager for the Cybersonic project, considers himself a “project manager’s project manager.” He claims to use the principles of good project management, starting with having a plan and using it to carefully track the project. He announces to his team leaders that status meetings will be held on alternate Mondays throughout the expected year-long project. All 18 project team leaders must attend and give rundowns of the tasks they are currently working on.

All the team leaders show up for the first status meeting. Seven are currently managing work for the project and are scheduled to give reports; the other 11 are not yet working on the project (as specified by the project schedule) but attend because Miles wants them to stay informed about project progress. The meeting is scheduled

for 3 hours, during which the team leaders are to report on whatever they think important. After 4 hours of reports by five of the leaders, Miles ends the meeting. Several major problems are reported that he tries to resolve at the meeting. Specific actions to resolve some of them are decided, and Miles schedules another meeting for 2 days later to address the other problems and hear the remaining two reports. Some of the team leaders are miffed because they'll have to change their schedules to attend the meeting.

Miles arrives 20 minutes late to the next meeting, which, after 2 hours, allows enough time to resolve all the problems but not enough for the two leaders to give reports. Miles asks them if they are facing any major issues or problems. When they respond "no," he lets them skip the reports but promises to start with them at the next meeting 2 weeks later. A few of the team leaders are assigned actions to address current problems. Some of the attendees feel the meeting was a waste of time.

Before the next meeting, some of the leaders inform Miles they cannot attend and will send representatives. This meeting becomes awkward for three reasons. First, several new problems about the project are raised and, again, the ensuing discussion drags out and there is insufficient time for everyone to give a status report; only six of a scheduled eight team leaders give their reports. Second, some of the leaders disagree with Miles about actions assigned at the previous meeting. Because no minutes had been taken at that meeting, each leader had followed his/her own notes about actions to take, some of which conflict with Miles' expectations. Third, people at the meeting who are "representatives" are not fully aware of what happened at the previous meetings, do not have sufficient information to give complete reports or answer questions, and are hesitant to commit to action without their team leaders' approval.

The next several meetings follow the same pattern: they run over schedule; fewer team leaders and more representatives attend; status reports are not given because of inadequate time; people disagree over problems identified and actions to be taken. The project falls behind schedule because problems are not addressed adequately or quickly enough.

Miles feels that too much time is being wasted on resolving problems at the meetings and that many problems should, instead, be resolved entirely by the team leaders. He instructs the leaders to work out solutions and changes on their own, and to report at status meetings only the results. This reduces the length of the meetings but creates other complications: some team leaders take actions that ignore project dependencies and conflict with other leaders' work tasks. Everyone is working overtime, but the Cybersonic project falls further behind schedule.

QUESTIONS

1. Why is Miles' approach to tracking and controlling the Cybersonic project ineffective?
2. If you were in charge, what would you do?

CASE 13.2 SA GOLD MINE: EARNED VALUE AFTER A SCOPE CHANGE¹⁴

The team at South African (SA) gold mine was tasked with sinking a 2,000-meter deep ventilation shaft and excavating space for a station at the bottom. The plan was to sink the shaft within 20 months at a cost of R65,000 (about US \$3,600) per meter of shaft depth. For the station at the bottom, 30,000 m³ (cubic meters) of rock would have to be excavated within 3 months at a cost of R700 per m³. The plan assumed a uniform earned value over time.

After the work had begun, the scope of the project was changed to include excavation for a new station halfway down the shaft (Figure 13.21) with a volume of 20,000 m³. It was agreed that the additional work would have to be done at the same excavation rate as the bottom station, but since removal of the rock required hoisting only 1,000 meters (instead of the 2,000 meters for the bottom station), the team agreed on the cost of R500 per m³ for the new station. Since limited working space and available resources would delimit the amount of work that could be done simultaneously, everyone agreed that the new station would delay the sinking of the shaft. After 13 months, the shaft had reached a depth of 1,400 meters below surface and excavation for the halfway station was completed. The actual cost at this time was R90 million, which was more than was budgeted for the period. This provoked a cash-flow problem at that stage, and executive management requested an earned value report. Information on the relative amounts of time spent on excavating the new station and sinking the shaft was not available.

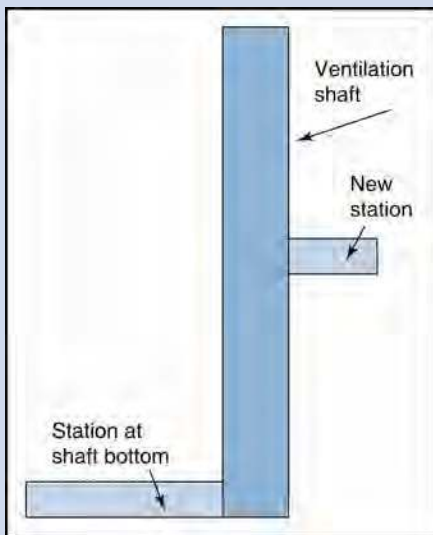


Figure 13.21
Mine shaft.

QUESTIONS

1. Calculate the CV, SV, TV, CPI, and SPI.
2. Prepare a graph to illustrate the initial plan for the work, including the excavation for the station at the shaft bottom, as well as the changed plan. Indicate the earned value and the actual cost after 13 months.
3. Regarding the cash-flow problem that was aggravated by the high rate of spending, discuss the desirability of performing projects faster than planned.

CASE 13.3 CHANGE CONTROL PROCESS AT DYNACOM COMPANY¹⁵

At Dynacom Company, any change that potentially affects project scope is subject to a rigorous review and approval process. Anybody requesting a project change must document and present it to the team lead. If the team lead approves the request she enters it into the company's change request system, which the project manager checks each day. The project manager then meets with the team lead and original requester to discuss the change's likely impact on the project. If they conclude the change is worthwhile, the project manager schedules a meeting with the entire team to discuss the need for the change, its impact on schedule and budget, and the risks. Sometimes a team approves changes immediately, other times it takes a few days or weeks of review. If the team approves the change, it sends a recommendation to the technical change management board (TCM) for a final decision. TCM has no association with the project and consists of upper managers and other project managers. If TCM accepts the recommendation, the project manager makes the necessary changes to work schedules, budgets, and other documents. Dynacom is a rather conservative company and the process has served it well in helping to avoid risks associated with changes.

A drawback is that the process takes at least 3–5 weeks to decide on a change request. As a result, project managers sometimes implement changes before they are approved. For example, Karen, the manager of a project on a very tight schedule and running behind, needed to make changes on the critical path. She worried that if she waited for approval of the changes the project would fall too far behind and might be cancelled. Intent on getting the project back on schedule and willing to risk breaking the rules, she made the changes immediately and assumed the TCM board would accept them, which it did.

QUESTIONS

1. What is your opinion of the change control process at Dynacom? What are the benefits and drawbacks?
2. What do you think about Karen bypassing the process to make changes?

CASE 13.4 STATUS REPORT FOR THE LOGON PROJECT

The LOGON project began as scheduled in May 2024. In late September—after the project had been underway 20 weeks—Midwest Parcel Distribution (MPD) Company, the customer, requested Frank Wesley, the project manager, to prepare a written summary status report about progress to date. Review Appendices A and B at the end of the book for background about the project, then see Appendix C for information on the budget and scheduled dates for milestones and deliverables. Prepare the report as if you were Frank. Note that the report is to MPD's top management and should address issues most important to them: deliverables and other requirements, schedule, and budget, as noted in Appendix C, end of book. The report should also note any problems encountered to date, anticipated challenges, and recommended suggestions or changes to the plan.

Notes

1. The terms "variance" and "deviation" are used here interchangeably, although sometimes variance refers to small changes in the project plan for which compensation or correction is expected, and deviation to large changes that require a formal contractual response.
2. See Turner J. and Muller R. Communication and cooperation on projects between the project owner as principle and the project manager as agent, *European Management Journal* 22(3); 2004: 327–336.
3. Muirhead B. and Simon W. *High Velocity Leadership: The Mars Pathfinder Approach to Better, Faster, Cheaper*. New York, NY: Harper-Business; 1999, pp. 40–42.
4. Cusumano M. and Selby R. *Microsoft Secrets*. New York, NY: Free Press; 1995, pp. 204, 221, 256–257, 417.
5. Thompson C. Intermediate performance measures in engineering projects. *Proceedings of the Portland International Conference on Management of Engineering and Technology*, Portland, OR, July 27–31; 1997: p. 392.
6. Methods to determine EV [BCWP] are explained in Pham T. The elusive budgeted cost of work performed for research and development projects. *Project Management Quarterly* March; 1985: 76–79; for EVM, see Fleming Q. and Koppleman J. *Earned Value Project Management*. Upper Darby, PA: Project Management Institute; 1996.
7. Lukas, J. Earned value analysis: why it doesn't work. *AACE International Transactions*, EVM.01.1–EVM.01.10; 2008, see <https://www.proquest.com/docview/208198026?sourcetype=Scholarly%20Journals>. Accessed June 20, 2024.
8. Issues of EVM vs. CCPM are discussed in Newbold R., Budd C.S. and Budd C.I. *Protecting Earned Value Schedules with Schedule Margin*. ProChain Solutions, 2010, <http://www.prochain.com/pm/articles/ProtectingEVSchedules.pdf>; downloaded Jan. 5, 2016.
9. For examples of analytical models used for TePM, see Eisner H. *Computer-Aided Systems Engineering*. Upper Saddle River, NJ: Prentice Hall; 1988, pp. 297–326.
10. Harrison, F. *Advanced Project Management*. Hants, England: Gower; 1981, pp. 242–244.
11. Harrison, *Advanced Project Management*, pp. 245–246.
12. Harrison, *Advanced Project Management*, p. 244; Archibald, *Managing High-Technology Programs and Projects*, pp. 187–190.
13. Roman, *Science, Technology, and Innovation*, pp. 327–328, 391–395.
14. Source: Mr. P. Joubert of Anglo Platinum.
15. Adapted from Quane J., Kosin M., Heinlen L., Mahoney S., and Quantraro F. *Cyberdyne Project Planning Management Investigation Report*. Quinlan School of Business, Loyola University Chicago; May 2007.

Chapter 14

Agile, Lean, and Hybrid Project Management

Some methods of project management are well-suited for projects such as construction, infrastructure, and product development and are not well-suited for projects such as software development. In 2001 a group of 17 developers met to address this problem and created **The Agile Manifesto** wherein they proposed a set of management principles better matched to their projects; they include:

- Individuals and interactions over processes and tools.
- Working software over comprehensive documentation.
- Customer collaboration over contract negotiation.
- Responding to change over following a plan.¹

In each of these principles, the implication is that the things on the left (e.g., individuals and interactions) are more important than and should take *precedence over* the things on the right (processes and tools). Innovative practices in project management arise constantly, and the *Manifesto* endorsed some that had previously been applied on a limited basis.

Accompanying the *Manifesto* was the rise of “agile software development,” which was the precursor of **agile project management, APM**, and management methods such as Scrum and Kanban. APM is a form of project management **methodology**, and contrasts to the **traditional project management** methodology as covered throughout this book. The PMI refers to APM methodologies as *adaptive methodologies* since they attempt to adapt the methodology to the uniqueness of the project, state of knowledge, and desired results. We believe that for some APM methodologies, “adaptive” is a more suitable term, although we will stick with the more widely recognized term, *agile*. Lean—short for **lean production**—refers to the Toyota Production System, the management system that helped propel Toyota to the forefront of the automotive industry. The creators of *The Agile Manifesto* were inspired by Toyota’s lean principles—continuous improvement, elimination of waste, and respect for people²—so it is no coincidence that APM and lean practices somewhat overlap. As this chapter discusses, lean and APM applications extend beyond software development and have taken hold elsewhere, especially in product development projects.³

14.1 Traditional Project Management

All projects can be conceptualized in terms of the life cycle phases of conception, definition, and execution, but exactly what happens in each phase depends on the kind of project and on how clear, complete, or well-understood are the problem/goals and solution/end-item of the project. When these are well understood and can be defined early in the project, the project moves readily through the phases: goals and the solution/end-item are established in conception; requirements and the project plan are drawn up in definition; and work toward the goals/requirements is undertaken in execution. This is the **traditional project management** methodology, **TPM**—also called **waterfall**. In TPM, the phases of definition and execution proceed sequentially and happen only once: the solution/end-item requirements and project plan are clearly laid out in definition, and the work is undertaken during execution (Figure 14.1). TPM projects are *plan-driven* projects; the execution phase means, simply, “execute the plan.”

Incremental TPM

A variant of this approach allows for implementing (or launching) the end-item in a series of increments. With **incremental TPM**, the end-item is implemented stepwise, piece by piece, allowing some elements of it to begin use before the entire end-item is completed (Figure 14.2). For example, accommodations at some parts of a resort are completed and occupied even though the rest of the resort is still under construction, or the lower floors of a high-rise are completed and occupied even though the upper floors are still being constructed. Decisions about when the increments are to be built and implemented are based upon dependencies among them (which pieces need to precede which others) and on market or financial considerations. Besides allowing the customer to begin using parts of the end-item earlier, incremental TPM provides opportunities to experiment with and learn about the end-item system and identify necessary changes or improvements. But incremental TPM is still TPM: it assumes that the solution/end-item is well-understood and clearly defined early in the project; the only difference is that the end-item is demarcated into discrete elements, to be implemented weeks or months apart in a series of steps.



See Chapter 13

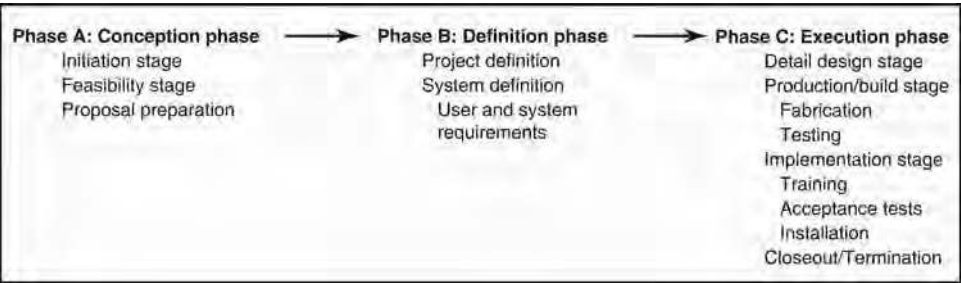
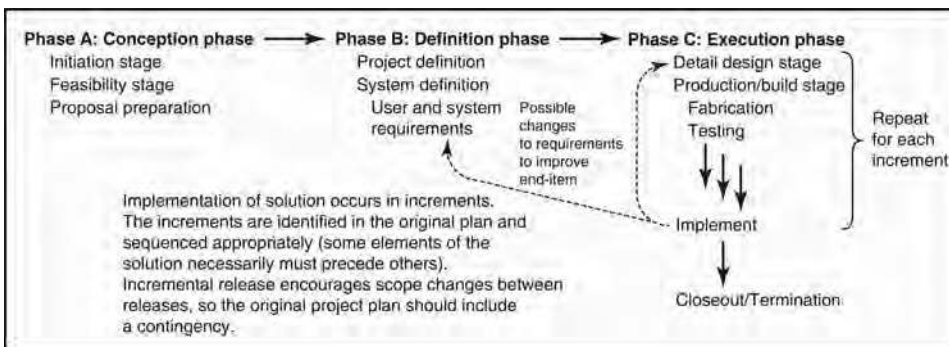


Figure 14.1
TPM methodology: applies when problem/goals and solution/end-item can be well-defined.

**Figure 14.2**

Incremental TPM methodology: applies when problem/goals and solution/end-item are well-defined and end-item is to be built/launched in increments.

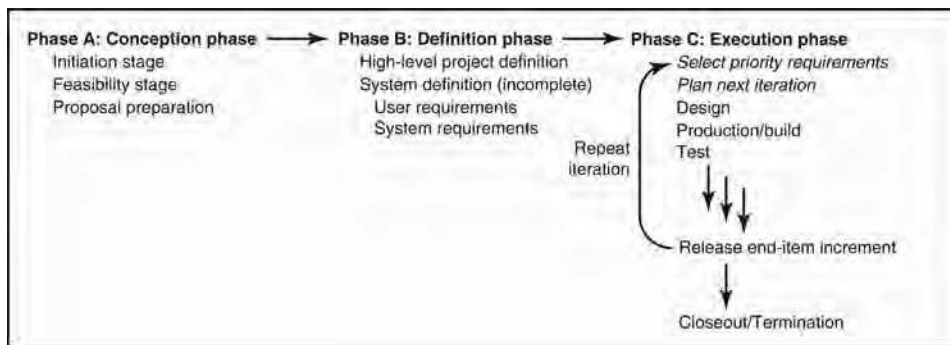
Inappropriate use of TPM

When the solution cannot be clearly identified or is uncertain (e.g., development of a revolutionary new technology or radically different product), the end-item requirements are very difficult to define, and the project is hard to plan. As a result, both requirements and plan must be developed over time, as things become better understood. The consequence of this is that the requirements and plan will change and expand, perhaps often. The traditional change control process described in Chapter 13 is ill-suited for such continuous change, as it tends to lengthen project duration, increase costs, and, often, yield mediocre or poor results.

When little is known about a project, however, usually there is *something* that is known, and that something is often enough to start a project. Unlike TPM wherein much or all of the project plan is created in the definition phase, an alternative is to hold off on planning, *acknowledge* that little is known, and use the project as the vehicle to *learn*. Rather than implementing a particular solution, the project focuses on *learning what the solution should be*. Rather than moving through the definition and execution phases once, they are cycled through iteratively in a learn-as-you-go fashion. Every project begins with initiation and ends with closeout, but in between whether the phases/stages happen once or repeatedly should depend on the certainty of the solution/end-item.

14.2 Agile Project Management

Project management methodologies that accommodate uncertainty in an iterative, learn-as-you-go manner are referred to as agile project management, APM. It is the preferred methodology when the customer/developer does not know or is unclear about the desired solution and cannot define the end-item requirements upfront. In terms of the diamond model in the Introduction, Figure I.6, APM is best suited to projects with relatively high novelty (newness to market), high technology (technological uncertainty) and low complexity (relatively small system scope). In such projects, the purpose of the definition phase is to identify, as best possible, the customer's wants, needs, or basic requirements, and create a *high-level* project plan. Then, in the remainder of the project, detailed requirements and features of the end-item are developed in a series of iterations; the high-level plan stipulates the anticipated number, length, and objectives of each iteration.

**Figure 14.3**

APM methodology: applies when problem/goals and solutions are not well-defined.

Learning from iterations

In the execution phase, most work happens in the iterations (Figure 14.3). Each iteration leads to discoveries and better-defined requirements; each concludes with a **release** or end-item “increment”—a partial solution that provides some value to the customer; it addresses the customer’s needs or problem, but in a limited fashion.

Detailed planning repeats on a just-in-time basis: at each iteration; the customer and team assess progress made so far and plan for upcoming iterations. Thus, each iteration includes not only the typical execution stages of design–build–test, but also aspects of the definition phase—requirements definition and project planning. The main emphasis is on creativity and delivering a solution of immediate value; budgets and schedules are of less or little importance.

To facilitate learning and discovery, the iterations tend to be short, in some cases a few months, in others 2–4 weeks each. At the end of each iteration the outcome is assessed: does it provide value and the expected benefits? Is it useful and usable? Where is it lacking or what is wrong with it? This enables subsequent releases to be improved so that by project closeout the cumulative increments will have fully or substantially met the customer’s needs and wants.

Agile variants⁴

APM comes in many forms, and the one to use depends on what we know about the solution or end-item. For example, do we know a range of different, possible solutions but not the best one to choose? Do we know the best solution, but not details about it? Or do we simply not know the solution or even the problem?

Take the second case—we know the solution but not the details. In that case, the execution phase is repeated iteratively, as described, and the purpose of each iteration is to discover and integrate newly identified details—features and functions—into the solution/end-item. The customer is involved throughout the project, assessing changes/additions to the solution with each iteration, and giving feedback to the development team. The **prototyping** methodology with its repeated design–testing cycles described in section 4.5, Chapter 4, is an example of this approach. The spiral model is another.



See Chapter 4

Example 14.1: The Spiral Model

In a spiral-model project, the end-item or product is released in a series of cycles, each cycle taking a few to several months.⁵ Also called “iterative prototyping,” each cycle includes the usual stages of analysis, design, develop, test, integrate, etc., and results in a prototype. The first cycle delivers, optimistically, a product prototype, which the customer uses and assesses. The process repeats, each cycle delivering an improved prototype version based upon customer feedback from earlier versions (Figure 14.4). After a few cycles—either as predetermined or whenever the customer is fully satisfactory—the prototype becomes the “production” or operational version of the end-item. Spiral is often used for reducing risk in developing large, complex, systems by breaking the project into pieces (each cycle of the spiral) and addressing functions and features of the system one at a time.

Although the spiral model is commonly applied to develop a prototype design that will ultimately lead to a final product, it can also be applied to successively improve an existing product. An example is a software product or software portion of a hardware product where every few months or annually an upgraded or enhanced version of the product is released. This “outward spiral” of product improvement (Figure 14.4) continues for as long as the product remains “improvable” and viable in the market.

To the extent that APM typically involves iterations, it is spiral, though some have argued that spiral is not APM but a form of incremental TPM since it involves simply repeating TPM stages within each cycle. There is however a difference between the spiral model and incremental TPM. With incremental TPM, which applies to all kinds of end-items, including one-of-a kind (e.g., space station, aircraft carrier, skyscraper, highway), the overall end-item does not “evolve.” Although portions or elements of the end-item might evolve (be upgraded, improved, expanded), the overall end-item does not; its fundamental structure/composition is set at the start of the project and remains so forever. With incremental TPM, the end-item “grows” but does not evolve. Furthermore, the

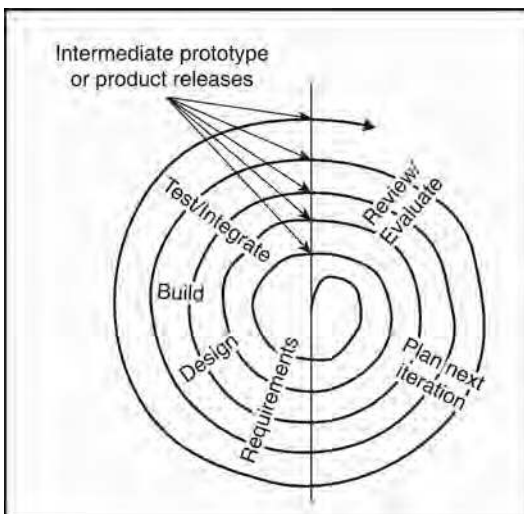


Figure 14.4
Spiral development methodology.

overall end-item/solution and the elements to be added incrementally are known and defined early in the project.

With APM, including the spiral model, details of the solution and end-item system evolve and expand with each iteration, usually of necessity since much about the solution is uncertain and a purpose of the iterations is to *discover* and *define* the details. Each iteration begins with a limited understanding of some aspects of the solution and builds on knowledge gained from the releases of previous iterations. The process iteratively *converges* on a complete solution.

Modern products like smart phones and cars are a combination of hardware and software. Often the hardware must be developed using the TPM approach, although the software might more-appropriately use an APM approach. A challenge in managing such projects is to coordinate the APM iterations so the software will be available when needed to integrate with the hardware.⁶

In APM, typically the duration of each iteration corresponds to how much is known about the solution: the less known, the *shorter* the duration. When the solution is somewhat well-known, as is common in spiral-model projects, an iteration can last several months. In the **Scrum** method of APM, an iteration lasts just weeks.

14.3 Scrum⁷

The Scrum version of APM is shown in Figure 14.5. The main focus of Scrum (the term Scrum derived from a facet of rugby football) is on creating a software product with the “desired” functions and features. Initially these functions and features are unknown, so the process starts with a list of customer needs and wants called a *product backlog*. The backlog, which is created by the customer or a representative called the *product owner* typically consists of a list of wants, needs, or “stories” that describe how a user might use or benefit from the product. Details will be flushed out in iterations, the number of which is set in advance in the project plan.

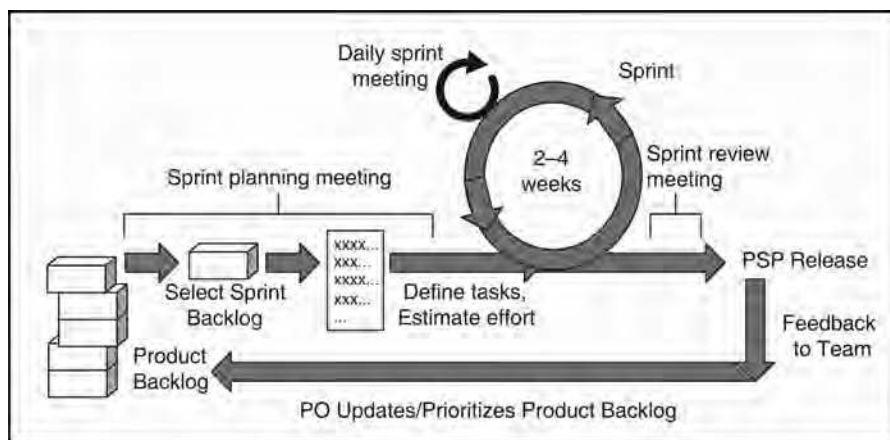


Figure 14.5
Scrum process.

Source: Adapted from Schwaber K. and Beedle M. *Agile Software Development with Scrum*. Upper Saddle River, NJ: Prentice Hall, 2002.

Just before each iteration—called a *sprint*—the development team and product owner review the product backlog and select from it a few items to be the focus the next sprint; these items constitute the *sprint backlog*. The team agrees on the functionality it must design and build to address those items, and that the work can fit within the sprint timeline or *time-box*, which is typically 2–4 weeks. During the sprint, the team meets briefly each morning to review progress and determine the day’s tasks. At the sprint’s conclusion the team releases a *potentially shippable product (PSP)*—an increment that provides value to the customer. Before the next sprint the team meets to reviews its work performance and lessons learned so far to apply to subsequent sprints. These terms will all be elaborated.

Scrum roles⁸

Management responsibilities in Scrum are shared among three roles: **scrum master (SM)**, **product owner (PO)**, and **project team**. There is no project manager, so the “traditional” project manager in a Scrum organization must change roles: if she is good at facilitating, she might be the SM; if she is knowledgeable about customers and the business, she might be the PO; if she likes being hands-on and in the thick of doing technical things, she might be a team member.

Scrum master

The SM is responsible for facilitating team collaboration and keeping the team project focused. The SM works full-time on the project and serves as coach, advisor, enforcer of Scrum rules, and sometime administrator. Unlike a project manager, he does not make decisions for the team. He can tell people what they need to do and help train or facilitate them, but he cannot tell them how they should do it. The SM should be familiar with the organization for which the project is conducted and know how to remove barriers and get the needed resources.

Product owner

The PO is responsible for ensuring the team knows what the product is or might evolve into, and who are the customers and competitors. The PO sets the “vision” of the end-item/product—how it should perform, when it must be completed, how much it should cost—and keeps reminding the team of it by frequently updating the product backlog and discussing backlog items with the team. The PO is the main contributor to creating and updating the product backlog.

The PO represents the “customer” and therefore must thoroughly understand the customers, users, the business, or the market. She must be good at communicating with and listening to customers, users, and the team. The PO is the decision maker regarding product matters, so she must have decision-making authority in the organization to address any product-related issues facing the team.

The SM and the PO should not be the same person, and a tension should exist between the roles: the PO pushes for more in the product, but the SM pushes back whenever he feels the PO is asking too much.

Project team

Team membership depends on whatever the project requires—analysts, programmers, testers, etc. Members work primarily in their own specialties but also contribute to all other specialties. Work assignments are tentative because members are expected to provide aid wherever it is needed. If a member finishes her task early, she is expected to start a new one or help others, regardless of specialty. Says Cohen, in Scrum there is no “my work” and “your work,” only “our work.”⁹

Through rotating and sharing roles, members develop an understanding of every task in the project. Since everyone is able to do and does a little of everything, and since everyone participates in all tasks, they know what has been and still needs to be done. This reduces the number of work “hand-offs” (transferring work from one person to another) and the need for formal documentation.

Team structure

Scrum teams are small, typically five to ten people; Amazon calls them “two-pizza” teams. This affords big advantages: smaller teams are better able to retain focus on project goals and enjoy greater interaction, participation, and satisfaction among members. They also tend to finish projects with less total effort and in less time.

Scrum teams are **self-managed teams**, and scrum leaders tend to be **servant leaders**, which is true of teams in virtually all agile and lean projects. The teams decide how they will achieve goals and divide up and monitor work. (Some outside management is still necessary, however, to set project-level goals and boundaries, and remove high-level barriers.) Characteristics of these teams are described later in this chapter and in Chapter 4, section 4.5.



See Chapter 4

Sprints are fast-paced and emphasize rapid learning, so real-time communication among team members is essential. Typically, Scrum team members work full-time on a project. They also work at the same location, although Scrum has also been successfully applied in projects where members are distributed across the world and rely on the Internet and technology to keep in touch. In large projects, the work is apportioned among many two-pizza Scrum teams.¹⁰

Product backlog

Scrum substitutes all-at-once, upfront definition at a single time with step-wise definition spread over multiple sprints. Customer needs and wants as listed in the **product backlog** are minimally defined—initially in the form of **user stories** or **epics**, and are refined and expanded over time. Since the desired functions/features and other details of the product are initially unknown, it is more practical to define the product in terms of stories than specific requirements. Each story is a simple statement from the user’s perspective about a product’s potential use, benefits, functionality, or capability, typically written on a 3 × 5 card or sticky note and in the format “As a <type of user>, I want <some goal> because <some reason>.” The focus of each sprint is to satisfy a few user stories. A user story that has multiple goals and is too large to be addressed in one sprint is called an epic. Epics must be broken down into simpler user stories that can each be handled in a sprint.

The stories and features that appear in a product backlog can be written at different levels of detail. Stories that are specific, well understood, and can be addressed in a single sprint are

placed at the top of the backlog. Stories that are broader and less well understood are placed farther down. Epics start out low on the backlog; as the project progresses, they are divided into user stories and moved up. The PO maintains the backlog—adding, deleting, updating, and reprioritizing items.

The backlog list is continually updated, refined, and reprioritized (“groomed”) so that the current most important needs and wants are always at the top. This grooming occurs during sprint planning meetings or before at special **refinement meetings**. Items are added to or deleted from the backlog according to current needs, stories are refined, epics are divided into stories, and new priorities are set.

The main system definition emphasis in Scrum shifts from documentation to discussion. Prior to each sprint the team discusses stories on the backlog with the PO and chooses items near the top it best understands and wants to attack in the next sprint; these form another list called the **sprint backlog**.

For each story selected, the team and PO define **conditions of satisfaction (COS)**, which state what the user does and does not expect. Unlike TPM projects that often have numerous pages of requirements, the COS and user stories constitute the sole “requirements documentation” for a sprint. When the product is required to comply with regulatory laws or perform complex calculations, the requirements are specified in terms of realistic examples that illustrate what the product must be capable of doing.

Sprints

The goal of each sprint is to deliver a working product—usually software. This product is called a **potentially shippable product (PSP)**, **minimum viable product (MVP)**, or **product release**. Although not a complete product or solution, and not necessarily without bugs, the PSP meets all the requirements set by the PO (e.g., the COS) and the project team (e.g., completion of all development stages) for the sprint. Each PSP is a stand-alone working product, and with it the customer receives something of value—a usable product, and the team can gauge how much it has done so far on the product and how much it has yet to do. Even if the project is terminated midstream, the customer receives value from the PSPs delivered so far.

Within each sprint, all steps needed to produce a PSP are performed (analysis-code-test-checkout), although they are *overlapped* and done in “chunks”; this means a little analysis, a little coding, and a little testing on one feature, then a little analysis-coding-testing on another, and so on. The credo is, “Do a little of everything, all the time.” At any given moment different features are in different stages of work—analysis, coding, testing, etc., which is why team members must be capable of doing anything whenever it is needed. Teams keep track of who is doing what by using whiteboards (or kanban boards), discussed later.

As mentioned, the sprint duration is fixed or **time-boxed**, usually 2–4 weeks, based on whatever time the team decides. Time-boxing simplifies planning and estimating since the team learns how much work it can do each sprint (its **work rhythm**) and is able to monitor its output rate. Knowing its work rhythm, the team selects user stories for each sprint backlog that it thinks it can achieve within the time box. The time-box duration is held constant throughout the project, is strictly reinforced, and rarely permits overtime. A typical project involves many sprints, and teams are most effective when they work at a uniform, sustainable work rhythm. If the team cannot finish everything in the sprint backlog, the PO decides which items to drop—and be picked up in later sprints.

Once a sprint is underway, the PO steps out of the way but is always available to respond to team requests.

Sprint increments versus production versions

Although each sprint produces a new end-item increment or PSP, rarely are companies able to absorb new increments of such frequency into their operational or production environments. Most companies can implement revised “production” or user-operated versions only a few times a year—although when they do, each version will incorporate improvements from all the sprint increments released up to that point. Meantime, so that the development team can receive frequent practical feedback, a focus group of about ten people representing the company and its key users reviews and critiques the released PSPs from every sprint. This provides the development team with useful information about needed changes and improvements to address in upcoming sprints. After, say, six sprints or six months, the accumulated improvements are incorporated into the production version of the product.

Scrum planning

Scrum plans are developed progressively. A high-level plan is prepared early in the project, but it contains only basic information about the solution/end-item and has few details. Any commitments specified in the plan tend to be flexible. The planning process in Scrum is analogous to the rolling-wave, phased approach described in Chapter 4, the difference being that the “goal” and its associated requirements in Scrum projects are somewhat fluid and the path to reach them somewhat unclear. As the team’s knowledge about the project expands, so do details about the project plan.



See Chapter 4

Sprint planning (aka iteration planning) meeting

Before each sprint, the team devotes 4–8 hours to preparing for the next few sprints. It selects user stories from the product backlog to comprise the next sprint backlog. For each item on the sprint backlog, the team determines the specific tasks to be done and the estimated time; this enables the team to know that the chosen work is “doable” within the sprint time-box and, later, to track its progress during the sprint. The team also selects stories for the next two sprints (e.g., as it plans for sprint 2, it creates backlogs for sprints 3 and 4, although the backlogs can be changed).

Daily sprint (aka daily scrum) meeting

The team gathers each morning during the sprint for a 15-minute standup meeting to review progress and update its status via a burn down chart and whiteboard (Examples 14.2 and 14.4). The meetings are no-nonsense and held at the same time and location every day. They address: What did you do yesterday, What will you do today, and Are you facing any obstacles? Team members discuss what they must do and the Scrum master learns what barriers he must remove.

Tracking and control

A significant feature of APM in general and Scrum in particular is the relative absence of documentation. In many cases, there are but three kinds of documents (or **artifacts**): the product

and sprint backlogs, which represent the “plan,” and a simple, visual tool like the burn down chart for tracking and controlling work. Following is an example.

Example 14.2: Burn Down Chart¹¹

Progress in each sprint is tracked with a *burn down chart*—a chart that shows remaining work effort (work backlog) on the vertical axis and sprint days on the horizontal axis (Figure 14.6). Work effort—the amount of “sweat” the team devotes to a sprint—can be measured by whatever way the team prefers—story points, hours or days of effort, and so on. During the sprint planning meeting the team identifies the jobs or tasks it will perform, allocates the jobs among the team, and assigns “story points” or estimated hours or days of effort to each. If story points are used, the team assigns points to each task in proportion to the estimated effort required to do it (the more effort, the more points); if hours of effort are used, the team estimates the number of labor hours needed to complete each task.

The total estimated hours of effort for all tasks must fit within the time-box—the number of hours available in the sprint. For example, a team with six members working nominal 6-hour days (= 36 labor hours per day) for 4 weeks (20 days) will have 720 labor hours of effort available, so the work required to complete user stories in the sprint backlog must be doable within 720 hours. In fact, in selecting stories from the product backlog, the team keeps a running tally of the hours of effort involved, and it selects no more stories than can be “fit” within the available labor hours of effort. Although a team will certainly work at least 8 hours a day, by using 6 hours a day in the estimate it is being conservative and somewhat guaranteeing that it will be able to complete the work in the sprint backlog.

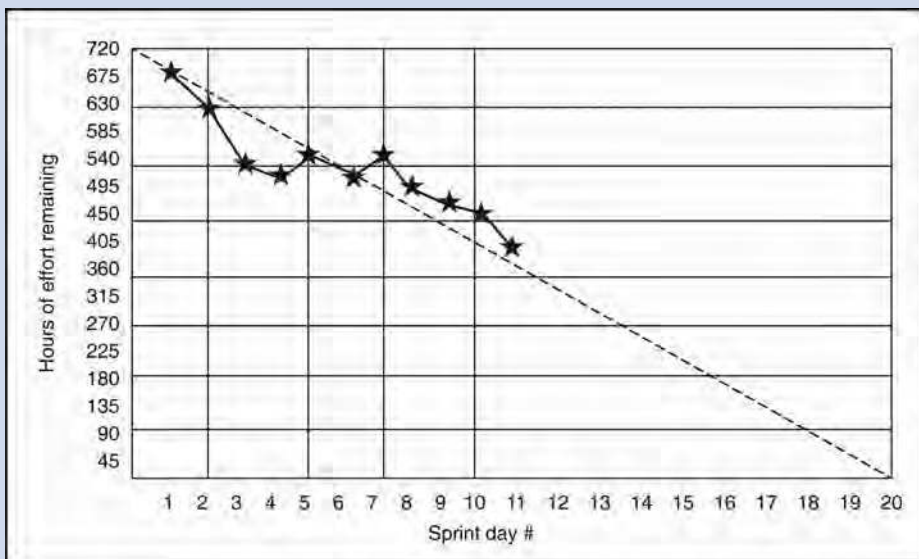


Figure 14.6
Burn down chart, progress as of 11 days.

Continuing the same example, at the start of the sprint the team will have 720 “hours of effort remaining”; if it were to complete every task exactly in the time estimated, then theoretically each day the hours of effort remaining on the burn down chart should decrease by 36 hours (dotted diagonal line in Figure 14.6).

In reality, of course, tasks take more time or less time than estimated. Whenever a task takes less time and finishes early, a new task is started. At the end of the day each team member records the tasks she completed and provides an estimate of the hours of effort still remaining to complete any task she is still working on. For example, suppose on Day 7 a team member starts Task A, which was estimated at 6 hours, and finishes it in only 3 hours. Immediately thereafter she starts Task B and works on it for the rest of the day. Suppose Task B was estimated to take 8 hours, but at the end of day the team member estimates 2 hours of work still remain. (This can be done using percent complete. If an 8-hour task is estimated at 75 percent complete, that equates to $0.75 \times 8 = 6$ hours completed and 2 hours remaining.) Thus, at the end of Day 7, considering only Tasks A and B, the hours of effort remaining on the chart will be reduced by 6 hours (estimated time) for Task A and 6 hours (8 estimated – 2 remaining) for Task B. Working faster than expected shows up on the burn down chart wherever the plotted line is steeper than the estimated, expected line. In Figure 14.6 this happened on Days 2, 3, 8, and 11.

If tasks take longer than expected, the hours of effort remaining will decrease less rapidly than estimated, and the slope of the plot on the burn down chart will be shallower than (or opposite) the estimated line. In other words, at the end of the day, even though the team may have worked well over 36 labor hours, if it did not complete its assigned tasks for the day, the hours of effort remaining will be reduced by less than 36 hours. For example, suppose one day some team members did not complete their assigned tasks, and they estimate that to do so will require 31 more hours. Thus, for that day, the hours of effort remaining will be reduced by only 5 hours (36–31), not 36 hours as expected. Of course, if at the end of a day the estimated hours remaining for tasks that should have been done that day exceeds 36 hours, then the hours of effort remaining on the chart will *increase*. In Figure 14.6 this occurs on Days 5 and 7.

By monitoring the chart, the team can readily identify problems with work scope (number and size of tasks), pace, quality, or errors in the estimated times. It can also gauge its day-to-day progress and whether it will be able to complete the sprint backlog within the time-box.

Work pace in Scrum is called **velocity**. Figure 14.6 shows roughly 405 hours of effort remaining at end of Day 11. Thus, the team has completed $720 - 405 = 315$ hours of effort and its velocity is:

$$\text{Effort Completed/Elapsed Days} = 315 \text{ hours}/11 \text{ days} = 28.6 \text{ labor hours/day}$$

If the team maintains this velocity, it will finish the remaining work in $405/28.6 = 14.2$ days, which means it will have taken $11 + 14.2 = 25.2$ or 26 days to do all the intended tasks—and exceed the 20-day time-box by 6 days.

An alternative to the burn down chart is the burn up chart, described in example 13.10. The two are similar, except the burn up chart shows *completed work effort* (instead of work remaining); thus, plots on the burn up charts move diagonally upward as the work progresses. Besides burn charts, scrum teams typically use other visual tools like whiteboards to monitor and control their work. Relying on visual tools like this, called **visual management**, is further described in section 14.6.



See Chapter 13



See Chapter 13

When a sprint or entire project falls behind schedule, i.e., the planned work cannot be completed within the time-box, then the project scope—not quality, resources, or schedule—is changed. Doing so enables the project to meet its most important goals without compromising quality, increasing costs, or extending schedules. This is as it should be. If the PO had put 50 features in the prioritized product backlog, and if 40 of those were delivered in the specified number of sprints, then by the end of the project the PO and customer will have gotten a fully functional and operational product with 80 percent of the highest priority features, and all within the time and money allotted.

Sprint review (aka iteration review)/retrospective meeting

Concluding each sprint is a one-day, two-part meeting. In the first half of the meeting the team reviews the work it had planned, completed, and not completed, and it presents or demos the completed parts to the PO or customer. In the second half it reflects on what went well in the sprint, what not so well, what it learned, and what it needs to do and improve in the next sprint.

DevOPs

The practice of **DevOps** has received much attention because agile teams sometimes fail to adequately address Phase D operations, in the system design. Between iterations, the design development team (the “Dev”) relies on intermittent feedback from users to learn how to improve the product. Problem is, the team must wait for that feedback and until the next iteration to improve the product, and how long this takes depends partly on the length of the iterations and sprints.

A better way is to incorporate users (the “Ops,” or system operators and support) into the development team, so their input is incorporated into the product as the product is being designed. This is the thrust of DevOps. The approach speeds up the design–build–release iterations because designers don’t have to wait until after the product is released for user feedback because much of that feedback has been already incorporated into the design.

The DevOps team can incorporate user feedback in different ways, by

- Including user “representatives” on the team.
- Including developers who have operational knowledge and experience (**DevOps engineers**).
- Calling from a list users who are available to respond to developers’ questions.

The first way is best because users can offer immediate suggestions regarding whatever they see the developers doing (the ball is in the user’s court). This provides more and quicker information than developers calling on users only when they have questions (the ball is in the developer’s court). Product owners in DevOps projects still direct and have final say about the product, although they often lack hands-on experience and detailed knowledge about user needs and requirements, and about matters such as system implementation, maintenance, upgrade, and ultimate decommission, i.e., the entire product life cycle.

DevOps is a form of **concurrent engineering** and **integrated product development**,

See Chapter 15 details about which are covered in Chapter 15.



14.4 APM Controversy

Many of the methods and tools of TPM were developed in the 1950s to address projects in construction and large-scale systems and product development—the kinds of projects illustrated in this and other project management books. But, as mentioned, applied to software development, the same methods and tools have limited success and often lead to failure.

Bits versus atoms projects

But it works the other way too: APM methods developed for software development (so-called “bits”) projects have only limited applicability in more traditional, hardware (“atoms”) projects. Paraphrasing Lohnes and Wilson:

When dealing with “bits,” which do not have physicality (weight, mass, form, length, height, etc.), requirements can readily be changed if the customer is willing to pay for the rework.

But when dealing with deliverables that involve “atoms,” which do have physicality, that is a different matter.¹²

For example, once an aircraft (a hardware product) has been designed with an 18-meter diameter fuselage, a later change in requirements to 18.5 meters would necessitate significant redesign to many of the fuselage’s components and to components in related systems, which involves significant time and cost. You cannot design and build an airplane (or an artificial heart, or a bridge, or a drug) through iterative sprints. Simply, APM doesn’t work there.

You can, however, design and build some of an airplane’s avionics software through sprints,¹³ which raises an important point. More and more, products thought of as hardware are actually hardware–software products, and the software part overshadows or dictates the hardware part. Airplanes are an example, some of which are able to fly only by virtue of software. Thus, although APM methods may be limited to software end-items, given the ubiquitousness of software-within-hardware, APM’s applicability is vast and growing.¹⁴

Project management principles

With the spread of agile practices, some proponents have prescribed that they replace TPM practices, no matter the kind of project. But *The Agile Manifesto* was not intended to debunk TPM, and a reasoned approach to project management says not to replace one methodology or set of practices with the other, but to choose selectively from both depending on the nature of the project, the end-item, and the stakeholders—in other words, to create *hybrid methodologies*; this is addressed later.

Whether the deliverables are bits or atoms, principles and practices associated with good project management remain the same—the differences being in when, where, and how they are applied. For example, defining requirements is always important: if it can be done early, it should be done that way, as in TPM; if it cannot, then it will have to be done iteratively and evolutionarily, as in APM.

Project control is important too: TPM uses Gantt charts, networks, earned value, and formal change-control methods; APM uses controlled iterations—time-boxed and tracked with

whiteboards and burn charts. Both sets of methods serve to keep projects on schedule and moving toward goals.

APM has been called “lightweight” project management because it emphasizes informal communication and minimum documentation. APM replaces documentation’s information storage and sharing functions with something better: face-to-face communication. People in small, co-located teams working on small tasks in short time bursts know most everything about the project that can be known, and they share it verbally. They do not need documentation. But APM does not eliminate the need for documentation: the end-item must still be documented for the same reasons as in TPM.

In summary, tools and methods of APM are clearly appropriate and desirable for software deliverables, though are less so or won’t work for most physical deliverables. APM has a proper place among project management practices, although many projects still require a more formal and structured approach to definition and execution. It would be incorrect to try to replace TPM with APM and be more appropriate to consider APM as an addition to the project management toolbox. The following shows one successful application of APM to a non-software project.

Example 14.3: Wikispeed Car Project¹⁵

Imagine what it would take to develop a safe, road-worthy, 4-passenger automobile that gets 100 miles/gallon (2.35 L/100 km) and do it in a relatively short time. The instigator of this challenge was a \$10-million X-Prize offered by Progressive Insurance. The Seattle-based Wikispeed Corporation was formed to accept the challenge, and, using APM methods it was able to design and build a fully operational prototype that met requirements in just *three months*.

The project team comprised 44 mostly volunteer members arranged into self-managing teams. Using the Scrum model, the teams worked in one-week sprints. Each sprint was devoted to deciding on the highest-priority aspects of the car to address, and then creating and testing the car components to address them. In every sprint, a team would (1) identify the customers’ wants, (2) define those wants in terms of clear requirements and tests, (3) prioritize the tests, (4) create car components that met the tests, and (5) show results to customers to learn if that is what they really wanted. The teams employed whiteboards and Kanban to track and control work progress.

The car was designed and built based on *modular design* methodology mentioned in Chapter 2. This enabled the project team to focus on different pieces of the car simultaneously and to rapidly try alternatives for each (e.g., substitute a gas engine with an electric one; strengthen door panels to meet federally mandated side-impact tests) and ensure that all the pieces would fit together.

Despite deviating from typical Scrum principles such as small development teams (Wikispeed team had 44 members, though individual sprint teams had fewer), co-location (members were distributed geographically across 4 countries), and daily sprint meetings (instead, hour-long weekly standup meetings were held), the team was able to effectively prioritize requirements, conduct sprints, communicate progress and issues, and get customer feedback, mostly via electronic media.

Wikispeed did not win the X-Prize, although it did very well, creating a fully-functional car on a shoe-string budget that placed 10th out of over 100 competitors, some from well-established automobile corporations. The company continues to innovate and improve a car that someday might be ready for the marketplace.



See Chapter 2

The purpose of the Wikispeed example is not to suggest that APM directly applies to or is widely used in hardware development projects, but rather to show that aspects of APM can be effectively applied there by adjusting it to the circumstances.

14.5 Lean Project Management

Lean project management (LPM) is derived from lean production, which is the concept of doing work for the least time, resources, and effort without diminishing quality or output. Lean equates to “no-waste,” where waste is anything that adds to the cost but not to the value of a product or end result. Toyota, the originator of lean production, identified particular kinds of waste. Those most relevant to project environments are overproduction (production beyond what is required), inventory (holding excess materials or unneeded resources), extra processing (doing more work or higher quality than needed), waiting (delaying work), defects (incorrect or inadequate work), and motion (unnecessary work, hand-offs, and transfer). An additional waste is non-utilized human talent, i.e., not taking full advantage of workers’ ideas, skills, and abilities.¹⁶

Lean production emphasizes “workflow” or “product flow,” the concept that anything moving through a multistage process should flow unhindered from one stage to the next. In manufacturing the “flow” is of materials moving through the production process and being transformed or assembled into a product. In projects the flow is of information and sequences of work tasks to create conceptual or physical end-items.

LPM also emphasizes incremental and continuous improvement. The collaborative effort of team members results in a continuous flow of suggestions. And, since it is easier to make small incremental changes than big ones, the LPM emphasis on dividing projects into small tasks enables the team to make small continuous improvements to whatever they are working on.

Despite originating in manufacturing, many lean production concepts and practices apply to projects; these include *small batch flow*, *minimal waiting*, *cycle timing*, *standardization*, *minimum hand-offs*, *self-managed teams*, *visual management*, and *Kanban*. Many of these concepts overlap with APM practices (hence APM is sometimes referred to as “lean project management”) and many lean practices are categorized as APM. Lean concepts and practices can be found in all kinds of projects, not just bits-oriented (software) projects but also atoms-oriented (construction)¹⁷ and bits- and atoms-oriented (product development) ones.¹⁸ In other words, concepts associated with lean project management can be applied to both TPM and APM projects. Despite commonality, lean and Kanban project management are often treated separately. We will do the same here, although the similarities will be readily apparent.

A thrust of LPM is to divide everything into small pieces—analogueous to executing a large project through a series of short sprints. Performing work in this way is like doing it in small “batches” where a batch represents a quantity of work to be done. What this means and the benefits it affords are described next.

Small batch flow¹⁹

Look at a Gantt chart or network diagram; what you see are project stages or activities linked in order of precedence. You might see “Requirements” followed by “Code,” “Code” followed by “Test,” “Test” followed by “Integration,” and so on. The implication is that the stages or activities are done in sequence. Upon completion of one, the results and work are transferred to the next.

In manufacturing, work moving from stage to stage like this is called “batch production” and the implication is that work moves through the process in *batches*. If the size of the batch is 40 items, 40 items are processed at each stage, and nothing moves to the next stage until

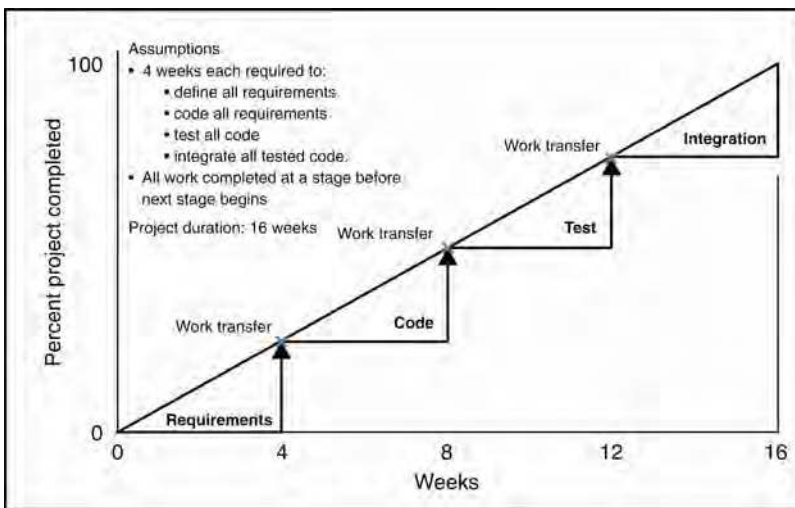


Figure 14.7

Four-stage process where work is transferred monthly at completion of each stage.

all 40 are completed. Forty items go to stage 1, then 40 go to stage 2, and so on. In a project the batch might constitute 40 requirements, 40 drawings, 40 coded modules, 40 tests, etc.

Figure 14.7 shows how this works. Upon completion of the Requirements stage, all requirements are released to the Code stage; upon completion of the Code stage, all coded modules are released to the Test stage. The key word is “completion:” all work in one stage must be completed before any results are released and the next stage can begin. The process in the figure requires 4 weeks per stage and results in a release and transfer every 4 weeks.

In a product-development company, you can sometimes see this happening as stacks of paper (requirements, drawings, work orders, and other documents) move from department to department. Digital information moves like this too but is not obvious since it is stored and transferred unseen.

An alternative way for this to happen is to transfer *some* of the work at each stage, *every* week. For example, instead of waiting until all requirements are completed, transfer whatever requirements are completed each week so the next stage can get started.²⁰ Figure 14.8 illustrates this, showing work flowing from stage to stage on a weekly basis. Now, it might be possible to transfer the work more often than this—daily or even item by item. Transferring individual items—requirements, coded modules, test result, etc.—like this, between stages, one by one, is called **one-piece flow**, meaning that everything moves through the project in batches as small as size one.

Project duration

With smaller batches, project stages overlap, and the project finishes sooner. In Figure 14.7, where results are transferred between stages every 4 weeks, the project takes 16 weeks to complete. In Figure 14.8, where results are transferred weekly, the project takes 7 weeks.

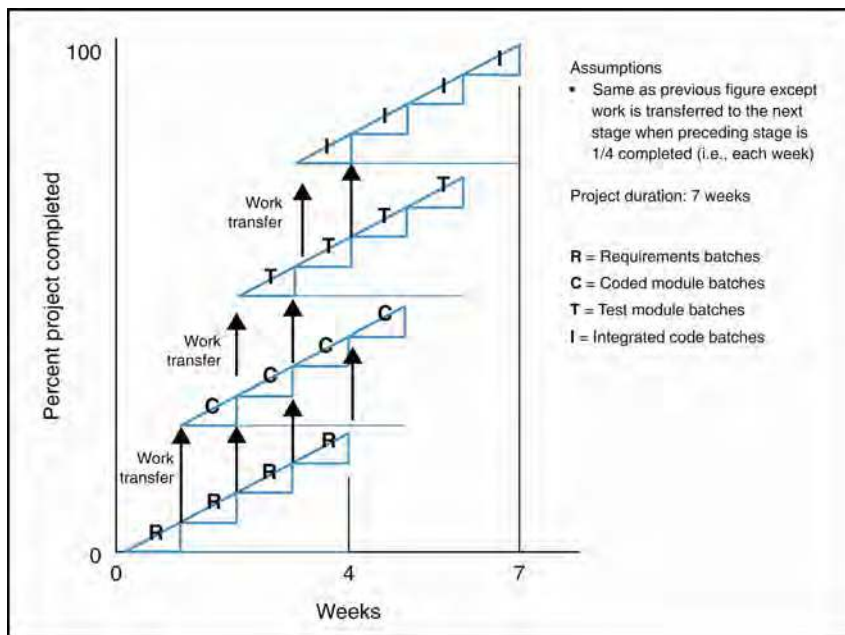


Figure 14.8
Four-stage process where work is transferred weekly.

Quality

Suppose a mistake made in Requirements is not detected until Test, which per Figure 14.7 happens at least 4 weeks after the Requirements stage is completed. If the mistake affected subsequent requirements and coding, many requirements and much code might have to be redone.

With the smaller batches in Figure 14.8, the mistake will be detected in the Test stage less than 2 weeks after it originated in Requirements. At Week 3 some requirements and half the coding will have yet to be done and the mistake will not have influenced them. Also, at Week 3 the Requirements and Code stages are still underway; the people in those stages, having recently worked on the affected requirements and code, will more readily be able to identify the source of the mistake and correct it.

Feedback

The quality benefit of smaller batches stems from accelerated feedback. In Figure 14.7 the Requirements and Code stages have no opportunity to receive feedback from the Test and Integration stages. Requirements and Code will have been completed; the only way for those stages to benefit from information gained in later stages is to restart them. People who worked in those earlier stages will have moved on to other work and it will be difficult to bring them back. Plus, people forget, and it might be hard for them to determine the nature of a mistake and to correct it (see efficiency, below).

In all projects, requirements are ideally firmed up as early as possible; paradoxically, the best way to do that, especially when the requirements are uncertain, is to *learn* from information

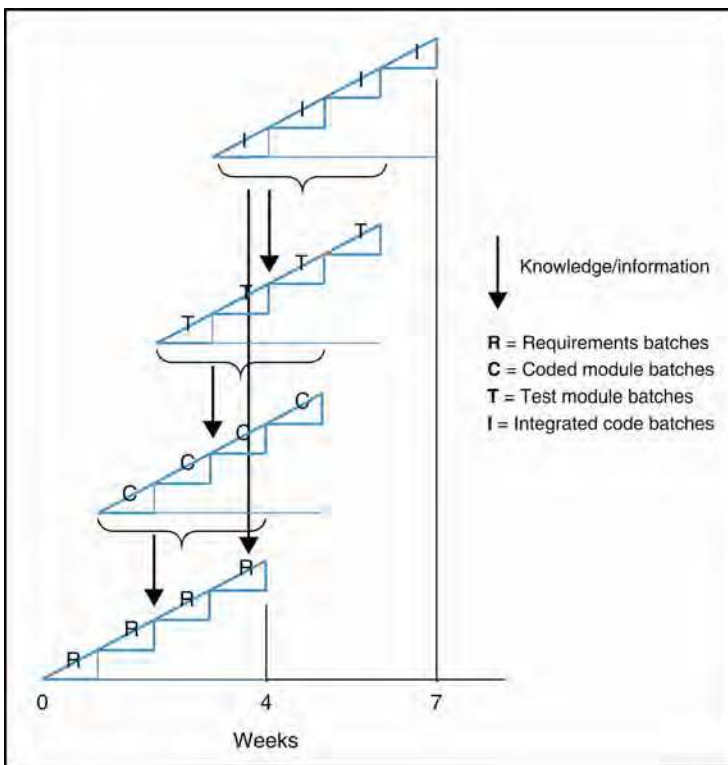


Figure 14.9

Feedback when work is shared weekly: sharing knowledge and information gained in later stages with earlier stages.

based on discoveries made later—in the Test, Integration, and Launch stages. Small batches permit this: if someone in the Integration stage determines a requirement must be changed, she can communicate that back to the people working in Requirements, which is still underway (Figure 14.9). If someone in Test determines the code must be changed, she can communicate that to Code, which is also still underway. Frequent feedback enables decisions to be revised based on the latest information.

Efficiency

Suppose the Code stage created programs from a batch of 40 requirements, but then it was discovered that all 40 requirements were based on an initial, incorrect assumption. This will require that all 40 requirements, and the code based upon them, be corrected. If, instead, the Code stage had received only the first 10 requirements, identified the incorrect assumption, and informed those creating the requirements, the next 30 requirements would not include that assumption and only the first 10 requirements and associated code would have to be redone. And, redoing them will take relatively less time because everything is still fresh in everyone's minds. In general, quicker feedback results in quicker fixes; when people get delayed feedback, they waste time trying to figure out what they did wrong.

Workload variability

Large batches between stages impose an up–down workload on project resources. An engineer who can readily test two items a day will be overloaded by the arrival of 40 items. As large batches move through a project, resources at each stage are stretched. Says Reinertsen, it’s like watching “an elephant move through a boa constrictor,” progressively overloading stages of the system along the way.²¹ Smaller batch transfers result in smaller workload variability and a more uniform workload for project resources.

Urgency and motivation

A programmer who must deliver written code for testing *tomorrow* is motivated and puts the code at the top of her to-do list. She won’t feel so motivated if the code has to be delivered 30 days from now as part of a batch that will include a big pile of code from many programmers.

Overhead cost

The larger the batch, the more items that must be checked and reported on. If software is tested in a large batch, there might be many bugs to identify and rectify—so-called “open bugs.”²² If there are 100 open bugs and one more is discovered, all 100 will have to be reviewed to see if the new one is unique or duplicates any of the others. If there are only ten open bugs, each new one will have to be checked against only nine others. Status reports would address 10 bugs instead of 100.

Risk

Much of the risk in development projects comes from the threats of changing customer needs and wants, preemption by competitors, and emerging new technology. Smaller batches provide quicker feedback about all of these, and enable the project to finish quicker and, thus, be vulnerable to risks for a shorter time.

Batch size versus iteration

Batch size and iteration are related concepts. TPM is large-batch, single-iteration: do each stage once and complete everything before moving to the next stage. APM is small-batch, many-iterations: repeat stages iteratively, and in each iteration address only a portion of the solution/end-item and take advantage of feedback from prior iterations. Scrum is iterations within iterations: do a little of everything each day to build a small piece of the system; combine the pieces during each sprint to create a stand-alone, usable result; in the last sprint, combine the results to create a completed product.

In general, the batch size and iteration frequency should depend on the uncertainty: projects with greater uncertainty should use smaller-sized batches (shorter tasks), repeated more frequently to allow quicker feedback and quicker changes. APM projects handle uncertainty in this way, but so can TPM projects, e.g., give the design group requirements as soon as each requirement is created; give the modeling group drawings as soon as each drawing is created;

and so on. This principle applies as long as the risk consequence of making an error or doing rework is less than the value of doing work quicker and getting feedback quicker.

Queue size and wait²³

Queues—which are **backlogs** of work waiting to be done—are the project equivalent of inventory or **work in progress (WIP)**. In general, the longer the queue or greater the WIP, the longer the waiting time (think of people waiting in line for something; the longer the line, the longer the wait).

Queue length and batch size are related: the larger the batch arriving at a stage, the longer the queue. Figure 14.10 shows the work queue size at each work transfer point. The shaded triangles represent growing queues as items are processed at each stage. The right-hand side of each triangle is the amount of work transferred to the next stage; think of it as a batch of items that must wait in line at the next stage to be processed. Figure 14.11 shows what happens with smaller batches: smaller triangles represent smaller queues and shorter waiting times.

In manufacturing systems, queues are easy to see—they are physical items—and when the queues get too large managers try to shorten them. But in projects, queues are often invisible because they consist of information stored on computers; unaware of them, managers don't feel compelled to do anything. But long queues—even invisible ones—have the same drawbacks as large batches, and the same benefits to reducing them. Two ways to reduce average queue size are to speed up the stage (move work through it faster) or reduce the batch size of items arriving at the stage. Whereas work at a stage often cannot simply be “sped up,” usually the sizes of work batches arriving at the stage can be reduced.

Cycle timing and standardization

Cycle time is the time to complete a job or produce a unit. It usually relates to repetitive or iterative work and the time to, say, complete an iteration. In a project, it can be the time to

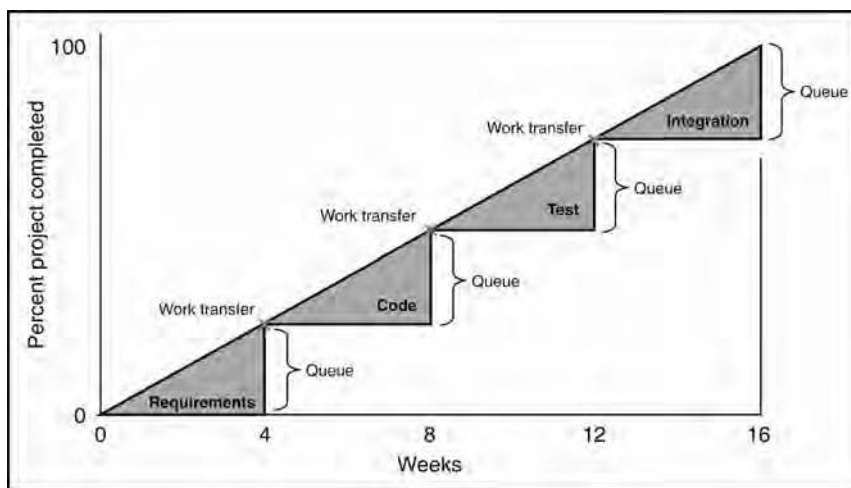


Figure 14.10

Queue length when work is transferred monthly.

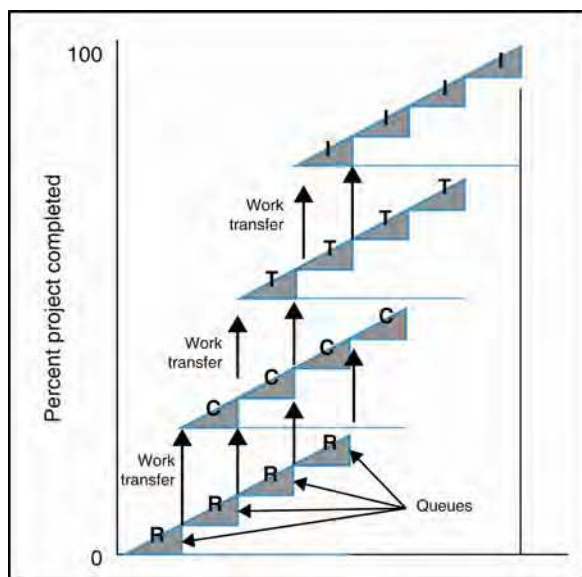


Figure 14.11
Queue length when work is transferred weekly.

build a model, code a module, complete a test, launch a new product, and so on. **Cycle timing** infers the concept of regularity or rhythm and devoting roughly the same time to build every model, perform every test, launch every product, etc. In terms of scheduling, it says, for example, “we will complete one coded module every day” or “we will complete one test every week” or “we will release a new version of the product every month.” Cycle time for each sprint, iteration, or test can be tracked on a chart to reveal trends.

Cycle timing is related to the concept of **smooth workflow**, or work that is performed uniformly to a beat, rhythm, or pace, without interruption. A plot of cycle times versus time should be roughly a flat line. Such uniformity reduces uncertainty about workload variability and enables managers and workers to better meet work expectations; for example, each day a programmer knows she is expected to complete one coded module. Smooth flow is achieved partly by periodically moving work stage-to-stage and in small batches. In APM projects, the workflow moves in this manner—the concept of **velocity**. Consider for example a project that involves daily cycles of code–test–integration. This means every day programmers will complete and submit code that is ready to test, and every day testers will complete tests and submit tested code that is ready to integrate. If code is not ready to test or to integrate, it is not submitted, but because everyone knows what is expected each day, most often they do it. Everyone receives, produces, and delivers known entities every day.

Cycle timing is aided by the principle of **standardization**, which refers to setting standards on processes, tasks, schedules, etc., so that everyone tends to do work in the same way, in the same amount of time, and use the same steps and resources. Standardization reduces uncertainty and variability regarding what has to be done and the resources needed to do it. In lean organizations, the people most familiar with the work—those who do it—set the standards; this is part of the self-management principle described later.

Meetings and reviews²⁴

The applications and benefits of small batches and cycle timing apply broadly; a good example is meetings and reviews. Instead of scheduling infrequent, long meetings, use standard,

frequent, short meetings—daily standups. Short, frequent meetings permit quicker feedback and attacking issues immediately and with urgency. The overall meeting time stays the same since the time for many short meetings is about the same as for fewer long meetings.

Schedules for meetings and project status reviews are standardized and cycle-timed, i.e., meetings are convened at the same time and place, every day or same day every week, with the same durations. This reduces planning and coordination effort and decision delays: whereas an ad-hoc meeting might take several days to arrange and, thus, delay decisions by several days, daily meetings require no arrangements and delay decisions by at most 1 day.

The benefits of short, regular, frequent meetings are seen everywhere. Reinertsen tells the story of Hewlett Packard where every morning and afternoon the coffee cart came rolling through the departments.²⁵ Twice a day engineers would gather around the cart, talk informally, and cross-pollinate ideas. When the coffee cart was terminated, engineers had to go to the cafeteria for coffee. Everyone went at different times; conversations and cross-pollinating dropped off significantly.

Similar standardized-cycle-timed rationale applies to resources that support multiple projects: they should be available to each project at a scheduled time every day or week. For example, make the manufacturing representative available to the project team from 9 to 10 am each morning. That way anyone who needs to work with the representative will have an hour during which the rep is fully available. If someone has an important issue at other times, she can request the representative's assistance; otherwise, she waits until the next day. The time the resource is available can be adjusted depending on demand or the stage of the project.

14.6 Kanban Project Management

LPM and **Kanban project management (KPM)** are similar, although the emphasis in LPM is on reducing work “batch” sizes while in KPM it is on the lean concept of Kanban for tracking and executing work. As with virtually all APM methodologies, both rely on self-managed teams and visual management.

Self-managed teams

Lean philosophy recognizes that problems and opportunities in a process are most readily identified by the workers in the process and, given appropriate skills, those workers are best-suited to make decisions and take action. This is the lean principle of *self-management*, which refers to empowering worker teams and providing them with the skills and information necessary to act without direction from supervisors.

In a **self-managed team**, everyone can assist everyone else, and each person is able to do a little of everything, regardless of her specialty. Reinertsen calls these **T-shaped resources**, i.e., people deep in one skill area but broad in many—Jack-of-all trades, master of one.²⁶ T-shaped resources are developed by hiring “I-shaped resources” (deep skill in one area) and giving them assignments to expand their skill set. In a project this starts by giving workers assignments in “adjacent” stages of the project—stages that ordinarily provide inputs to or receive outputs from the worker. For example, analysts are cross-trained to do programming, and programmers are cross-trained to do analysis and testing. When the programming queue gets too long (perhaps as revealed on a whiteboard), analysts stop doing analysis and start doing programming; when the testing queue gets too long, programmers stop coding and begin testing. Same goes for everyone. The ideal leader in a self-managed team—whether scrum

master or project manager (if the team has one, which agile-lean teams often do not) is the typical **servant leader**, a person who is able to bring out the best in people while motivating them toward a collective goal. Among the attributes of servant leadership, says Greenleaf—the concept’s founder—are empathy, careful listening, gentle persuasion, foresight, honesty, integrity, caring, and stewardship—concepts echoed elsewhere in project leadership as covered in Chapters 16 and 17.²⁷



See Chapter 16 and 17

In many projects, team members commonly work in different departments, buildings, or wherever they are needed. For a self-managed team, however, it is best to physically co-locate everyone—this is a maxim for APM projects but applies also to concurrent engineering in TPM projects. **Collocation** maximizes information exchange, feedback, and cooperation. Team members share information constantly—and in small batches. They learn about each other’s interests, backgrounds, and families, which builds team cohesion.

Collocation also eliminates a big source of inefficiency in projects—**handoffs**, which is the transfer of tasks and work items between workers, departments, and contractors. Handoffs are wasteful: information is transferred inadequately or incorrectly, and jobs are held up. Collocation minimizes this because everyone already knows what everyone else is doing.

Visual management

Self-managed teams are aided with **visual management**. As discussed in Chapter 13, this refers to using visual tools such as charts and diagrams for tracking work status and problems and making decisions about what to do. In combination with daily standup meetings, visual management enables self-managed teams to monitor their work and measure work progress.



See Chapter 13

One kind of visual tool is the burn down chart, described in Example 14.2 and shown in Figure 14.6; another is the **whiteboard** (or **kanban board**), which is a main feature of KPM. *Kanban* is the Japanese word for “card” or “signal,” and in lean manufacturing all work is controlled via cards to signal when work at each stage should begin. In essence, work is “pulled” through the process and proceeds only with a signal; no work is permitted to move to the next stage until a signal is received that the stage is ready to receive it. Although KPM somewhat twists this principle, it yields the same benefit: expedited workflow through the process. With the help of a whiteboard, a team can see the work in progress and adjust its capacity to meet demand, which ultimately speeds up workflow.

The whiteboard (example in Figure 14.12) is divided into columns to represent the stages of a process (e.g., to-do, in process, verify, done) or a project (e.g., analyze, design, build, test, install). The columns and titles are tailored for each project to represent chronologically the steps, stages, or phases of the project. Additionally, the board can be divided into rows, called swim lanes, to differentiate tasks for different teams, projects, customers, etc.

Each task, job, user story, or assignment to be done is represented by a card or note and is assigned an owner—someone on the team responsible for the work. The card (typically a sticky note) is attached on the board in a column to represent its current status (e.g., the card for a particular requirement would be attached in the Verify column after work for it had been completed, represented by the In-process column).

To do	In process	Verify	Done

Figure 14.12
White (or kanban) board.

Tasks are moved through the process, stage by stage (on the board, column by column) whenever they are ready; in Kanban there are no schedules or due dates. The team (or, sometimes, project manager) updates the board on a regular basis, typically daily, during or before standup meetings when team members report their progress. Use of the whiteboard is illustrated in Example 14.4.

Example 14.4: White (aka Kanban) Board²⁸

Each job, written on a card or sticky note, is placed on the *To do* column of the board, which represents the first of a hypothetical four-stage process that includes *In-process* (being done), *Verify* (done correctly), and *Done*. Each job is a task that must be done (e.g., a requirement or function, or piece of a user story in the sprint backlog, etc.), and different kinds of tasks and their owners can be represented by different color cards.

Figure 14.13 shows project status at three separate times. As jobs progress, the cards are moved on the board from one column to the next.

To do	In process	Verify	Done
A B C	D E	F	
A, B, and C are ready to do. D and E are being worked on. F is being verified.			
To do	In process	Verify	Done
H I J	A B C	E D	F
F is done. E and D are being verified. A, B, and C are being worked on. H, I, and J are ready to do.			
To do	In process	Verify	Done
J →	C → H I →	E D B	F A
A is also done. E, D, and B are being verified. C and I are ready to be verified but cannot due to the kanban constraint of three jobs per section. They can be moved only after space opens up in Verify. Similarly, J can be moved when space opens up in In-process.			

Figure 14.13

White board showing progress at three times.

An important part of Kanban is *delimiting* the work buildup at each stage of the process. Suppose the team has decided that no more than three jobs at a time can be in a column—this is the work in progress (or WIP) limit; when a column has three cards, no more are allowed. This will prevent the team members at that stage from taking on more tasks before completing existing tasks. By scanning the board, the team can see at any given time the process stages that have reached the limit and need additional resources.

Work in Progress (WIP) represents work backlog, and stages with large backlogs become bottlenecks that slow or stop workflow. In the example, whenever a stage accumulates three jobs, that means the stage is overloaded, in which case team members at other stages join to reduce the WIP. When the number of jobs at a stage drops below three, the stage is ready to receive another job. Note that for this process to be effective, team members need to be familiar with the works at all stages of the process—T-shaped resources. To promote workflow uniformity, individual jobs should be sized roughly the same and require about the same time.

WIP limits are usually set by trial and error after monitoring a few projects or iterations. They must be high enough so that everyone is kept busy, yet not so high that members have too much work and must struggle to complete it. The WIP limit can vary for each stage; as a rule of thumb, it should be less than the number of members on the team, though that depends on the skill and experience of the team: the better the team, the more WIP it can accommodate.

Although kanban boards created and updated manually with sticky notes are usually adequate, digital, software-managed kanban boards are also available.

Use of the kanban board is sometimes accompanied with the cumulative flow diagram (CFD), which is analogous to the burn chart and is used to display progress. The diagram corresponds to the kanban board and shows cumulative tasks completed at each stage as of a given date. Whenever the kanban board is updated, so is the CFD. For example, each day, whenever cards are moved from one stage to the next on the kanban board (Figure 14.13), lines on the CFD for those stages are updated (Figure 14.14). If, for example, two cards are moved to the Verify column on the kanban board, the Verify line on the CFD is increased by two jobs. If, say, the project consists of 18 jobs, when all 18 have moved through all stages of the project (and, thus, the Completed stage), then the project is completed. In Figure 14.14, the “Completed, projected” line represents a crude estimate as of the status date of when the project will be completed.

The whiteboard, CFD, burn chart, and other information about problems, solutions, and actions are most effective when displayed together on a **big visible chart**, located prominently at the workplace, and used as a gathering point for team stand-up meetings or **huddles**, which are brief meetings, convened early each day, for the purpose of reviewing progress, identifying obstacles, deciding next steps, and updating information on the chart.

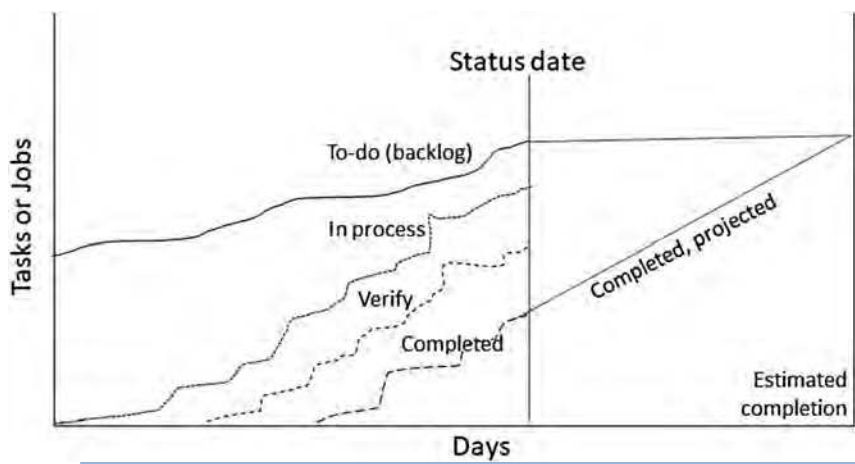


Figure 14.14
Cumulative flow diagram.

14.7 Hybrid Methodologies

Kanban boards are frequently used in Scrum to monitor and control the progress of sprints. Joining methodologies like this—Kanban and Scrum—results in a **hybrid methodology** (in this case called **Scrumban**), which can provide the combined benefits of both.

Determining which methodologies to combine for a project depends on the project, team, and customer and whatever methodologies are deemed practical and able to provide the best means to conduct and manage the project. For example, kanban boards can sometimes be used in TPM projects as an additional tool for tracking work progress. They help project managers to identify bottlenecks, allocate resources to work tasks, and enable teams to better complete work through delimiting the number of work tasks they must do. But they are most effective in small projects and where the work moves linearly, but not on large, complex projects that require the coordination of numerous tasks among multiple teams and contractors. Also, they cannot be used in projects that require the back-and-forth repetition of stages (e.g., build–test–build).

Often, different combinations of TPM and APM tools and practices can beneficially be employed in different phases of the project. For example, TPM practices for contracting, planning, and procurement can be employed during project conception and definition, followed by APM practices for design and development work during project execution. Or, APM practices could be employed early in the project to develop the product concept, followed by TPM practices for the remainder of the project. Determining the best hybrid methodology for each project requires complete understanding of the project, the team, its customers, and the various methodologies, plus a willingness to experiment with them.

14.8 Summary

Traditional project management (TPM) methodology applies to projects where the problem/goals and solution/end-item are well-understood and can be well-defined. The project phases/stages are largely completed in sequence. In incremental TPM, a variation of TPM, the end-item is implemented in stages or pieces, which enables portions of it to be put into use sooner.

Agile project management (APM) applies to projects where the solution/end-item is somewhat or largely uncertain. It accommodates this uncertainty through a learn-as-you-go process of iterative steps in the execution phase, where each iteration leads to the release of an end-item “increment” and a better understanding of the ultimate end-item.

In the spiral model form of APM, the end-item is released through a series of multiple cycles, where each cycle consists of the stages of analysis, design, develop, test, etc., and results in a prototype. With each cycle the customer provides feedback to the developer, who creates an improved version of the prototype and, ultimately, a final product.

Scrum, another form of APM, is commonly applied to software development. The scrum process starts with the product owner listing customer/user needs and wants on the product backlog, and the team addresses these through a series of sprints. Each sprint focuses on a subset of items in the backlog and results in the release of “something of value” to the customer—a potentially shippable product. The team receives feedback on the release as input to subsequent sprints.

Lean production implies performing work for the least time, resources, and effort without diminishing work quality or output. It emphasizes smooth workflow—that anything going through a multi-stage process should move unhindered. APM is sometimes called “lean project management,” LPM, although many practices of LPM apply to TPM projects as well.

One lean concept is the use of small batches. By decreasing the batch size, project stages can be overlapped: the project finishes sooner, mistakes are detected earlier, and the team gets quicker feedback from later stages and from the customer. Batch size and iteration are related concepts. APM is small-batch, many-iteration: stages are repeated iteratively; each iteration addresses only a portion of the solution/end-item and takes advantage of feedback from prior iterations.

Lean philosophy recognizes that the problems and opportunities in a process are often first identified by workers in the process; given the right skills and information, they are often the best-suited to make decisions and take action. Kanban project management is an aspect of APM and LPM that relies almost entirely on self-managed teams and visual tools like whiteboards, self-managed teams and visual management are also employed in TPM, but less frequently.

Hybrid project management methodologies are created by combining tools and practices from both APM and TPM. Creating a hybrid methodology to serve the unique circumstances of a project can provide a superior way to manage and conduct the project.

The preceding sections of the book described how project managers, organizations, and teams plan, organize, execute, and control projects, though said little about the project managers, organizations, or teams themselves. The following section addresses project *organizational behavior*. It focuses on managers and teams and the topics of organization structure, leadership, participation, and teamwork.



Review Questions

1. What is the main characteristic of projects for which TPM (*waterfall approach*) applies?
2. How are changes in requirements handled with TPM?
3. How does incremental TPM differ from TPM? What is “incremental” about it?
4. For what kinds of projects is TPM inappropriate or poorly suited?
5. Describe how APM can be described as a “learn-as-you go” approach.
6. What happens during each iteration in APM? What are the expected outcomes of an iteration? How long are the iterations?
7. Describe the planning process in APM.
8. How is APM different than incremental TPM?
9. Describe how the spiral model works and the outcomes of each cycle. In what ways does the spiral model differ from Scrum?
10. Can you imagine projects where both APM and TPM would apply? Describe them.
11. Define each of the following: product backlog, user stories and epics, sprint backlog, sprint, time-box, conditions of satisfaction (COS), potentially shippable product (PSP).
12. Define the roles and responsibilities of the scrum master and product owner.
13. Describe significant features of a Scrum team—roles, structure, size, responsibilities of team members.
14. How are the results of “sprint increments” turned into production or operational product versions?
15. What happens during the following: sprint planning meetings, daily scrum meetings, and review/retrospective meetings?
16. Describe how a whiteboard is used for tracking jobs/tasks.
17. Describe how a burn down chart is used for tracking work progress.
18. If a three-person team is to work 6 hours a day for a 15-day sprint, what is the total number of labor hours available in the sprint? What will the hours of effort remaining be at the start of the sprint?

19. On Day 7, Helena starts and completes Task A, which was estimated to take 4 hours. She also starts Task B, estimated to take 5 hours, but at the end of the day she guesses that the task is about 60 percent complete. For Day 7, by how many hours has Helena reduced the hours of effort remaining?
20. As of Day 7, the sprint has 59 hours of effort remaining. The sprint started out with 120 hours of effort remaining and was time-boxed to 15 days. What is the velocity? Will the project finish within the time-box?
21. If a sprint is falling behind, why not work the team overtime to complete the backlog?
22. Why is application of APM methods limited to certain kinds of projects? Why might it be difficult to apply APM to construction, large-scale infrastructure, and hardware-product development projects?
23. What does “lean” in lean production refer to?
24. What does “flow” refer to in projects? What is it that is flowing?
25. What are the “batches” in projects? What is small batch flow?
26. Explain how small batches: reduce project duration; improve quality; increase feedback, efficiency, and motivation; and decrease workload variability, overhead cost, and risk.
27. There are also drawbacks to small batches (hence, benefits to large batches). What are they?
28. Describe the connection between batch size and iteration in projects.
29. Describe the connection between batch size, queue size, and waiting time.
30. Where are the queues in a project, and what, exactly, is waiting in the queues?
31. Is the gating process large batch or small batch?
32. What is cycle timing? How does it apply to projects and what are the benefits?
33. What does standardization mean in a project context? Give some examples.
34. Describe self-managed teams. Are they the same or different than the teams in APM?
35. How do self-managed teams reduce the problem of handoffs?
36. What is visual management? Give examples.
37. What is Kanban project management? Explain how it employs visual management tools (e.g., whiteboard and cumulation flow diagram) to track and control work progress.



Questions About the Study Project

Do you think or know as a fact that incremental, iterative, agile, or lean practices were used in the project? If so, answer the following questions.

1. Was the project performed in iterations? If so, why? What aspects of the project (phases or stages) were being iterated?
2. If the project was performed in iterations, what were the outcomes of each? Would you say they were “increments”?
3. If the spiral model was used, how was it similar or different than the model described in the chapter? If the Scrum approach was used, how was it similar or different? In answering this, refer to the terms mentioned in Question 11 above.
4. Were there a scrum master and a product owner? What were their roles?
5. Was there a project manager? What was her role?

6. Describe the project team, roles of team members, responsibilities, and how the team functioned.
7. Who were the other stakeholders and what were their roles?
8. How was the project planned? How was it tracked and controlled? Were whiteboards, burn down charts, cumulative flow diagrams, or other methods used?
9. Is the project manager aware of “lean methods”? Does she apply them to project management?
10. Would you say that the work on this project moved in small batches or large batches? (What are the “batches” made of?) If large, explain how/where the project might have benefitted from small batches.
11. Did you observe work tasks waiting or being delayed that could have benefitted from smaller batch transfers?
12. Did you observe cycle timing or task standardization in the project?
13. Were visual management methods or the Kanban concept applied in the project?
14. Was the team self-managed? If yes, discuss in what ways it managed itself.

CASE 14.1 GRAND ENTRY FOR ACCENT, INC.

The goal of the Grand Entry project was to provide a new web portal for employees of Accent, Inc., a replacement for the existing portal, which employees found cluttered and difficult to use. The project’s mission statement was to “Create an improved user experience by developing a simple and intuitive system that allows users to navigate and access the desired contents quickly and efficiently.” Among its objectives were “innovative design, simple and intuitive interfaces, features to encourage users to return to the site, and use of an agile process to develop the site.”

The CIO had heard about agile methodology and thought Grand Entry would be good place to try it out. He assigned the project to Theodora Lamar, a software project manager with much experience, although none in agile. Senior management provided no budget but told Theodora to hold costs to “between \$400,000 and \$600,000” (to cover salaries of workers assigned to the project) and shoot for completion in 16 months. She was to conduct the project in “agile fashion” wherein each sprint was to deliver “additional functionality” that “stakeholders” would review and approve.

Theodora read some articles on agile and Scrum, appointed herself Scrum master, and selected two analysts for the Grand Entry (GE) team. Their first action was to form a focus group through which portal users could vocalize their dissatisfaction with the current system. The team chose for the focus group two Accent employees, recently hired and not yet assigned to specific work. Although new to the company, they had recent experience with the portal and would know its limitations. Their role was twofold: (1) talk to as many fellow employees as possible to learn the problems of the current portal and get ideas for improvement; (2) use the deliverables from each sprint and make suggestions. Initially they would devote all of their time to the project; later they would split their time between the project and other work assignments.

Theodora and the analysts first interviewed the two employees and then three senior managers, including the CIO. Their comments and suggestions formed the basis for the original list of “user requirements” for Grand Entry. Theodora reviewed the list and chose the ones she thought would be the most realistic to implement. She then

selected three more people from different departments to join the GE team—an architect, a developer, and a support resource. None of them had worked together, but Theodora knew them all and felt they were technically “the best.” Besides the GE team, the other technical party involved in the project was Metasoft, the developer of the browser platform upon which the portal resided. Metasoft would handle all project issues relating to the browser.

The GE team identified itself, the focus group, Metasoft, and the three senior managers as the project “stakeholders.” It did not include or communicate with the group charged with maintaining and updating the existing portal and, as a result, was not aware of problems that that group had already discovered. This resulted in some duplication of effort as both the GE team and the maintenance group worked on the same problems; the GE team even tried using features that the maintenance group had already found unsuccessful. When the maintenance group eventually learned of the project, they initially resisted the GE team’s requests for assistance. Only several weeks later did they start to cooperate.

Theodora planned the project in a 4-month, rolling wave fashion; that is, she prepared a plan to address the requirements in an upcoming 4-month period, and intended to repeat this four times during the 16-month project. The plan did not specify the expected number of sprints or their durations. Each sprint was to last 2–3 weeks, depending on Theodora’s estimate of how long it would take to complete the requirements she had selected. Some sprints originally planned for 2 weeks stretched to 3 weeks when the work took longer than anticipated.

During the project, the browser shut down twice and halted the project. The GE team was at the mercy of Metasoft, which had assigned no special priority to the project and took several days to fix each shutdown.

Each sprint resulted in a beta version that the GE team demonstrated to the focus group. The focus group’s typical response consisted of suggested improvements beyond what the team was capable of addressing in the next few sprints. This created a backlog of new requirements and open issues, from which Theodora could select only a few as focal points for the next sprint. Since so many previously identified issues had not been addressed, the focus group kept re-identifying them; thus, instead of identifying new, more pressing issues, the group kept pointing out issues the GE team was aware of but hadn’t had time to fix.

A few months into the project, additional senior managers started to sit in on the beta demonstrations and add their own suggestions for improvements. Consequently, issues intended to be resolved or functionalities to be added in upcoming sprints were superseded by new requirements. The list of stakeholders grew as more managers learned about the project and attended the demos.

Theodora’s response to the growing list of requirements was to impose more work on the team. She had tried to avoid overtime, but not ever being completely aware of what the team was working on, she simply requested them to do more. They never said no, and only after several weeks of the team working overtime—with a noticeable decline in morale and increase in mistakes—did Theodora realize she was asking too much. The reason she didn’t know exactly what the team was doing was because her tracking method was entirely verbal. She had told team members they should inform each other about when they had completed or were stalled on a task. Team members had different perceptions about work progress, and only Theodora’s constant checking prevented work from falling through the cracks.

The project was completed within 16 months and the target dollar range. The most significant requirements as identified by the focus group and senior managers were incorporated into the portal, but opinions about the portal's effectiveness from the broader employee population are still pending. The operation and future upgrade and repair of Grand Entry will be handled by the maintenance group, which, said the group, might prove challenging since the GE team's "agile process" had produced little documentation—so little that code for portions of the site was hard to comprehend. The GE team, following *The Agile Manifesto* creed of "value-added work over documentation" had done practically nothing to document its work or the system it had created.

QUESTIONS

1. Prepare a "lessons learned" section for the Grand Entry project closeout report. Consider at least the following points:
 - a. The customer and customer representation
 - b. Requirements definition and prioritization
 - c. Tasks selected for each sprint
 - d. Participation of Metasoft
 - e. Theodora's project planning and sprint planning
 - f. Sprint duration and overtime
 - g. Project tracking and control
 - h. Documentation
2. Describe how the case illustrates the benefits and pitfalls of agile. Which of the mistakes made resulted specifically from the agile method and which might have been made with a more traditional approach as well?

CASE 14.2 TECHNOLOGY TO TRACK STOLEN VEHICLES²⁹

Track & Found, Inc. (TFI), a leading vehicle tracking and recovery company, had grown significantly over the past 10 years due to high demand for its systems. The company was aware that its entire product offering was reliant on information and communication technology capabilities, so it restructured its IT department into a DevOps organization to promote the DevOps software development methodology. The restructuring split the IT department into two teams, one for development (50 employees) and one for operations (30 employees). The IT department's more experienced and better qualified programmers and engineers were placed in the development team; its less-qualified and newly-hired college grads were put in the operations team. Because the company was growing so quickly, the two teams had to be placed in different buildings located a considerable distance apart.

TFI executive management approached SoftTech Engineers, a reputable software engineering company with which it had had a longstanding relationship, to assist its own development team in developing a new tracking application called "TFIApp" to be used by vehicle response teams in conjunction with TFI's existing tracking solution to recover stolen vehicles. SoftTech was selected because it possessed development skills and knowledge that TFI did not, plus it had agile project management experience that

TFI management felt could benefit the project. TFI's development team had no such experience and expected SoftTech would provide guidance.

In the TFIApp project, SoftTech would create software for the GPS-based tracking application, the TFI team would create hardware to add functionality to the vehicle receiver unit, and SoftTech would integrate the hardware and software. SoftTech was capable of also developing hardware, but in the interest of the DevOps methodology TFI management gave the task to its own development team to encourage a close working relationship with the operations team.

SoftTech divided work for both hardware and software components of the system into a series of two-week sprints. Completion of each sprint would be followed by a demonstration to TFI management. In dividing up the work, the TFIApp project manager, who worked for SoftTech, made sure that all of his own team's work could be done independently of the development team's work (he had learned from prior experience that TFI's IT department could not be expected to stick to plans). The teams would not have to work together until integration at the end of the project.

The SoftTec team's work progressed smoothly and it completed the intended deliverables of every sprint for the software on time, on budget, and according to specifications. Unfortunately, however, the TFI development team completed its deliverables for the first sprint only. Thereafter it slipped behind because it needed input from the operations team, which, being understaffed, was in constant fire-fighting mode and had no time to meet with the development team. As one consequence of "splitting the work," the development team had received no training and little guidance in agile. And the fact that the team was not demonstrating deliverables every two weeks was apparently overlooked by the SoftTec project manager and TFI management. Neither had shown much interest in the development team until SoftTec completed its software application and was preparing for integration with the development team's hardware. Upon hearing from SoftTec that the hardware was not ready, TFI's IT executive raced to the development team and demanded an explanation.

QUESTIONS

1. What is DevOps? List characteristics of the DevOps software development methodology.
2. What might the SoftTec project manager have done differently to ensure a more successful outcome for the entire project?
3. What parts of the agile project went well and what parts of the project did not adhere to agile methodology?
4. What should TFI's management have done differently? What advice do you have for them to manage their IT resources better?

Notes

1. Fowler M. and Highsmith J. *The Agile Manifesto*, August 2001. See <http://www.pmp-projects.org/Agile-Manifesto.pdf>.
2. Nicholas J. *Lean Production for Competitive Advantage*, 2nd edn. Boca Raton, FL: CRC/ Productivity Press; 2018.

3. For basic principles, see Dennis P. *Lean Production Simplified*, 2nd edn. New York: Productivity Press; 2007. For applications to project management see Blackburn J. *Time-Based Competition*. Homewood, IL: Business One Irwin; 1991; Reinertsen, D. *Managing the Design Factory*, New York: Free Press; 1997; Leach L. *Lean Project Management: Eight Principles for Success*. Boise, ID: Advanced Projects; 2005.
4. For thorough coverage of APM and its variants see Wysocki R. *Effective Project Management: Traditional, Agile, Extreme*, 6th edn. New York: Wiley; 2012. Much of the following discussion is adapted from that book, particularly from pp. 44–47, 380–445.
5. The spiral model was introduced in Boehm B. A spiral model of software development and enhancement, *ACM SIGSOFT Software Engineering Notes* 11(4) August; 1986: 14–24.
6. The problem of managing hardware–software integration and simultaneous waterfall–spiral methodologies is addressed in Maier M. and Eberhardt R. *The Art of Systems Architecting*, 3rd edn. Boca Raton, FL: CRC Press; 2009, pp. 96–98.
7. Much of this section is adapted from Cohen M. *Succeeding with Agile: Software Development Using Scrum*. Upper Saddle River, NJ: Addison Wesley; 2010.
8. Cohen, *Succeeding with Agile*, pp. 117–140.
9. Cohen, *Succeeding with Agile*, p. 202.
10. Cohen, *Succeeding with Agile*, pp. 355–388. For discussion on scaling agile see Gower B. and Rally Software. *Agile Business: A Leader's Guide to Harnessing Complexity*. Boulder, CO: Rally Software; 2013.
11. See Schwaber K. and Beedle M. *Agile Software Development with Scrum*. Upper Saddle River, NJ: Pearson; 2001.
12. Lohnes P. and Wilson C. Can agile and traditional project management be partners? Integrating agile methods with project management best practices. <http://pppm.mclmg.com/docs/AgileVsCBPwhitePaperFinalB.pdf>. Accessed Sept. 10, 2014.
13. See Wils A., Van Baelen S., Holvoet T., and De Vlaminck K. *Agility in the Avionics Software World*. K.U. Leuven DistriNet, Department of Computer Science. Leuven, <https://distrinet.cs.kuleuven.be/legacy/publications/42148.pdf>. Accessed Sept. 17, 2014.
14. Lohnes and Wilson. Can agile and traditional project management be partners? [Most APM methods originated from and are used in software development projects, but one, Adaptive Project Framework, purportedly can be used in all kinds of projects; see Wysocki, *Effective Project Management*, pp. 408–437.]
15. From Denning, S. How agile can transform manufacturing: the case of Wikispeed. *Strategy & Leadership* 40(6); 2012: 22–28. Accessed 25 June 2019.
16. Nicholas, *Lean Production for Competitive Advantage*, p. 66.
17. Lean concepts applied to construction projects, see: Ballard G. The last planner. *Northern California Construction Institute Spring Conference*. Monterey, CA: Lean Construction Institute: April 1994; Koskela L., Howell G., Ballard G., and Tommelein I. Foundations of Lean Construction, in Best R. and de Valence G. (eds). *Design and Construction: Building in Value*. Oxford, UK: Butterworth-Heinemann; 2002.
18. Lean concepts applied to product development, see: Smith P. *Flexible Product Development: Agile Hardware Development to Liberate Innovation*, 2nd edn. Preston G. Smith; 2018; Reinertsen D. *Managing the Design Factory* and *The Principles of Product Development Flow: Second Generation Lean Product Development*. Redondo Beach, CA: Celebras; 2009. The following sections draw from the last, especially Chapters 5 and 7.
19. There are two kinds of batches: production and transfer. Production is the volume of work done in a stage without interruption; transfer is the volume of work moved from stage to stage. Our discussion of batches focuses solely on transfer batch size. For further discussion of production vs. transfer batch, see Nicholas, *Lean Production for Competitive Advantage*, pp. 123, 128.
20. Reinertsen, *The Principles of Product Development Flow*, p. 112.
21. Reinertsen, *The Principles of Product Development Flow*, p. 112.
22. Reinertsen, *The Principles of Product Development Flow*, p. 115.
23. Reinertsen, *The Principles of Product Development Flow*, pp. 111–120.
24. Reinertsen, *The Principles of Product Development Flow*, pp. 180–185.
25. Reinertsen, *The Principles of Product Development Flow*, p. 185.
26. Reinertsen, *The Principles of Product Development Flow*, pp. 155–156.

27. Greenleaf R. *Servant Leadership: A Journey into the Nature of Legitimate Power and Greatness*. Mahwah, NJ: Paulist Press; 2002; Martin S. A. To serve or not to serve: there is no question for a leader. Paper presented at PMI Global Congress 2015—EMEA, London, England. Newtown Square, PA: Project Management Institute; 2015. <https://www.pmi.org/learning/library/servant-leadership-cures-organization-9856>. Accessed April 4, 2024.
28. Adapted from Ries E. *The Lean Startup*. New York: Crown Business; 2011, pp. 138–140.
29. Bond-Barnard T., University of Pretoria.

Part IV

Organizational Behavior

15	Project Organizational Structure and Integration	537
16	Project Roles and Stakeholders	571
17	Leadership, Teamwork, and Conflict	601

Projects are organizations of individuals and groups created for the purpose of delivering results or end-items, and project success depends on how those organizations are structured and how well the people within them work together as teams.

The three chapters in this section focus on organizational and behavioral issues inherent to projects. They describe the ways that projects are organized and integrated, leadership styles of project managers, roles and responsibilities of project team members, and ways teams are managed to maximize teamwork and minimize the negative personal consequences of working in projects.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Chapter 15

Project Organizational Structure and Integration

Organizations are systems of human and physical elements created to achieve goals. As with all of systems, they are partly described by their structure—the form of relationships that bond their elements. In all organizations two kinds of structures coexist. One is the *formal structure*, the published one that describes normative superior–subordinate relationships, chains of command, and subdivisions and groupings of elements. The other is the *informal structure*, the unpublished one that consists of relationships that *evolve* through the interactions of people. Whereas the formal organization prescribes how people are supposed to relate, the informal organization is how they actually do relate. This chapter deals primarily with the formal organizational structure of projects and how project organizations are structured, depending on project goals and available resources.

The chapter also deals with project integration, which is the way that individual functional groups, subunits, project phases, and work tasks are interlinked and coordinated to achieve project goals. The discussion covers various means of project integration and the special case of integration within large-scale development projects.

Important to note is that projects are sometimes conducted without any formal project organization, per se. In other words, a manager and people are tasked with doing a project, but no project group or organization is formally recognized. Lack of a formalized project organization makes everything more difficult to manage because of uncertainty over reporting relationships and who, exactly, is on the project. Also noteworthy is that most project organizations are “superimposed” on the existing organizational structure; although this better enables them to achieve project goals, those in the formal organization sometimes view the project organization as disruptive to their business-as-usual.

15.1 Formal Organizational Structure

Concepts of organizational structure apply to all kinds of organizations—companies, institutions, agencies—and to their subunits—divisions, departments, projects, and teams. The **formal organization structure** is often publicized in a chart such as the one for NASA in Figure 15.1. A quick glance at it reveals both the organizational hierarchy and groupings for specialized tasks. The chart in Figure 15.1 shows, for example:

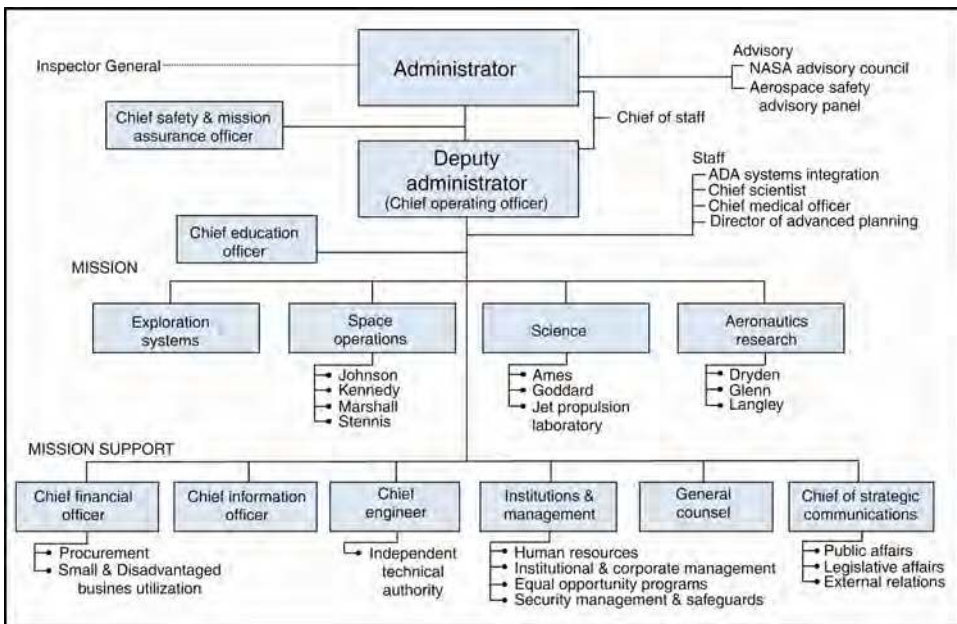


Figure 15.1
NASA organization and program chart.

Source: NASA.

1. The range of activities in which the organization is involved and the major subdivisions of the organization (exploration, space operations, science, aeronautics research).
2. The management hierarchy and reporting relationships (under “Mission,” e.g., directors at Ames, Goddard, and Jet Propulsion Laboratory all report to the administrator for science).
3. The type of work and responsibility of the subdivisions (e.g., research centers focus on specific disciplines or goals such as space exploration and space operations).
4. The official lines of authority and communication (the administrator is the highest authority, the deputy administrator is next highest, and so on; communication moves vertically and horizontally along the lines from one box to the next).

15.2 Organizational Design by Differentiation and Integration

There is no “best” kind of organization structure. The most appropriate structure depends on the organization’s goals, type of work, available resources, and environment. Organization structures typically develop through a combination of planned and evolutionary responses to ongoing problems. To deal with certain classes of situations and problems, organizations create specialized subdivisions and groupings, each with the necessary expertise and resources. As they grow or the environment changes, organizations add new subdivisions and groupings to handle new situations and emerging problems. Subdividing an organization like this into specialized areas is called **differentiation**.

Ordinarily, the subunits of an organization do not act independently but interact and support each other—at least in theory. The degree to which they interact and coordinate their actions to fulfill organizational goals is called **integration**.

Traditional forms of organization

There are six ways that organizations differentiate into subunits: functional, geographic, product, customer, process, and project. We will start by looking at the first five forms and then delve more deeply into the project form.

Functional differentiation

Functional differentiation is so-called because the organization is divided into functional subunits such as marketing, finance, production, and human resources; the structure of the Iron Butterfly Co. in Figure 15.2 is an example. Most of the integration between subunits is handled by rules, procedures, coordinated plans, and budgets. When problems occur that cannot be handled by these measures, the managers of the affected subunits work together to resolve them.

Functional differentiation works well in repetitive, stable environments because there is little change, and the rather low level of integration afforded by rules, procedures, and chain of command gets the job done. The functionally differentiated organization has a long history going back to the Roman army and the Catholic Church and remains today the most prevalent form of organization.

Geographic differentiation

Most organizations have more than one basis for differentiation. The Roman army was also geographically differentiated; that is, it was subdivided into legions that oversaw different locations. Organizations subdivide this way (e.g., Atlantic branch; European division; East Asian command; etc.) to adapt themselves to the unique requirements of local customers, markets, suppliers, adversaries, and so on. Within each geographic subunit, functional differentiation is often retained (e.g., marketing department of the Indian branch). Regional subunits may operate relatively autonomously and be integrated through standardized financial and reporting rules and procedures.

Product differentiation

Firms with a variety of product lines use product-based differentiation. Corporations such as General Motors, General Foods, and Taha are split into subdivisions wherein each one designs,

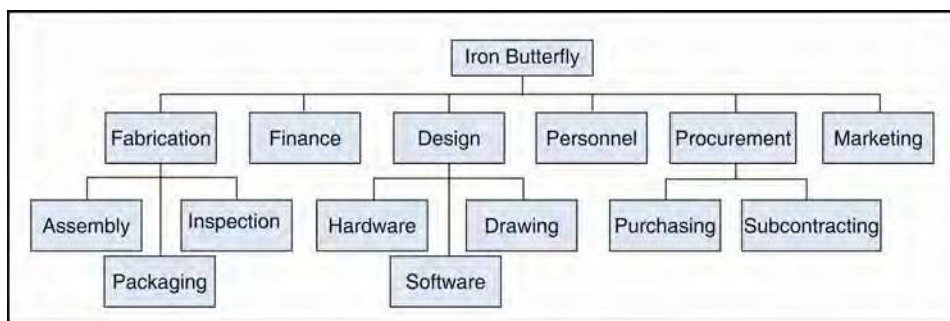


Figure 15.2

Functional differentiation in the organization structure for Iron Butterfly Company.

manufactures, and markets its own product line. Within each subdivision is a functional, geographic, or other form of breakdown. As with geographically differentiated organizations, integration between product subunits tends to be limited to standardized financial and reporting rules and procedures.

Customer differentiation

Organizations may also differentiate by customer type. For example, companies with both military and commercial sales often establish a separate military division because federal requirements for proposals, contracting, and product specifications differ substantially from those for commercial customers. The level of integration between divisions depends on the interdependency of the product lines.

Process differentiation

Organizations also differentiate according to process or sequence of steps. This is illustrated in Figure 15.2 for the Fabrication division of Iron Butterfly Co., which includes departments for assembly, inspection, and packaging, which are three steps in the fabrication process. These subunits require high-level integration because they are sequentially interrelated and problems in one unit directly affect the others. The subunits are integrated through coordinated plans, schedules, and task forces and teams, discussed later.

Drawbacks of traditional forms of organization

By their very design, traditional forms of organization can address only certain anticipated, classifiable kinds of problems. As the environment changes and new kinds of problems arise, they react by further differentiating subunits and adding more rules, procedures, and levels of management; in other words, they add more bureaucracy and pay the price of less flexibility and greater difficulty in integrating the subunits.

Traditional forms of organization work on the assumption that problems and tasks can be neatly classified and resolved within specialized units that can work somewhat independently. But what happens when a problem arises that does not fit any of the subunits? There may be no place for it, and such problems fall through the cracks.

The usual way to handle new, previously unanticipated or unclassifiable problems is to redesign the organization and add new subunits. However, the process of redesigning an organization to suit unique problems is slow and expensive. An alternative is to bump problems up the chain of command and involve managers of the functional units best suited to resolve them. This works as long as it is not done too often since it sometimes overwhelms the chain of command and encourages hiring more managers (i.e., and increasing the size of the bureaucracy). In short, traditional organization forms are not well-suited for situations with frequent change, unique problems, and high uncertainty. Nonetheless, most projects are conducted within or use resources provided by organizations with traditional forms of structure.

15.3 Requirements of Project Organizations

Each project is a new undertaking, somewhat unique, aimed at a new goal and, because of that, is inherently risky and uncertain. Changes, mistakes, or delays in one subunit have consequences for all others, and because of that the subunits must be able to work together—they

must be integrated. Project organizations are created around projects and, ideally, their structure and composition is whatever best suits the project. And like projects, they are temporary. When the project ends, the project organization is disbanded.

Projects in software development, pharmaceuticals, biomedicine, space exploration, product development, and even construction routinely encounter unexpected changes in goals, customer needs, environmental conditions, and resources. Consequently, project organizations must be adaptable; they must be both differentiated *and* integrated to accommodate a variety of problems and situations. To achieve this, project organizations share two properties:

- Subunits are differentiated to suit the unique requirements of the project and the environment.
- Subunits are integrated using horizontal relations.

These properties are discussed next.

15.4 Integration of Subunits in Projects¹

Traditional organizations are characterized by their “verticalness” or reliance upon up-and-down patterns of authority and communication. This makes them slow and ineffective in dealing with uncertainty or quickly changing situations. In contrast, project organizations are characterized by “horizontalness,” which means cutting across formal lines of authority and moving decision-making to the level of the parties affected.

Horizontalness helps expedite communication and resolve problems between units through informal relationships. For example, whenever the assembly department in Figure 15.2 experiences a minor parts shortage, George, the assembly supervisor, phones Helen in the purchasing department for a “rush order” favor. The informal call bypasses the formal structure (involving George’s and Helen’s respective managers) and speeds up the order.

A drawback with informal processes is that they do not ensure everyone who should be involved is, and they do not guarantee that people will get the information they need.

Project organizations improve upon informal processes by building horizontalness into the formal organization structure through the use of functions called **integrators**. Integrators are people or groups who facilitate communication between the subunits working on a common task. Integrators bypass the traditional lines of authority and speed up communication, but they also ensure that everyone affected by a problem is involved and has the necessary information.

Several kinds of integrators are used in projects. They are listed below in order of increasing authority, importance, and cost; in the list, the latter kinds take on all the authority and responsibility of the former:²

- Liaison roles
- Task forces and teams
- Project expeditors and coordinators
- Pure project managers
- Matrix managers
- Integrating contractors.

15.5 Liaison Roles, Task Forces, and Teams

The **liaison** role is a specialized person or group that links two or more organization subunits. In Figure 15.3 the dotted line represents the liaison role of “inventory controller.” The role

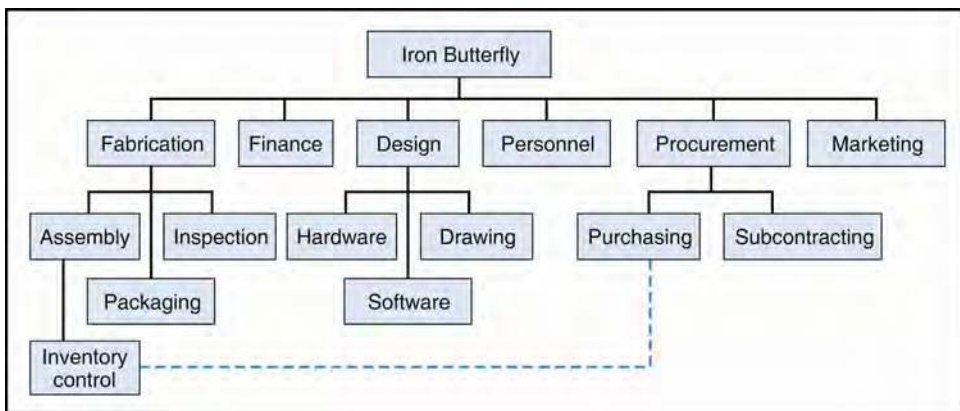


Figure 15.3

Inventory control liaison role linking Assembly and Purchasing Departments.

serves as a bridge connecting the assembly and purchasing departments, communicating, and coordinating efforts to ensure smooth operations. But the liaison role is not always effective. Although the liaison connects the two departments, it is not involved in the decision-making of either group. This is where the next kind of integrative role, a multifunctional **task force** or **team**, comes into play.

A **task force** is a temporary group with representatives from several subunits (multifunctional) that are formed to address a need or solve a problem. When the group begins addressing the problem, they are, in fact, conducting a project. The task force meets as often and as long as necessary to solve the problem, and then it disbands. The most effective task forces have ten or fewer members, a team leader or coordinator, and are short-lived.³

Both the team leader and members are selected by whoever initiated or sponsored the project. Team leaders are responsible for expediting and coordinating the efforts of the team, although usually they have little formal authority over team members, who must divide their time between the task force and their “usual” work. Task forces undertake an unlimited variety of projects and special assignments, including:

- Company reorganizations
- Mergers, acquisitions, or divestitures
- Special studies, surveys, or evaluations
- Major audits
- Efficiency, modernization, and cost reduction efforts
- Geographic or marketing expansions
- Facility relocations or changes in facility layout
- Management and organization development programs
- New equipment or procedures installation.

Ideally the task force members have access to information relevant to the project plus authority to make commitments for their functional areas. Lacking information, the group cannot make good decisions; lacking authority, the group will not be able to act on its decisions.

For problems that are novel but need continuous attention, **permanent teams** are formed. These teams have the same characteristics as task forces except that they convene periodically

on a regular basis, indefinitely. For example, if Iron Butterfly Company makes products that require design changes throughout the year in response to changing markets and competition, then representatives from design, fabrication, procurement, and other areas will need to meet face-to-face on a regular basis to make decisions regarding these changes. Members work on the team either part-time or full-time.

Sometimes it is difficult to find people with the requisite knowledge, authority, and inclination to serve on multifunctional tasks forces and teams. People develop attitudes and goals oriented toward their specialization, and although this helps them be effective in their own areas of work, it delimits their ability to work with people from other areas. For multifunctional projects, the teambuilding methods described in Chapter 17 help break down barriers and forge bonds between project team members.



See Chapter 17

15.6 Project Expeditors and Coordinators

The simplest kind of project organization is a single, small group of people, a task force or team formed on a full- or part-time basis to perform an assignment. Such a group can exist inside one functional area or span across multiple functional areas.

Projects within one functional area

It makes sense that a project that affects or requires expertise from only one functional area should be located in that area. For example, a project to survey customer attitudes about a new product would ordinarily be placed entirely within the marketing department, as indicated in Figure 15.4 because all the necessary resources and expertise are there. The team does everything—prepare the survey instrument, obtain mailing lists, distribute the survey, and process the results. A project team like this is managed by a **project expeditor**,⁴ someone selected by the manager of the area wherein the project lies. The expeditor coordinates decisions, creates and monitors schedules, and keeps the project moving and the manager updated about progress. The expeditor, however, typically has no authority over team members and so must rely on persuasion, personal knowledge, and information about the project to influence team members.

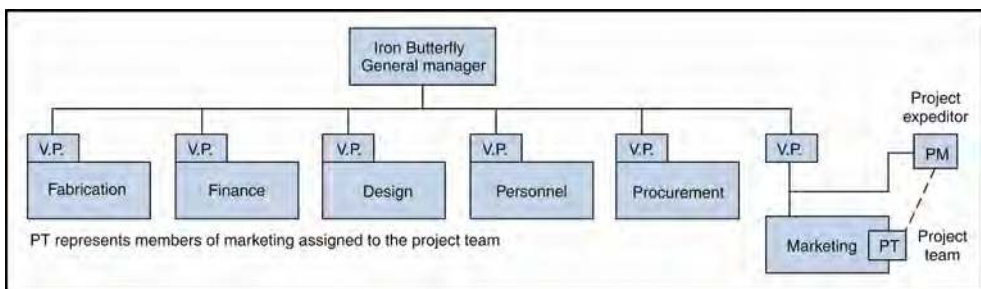


Figure 15.4

Project expeditor within a single functional area, the marketing department.

Multifunctional project teams

An example of a project that might use a **multifunctional team** is one to develop an enterprise resource planning (ERP) system, which is a companywide system that connects information about forecasting, order entry, purchasing, and inventory. The team would include representatives from all the departments that must provide inputs to the system or would utilize its outputs, such as accounting, inventory control, purchasing, manufacturing, engineering, and IT. Representatives from suppliers and customers might also be on the team. The team is responsible for defining the system requirements and overseeing the development and installation of the system.

Multifunctional project teams are not associated with any one particular functional area (they are multifunctional), and they report to a higher-level manager as shown in Figure 15.5, which imputes greater importance to the project. The person managing such a project is designated the **project coordinator**. The coordinator has no line authority over team members but does have authority to make and execute decisions about project budgets, schedules, and work performance. The coordinator's influence originates in his reporting to a high-level general manager and, like that of the expeditor, his knowledge of and central position in the project.

Multifunctional teams are commonplace in product development. By forming closely knit teams, phases of the systems development cycle can be done simultaneously instead of sequentially. This approach, called **concurrent engineering**, eliminates cross-functional barriers and can result in higher quality and lower cost. Concurrent engineering is discussed later in this chapter.

15.7 Pure Project Organizations

Projects that involve much complexity, major resource commitments, and high stakes require a **pure project** or *projectized* form of organization. A pure project is a separate organization, sometimes its own company, created especially for and singularly devoted to achievement of the project goal. Whatever is needed to accomplish project goals—all necessary human and

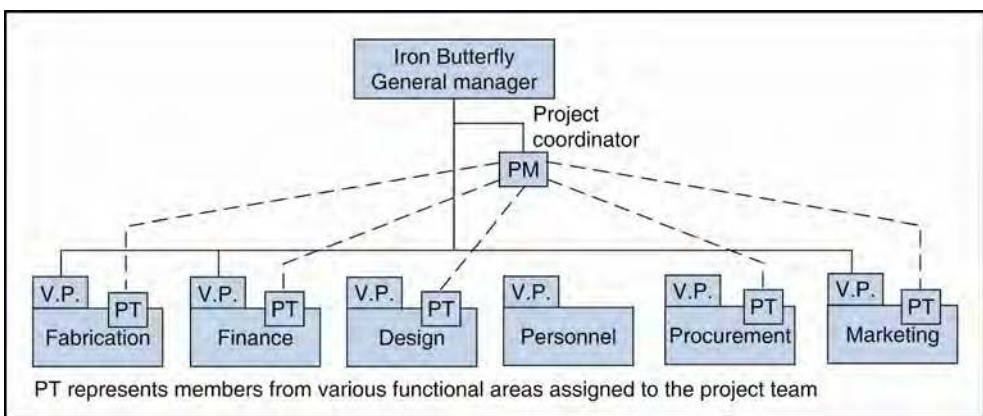


Figure 15.5

Multifunctional project team with project coordinator.

physical resources—is incorporated into the pure project organization. Often, within the pure project organization are liaisons, task forces, and teams.

Heading the pure object organization is the **pure project manager**, a manager who has formal authority over all human and physical resources assigned to the project. The project manager can bring in resources from internal functional areas as well as contract out with external subcontractors and suppliers. The pure project manager engages in the project from start to finish: during proposal preparation she requests and reconciles plans from functional areas and prepares preliminary budget and schedule estimates; after project acceptance, she hires personnel; during project execution, she allocates resources and approves changes to requirements and the project plan. When personnel must be “borrowed” from functional areas, she negotiates to obtain them.

When external resources are required, the project manager heads selection of and negotiations with contractors. She oversees and coordinates their work with other areas of the project. The project managers in the disaster recovery and NASA examples in Chapter 1 are pure project managers.

Besides providing for project focus and control, the potential advantages of pure project organization include higher team spirit, speedier execution, fast decision making, flexibility, and autonomy (less subject to bureaucracy).



See Chapter 1

Pure project variations

Three common variations of the pure project structure are the project center, the partial project, and the stand-alone project.

In the **project center**, the structure of the parent organization remains the same except for the addition of a separate “project arm” and project manager. This form is shown in Figure 15.6 for the Iron Butterfly Company, which has two pure-project arms, LOGON and SPECTOR. Resources and personnel are borrowed from functional and staff areas to work in the project center for as long as needed. General Motors used a project center when it chose 1,200 key people from various divisions for the task of downsizing vehicles in all of its automotive lines. The project center developed suggestions, turned them over to the automotive divisions for implementation, and then disbanded. In another corporation, a project center oversaw the relocation of the corporate offices. The center worked full-time to address the tricky problems of relocation while the rest of the organization continued to work as usual.

In a **partial project**, the functions critical to the project (such as construction or engineering) are assigned to the project manager while other, support-oriented functions (such as procurement and accounting) remain within functional areas of the parent organization. The manager of a partial project directly controls all major project tasks but receives assistance from functional areas in the parent company providing support. In Figure 15.6, for example, the LOGON project manager might fully control design and procurement but rely on the functional areas of finance, human resources, and marketing for support.

The **stand-alone project** is an entire organization created especially for the purpose of accomplishing the project. It is typically used for large-scale government, public works, or development and installation projects that involve one or more prime contractors, dozens of subcontractors, and numerous supporting organizations, suppliers, and consultants. It is typical of megaprojects such as the International Space Station development program, Quebec’s La Grande hydroelectric complex, the Channel Tunnel, China’s Three Gorges Dam, and Boston’s Big Dig. When the project is completed, all that might remain of the organization is a function to operate the system; the rest is dissolved. Stand-alone projects are discussed later in this chapter.

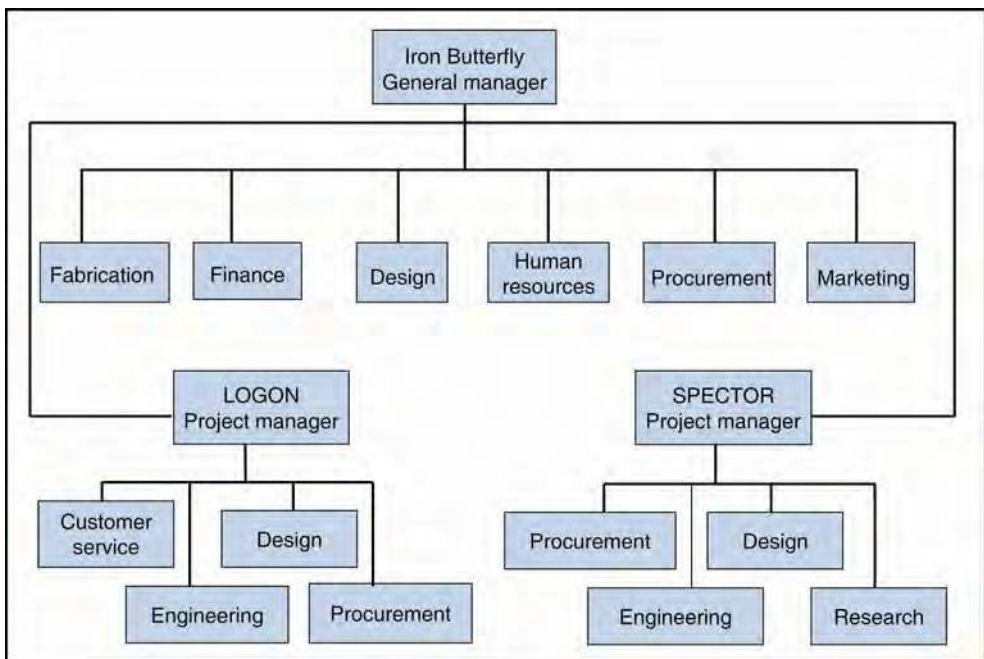


Figure 15.6

Pure projects: the LOGON and SPECTOR projects as “arms” to the functional organization.

Disadvantages

The chief disadvantage of pure project organization is its cost. Because each pure project is a completely or partially independent organization, it must be completely or substantially staffed. Each project becomes a self-contained empire, and there is often little sharing or cross-utilization of resources with other projects. Companies conducting multiple pure projects may incur considerable duplication of effort and facilities, and high overhead costs.

To ensure that resources will be available when needed, pure project organizations must begin acquiring them in advance of the project. One of the authors was among numerous engineers hired in anticipation of a large government contract to ensure the project could begin as soon as the contract was signed. But the contract was never signed, and everyone had to be transferred elsewhere or laid off. The payroll loss amounted to many hundreds of person-months.

In most organizations, the functional manager is the driving force behind workers developing technical competencies. However, when there are no functional managers to encourage workers to expand their capabilities, the workers become so specialized that after the project they cannot be placed in projects requiring more generalized or up-to-date skills. This leads to still another cost: outplacement. Whenever there is no follow-up work, the pure project organization faces the problem of what to do with its workforce when the project ends.

As the project draws to a close, uncertainty about the fate of the team grows, and morale and enthusiasm decline. A project manager may become so preoccupied with generating new contracts or finding a job for himself and his team that he becomes neglectful of his closeout responsibilities for the current project.

15.8 Matrix Organizations

The pure project form often provides the only way to do large-scale, one-time projects, however its disadvantages make it impractical for industries that continually operate on a project basis. Such industries include:

- Architecture and construction, where every building, bridge, or highway is a project.
- Product development, where every product design, manufacture, and promotion is a project.
- IT, where every hardware and software installation is a project.
- Law and accounting, where every litigation and audit is a project.
- Aerospace, where every new aircraft and space system is a project.

Most of these projects are too large, too complex, and have too much at stake to be handled by task forces. In addition, businesses in these industries engage in many projects at a time—they are **multi-project** organizations—and they need the capability to create large project teams quickly without the personnel and cost disadvantages associated with pure project organizations. To achieve that capability, the **matrix** form of organization evolved. First adopted in the aerospace industry, the matrix form, illustrated in Figure 15.7, is a grid-like structure of authority and reporting relationships created by the overlay of a project organization onto a traditional, functional organization.⁵ This overlay gives the matrix four unique capabilities.

First, the functional part provides the pool of technical expertise and physical resources needed by projects. The number of required resources is determined by the project manager, who negotiates with functional managers about who and how many resources to borrow.

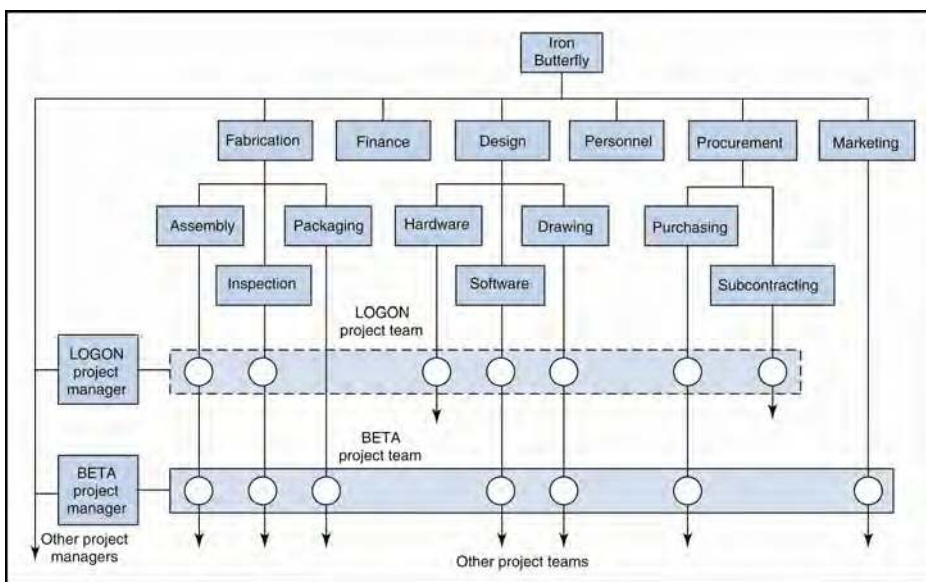


Figure 15.7

The LOGON and BETA project teams in a strong matrix form of project organization.

Second, while in their “functional homes” workers associate with colleagues in their fields of specialization, which makes them more assignable to new projects. Each functional area has at a given time many individuals working on different projects, sharing ideas, and exchanging points of view.

Third, when their assignments are fulfilled or their projects are completed, workers return to their functional areas for new assignments. This eliminates anxiety and reduces fluctuations in workforce levels and worker morale.

Finally, while managers of functional areas provide resources and technical support to each project, one person, the project manager (or **matrix manager**) oversees the resources and unifies and integrates their efforts to achieve project goals.

The matrix structure has the virtue of dedicated project resources and a dedicated manager to direct the project, but the project manager’s range of authority within the matrix can vary considerably. The Project Management Institute distinguishes matrix organizations as “strong,” “weak,” or “balanced.” In a strong matrix, project managers have substantial authority; they control project funds and other resources, and devote most or all of their time to managing each project. In a weak matrix, project managers are actually coordinators or expeditors who, as explained before, are not fully-fledged project managers and must fit the role into their other, usually non-project work. They coordinate project work performed by the contributing functional areas, but have little authority, no budget responsibility, and no ability to command resources on their own; project work within the functions is overseen by the functional managers. In the balanced matrix, the managers are fully-fledged project managers, but their level of authority and control over budgets and resources is less than in a strong matrix and is shared with functional managers.

In a strong matrix, prioritizing and balancing the resources shared by the different projects is the responsibility of the **manager of projects**, the **PMO director**, or in Figure 15.8 the vice

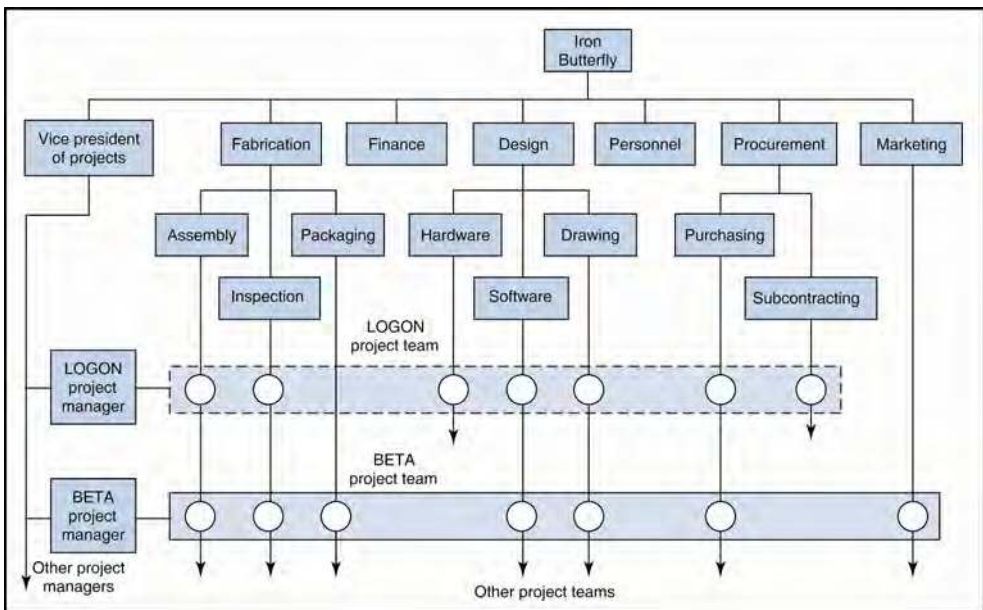


Figure 15.8

Location of the vice president of projects in a matrix organization.

president of projects. The manager of projects attempts to meet the requirements of current and upcoming projects, resolves resource conflicts between projects, and relieves top management of project operations responsibility. The PMO is discussed later.

Problems with matrix organizations

The main benefit of the matrix organization—its combined vertical–horizontal structure—is also the main source of its problems.⁶ The matrix is not just a structure but a whole different way of doing things. Most organizations are accustomed to hierarchical decision making and vertical information flow, but the matrix utilizes horizontal relations, lateral information flow, and decentralized decision making. It superimposes a horizontal-team system on a vertical-functional system, and companies that adopt the structure must add horizontal information processing systems to existing vertical accounting and command systems.

In human terms, the drawback of the matrix is that it induces conflict. Theoretically, the matrix promotes coordinated decision making among functional areas and enables tradeoff decision-making for the benefit of the projects. The matrix assumes a balance of power between functional and project managers. But functional managers control project resources, so conflicts arise over which projects get priority and which project managers get the best resources.

The matrix structure attempts to be the best of both worlds—functional and projectized, but it falls short. Its main problem is rooted in something few people like to admit—fear and power: functional managers fear that project managers might take control of “their” resources and reduce their roles to mere “support/staff” functions. They become even more worried when the vice president of projects controls project funding and threatens to outsource work normally provided by their areas. Project managers get frustrated too, because, unlike functional managers, they have little or no control over workers because they have no say over worker incentives such as promotions, salaries, and bonuses.

There are no easy solutions to these problems, but as a start everyone must understand their roles: the project manager should have a say over *what* must be done, and the functional managers should have a say over *how* it must be done (and, to considerable extent, *who* within the function should do it).

Here is another problem: each project worker in the matrix has at least two bosses, a functional manager and a project-matrix manager; this violates a major principle of management: single chain of command. The project manager directs the worker on a project, but the functional manager evaluates the worker’s performance. The inevitable result is role conflict and confusion: to whom should the worker give allegiance, the project manager or the functional manager?

To avoid conflict in the matrix, everyone—managers and workers—must be clear about their roles, and the organization must establish clear rules specifying, for example, days of the week that individuals are to work in a project team and days they are to work in their functional areas.⁷

The matrix can lead to still other dilemmas, explained below.

Example 15.1: Two-hat Problem

The matrix structure requires a lot of managers, yet in many organizations managers are scarce. One solution is for managers to wear two hats—one as project manager, the other as functional manager. While wearing the functional hat, the manager allocates resources

to different projects; the problem is, while wearing the project hat it is hard to convince other project managers that she has not grabbed the best resources for the projects that *she* is managing. Also, to people from her department who are working on her projects, the “project hat” is invisible. All they see is that “functional hat,” ever mindful of the fact that she controls their wages and promotions; as a result, they *always* prioritize her projects over others they are working on.

Any attempt to adopt a matrix structure must be accompanied by both attitudinal and cultural changes, which are difficult to achieve. In many companies, conflicts over priorities and resource allocation are eliminated or reduced by the PMO, which sets the priorities and assigns resources. Even with a PMO, however, anxiety and conflict remain common maladies of the matrix structure.

15.9 Selecting an Organizational Form for Projects

Project managers are seldom involved in designing the organizational structures of the projects they lead, yet they can offer suggestions to the managers who do. It is impossible to state which organization form is always best, but general criteria help specify which form is most appropriate for a given project. Figure 15.9 shows the approximate applicability of different project organizational forms based upon four criteria:

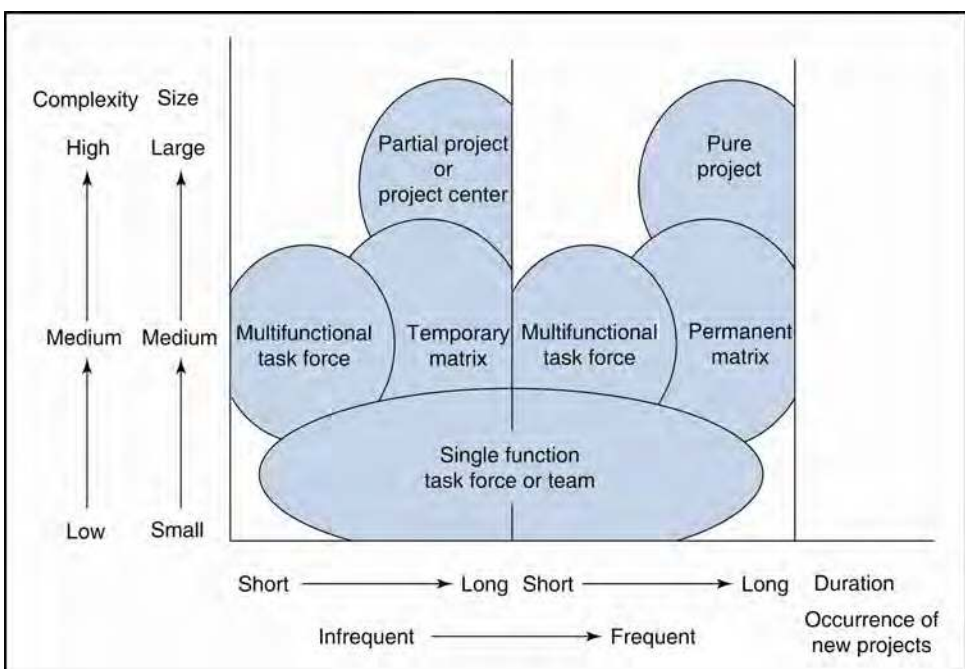


Figure 15.9

Criteria for selecting the appropriate project organizational form.

- Number of projects underway at any given time.
- Duration of projects.
- Size of projects in terms of human, capital, or other resources.
- Complexity of relationships among projects and with functional areas.

Matrix and pure project forms are applicable for projects of medium and higher complexity and of medium or larger size, and in companies that are always doing projects. In particular, the matrix works best where a variety of different projects are being done concurrently and all can share functional resources on a part-time basis. In contrast, when there is less variety between projects, when specialists must be devoted full-time, and when high-level project authority is desired, then the pure project form is better. Both forms are applicable where projects are the organization's "way of life."

For small, short-term projects that involve multiple functional areas, task forces and cross-functional teams are appropriate. Projects of longer duration, but small in scope, are best handled by full-time project teams with project coordinators. For large, complex projects with large teams, a temporary matrix or partial project is appropriate.

In selecting a project form, consider the relative importance of the following criteria: the stake of the project, the degree of technological uncertainty, the criticalness of time and cost goals, and the uniqueness of the project.⁸ For example, task forces and teams are generally appropriate when the project task is highly certain and involves little risk, and when time and cost are not major factors. When risk and uncertainty are great, when time and cost goals are critical, or when much is at stake, matrix and pure project forms offer better integration and control. When a project differs greatly from the normal business of the firm, it should use a partial or full pure project form.

Top management's attitudes about the appropriate level of responsibility and authority for the project manager also matter. The most important factor is the company's experience with projects and management's perception of which project forms work best. Firms with little project experience avoid the matrix because it is difficult to adopt. Faced with a complex project, they adopt a partial or project center approach.

Most organizations engage in a variety of projects and use different project forms, whatever best suits each project. In a given organization much or most of the work might be done by matrix teams, but a few high-visibility projects will be set up as pure projects. Meantime, within functional departments, innumerable small, single-function projects are being conducted, and scattered elsewhere are project task forces. Within a given project, a composite structure might be created, i.e., a structure that combines features of functional, matrix, and pure-project forms, depending on the scope and kinds of work in the project. At Microsoft Corporation, for example, the organization structure of development projects mirrors the products they produce.

Example 15.2: Product Development Organization at Microsoft⁹

A software product-development project at Microsoft might involve 300 to 400 people, including specialists in product specification, development, testing, user education, and planning. Program managers and developers divide the product into "features," where each feature is a relatively independent building block that will be apparent to customers (e.g. printing, adding a column of numbers, or interfacing with a particular brand of hardware). They then

divide the project organization into development teams where each one concentrates on one or some of these features. In essence, the project is divided into small, feature-driven projects that mirror the structure of the overall product. Through this feature-driven organization, product functionality can be increased simply by adding more development teams: the more features desired in the product, the more teams assigned to the project.

Each team consists of three to eight developers, one of whom is the “lead.” The lead reports to the project’s development manager, who has a broad view of the product and interconnections among its features. A recent version of Excel had ten feature teams: eight worked on the basic Excel product, one on a graph product, and one on a query tool product. Paired with each feature development team are parallel teams responsible for feature testing and user education.

Each feature team has considerable autonomy, though it must follow rules so its work stays coordinated with the other teams. Each team is expected to “build” and have checked a certain amount of code each day. This forces the teams to synchronize their work at the pace of the overall project.

Microsoft’s philosophy for organizing projects is that a product tends to mirror the organization that created it. A big, slow organization will create a big, slow software product. A small, nimble group in which everyone gets along well will produce pieces of code that work together well, which is why Microsoft uses small, flexible teams.

15.10 Project Office and PMO

The term **project office** has dual meaning: it can refer to a support staff group that assists in project management and reports to the project manager, or it can be a physical place where the project team meets. Our discussion here will focus on the **project support staff**.

The purpose of the project office is to *coordinate* work efforts and advise the different functional areas and subcontractors on *what* they should do in the project (but not *how* they should do it). The office is responsible for planning, directing, and controlling project activities and for linking the project teams, users, and top management. When a project is small and procedures are well established, the office might consist of just one person, the project manager. When the office must coordinate multiple projects, the staff is larger and comprises what is called the *office of projects* or, more commonly, the **project management office** or **PMO**. The PMO is a *support office* that develops project management policy and methodology, offers training, and provides services to project managers, as described in



See Chapter 18 Chapter 18.

Functions of the project office

The functions and composition of the project office depend on the authority of the project manager and the size, importance, and goal of the project. The project office shown in Figure 15.10 is for a large-scale engineering development project. Among the functions shown is planning and control. During the conception and definition phases this function prepares the WBS, schedules, and budgets. During execution it monitors work, forecasts trends, updates schedules and budgets, and distributes reports to functional, upper-level, and customer management.

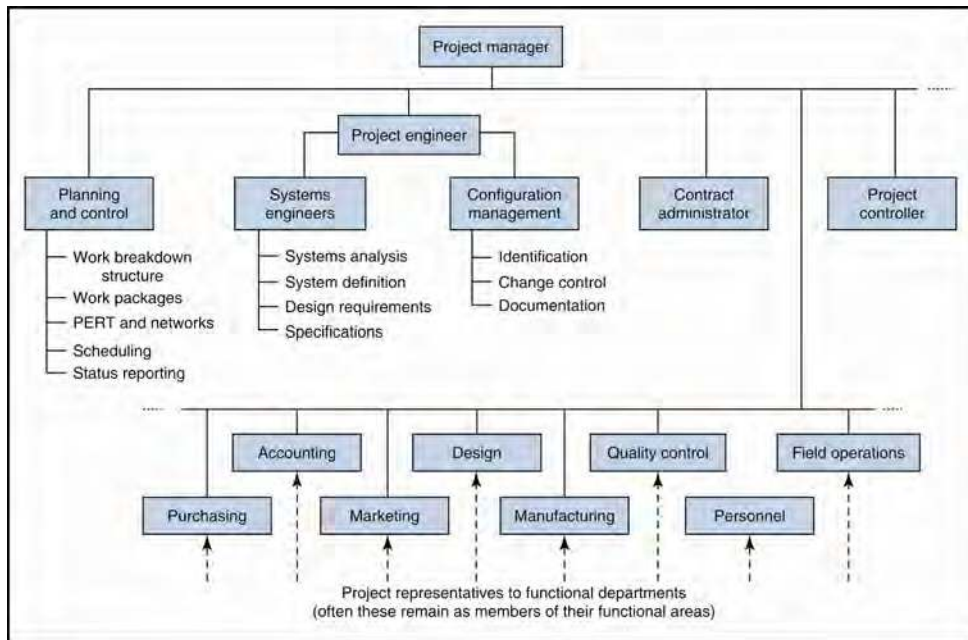


Figure 15.10
Project office for a large technical development project.

Also shown are functions for systems engineering and configuration management, both headed by the project engineer. The systems engineering function oversees systems analysis, requirements definition, and end-item specifications (discussed in Appendix A to Chapter 4), and furnishes inputs for planning and control. As shown, the project office also manages contracting and financial control.

Integrating the functions within a project is achieved by structuring the project office to mirror the functional areas the office must integrate.¹⁰ This happens by including in the office a representative from each functional area working in the project (in Figure 15.10, purchasing, accounting, etc.). Although each representative is a specialist in a functional discipline, while in the project office his primary role is to integrate that discipline with the others. As a result, through *coordinating and integrating the functional representatives* in the project office, the work of the functional areas in the project is coordinated and integrated. Usually, members of the project staff are co-located in the same physical office where they can intermingle face-to-face. In smaller projects the size of the project office can be minimized by allowing some or most of the functional representatives to remain in their functional areas.

Office of projects, PMO, and program office

Multiproject organizations also have an **office of projects** (not to be confused with the project office), PMO, or program office. This was shown in Figure 15.8 as the “vice president of projects.” In pure project organizations, the office is located at a level between senior management and project managers (in Figure 15.6, it would be located below the general manager and on the line between the LOGON and SPECTOR projects). When projects are small, the office of



- See Appendix 4



See Chapters
1 and 18

projects substitutes for individual project offices and oversees proposals, contracting, scheduling, cost control, and report preparation for every project. When projects are large or overlap, the office of projects or PMO is used in *addition* to project offices and coordinates the combined requirements of all the projects.¹¹

When projects are part of a program, a **program office** is set up to ensure that the projects supplement one another and “add up” to overall program goals. The program office (discussed in Chapter 18) handles interfaces and integration between projects and with external resources for each project, maintains customer enthusiasm and support, and keeps project managers informed of potential problems. The NASA program office described in Chapter 1 is an example. When programs are very large, the integration work of the program office is supplemented by outside “integration contractors,” discussed next.

15.11 Integration of Large-scale Projects

In a large-scale project (LSP) or **megaproject**, numerous parties all contribute to one effort. Figure 15.11 shows the principal contributors and relationships in an LSP. Relationships are complex and lines of authority connecting the parties are often weak, sometimes based

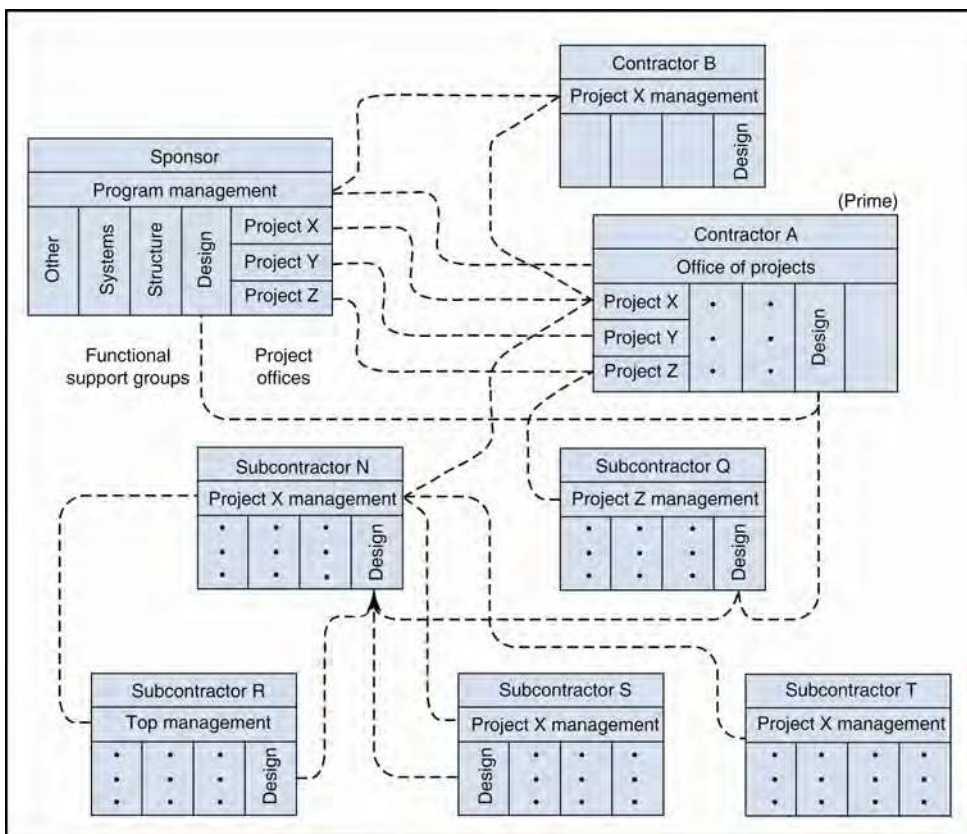


Figure 15.11
Integration relationships in a large-scale project.

entirely on contracts and purchase orders. If Figure 15.11 appears confusing, well, that simply reflects the fact that relationships in LSPs can be confusing. Examples of LSPs include space systems (e.g., the International Space Station), construction projects (India's Navi-Mumbai International Airport), company relocations (involving client, movers, and suppliers), and corporate mergers (dual sets of clients, consultants, and attorneys).

Notice in Figure 15.11 the direct relationships, both horizontal and hierarchical, among different contributors' managers as well as between their functional areas. Such relationships between, for instance, design groups from the sponsor and its contractors and subcontractors help speed up decision making and tighten integration. The relationships between contributors are facilitated by project managers, coordinators, expeditors, liaisons, and task forces.

Most LSPs are devoted to development and/or construction of complex systems. The total effort is subdivided among several contributors, each responsible for a specific subsystem or component that must be integrated with others to form the overall system. Figure 15.12, for example, shows the major components in the International Space Station. The figure is simplified and does not show major elements of the program such as launch vehicles to place the components into Earth orbit, ground support systems, and the numerous organizations that work to develop, produce, integrate, and operate the components (prime contractors, subcontractors, and suppliers).

Oversight and integration contractors

In public works and government projects, integration is usually the responsibility of the sponsoring agency. Sometimes, however, the engineering and management tasks are quite difficult or extensive, and outside help is required.

Among the first LSPs to experience the integration problems inherent to large systems were weapons system development projects during World War II.¹² For instance, separate offices within the Army Air Corps purchased the components that made up a bomber aircraft, and these components—airframe, engines, and electronics—were then furnished to and assembled by the aircraft manufacturer. As systems grew more complex, this approach no longer worked. Sometimes the component interfaces (connections) were different, so plugs and fasteners would not fit together, or the sizes of the components were different than planned and the entire system had to be redesigned to accommodate them. To overcome these difficulties, the military formed technical groups to coordinate component interfaces, but this resulted in massive red tape, as explained by Livingston:¹³

A contractor wanted to change the clock mechanism in an airplane cockpit from one-day to eight-days. It submitted a written request to the military representative, who forwarded it to the military tech group. The tech group reviewed the request and asked the contractor for more details. The contractor complied, after which the tech group approved the request and forwarded it to the change committee. The change committee reviewed and accepted the request, and sent written authorization to the representative, who notified the contractor. This simple request took *three months* to approve.

Today the process is expedited by giving integration responsibility to a single “oversight” body, usually the **lead** or **prime contractor**. The other contractors in the project (**associate contractors**) become subcontractors to and take orders from the prime contractor. The project sponsor is still responsible for contracting with associate contractors, and for making major decisions, and resolving conflicts between the prime and associates. Figure 15.13 shows the relationships among the sponsor, prime, and subs for an urban transit project.¹⁴

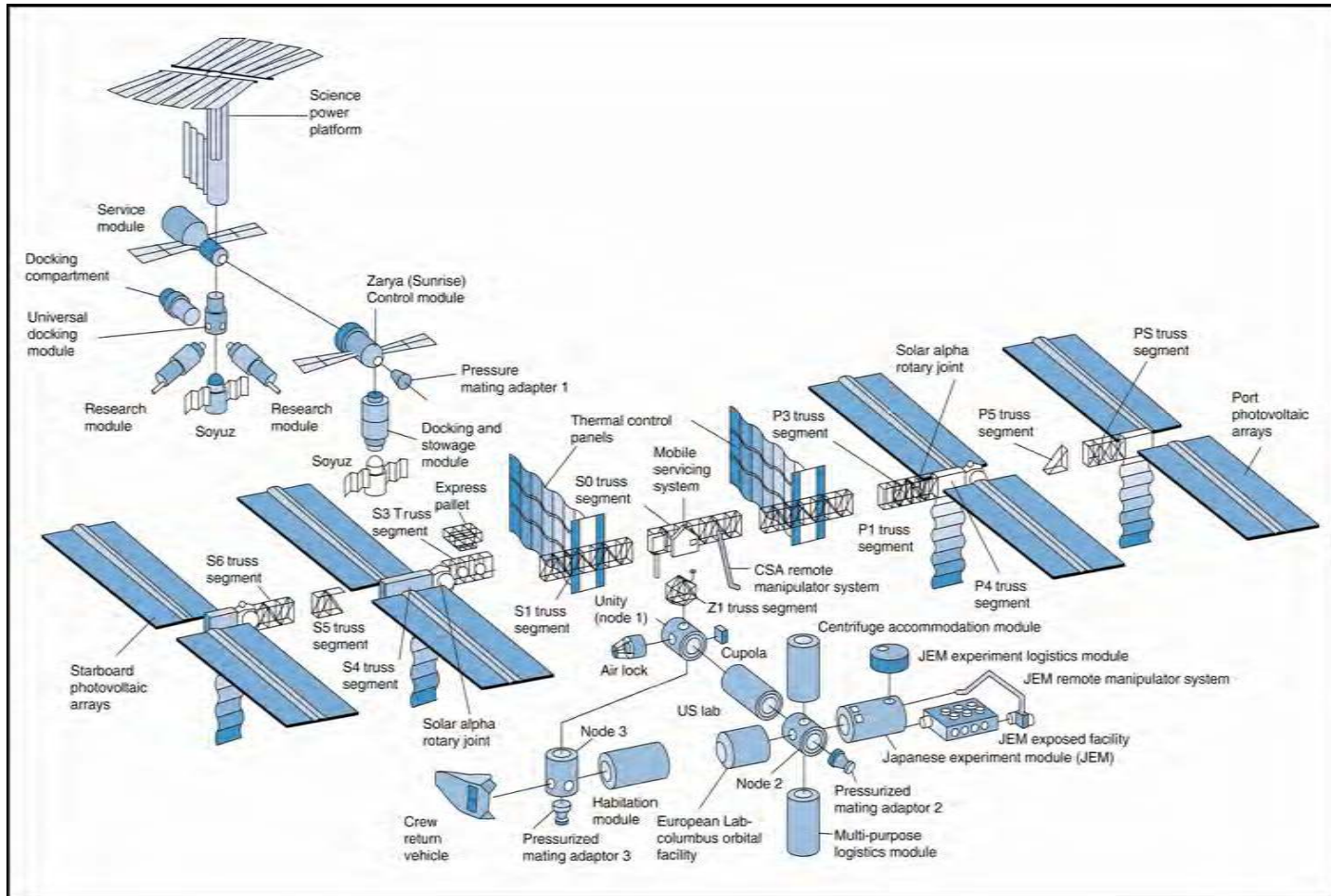


Figure 15.12

Major components in the hardware and assembly of the International Space Station.

Source: NASA.

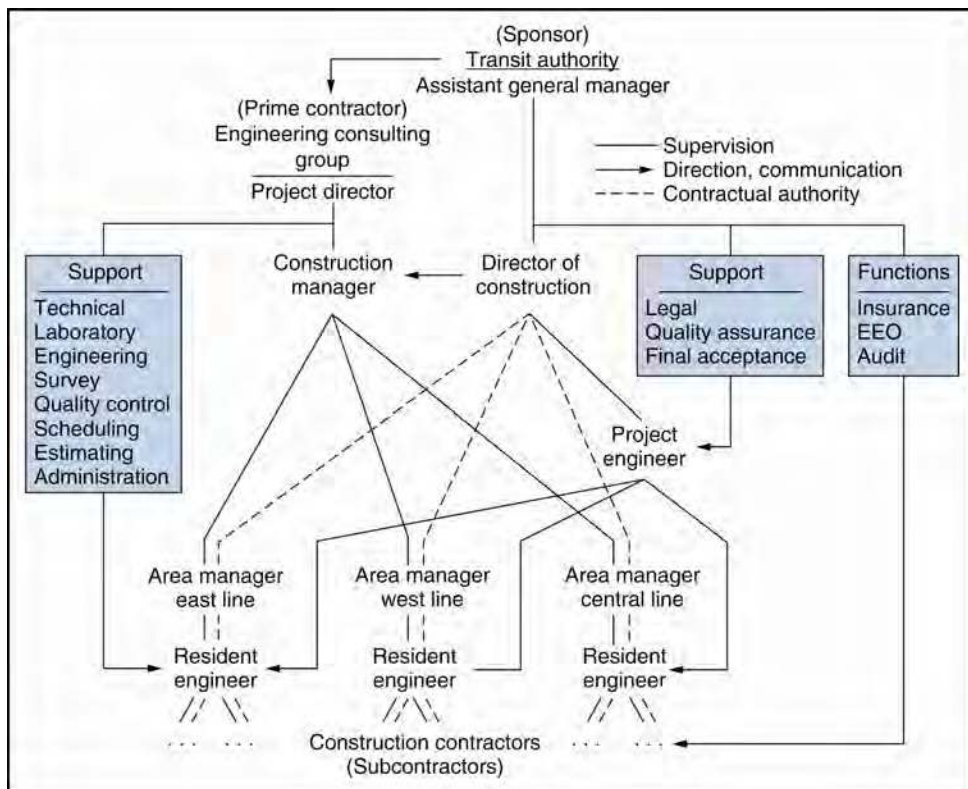


Figure 15.13
Management and authority relationships in a large construction project.

Sometimes the prime contractor is conferred even greater responsibility, such as assisting the sponsor in selecting associates, pricing of subsystems, and allocating project funds. This situation poses a problem when the prime and the associates are competitors since, understandably, associates are hesitant to divulge innovative design concepts or subsystem details, even though the prime needs to know those things to integrate the subsystems.

Sometimes even the largest prime contractors are not big enough. At such times they form teams called **joint ventures** and submit joint proposals. This one organization serves as leader and takes on responsibility for systems engineering and management of all the others. This appeals to small- and medium-sized firms that ordinarily would not have the resources to contract independently. With this approach, however, unless the lead company is strong there could still be interface problems.

When the prime contractor lacks the capability to perform all the integrating work, a separate consulting firm or **integration contractor** is engaged entirely to provide integration and engineering advice.¹⁵ These contractors, which sometimes employ thousands of workers, are able to quickly pull together all the necessary resources. The problem is that they often operate in the same business as the contractors they are responsible for integrating, which puts them in the awkward position of managing their competitors and being able to learn their secrets. System integration in megaprojects is discussed in Chapter 20.



Example 15.3: Corporate Merger—Large Scale Non-Technical Project¹⁶

Special integration management is necessary not only for technical projects but any *large, complex* project. When one of the US's largest pharmaceutical firms acquired one of Europe's largest pharmaceutical firms for \$6.9 billion, an acquisition that involved 10,000 people, 18 manufacturing locations, and 30 international affiliates, it engaged a well-known global consulting firm to oversee the integration effort. The consulting firm established a program management office and a global acquisition integration management (AIM) team with 18 full-time director-level individuals from the US corporation. The AIM team's purpose was to plan, manage, and execute the integration of all divisions and functional areas of the corporation. This team went on to create other teams, eventually numbering 24 and including more than 500 people from both companies. The consulting firm formed the teams, structured the work of the teams, participated in their major decisions, monitored critical path activities, and consulted with the European firm's managers and functional areas. By engaging the consulting firm as integration contractor, the project benefited from the best practices and lessons learned through the consultant's many years of merger and acquisition experience—experience that the two pharmaceutical giants lacked. The project structure, consisting of the consulting firm, AIM teams, and other teams was a pure project organization devoted entirely to the corporate merger.

15.12 Integration in Systems Development Projects

Project integration can be conceptualized in two ways: integration of the functional *areas* of the project organization, and integration of the *phases* of the project. The former, and the subject of the chapter thus far, is called *horizontal integration*; the latter is called *vertical integration* (Figure 15.14). The two ways are interrelated because integration of the project phases also usually requires integration of the functional areas working within the phases.

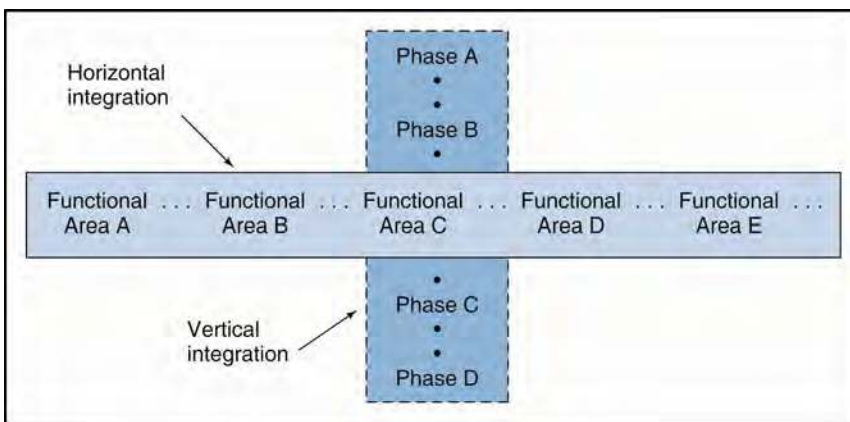


Figure 15.14

Horizontal and vertical integration in a systems development project.

Large-scale product and software development projects require the integrated efforts of many functional units throughout conception, definition, design, testing, production, and installation. Achieving the necessary high-level integration can be difficult and is not always successful.

Serial systems development

In a traditional, **serial development** project, a different functional group is responsible for each project stage. For example, the marketing group specifies the initial concept and user requirements, the design group then defines the technical specifications and creates the system design, the manufacturing group then determines how it will make the product, and so on. The project is a series of “handoffs” where one functional group hands off its work to the next. At each stage, a new functional group takes over, “inheriting” and being forced to accommodate the output of previous areas. In this serial development process, illustrated in Figure 15.15, there is little interaction and knowledge sharing between the participating groups.

Since the different functional groups work somewhat independently, their decisions are often incomplete or incorrect because they do not address the needs and requirements of other functional areas involved in the project. Each functional group entering the process must struggle to accommodate commitments made by earlier groups. When it encounters a prior commitment (about a requirement, design, procedure, etc.) that is wrong or difficult to implement, it must request a modification to the commitment from the other groups. This back-and-forth exchange between groups results in numerous **change requests**, which delay progress and increase costs. The problem is lack of horizontal and vertical integration—a failure of groups involved early in the process to address the complete life cycle of the system and the needs of functional groups that will later inherit and have to live with the consequences of their decisions.

One integrated approach to systems development is concurrent engineering.

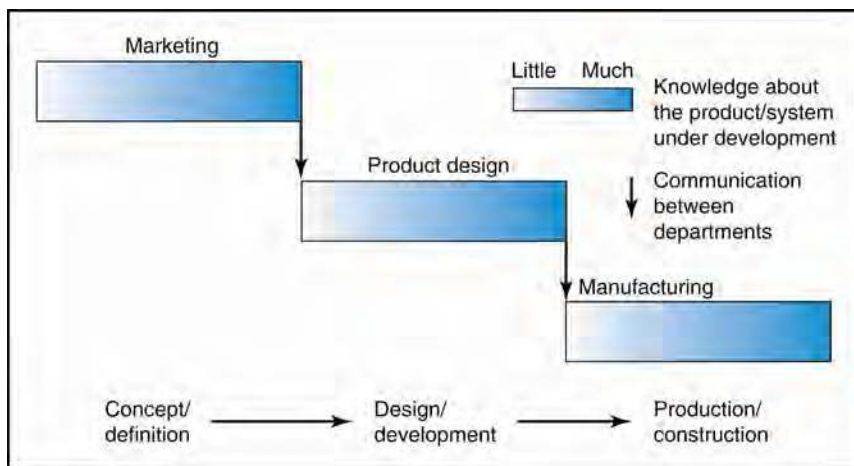


Figure 15.15

Traditional, sequential interaction between functional areas during phases of systems development.

Source: Adapted from Wheelwright S. and Clark K. *Revolutionizing Product Development*. New York, NY: Free Press; 1992, p. 178.

15.13 Concurrent Engineering and DevOps

Concurrent engineering (CCE) is implemented with a cross-functional team structured as a matrix team or pure project organization. Every group, department, or contractor responsible for or influenced by some piece of the project has the opportunity to participate *early* in the project and to contribute to key decisions long before they actually begin to design, produce, test, and operate the system (Figure 15.16). Unlike serial systems development, in which departments and contractors tend to work sequentially and handoff work to one another, handoffs are nonexistent in CCE because all parties have a hand in everything. Horizontal integration and vertical integration are achieved in one fell swoop.

In CCE, decisions about design, development, production, and operation overlap. For example, the process of making dies for stamping automobile body-panels, which is expensive and time-consuming, does not usually occur until near product launch. The body-design group does not talk to the die-producing group until the last minute. With CCE, however, the groups work together so the dies are designed and produced soon after the body panels are designed, which happens months before launch and allows ample time to work out bugs in the dies and make changes if needed. Concurrent design of the panels and dies alone can reduce the production preparation time for a new automobile by more than a year. When concurrent engineering is applied to the design and development of new products, it is called **integrated product development (IPD)**, which refers to a cross-functional team effort to integrate the wants, needs, requirements, and constraints of stakeholders representing the entire system's development life cycle.

Concurrent engineering and IPD improve tradeoffs between product features and production capabilities. Working together, product and process designers can discover subtle changes to the product design that are transparent to the customer but that take advantage of existing

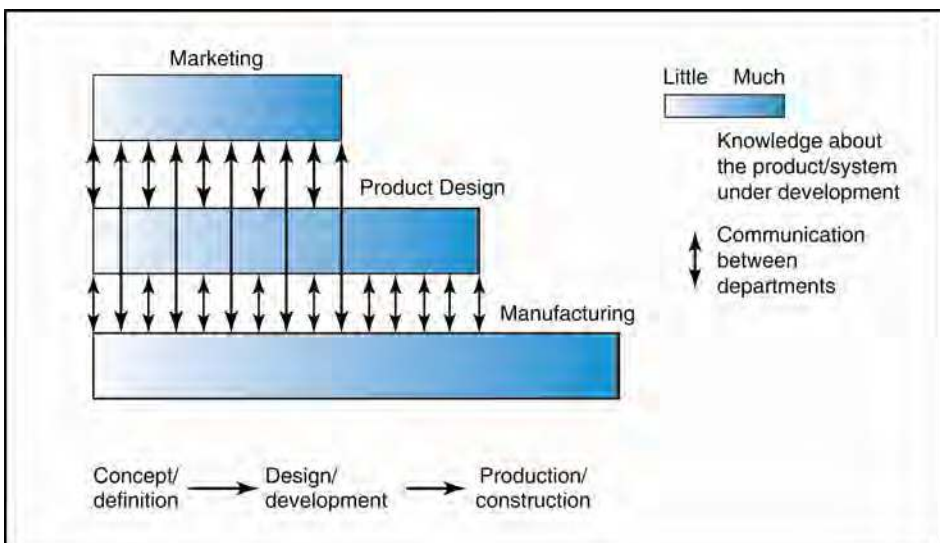


Figure 15.16

Concurrent interaction between functional areas in systems development.

Source: Adapted from Wheelwright S. and Clark K. *Revolutionizing Product Development*. New York, NY: Free Press; 1992, p. 178.

production capabilities, resulting in lower production costs from fewer production bugs and rework, and fewer customer usage problems and warranty claims.

Closely related to CCE is **DevOps**, a software engineering methodology wherein software developers and system users and operators collaborate to create a product that meets the needs and requirements of the users and operators. As discussed in Chapter 14, when combined with agile approaches that employ iterative, rapid product releases, DevOps enables agile teams to quickly understand users' needs and ideas and incorporate them into software features.  See Chapter 14

Although DevOps is relatively new (introduced around 2009), the concept of designers and users working together in product design is identical to CCE and IPD, practices that have been in use for over 50 years.

The goal of CCE, IPD, and DevOps is to foster a **culture of collaboration** and **shared responsibility** between designers and users. A common issue with all these practices is that none of them is easy to implement. The fact is, experts in each discipline do not always look kindly at people from other disciplines offering suggestions or critiquing their work. The success of any of these practices requires building a culture of collaboration and openness to the input of others, no matter their discipline, and willingness to work with them as a team. Ways of doing this and for building effective teams are covered in Chapter 17.  See Chapter 17

Team organization

A CCE team is organized to maximize team member communication and control over design decisions. This is achieved by:¹⁷

- **Autonomy.** Members of a concurrent engineering team are relieved of unrelated obligations and commit fully to the development effort.
- **Full-time, full-duration participation.** Ideally, members participate in all decisions throughout the entire systems development process. That is what “concurrent” means.
- **Co-location.** Team members work in close proximity and share one office, which encourages frequent, spontaneous informal chats. Periodic formal weekly meetings are largely replaced with continual informal meetings and standups.
- **Small size.** The team is small enough to allow open communication and encourage team commitment, yet large enough to represent all the affected functional areas, customers, and key suppliers. About six team members is optimal, although as many as 10 to 20 can be effective. If the size exceeds 20, smaller sub-teams are formed and coordinated by an inter-team steering group.
- **Team of doers.** Each member is a specialist in some area (design engineering, manufacturing, marketing, purchasing, etc.), yet is expected to take on a wide range of responsibilities and obligations. Members are “can do” folks, willing to visit customers and suppliers, work on design, modeling, assembly work, and whatever else that needs to be done.

These features mirror those of the **sprint teams** described in the previous chapter. Although CCE teams usually have a team leader or project manager, in other regards they share characteristics with **self-managed teams**, also described in the previous chapter.

But concurrent engineering is more than just organizing people into teams. It is taking people normally involved in only one stage of the system development process and engaging them in the other stages. Product designers wander the factory to see how their designs are manufactured and what features of the design make it hard or easy to produce. At the same time, production engineers and assembly workers talk to designers to understand why a design

feature is important and has to be retained. Some manufacturers require their designers to spend a full day every few months on the assembly line producing the portion of the product they helped design.

There are other ways in which organizations organize teams to achieve vertical and horizontal integration. Example 15.4 describes two.

Example 15.4: Systems Development at Motorola and Lockheed-Martin¹⁸

Motorola's systems development cycle includes the phases of product definition, contract development, product development, and program wrap-up. The process emphasizes integration of functions to develop innovative products and effective resource utilization for speedy product development. Projects are conducted by a core cross-functional team responsible for most development decisions and detailed design work, as well as for specifying resource requirements and setting performance targets. Functional units provide support to the core team.

The core team approach is used when speed is critical, such as in projects to create systems with entirely new architectures or in markets that quickly change. For a project to develop a new phone, the team consisted of a project manager and eight members from engineering, robotics, procurement, product design, manufacturing, human resources, and accounting/finance, plus a member from Hewlett Packard, the vendor for a crucial component in the system. To encourage outsiders to "look in" and offer suggestions, they worked in a glass-enclosed office located in the middle of a manufacturing facility. The team completed the project in 18 months (half the usual time for a project of that size), met the cost objective (which was much lower than normal), and yielded a product of high quality and reliability.

Lockheed-Martin's advanced development division, called "Skunk Works," has a reputation for developing radical designs and breakthrough aircraft and space vehicles.¹⁹ The term "Skunk Works" is trademarked by Lockheed but in common usage refers to an autonomous project team working on advanced technology that can achieve results more quickly and at a lower cost than traditional development projects. For each development effort Skunk Works hand picks the project manager and cross-functional team. Each Skunk Works team is fully autonomous and controls virtually all the resources it needs. The team is similar to a separate business unit: it works on its own and has the authority to requisition resources and subcontract work. Although projects tend to broadly follow the familiar phases of conception, definition, and so on, the team is free to create procedures and standards that best suit a project's goals. Team members are selected for high competency, broad skills, strong commitment, and ability to think on their feet. They are co-located, usually at an isolated site to increase motivation and teamwork and to maintain secrecy. Aside from budgets and general procedures, the team gets minimal direction from senior management. Since its inception in World War II, Skunk Works has become a model for creating highly innovative, leading-edge aircraft and space vehicles quickly and on budget.

An example is the F117 Stealth fighter project mentioned in Chapter 11.²⁰ The Air Force wanted a relatively low-cost production aircraft that would be difficult to spot on radar (stealth). The Skunk Works team created a radical design that used new materials but minimized costs by using an engine, computers, flight controls, and other parts from



See Chapter 11

pre-existing aircraft (off-the-shelf). The project was completed in record time—31 months, and for \$6.6 billion (considered low at the time). Efficiency and low cost were partly attributed to the small-sized development team (a few hundred people in the design phase), which minimized red tape and maximized communication and project control.

Heavyweight teams

The teams in the approach used by Motorola as described in Example 15.4 are an example of what Wheelwright and Clark²¹ call “heavyweight” teams; they are the systems development equivalent to the pure project organizations described before. The project managers are heavyweights too because they are minimally on the same level as functional managers and have the organizational authority to exert strong influence over everyone involved in the development project. The Lockheed-Martin Skunk Works teams are fully autonomous and even “heavier” in that they control all of whatever resources they need to get the job done. Being autonomous, of course, the team has only itself to blame if the project fails.

Cross-functionality and autonomy enable strong team emphasis on the project goal, discipline in coping with complexity, and consistency between design details. The teams define and bring into focus the customer requirements, and translate them into terms everyone on the team understands. Steps of the development process and details of the system are handled in a coherent fashion, minimizing inconsistencies and changes later on.

A disadvantage of heavyweight teams is that individual components or elements of the end-item system might not reach the same high-level technical excellence as they would if they had received attention from a traditional functional area. Although a cross-functional team might design a component that meets requirements and integrates with the entire system, the component might contain flaws that only a functional team of specialists could have prevented. A way around that problem is to involve specialists in formal reviews, as mentioned in Chapter 10.



See Chapter 10

15.14 Summary

Structure refers to the way organizations attempt to achieve goals and respond to problems in the environment. Two key features of structure are differentiation and integration; the former is the way organizations subdivide into specialized subunits; the latter is how they link the subunits to coordinate their actions. Organizations traditionally differentiate subunits along functional, geographic, customer, and process lines, and integrate the subunits with rules, procedures, coordinated plans, and chains of command. These means are effective when the environment is stable and tasks are certain, but less so when goals and tasks involve frequent change, high complexity, and uncertainty—the case with many projects.

Each project organization is uniquely structured to suit the project’s goals and environment. It is formed to include all the functions needed and to integrate those functions through management roles that emphasize horizontal relations. When the project goal involves just one specialty, the project team is comprised of staff from one functional area. More common is when it requires multiple functions, and the team is comprised of members drawn from all the functional subunits involved or impacted by the project; this form of organization, called a task force or cross-functional team, is managed by a project expeditor or coordinator. Expeditors

and coordinators direct project work but lack authority to command resources or strongly influence the behavior of team members.

For projects that have much at stake and require sizeable resource commitment, the appropriate form of organization is the pure project. This form gives the project goal the highest priority and the project manager greater ability to command and control whatever resources are needed, although it tends to be costly in terms of duplication of effort and project startup and shut down.

The matrix organization form creates project teams by sharing members and resources from across functional subunits. It is effective for creating a continuous stream of project teams; however, it can be difficult to implement and induces organizational conflict.

Many companies use a composite of multiple forms—the matrix and pure project for large projects, cross-functional teams and task forces for smaller ones. Most project organizations are hybrids and combine combinations of the task force, pure project, and matrix forms.²²

The project manager of a large project is often assisted by specialists and functional representatives in the project office. This office handles contracting, planning, scheduling, and control, but its major role is integrating functional units. In a large-scale project that involves multiple companies, integration of all their efforts is sometimes taken on by the project sponsor. For a large, technically complex project, responsibility is usually handled by the prime contractor or a special integration contractor. In companies that undertake multiple, simultaneous projects, oversight and integration of the projects is overseen by the office of projects, PMO, or program office.

Project integration involves coordinating both the efforts of multiple units (horizontal integration) and the phases of the project (vertical integration). In system development projects, this integration is achieved by combining representatives from all parties affected by the end-item system into a single team for the duration of the project, a practice called concurrent engineering or DevOps. The team is formed early in the project and has the resources and authority to make decisions that affect the project and the full life cycle of the end-item product or system.



Review Questions

1. What do the terms “differentiation” and “integration” mean?
2. What are five traditional forms of differentiation? List some companies that presently use each.
3. List the various forms of integrators. Give examples of each.
4. What are the advantages of functional organizations? What are the disadvantages?
5. What distinguishes project forms from other forms of organization?
6. Describe the responsibility and authority for each of the following:
 - a. Project expeditor
 - b. Project coordinator
 - c. Project leader in a pure project
 - d. Project leader in a matrix.
7. Describe the applications, advantages, and disadvantages for each of the following:
 - a. Project task force
 - b. Project team
 - c. Pure project and project center
 - d. Matrix.

8. Give some examples of organizations where each of these project forms has been used.
9. What is the project office? Describe its purpose. Who is in the project office? How should members be selected for the project office?
10. What is meant by the “informal organization”? Give some examples. How does it help or hinder “the formal organization”? How can the project manager influence beneficial aspects of the informal organization?
11. Describe the role of the prime contractor and integration contractor in large-scale projects.
12. One form of integration contractor is the wedding consultant; another is the consultant who organizes high school reunions. For each:
 - a. List the various groups, organizations, and individual parties that are involved and must be integrated.
 - b. Describe the relationship among these parties and how the consultant coordinates their efforts, both prior to and during the wedding or reunion.
13. What parties should or might be included in a concurrent engineering team? What are the contributions of each? How does their inclusion in the team improve (a) the systems development process and (b) the resulting, final product?
14. What do you think are some major difficulties in changing from a traditional non-integrated development approach to a concurrent engineering approach?
15. What do concurrent engineering, integrated product development, and DevOps have in common?



Questions About the Study Project

1. In your project, how is the parent organization organized—for example, functionally or geographically? Show the organization chart, its overall breakdown, and relationships.
2. How does your project fit into the organization chart of the parent organization?
3. What form of project structure is used in your project? Show the project organization chart; indicate the key roles and the authority and communication links between them.
4. How did the project structure develop? Did it “evolve” during the project? Who designs or influences the project structure? What role did the project manager have in its design? Is the design similar to those used in other, similar projects in the organization?
5. Critique the project organization structure. Is it appropriate for the project goal, the parent organization, and the environment?
6. Is there a project office? Is there also an office of projects, program office, or PMO? In each case: (a) describe the office and how it is used; (b) describe the members of the project or program office staff—representatives, specialists, and so forth. What is the purpose of the project office staff? Describe the various tasks and functions handled by the project office. What is the members’ participation in the project office (full-time, as needed, etc.)? What is the reporting relationship between the project manager and members of the project office?
7. How does the project manager integrate functional areas?
8. Are prime and associate contractors involved? If so, what is the function of the company you are studying (prime, subcontractor, or supplier) and how does it fit

- into the structure of all the organizations contributing to the project? If applicable, discuss the involvement of integration contractors or team leader contractors.
9. Did the project apply concurrent engineering or DevOps? If so, discuss how it was applied.

CASE 15.1 ORGANIZATION FOR THE LOGON PROJECT

Iron Butterfly Company is a medium-sized engineering and manufacturing firm specializing in warehousing and materials handling systems. The company purchases most of the subsystems and components for its product systems and assembles them to meet customer requirements. Every IBC system is made to customer specification and most of the firm's work is in system design, assembly, installation, and checkout. The firm's 250 employees are roughly equally divided among five divisions: engineering, design, fabrication, customer service, and marketing. Recently, competition has forced the firm to expand into computerized-drone warehousing systems despite its rather limited experience and expertise in that field.

IBC has been awarded a large contract for a drone system for placement, storage, retrieval, and routing of shipping containers for truck and rail by the Midwest Parcel Distribution Company. This system, called the Logistical Online System, LOGON, is to be developed and installed at Midwest's main distribution center in Chicago. The contract is fixed price of \$14.5 million and includes design, fabrication, and installation at the center. IBC was awarded the contract because it was the lowest bidder and has an outstanding record for quality and customer service. A clause in the contract imposes a penalty of \$1,000 daily for failure to meet the specified delivery date.

At various times throughout the estimated 47-week project, personnel from the functional divisions of design, fabrication, procurement, and customer service will be involved, most on a full-time basis for between 4 and 18 weeks. In the past, the company set up ad hoc project teams comprised of a project coordinator and members from the functional areas. These teams were then responsible for planning, scheduling, and budgeting the actual work to be done by the functional departments. Team members served primarily as liaisons to the functional areas and work part-time on the teams.

The LOGON contract differs from other IBC systems, both in its heavy usage of computerized, real-time operation via drones, and size. Although IBC has some prior experience with real-time warehousing systems, the technology involved is continuously evolving. IBC recently hired people with the backgrounds needed for the project. In addition, it has signed contracts with CRC and Creative Robotics to provide computer software and drone hardware, and assistance with system design, installation, and checkout.

The LOGON contract is among the largest IBC has ever undertaken. The company is presently in the middle of two other projects that absorb roughly three-fourths of its labor capacity, is winding down on a third that involves only the customer service division, and has two outstanding proposals for small projects under review.

QUESTIONS

Discuss how you would organize the LOGON project if you were the president of IBC. Discuss the alternatives available for the LOGON project and the relative advantages and disadvantages of each. What assumptions must be made?

CASE 15.2 PINHOLE CAMERA AND OPTICS, INC.: WHY DO WE NEED A PROJECT MANAGER?

Beverly is the newly appointed vice president of strategy for Pinhole Camera and Optics, Inc. (motto: "See the World through a Pinhole"), a medium-sized, privately owned, manufacturing firm. Until recently the 24-year-old company had experienced rapid growth through developing new products and optical manufacturing processes. Beverly believes that the company's market position has slipped because Pinhole has not been able to react quickly enough to changing market requirements and increased competition. The company is divided into the traditional functional departments of research, marketing, sales, production, and so on. New product development projects are managed by handing off responsibility between managers of the departments. Beverly believes this is the greatest contributor to Pinhole's inability to identify and respond to market opportunities, and she would like to create a new position, manager of new products, for the purpose of integrating departments during product development projects. This position would be the project manager of new product development.

The owner of the company, Ovid Pinoli, disagrees. He contends that the managers of the functional departments, most of whom have been with the company since it started, are excellent managers, really know their specialties, and usually are able to work together. He feels there is no need to create the position, although he wonders where such a person would come from. Mr. Pinoli instead suggests that for each new project one of the department managers be picked to coordinate the efforts of all the departments. The manager would be selected from the department that has the biggest role in the project; in other words, according to whether the project primarily involves research, marketing, or production.

Beverly is convinced that Mr. Pinoli's idea will not improve the situation. She decides to prepare a formal written report that will address the pros and cons of Mr. Pinoli's suggestions and persuade him that the new position of manager of new products must be filled by someone other than a functional department manager. She also wants to describe how Pinhole's new development projects could be better organized and staffed.

QUESTIONS

If you were Beverly, why would you disagree with Mr. Pinoli's suggestion that the existing departmental managers serve as the project manager? What would you say in the report to argue for the position of manager of new products?

CASE 15.3 IMPLEMENTING A MATRIX STRUCTURE IN AN R&D LABORATORY²³

The R&D laboratory of a Dutch multinational corporation served two roles, split 50/50 between product/process development (PPD) and support service to product/process development, production, and other areas of the corporation. The lab's employees were grouped into 13 departments, seven (with 85 employees) devoted to PPD, six (with 84 employees) to support services.

The decision was made to restructure the lab to a matrix operation, and a policy committee was appointed to draft a proposal for the restructuring. After a year of discussion, a “balanced matrix” was introduced and five project managers, recruited from the lab, were appointed to coordinate PPD projects. The functional managers of the restructured R&D departments (who were excluded from strategic decision making) felt uneasy about the balanced matrix and suggested instead a “weak matrix,” but they were overruled. Functional managers responsible for PPD complained about loss of operational authority. Those responsible for support services had a different grievance: in the past, their work supporting stakeholders outside the R&D lab (e.g., production and marketing) always took precedence over PPD activities, which they performed in whatever time remained. The matrix now changed that, giving priority to PPD activities.

The functional managers, who “didn’t feel called upon to cooperate much,” rebelled and ceased making constructive contributions to the projects for their departments. This forced the project managers to attempt to manage these projects single-handedly, which resulted in big work overloads. Trying to speed up project work, they stealthily bypassed functional managers whenever they visited the functional departments. Exacerbating the rift was the fact that project managers were paid better and given nicer company cars than the functional managers.

Twice the functional managers requested that more projects be delegated to their departments. The first time they created a list of 22 big and 26 small projects (“small” defined as requiring fewer than 1,000 labor hours per half-year and involving only one or two departments) and proposed that the small ones be delegated to their departments. The policy committee (under the leadership of one of the project managers) countered this by cancelling some small projects and integrating others into the big projects. Six months later, the functional managers noted that some of the biggest projects involved seven to eight departments, and that coordinating work among them was difficult. They proposed that big projects be divided into sub-projects with responsibility delegated to sub-project managers within their functional departments. The project managers rejected this proposal.

Frustrated that personnel were often shifted between projects by the functional managers who supplied the personnel, the project managers proposed that personnel on larger projects be assigned to projects “semi-permanently,” for a period of, say, 6 months, during which they would not be reassigned. After a year of deliberation, the proposal was approved despite the opposition of managers of the service support departments, production, and marketing because it compromised their ability to give highest priority to service requests.

Initially, service support tasks requiring more than 300 hours were handled by project managers; those requiring fewer were handled by functional managers. Later the 300-hour threshold was lowered to 100 hours. All service requests were sent directly to the departments for subsequent assignment to project managers or functional managers, but the project managers suspected that functional managers manipulated the rule by creating service projects such that they required fewer than 100 hours. They proposed that service requests be sent directly to them, but this was rejected.

Three years after initiating the matrix structure, the caustic behavior slowly decreased; disagreements still existed, but the atmosphere improved. Managers of the support-oriented departments admitted that the matrix structure improved project objective-setting and control, but they still favored a weaker matrix form over the balanced matrix. Managers of the PPD-oriented departments came to accept the

balanced matrix, although the general manager remained unsatisfied and suggested that all departments should split their staffs into two groups, one for product-process development, and the other for support work.

QUESTIONS

1. What do you think were the roles and responsibilities of functional managers before implementation of the matrix structure? How did the changes in roles contribute to conflict after the implementation?
2. Comment on the complaint of functional managers of the support-orientated departments about due dates being enforced for product development work. What solutions do you suggest?
3. Many of the projects involved only one or two departments. Comment on how this fact should have been accounted for in the design of the organizational structure.
4. Why would it make sense for smaller projects to be handled by functional managers?
5. Why would the proposal about sub-project managers in functional departments for the biggest projects make sense?
6. Sometimes reasonable proposals get "politicized." What role do you think mutual suspicion plays in this? How could it have been prevented in this case? Comment on the idea that project managers should display "integrative leadership."
7. It took a year of discussions prior to implementing the matrix structure and three years more before the matrix worked reasonably well. Comment on the time you think it should take to implement a matrix structure. What factors must be considered?

Notes

1. Discussion of the organization structure and integrating mechanisms in high-technology environments is given by Galbraith J. Environmental and technological determinants of organizational design. In: Lorsch J.W. and Lawrence P.R. (eds.). *Studies in Organization Design*. Homewood, IL: Irwin-Dorsey; 1970, pp. 113–139.
2. See Galbraith J. *Designing Complex Organizations*. Reading, MA: Addison-Wesley; 1973.
3. Peters T. and Waterman R. *In Search of Excellence*. New York: Warner Communications; 1984, pp. 127–130.
4. See Davis K. The role of project management in scientific manufacturing. *IEEE Transactions of Engineering Management* 9(3); 1962: 109–113.
5. The matrix organization, its applications and implementation is described in Davis S. and Lawrence P. *Matrix*. Reading, MA: Addison-Wesley; 1977.
6. McCann J. and Galbraith J. Interdepartmental relations. In: Nystrom P. and Starbuck W. (eds.). *Handbook of Organizational Design II*, no. 61, New York: Oxford University Press; 1981.
7. Peters and Waterman, *In Search of Excellence*, pp. 307–308.
8. See Thomas R., Keating J., and Bluedorn A. Authority structures for project management. *Journal of Construction Engineering and Management* 109(4); 1983: 406–422.
9. Cusumano M. and Selby R. *Microsoft Secrets*. New York: Free Press; 1995, pp. 74, 235–236, 248–249.
10. Burns J. Effective management of programs. In: Lorsch J., Lawrence P. (eds.). *Studies in Organizational Design*. Homewood, IL: Irwin-Dorsey; 1970, pp. 140–152.
11. Archibald R. *Managing High-Technology Programs and Projects*. New York: John Wiley & Sons; 1976, Chapter 4.
12. This discussion is based on Livingston J. Weapons system contracting. *Harvard Business Review*. July–August; 1959: 83–92.

13. Livingston, Weapons system contracting, 85.
14. Adapted from Lammie J. and Shah D. Construction management: MARTA in retrospect. *Journal of Construction Engineering and Management* 110(4); 1984: 459–475.
15. Complete discussion of integration contractors provided by Sayles L. and Chandler M. *Managing Large Systems: Organizations for the Future*. New York: Harper & Row; 1971, pp. 253–271.
16. Arndt D., Clampett L., Fedder W., Foxx K., Lorenz C., Ward N., and Worthington J. *Analysis of the Role of the AIM Team in the Integration of Knoll Pharmaceuticals into Abbott Laboratories*. Loyola University Chicago; July 13, 2002.
17. From Nicholas J. Concurrent engineering: overcoming obstacles to teamwork. *Production and Inventory Management Journal* 35(3); 1994: 18–22.
18. Portions adapted from Wheelwright S. and Clark K. *Revolutionizing Product Development*. New York: The Free Press; 1992, pp. 159–162, 200–203.
19. Its SR-71, a high-performance reconnaissance aircraft developed in the 1960s, still holds the world's speed record five decades later. See Rich B. and Janos L. *Skunk Works*. Boston: Little, Brown & Company; 1994.
20. See Aronstein D. and Piccirillo A. *Have Blue and the F-117A: Evolution of the "Stealth Fighter."* Reston, VA: American Institute of Aeronautics and Astronautics; 1997; Stroud M. How the F-117A flew on budget, on time. *Investor's Daily* February 1; 1991.
21. Wheelwright and Clark, *Revolutionizing Product Development*, pp. 194–196, 202–112.
22. Strengths and weaknesses of different project forms are described in Bannerman P. *Risk Implications of Software Project Organization Structures*, 20th Australian Software Engineering Conference, IEEE Computer Society, April 14–17 2009, Gold Cost, Australia.
23. Adapted from De Laat, P. Matrix management of projects and power struggles: a case study of an R&D laboratory. *IEEE Engineering Management Review* Winter; 1995.

Chapter 16

Project Roles and Stakeholders

When a project is approved and a project team formed, often that team is comprised of “outsiders,” people from various functional and professional groups scattered throughout the parent company and outside subcontractors and over which the project manager might have little authority.¹ As Sayles and Chandler describe, project management:

requires a capacity on the part of the manager to put together an organizational mechanism within which timely and relevant decisions are likely to be reached, [as well as] a conceptual scheme for “working” interfaces . . . [It is a] dynamic, interactive, iterative, and intellectually challenging concept of the managerial role.²

Being an effective project manager requires the ability to influence people without giving orders and making decisions in the same way as other managers. Most project managers have a great deal of responsibility but not much formal authority, so they need a different skill set than traditional managers. This chapter describes the responsibilities, authority, and necessary skills of the project manager. It also describes other parties whose involvement and support are necessary to manage projects.

Important to note is that while many of these roles are typical in **traditional projects**, especially large, complex ones, they are not in **agile projects**, where the number of roles is intentionally limited to very few. Team roles in agile projects are covered in Chapter 14.



See Chapter 14

The project manager and project team are just some of many individuals and groups who are involved in, exert influence on, or are affected by a project—collectively referred to as **project stakeholders**, shown in Figure 16.1. All projects have stakeholders, and this chapter discusses who they are and how their interests and expectations are addressed. Shown in the figure, **internal stakeholders** are those individuals and groups who actively work to achieve project objectives—primarily the project team. **External stakeholders** are parties outside the project that (a) also contribute to project objectives or (b) are impacted by or have influence on the project and its results. In Figure 16.1, group (a) is in the gray box, and the (b) group is all the rest.

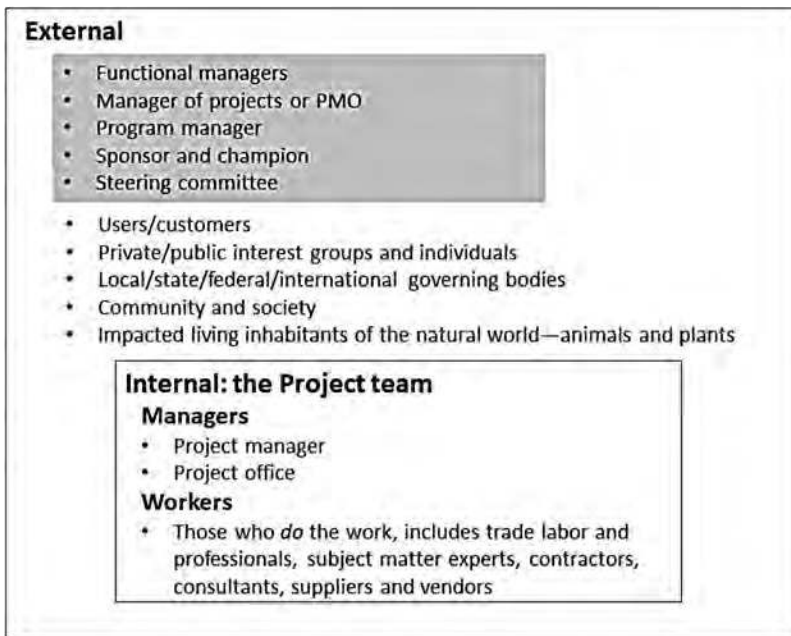


Figure 16.1
Project stakeholders.

16.1 The Project Manager

Project manager's role

The project manager is the centerpiece of project management, and being one manager requires wearing different hats, some at the same time; they are the hats of an integrator, communicator, decision maker, motivator, evangelist, and entrepreneur.

As the central figure in the project, the project manager's prime role is to *integrate everything and everybody* to accomplish project goals. The project manager has been called the organizational "metronome," the person who keeps the project's diverse elements responsive to a single, central beat.

The project manager is the *communication hub*, the funnel through which most reports, requests, memoranda, and complaints flow. She receives input from more sources and directs information to more receivers than anyone else in the project. Between sources and receivers, she refines, summarizes, and translates information to keep all key stakeholders well-informed about project plans, progress, and changes. Many project managers say they spend 90 percent of their time communicating; while all project managers might not agree with that figure, virtually all agree they spend most of their time communicating. (Of course, not all project information needs to be funneled through project managers and, knowing this, they encourage direct informal contact between the individuals involved.)

The project manager is the central *decision maker* for setting project scope and direction, allocating resources, and balancing schedule, cost, and performance criteria. Even when lacking authority to make high-level decisions, she often is the best situated to influence others who do make those decisions.

The prime motivational factor in any diverse group is strong commitment to a central goal; in this regard, the project manager fosters enthusiasm, team spirit, and confidence to drive the team, even when the work is stressful and frustrating.

You could say the project manager is a sort of *evangelist* who builds faith in the project, its value, and workability. During the conception phase, she is often the only person who sees the big picture, and whether the project gets funded often depends on her ability to gain the endorsement of influential stakeholders.

The project manager is also an *entrepreneur* of sorts, driven to procure the funds, facilities, and people needed to get the project off the ground and keep it flying. Early on she must win over reluctant stakeholders who are hesitant about the project. After the project is underway, she must continue to champion it and sometimes fight for its existence. In the end, whether project success or failure, the project manager is the one held ultimately accountable.

Example 16.1: Gutzon Borglum: Sculptor and Project Manager³

If you are familiar with the carvings pictured in Figure 16.2 then you know the handiwork of Gutzon Borglum. Well over two million people a year visit Mount Rushmore National Memorial. Many of them who hear the name Gutzon Borglum think that it was he who *sculpted* the faces. He was of course the sculptor, though not of the actual faces on the mountain.

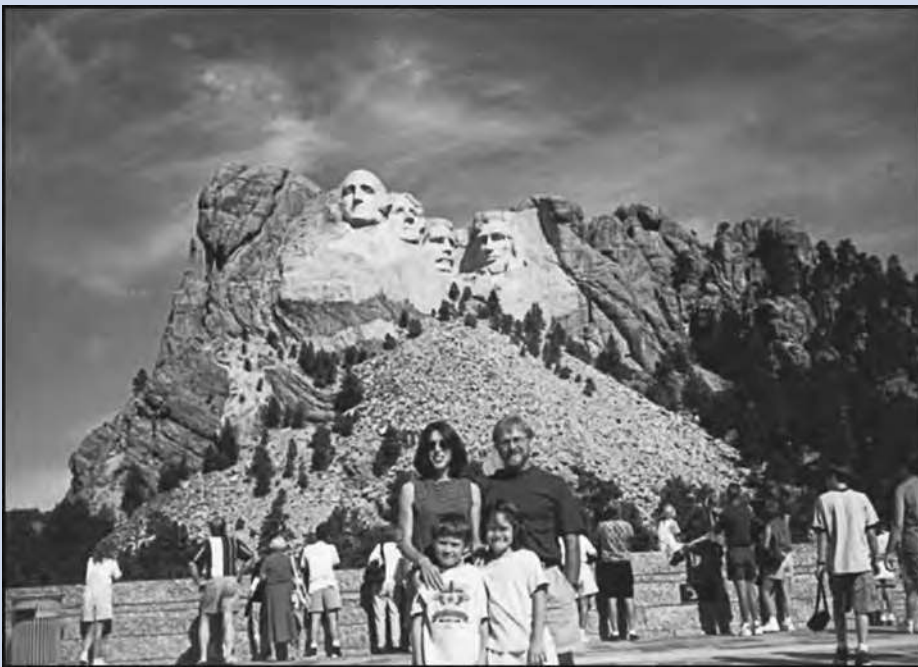


Figure 16.2

Gutzon Borglum's most famous work attracts millions of visitors a year.

Source: Photo courtesy of John Nicholas.

The contract for the project specified that the memorial was “to be carved by and/or under the direction of Gutzon Borglum” and that Borglum was to enjoy “full, final, and complete freedom of authority in the execution of the monument’s design.” He did carve the faces, but on a miniature model 1/12 the size of the ones on the mountain to serve as a guide for workers who did the actual “sculpting,” much of which consisted of removing huge quantities of granite using dynamite, heavy drills, and pneumatic jackhammers.

Projects of such grandiose size are never the work of just one person, although in the case of Mount Rushmore if anyone should get credit, it would be Gutzon Borglum. Many others contributed to the project in important ways, but it was Borglum’s tireless efforts that yielded much of the project funding, and his genius and dedication that made it happen. He picked the site; he wrote letters and spoke personally to businessmen, wealthy industrialists, senators, congressmen, and US presidents; he determined that the faces would be of Washington, Jefferson, Roosevelt, and Lincoln; he hired and directed the work crew; he created the innovative means for transferring the design from the model to the mountain; and, *in addition*, he attended to myriad details, from designing the scaffolding, work platforms, tramway, hoists, and grounds buildings to orchestrating the pageants for the initial dedication and final unveiling ceremonies. People wondered when he ever slept. Of course, he was not perfect; he did not always have project problems under control, and his efforts in the early years were criticized for being disorganized. When the project began in 1927, Borglum wasn’t completely sure what the monument was going to look like. People familiar with him were impressed with his artistic talent, but they were even more impressed with his “capacity for affection, wrath, generosity, stinginess, nobility, pettiness, charm, and sheer obnoxiousness.”⁴ He was short on modesty and humility, and long on “mulish stubbornness.” He thought big, dreamed big, talked big, and was not afraid to tackle any undertaking. His enthusiasm was contagious.

The project work crew consisted of 22 men. Most of the sculpting they did using 80-pound drills and jackhammers while dangling on the side of a cliff. They sat in harnesses designed by Borglum that were lowered down the mountain face with hand winches. Imagine their feelings, as described by biographer Rex Smith:

You do your drilling while hanging on the side of a stone wall. From where you sit you can look down upon mountains and plains that stretch farther than the eye can see. Surrounded by these vast spaces, suspended against a stone cliff, you feel dwarfed and insignificant . . . and uneasy.⁵

Borglum was a stickler for safety, so despite the dangers there were few accidents and no fatalities throughout 14 years of work. Borglum was never chummy with his crew, but he cared and looked out for them, and, in return, they were loyal to him and to each other.

Seeing the monument today, we realize that its construction must have posed great challenges, but that, obviously, those challenges were overcome. Borglum, however, was never sure they would be overcome. Although he had selected the mountain, he knew there was the risk that it might contain some disastrous hidden flaws—cracks or bad rock—that could not be worked around. Besides funding, it was the shape of the mountain and its deep fissures that determined that the number of presidential busts had to be just four. Time and again obstacles arose, funds ran out, and the project had to be stopped. But Borglum and other supporters persevered so that the project would again be revived. In the end, the carving was abandoned, and the monument left uncompleted because the nation was about to

become embroiled in World War II and would no longer support the effort. Borglum died just months before the project was canceled, and you have to ask, had he lived, how much more of the monument would he have completed? Borglum was a sculptor, but when it came to turning a mountain into a monument, he was the ultimate project manager.

Job responsibilities

The project manager's principal responsibility is to deliver value to stakeholders through project end-items that satisfy goals, requirements, and contract terms. Other, specific responsibilities vary depending on size and nature of the project, stage of the project, and responsibilities delegated by upper management, which range at the low end from the rather limited influence of a project expeditor up to the centralized, almost autocratic control of a pure project manager.

The usual responsibilities include:⁶

- Planning project tasks and end results, creating the WBS, schedule, and budget; coordinating tasks; and allocating resources.
- Creating and organizing the project team.
- Negotiating and working with influential stakeholders (customers, functional managers, contractors, supporters, and top management) and managing their expectations.
- Monitoring project status and communicating status to stakeholders.
- Identifying technical and functional problems.
- Dealing with crises and resolving conflicts.
- Anticipating risks and working to mitigate them.

Most managers of medium and large-sized projects report in a line capacity to a senior-level executive. They are expected to monitor and narrate the technical and financial status of the project and to report current and anticipated problems, or overruns.

Domain competency and orientation

Project managers work at the *interfaces* between top management, the customer, subject-matter specialists, and contractors, so they must have managerial ability, technical competency, and other broad qualifications. They must feel at home whether in an office talking with executives and customers about policies, schedules, and budgets, or on-site talking to subject matter experts about technical matters.

A broad background is also essential. Regardless of how differentiated the project and backgrounds of the people working on it, the project manager needs to understand the contributions of every area of the project, and the jargon, techniques, and procedures that people use in each. Called **domain competency**, the project manager must have a good understanding of *all areas* of the project domain. Put another way, the project manager's technical and administrative competency must cover the *full scope* of the project, i.e., all areas within the project scope statement and tasks at the first or second level of the WBS. Although most project managers cannot be experts in all areas of the project, they must be sufficiently familiar with those areas to be "credible"—to be able to intelligently discuss issues, ponder ideas offered by specialists, and evaluate and make good decisions. And whatever the core technology or main thrust of the

project—product or systems development, construction, operations, and so on—the project manager must be especially strong.

To deal effectively with top management and the customer, project managers must be familiar with the workings and businesses of the parent and customer organizations. Overall, they should know more about the project than anyone.

Of course, project managers cannot know everything. When they are responsible for areas about which they are ignorant, they admit it and seek input and advice from people they trust and who are competent. As a rule of thumb, a project manager tries never to bluff since doing so risks losing credibility with stakeholders.

Studies show that the most effective project managers adopt goals, time, and interpersonal **orientations** that are *intermediate* to the functional units in their projects, i.e., they take a balanced outlook.⁷ For instance, in projects that require a production department and a research department to work together, the project manager adopts a time perspective intermediate between the production group's short-term, weekly outlook and research group's long-term, futuristic outlook.

The relative importance of technical ability versus managerial competency depends on the project. Project managers in most research and engineering projects need some or much technical competency because the problems and orientation of the project team are technical. In non-technical projects, project managers might get by with little or no technical competency. But to manage projects that involve multiple, diverse functional areas, the project manager needs to know something about all of them, plus the managerial ability to integrate them. In general, they must be sufficiently technical to be able to understand project issues, but not so technical as to neglect their managerial role. In project management, there is no substitute for strong managerial competency.

16.2 Project Management Authority

Authority refers to a manager's power to command others to act or not to act. The most familiar kind of authority is that conferred by the organization and written in the manager's job description, called **legal authority**. With legal authority, people in higher organizational positions are viewed as having the "right" to control the actions of people who report to them. Associated with legal authority is **reward power**, the power to evaluate and reward subordinates. Another kind of authority, **charismatic authority**, stems from the power one gains through personal characteristics such as charm, appeal, personality, and appearance.

Traditional authority

Management theory says that authority is always greater at higher levels in the organization and is delegated downward, presumably because managers at higher levels "know more" and, therefore, should be able to make decisions and "command" workers at lower levels—the concept of legal authority. This point has been challenged, particularly in technology-based organizations, where managers cannot possibly know everything needed to make decisions. They often lack complete technical expertise and so, increasingly, rely on subordinate specialists for advice. Even managers with strong technical skills cannot manage alone; they rely upon staff groups for technical, personnel, and budgetary assistance. This so-called *participatory*



See Chapter 17 *management* or *shared leadership* is commonplace in projects (described in Chapter 17).

Influence

Legal authority is not the same as *ability to influence*. Although managers with legal authority can influence subordinates by barking orders and controlling salaries and promotions (called downward influence), the most effective managers exert influence without giving orders or making an issue of their superior–subordinate relationship. In fact, managers who rely solely on legal authority are often relatively unsuccessful. This is especially true with subordinates who are well-educated or highly experienced, who respond better to other sources of influence, namely knowledge and personality.⁸ The first source, expert power, refers to a person’s special level of knowledge or competency, plus the belief by others that knowledge and competency is important and greater than what they possess. Simply, the expert power holder is viewed as being right because he knows more, and others readily defer to his decisions.

The other, **referent power**, derives from rapport, personal attraction, friendship, alliances, and reciprocal favors. The subordinate in some way identifies with the power holder and defers to him. Through referent power, a person has charismatic authority.

Someone with expert and referent power can influence others regardless of the formal hierarchy, for example peers or colleagues (**sideward influence**) or clients and outsiders (**outward influence**). These forms of power can even subtly reverse the authority relationship. A subordinate may exert considerable influence over her superior (**upward influence**) if the superior comes to rely upon the subordinate for information or advice, or if a bond of trust, respect, or affection develops between them. Everyone has seen this, and history is replete with examples of people of “lower” social or organizational stature controlling people of higher stature: Alexandria was Empress of Russia; Rasputin was a lowly priest.

Authority in projects

Traditional managers rely on different forms of influence—knowledge, expertise, persuasion, and personal relationships; but when these fail, they can fall back on their legal authority. But project managers, however, seldom do this. Except in the case of the pure project manager, the typical project manager *lacks any form of legal authority*.

In traditional organizations, influence and authority flow vertically, but in projects they flow horizontally and diagonally. Often, the project manager role exists *outside* the traditional hierarchy. In pure projects and task forces, the role is temporary, superimposed on the existing structure, so, consequently, it is not afforded the leverage inherent to a hierarchical position. Project managers work across functional and organizational lines and, except perhaps for members of the project office, have no subordinates reporting to them in a direct line capacity. The issue is further complicated in matrix organizations wherein project managers share authority with functional managers.

Thus, despite the heavy responsibility, most project managers lack a comparable level of formal authority. They have instead what is called **project authority**, which is authority to make decisions regarding the project, but no authority to give orders backing up those decisions. (Note, there are exceptions, depending on the contract. In the so-called **EPC contract**, the project manager has considerable legal authority to make decisions and direct work. This is explained in Chapter 12, Example 12.1.)

The disparity between high formal responsibility and low formal authority has been referred to as the **authority gap**.⁹ The gap implies that project managers must strive to develop other forms of influence in the absence of legal authority. “How to make friends and influence people” is not an academic issue for project managers.



See Chapter 12

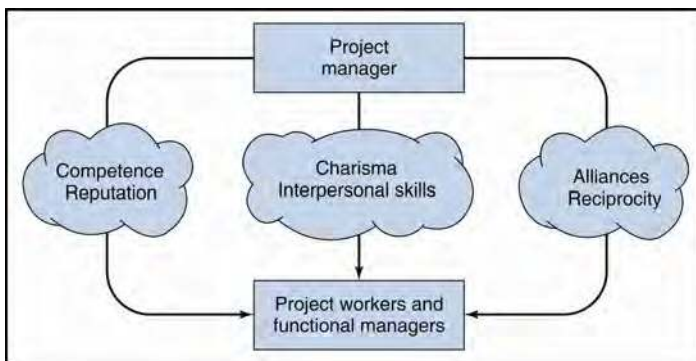


Figure 16.3
Project manager's
sources of
influence.

Project manager's authority

Most project managers handle the authority gap in a similar way: having no legal authority, their recourse is to rely on influence derived from *expert power* and *referent power*. They do this also because no matter what the project, they depend on others to get the job done. Even in rare cases where project managers have legal authority, they use it sparingly because making unilateral decisions and barking orders are usually not the best means to motivate people to do the work.

Project managers also gain influence through **networks of alliances** and **informal connections** they build with stakeholders. The strength and breadth of these networks increase with the project manager's perceived competency, reputation from prior project accomplishments, and charisma. Charisma, mentioned before, refers to the project manager's personal appeal—something about the project manager's demeanor, behavior, or personality that people like. Why should this matter? Well, project stakeholders are more likely to support, assist, or do what a project manager requests if they like her than if they do not; it's as simple as that. If the project gets into trouble, a project manager that people like will have many alliances and friends to call upon for help.

In summary, most project managers rely upon knowledge, experience, personal relationships, and personality to influence others (Figure 16.3). To build expert-based power, they must be perceived as technically and administratively competent. To build referent-based power, they must develop effective interpersonal, persuasion, and negotiation skills. The way project managers employ these sources of influence is illustrated in Example 16.2.

Example 16.2: Effective Project Managers, Contrast in Styles¹⁰

Two examples of how different project managers uniquely influence people are Kelly Johnson and Ben Rich, both former head managers of the advanced projects division of Lockheed-Martin Company, the "Skunk Works."

Kelly Johnson was a living legend, not only at the company but throughout the aerospace industry. With the help of a highly cohesive team of engineers and shop workers, he created over 40 airplanes, including the fastest, highest-flying ones in the world. Yet, he was strictly business, without humor, hot tempered, and reputed to "eat young engineers

for between-meal snacks.” He made the bureaucrats and engineers with whom he had dealings sweat—particularly excuse makers and faultfinders, so he had as many detractors as friends. Nonetheless, when management needed someone to head up the most difficult and challenging projects, they repeatedly selected Kelly. Why? Beneath his bad temper was a sure-fire genius. He knew everything, it seemed, and his ability to make accurate, on-the-spot deductions amazed everyone. For a new engine inlet, Kelly simply glanced at the initial design and pronounced it wrong, about “20 percent too big.” His engineers worked a full day to recompute the design only to discover that, sure enough, the engine inlet was 18 percent too big. Another time he looked at a design and said, “the load here is 6.3 psi.” After an hour of complicated calculations, his people measured it as 6.2 psi. When he retired, Kelly Johnson was recognized as the preeminent aerodynamicist of his time.

Kelly chose as his successor Ben Rich. Ben was the first to acknowledge that he didn’t possess Kelly’s genius and would rely on his teams for most decisions. His first move was to loosen the reigns and allow his teams to make most calls on their own. Ben was decisive in telling a team what he wanted, but he then let them decide which methods to apply. He used a non-threatening approach—schmoozing, cheerleading, and one-liners. As one employee said, “Whereas Kelly ruled by his bad temper, Ben ruled by his bad jokes.” Whereas he didn’t shirk from scolding deserving individuals, he preferred complimenting people and boosting morale. Said a colleague, he was the perfect manager—there to make the tough calls, defend and protect his project teams, obtain more money and new projects, and convince the government and senior management of the value of his teams’ work.

Despite using different styles, Johnson and Rich both have been acknowledged as exemplary project managers. Kelly Johnson accomplished great things, despite his temperament, and most engineers considered it an honor to have worked with him. Competency and reputation were his strengths; people tolerated his personality. Ben Rich was no technical slouch, but he acknowledged that he had some smarter people working for him. He had charisma and many personal friends, and with that he too was able to accomplish great things at the Skunk Works.

Project-functional shared authority

Typically, project managers and functional managers share authority, but in the best-performing projects that authority is clearly differentiated.¹¹ Project managers have authority to decide what work must be done, procure resources, coordinate work, and mediate conflicts; in contrast, functional managers have authority to decide how the work must be done (the technology to use) and to resolve technical problems.

Although project managers rarely have any form of reward power, workers tend to be more responsive to project managers if they perceive the project managers as having some influence over their performance reviews, salaries, and promotions.

16.3 Project Manager Qualifications

The qualifications of successful project managers fall into four categories: personal characteristics, interpersonal skills, business skills, and technical skills.

Personal characteristics

Archibald lists the following personal characteristics as essential for project managers:¹²

- Flexible and adaptable
- Preference for initiative and leadership
- Confidence, persuasiveness, verbal fluency
- Effective communicator and integrator
- Able to balance technical solutions with time, cost, and human factors
- Well-organized and disciplined
- Generalist rather than specialist.

These characteristics make sense given the project environment and the responsibilities and restrictions placed on the project manager role. Obviously, project managers must also be able to handle the pressure of constant deadlines, great uncertainty, startups and closeouts, and constant changes in goals, tasks, people, and relationships.

The typical project manager likes to be out and about, mingling with the project team on-site and elsewhere with stakeholders. In this way she connects with people, stays informed, and builds morale.

Interpersonal skills

The project manager must possess strong interpersonal skills and the ability to “actively listen to” and “read” people.¹³ Active listening means asking for clarification and paraphrasing to make sure you understand what people are saying, it means:

- Asking leading questions.
- Remaining quiet and allowing the other person sufficient time to talk.
- Reflecting on the person’s answer and checking for correctness.
- Reflecting on the person’s emotions.

The acronym for active listening is LEAR: listen, explore, acknowledge, respond.

The project manager must be sensitive to the perspectives of project stakeholders. Specialists on the project team often disdain anything non-technical and resent the imposition of schedules and budgets on them. The project manager must be able to convince them why such things matter.

Much of this derives from what is called **emotional intelligence (EI)**—which is a person’s awareness of others’ and one’s own emotions, and one’s ability to control emotions. EI has four components:

- **Self-awareness** is knowing one’s strengths, weaknesses, and emotions, and the impact those have on others. It is developed by seeking feedback from others, and consciously practicing behaviors that set others at ease and bring out the best in them.
- **Self-management** is exercising control over one’s emotions, especially in difficult or stressful situations. It enables managers to set the right tone in all situations—to remain calm and focused on the task, and not place blame in demanding situations.
- **Social awareness** is the ability to read others’ emotions, strive to understand their feelings and perspectives—to *empathize*, which is an important means for providing **social support** to individuals and reducing stress in project teams. It includes the

ability to know through non-verbal cues and instinct whether to trust a person's actions or words.

- **Relationship management** is the ability to coach and mentor others and resolve conflict, which are necessary leadership skills for developing team members and building teams.

The project manager must also be able to build trust, promote team spirit, and reward cooperation; often she does this through the only form of reward she is able to give: praise and credit. A good project manager understands the personalities, attitudes, and strengths of the team and knows how to utilize their talents, even when they do not measure up to her standards.

None of these abilities is easy to acquire, but all are necessary for effective leadership. It is a tall order, but a good project manager can do all of it.

Business skills

The project manager is, after all, a *manager* and so must also have general business knowledge and skills that include:

- Understanding the organization and the business.
- Knowledge of *management*—marketing, accounting, contracting, purchasing, human resource administration, and business concepts.
- Ability to translate business requirements into project and system requirements.
- Strong, active, continuous interest in teaching, training, and developing subordinates.

Most project managers have budgetary responsibility, so they must understand the concepts of cost estimating, budgeting, cash flow, overheads, incentives, and cost sharing. They are involved in contract agreements, so they must know contract terms and implications. They are responsible for scheduling work tasks to meet delivery dates, so they must be familiar with the tasks and the resources for executing them. And they are responsible for enforcing plans and schedules, hence must be knowledgeable about tools and techniques for tracking, and control.

Technical skills

To make informed decisions, project managers must have a strong grasp of the project's technical aspects. As mentioned earlier, their **domain competency** must span the full scope of the project. In low-technology projects, this grasp can be developed through experience and informal training. In high-technology projects, technical qualifications are more rigorous and usually require a career molded in the technology environment and a degree in a science, technology, engineering, or math (STEM-related) field.¹⁴

Although project managers seldom do technical analysis, they must be qualified to incorporate concepts from different disciplines to make technical judgments. Technically qualified people often are not good at integrating concepts from other disciplines because undergraduate training in engineering and other technical fields tends to emphasize analysis and ignore integration.¹⁵ The project manager must be able to understand and speak the technical jargon of different subject-matter experts on the team, regardless of their specialty; this is minimally necessary for communicating with and integrating the work of those experts.

Selection and recruiting

The manager to head up a forthcoming project is selected from among the ranks of product and functional managers, subject-matter experts, and experienced project managers. The last source is the best though not always feasible, since it might be difficult to find an experienced project manager who has the right mix of qualifications and is available for the project. As a result, when an experienced project manager is needed, often she is recruited from the outside; this is readily observable in job listings online and in newspapers (Figure 16.4 shows a sampling). The downside with choosing an outsider is that it takes time for the manager to

Senior Project Manager

We need aggressive, innovative, top-flight professionals to manage large-scale, complex building design projects. Degree and registration, plus a minimum of 10 years solid U.S. experience in putting buildings together.

Complete management responsibilities for scheduling, resource planning, feasibility studies, design and design/build for multi-million dollar projects.

PROJECT ENGINEERS

5-6 years of project management experience in the design of electro-mechanical products. Your track record of managing projects from inception thru completion should include: project and material planning; costing and product testing. A BSME is required with exposure to electrical engineering preferred. Supervisory and decision making skills.

PROJECT COORDINATOR

Move into this key position with the company that moves America...

Here you will assist/support a project team which analyzes the work flow/efficiency of several departments. Position involves: Heavy utilization of a PC; analyzing and assembling reports; independent research; identifying information pertinent to various projects; maintenance and control of all files; designing and developing PC forms and reports. Candidate will also perform various other duties as needed.

Requirements include 3-5 years of equivalent project coordination experience, preferably within a corporate environment, highly developed PC, communication and organization skills.

PROJECT MANAGER

We are a rapidly growing ENR 400 firm with an outstanding opportunity for a Project Manager. Responsibilities relate to activities for environmental audits and site assessments.

The successful candidate should have project management capabilities and hazardous waste background. Business development experience is also desirable.

SYSTEMS PROJECT ANALYSTS/LEADERS

Federal Savings and Loan Association has excellent professional opportunities for experienced Systems Project Analysts/Leaders. Immediate opportunities are available in the areas of telecommunications, McCormack and Dodge financial systems, and savings and loan financial systems, including installment lending and mortgage banking applications.

Ideal candidates must possess a college degree in business administration, or computer science with a business administration background, and have at least 5 years of experience in systems project implementation, operations analysis, and finance/accounting. Direct exposure and/or experience in savings and loans, other financial institutions, or mortgage banks is desirable.

Specific systems experience should include structured project management experience, through implementation. Candidates must also possess project supervisory and excellent communication skills.

CORPORATE PROJECT MANAGER

National manufacturer of transportation equipment is seeking an experienced individual to assume responsibilities as their Corporate Project Manager.

Candidates must possess a Bachelor of Science degree including some programming training. Three to five years experience within Production or Engineering Management within the primary metal fabrication industry is desired with experience working with mini and micro computers. Additionally, the preferred candidate will be experienced in modern inventory control techniques within a heavy manufacturing environment. Initial responsibilities of this function will be the analysis and installation of systems within a manufacturing environment with the successful incumbent becoming a candidate for executive management responsibility.

PROJECT ENGINEER

Pharmaceutical Company, the nation's leading manufacturer of private label vitamins, over-the-counter drugs and beauty products has an immediate need for a Project Engineer.

RESPONSIBILITIES. This position will be responsible for managing a variety of specification, design, and implementation projects in our manufacturing process area, and will also include providing assistance to the maintenance group and other engineering departments as needed.

QUALIFICATIONS. A BS in Mechanical Engineering, Chemical Engineering or related degree will be required, as well as 5 years project experience, preferable in a processing manufacturing environment.

PROJECT MANAGER TECHNICAL SERVICES

Our client, a financial services organization located in Texas, has retained us to assist them in recruiting a top level ELECTRICAL ENGINEER to manage hardware and software projects which support the services provided by the company's operating units.

The successful candidate will report to corporate manager of Technical Services and must possess a BS degree in Electrical Engineering and a minimum of 5 years experience. Experience should include: electronic hardware design, assembly language programming.

Project Supervisor

Looking for a challenging career with the leader in the communications industry? An opportunity is currently available for the qualified professional with technical, communication and management skills.

As a member of the Telecommunications Service Organization, the successful candidate will be involved in coordinating the installation of communications technology.

Strong interpersonal skills are essential as is the ability to work productively in a fast-paced, unstructured environment. This position requires a background in coordinating project installations and/or the communications industry.

PROGRAM DIRECTOR

A major social service agency in Chicago is seeking a Program Director for an adult day training program for persons who are blind and developmentally delayed. Responsibilities include planning, organizing and supervising the activities.

PROGRAM DIRECTOR

A major social service agency in Chicago is seeking a Program Director for an 85 bed inpatient chemical dependency treatment center. Doctoral level psychologist or MSW with extensive management experience in large inpatient facility required.

Figure 16.4
Advertisements for project management positions.

build friendships and alliances and learn organizational policies. On the plus side, she is likely better suited to objectively take on the task (without political influence), and less likely to have any enemies—at least initially.

The project manager can also be selected from inhouse functional managers, although functional managers sometimes have difficulty shifting to a project perspective, which requires overseeing and integrating the work of many areas rather than managing just one. And, without abundant well-rounded experience, each tends to view the world from her own, functional perspective, and everyone perceives her as just another functional manager.

Project managers can also be “created” by promoting subject-matter experts (SMEs—engineers, scientists, system analysts, designers, etc.), although this has the same drawback as with promoting any non-manager into a managerial role: the person has to first learn how to manage. Being a good SME is no assurance the person will be a good project manager. In addition, the specialist must learn how to remove herself from her area of specialty and become a generalist.

Ideally, the project management assignment will not conflict with existing lines of authority. It is a bad idea, for example, to promote a specialist to the position of project manager that gives her authority over her former boss.

Training

Project management skills cannot be learned quickly, so organizations devote significant time and expense to preparing individuals for the role. Some sponsor internal training programs that focus on the special requirements of their organizations; others use external seminars and university programs. Recent years have witnessed a proliferation of both kinds of programs and a rise in training directed toward professional certifications such as the PMP, APMP, and ICB-IPMA. Often, a project support office or PMO assists with this training and professional development.

But there is no substitute for experience. Many organizations allow promising people who aspire to become project managers the benefit of on-the-job training.¹⁶ As part of their career paths, they rotate assignments for aspiring project managers throughout all areas of the organization to develop sufficient domain competency to enable them to manage projects in those areas. Meantime they work full- or part-time as assistants to experienced project managers; while this exposes them to management, it also tests their aptitude and talent for being managers. Valued technical specialists with little managerial aptitude or ability are given other, non-managerial career opportunities commensurate with their skills and interests.

Example 16.3: On-the-job Training of Project Managers¹⁷

Microsoft Corporation’s approach to preparing project managers (which they term “program managers”) is typical. There is neither an official program for training program managers nor guidelines that spell out job requirements: people learn the job by “doing” it. Microsoft carefully selects and mentors the right people, then expects them to learn on the job. For about 90 percent of program managers, training happens by pairing a new program manager with an experienced, successful program manager; the other 10 percent receive formal training that includes a 3-week training session. Microsoft occasionally holds video-recorded luncheons where managers present their experiences, and then post the recordings online.

Moving into the role

Project management responsibilities range from few and mundane on simple projects to extensive and challenging on complex projects. Regardless of the project manager's qualifications, the burden of transitioning to the role is eased when the project manager:¹⁸

- Understands what must be done.
- Understands his authority and its limits.
- Understands his relationship with others in the project.
- Knows the specific results that constitute a job well done.
- Knows what he can do and where he falls short.
- Knows how to mitigate risks and resolve conflicts.
- Understands that his superiors have an interest and confidence in him and want to see him succeed.

Ideally senior management demonstrates interest and confidence, although sometimes the project manager must seek out, request, or demand them.

16.4 Filling the Project Management Role

Organizations use various titles for the project manager role including “project director,” “project leader,” and “task force chairman.” The titles “task force coordinator,” “project supervisor,” and “project engineer” are also used, though these usually imply more focused roles with less responsibility than other forms. When no one is available or competent enough to manage the project, the role is filled in other ways, for example by the general manager or plant manager, though these managers usually have neither the time to devote to the project nor the flexibility to shift roles. Alternatively, the role may be assigned temporarily to a functional manager. Here, the manager must divide her time between the project and her department, and both may suffer. Also, this combination functional-project manager may have trouble gaining cooperation from other functional managers when they see him as a competitor for resources—the “two-hat” problem mentioned in Chapter 15.



See Chapter 15

In long-term projects, responsibility may pass from one functional manager to the next as the project moves through stages. In that situation, however, there is no one to provide the managerial or technical continuity to integrate the stages. The project moves through a series of handoffs, and managers of later stages are forced to inherit problems created by managers of earlier stages.

Sometimes project management responsibilities are divided among two or more people, as in construction projects where the architect is responsible for technical matters while the so-called project manager handles administrative “paperwork.” Two managers tend to complicate issues of coordination, communication, and authority because both share responsibility. Further, when the project manager becomes subservient to, say, the architect, her ability to manage the project is compromised. A similar split is common in the motion picture industry. The movie producer manages the resources, schedules, and budgets (in essence, the project manager), while the director oversees technical-artistic matters; occasionally they are the same person. Because the shooting of a motion picture is an artistic pursuit, directors need flexibility in budgets and shooting schedules, but costs also matter and the producer faces the question of “what is the price of creativity?” It is no surprise that the two do not always have a happy relationship.¹⁹ Nonetheless, the movie industry holds the role of project manager in high

regard. When an Academy Award is given for “Best Picture,” it is awarded to the picture’s producer—the person who managed the resources, budgets, and schedules.

Some projects, especially large ones in the public sector, require exceptional presentation, negotiation, and political skills to deal with broad stakeholder constituencies and powerful public- and private-interest groups. In such projects it is also common to see two people heading up a project—one to deal with technical matters, the other with stakeholders.

Ideally, there is but one project manager, and any others also serving in a managerial or administrative capacity (engineers, architects, directors, etc.) report to her. The project manager becomes involved during proposal preparation and remains through project closeout.

Although ideally the person filling the project management role is committed full-time to managing the project, it is common for managers to oversee multiple projects. This is acceptable as long as the manager can adequately fulfill the responsibilities for all of them. In fact, managing multiple projects can be advantageous because it puts the project manager in a position to resolve resource and priority conflicts and to negotiate resources among all the projects he oversees.

16.5 Other Management Roles in the Project Team

Although the project manager is responsible for and oversees the project, a typical project involves the active or supporting management of other parties and individuals. Early in a project, the project manager works with these other parties, which include functional managers and/or contractors to divide the project into work packages; this will determine skill requirements and serve as the basis for personnel selection and subcontracting. Those who contribute to the project’s management are part of the **project management team**. This section describes their roles.

Members of the project office

An example **project office** (not to be confused with a **project management office** or PMO described in Chapter 15) for a large engineering-development project is shown in Figure 16.5. Typically, these members report to the project manager and include the following:



See Chapter 15

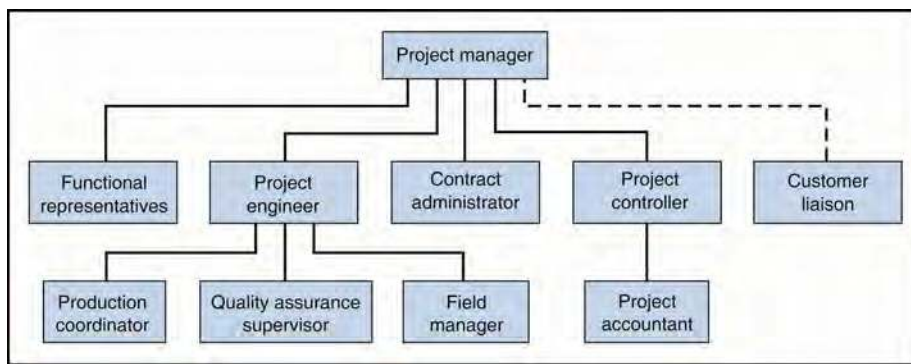


Figure 16.5

Members of the project office in an engineering development project.

The **project engineer** (aka systems engineer or systems designer) is responsible for coordinating the technical areas and assuring integrated design of the end-item. Sometimes the title “project engineer” denotes a person having full project manager responsibilities, but more commonly it refers to the more limited role described here. When several functional areas or subcontractors are involved, the project engineer:²⁰

1. Oversees product or system design and development.
2. Translates performance requirements into design requirements.
3. Coordinates and directs the work of the functional areas and subcontractors.
4. Plans, monitors, evaluates, and documents progress in the design and testing of subsystems and the overall system.
5. Oversees configuration management and the change control system.



See Chapter 12 are discussed more fully in Chapter 12.

The **contract administrator**²¹ is responsible for project legal aspects and subcontracting with outside firms. These responsibilities, which include preparing proposals, defining and negotiating contracts, integrating contract requirements into project plans, ensuring fulfillment of contractual obligations, and monitoring and communicating project changes to the customer,

The **project controller**²² works with functional managers to define tasks on the WBS and to identify individuals responsible for controlling tasks. She maintains work package files and cost summaries, releases approved work authorization documents, monitors work progress, evaluates schedule and cost progress, and revises estimates of time and cost to complete the project. She also prepares revisions to budgets, schedules, and work authorizations, drafts progress reports to users and management, and closes cost accounts upon project completion.

The **project accountant** provides accounting support. He sets procedures for using the PCAS, assists in identifying tasks to be controlled, establishes control accounts, prepares cost estimates, validates reported costs, and investigates financial problems.

The **customer liaison** maintains amicable contractor–customer relations. She participates in technical discussions and reviews as specified in the contract and helps expedite contract changes.

The **production coordinator** plans and coordinates production aspects of the project. Responsibilities include reviewing engineering documents released to production; developing requirements for equipment and parts; monitoring parts procurement and assembly processes for the end-item; monitoring production costs; scheduling production-related activities; and serving as the project manager’s liaison to the production department.

The **field or site manager** oversees construction, installation, testing, and hand-over of the project end-item to the customer. Responsibilities include scheduling field operations, monitoring field operations costs, and supervising field personnel.

The **quality assurance supervisor** establishes and administers quality assurance procedures. His responsibilities are to raise quality awareness and institute means for improving work methods and producing zero defects.

The project office also has **representatives** from participating functional departments and subcontractors. These people work with the project manager and each other to coordinate the activities of their functional areas with the overall project.

The number of staff in the project office should be as small as is practical. This makes the office simpler for the project manager to manage and minimize personnel costs and assignment problems. Members of the office staff contribute full- or part-time as needed and might be physically located in different places.

Functional managers

Often in project work, project managers get most of the glamour while functional managers feel their roles are diminished. But if earlier discussions have led you to believe that functional managers are somehow subservient to the project manager, be advised that that is rarely the case. The project manager relies on functional managers for maintaining technical competency and staffing and executing project tasks within their disciplines and functional areas. They work with the project manager to define the tasks and to plan, schedule, and budget them.

Personnel in matrix organizations shift from one project to another, and their only permanent “home” is their functional department. The functional manager is responsible not only for the hiring, performance reviews, and compensation of the people in his area, but also their career development. Unlike project managers who tend to solicit “human resources” solely in terms of what is best for their projects, functional managers are more likely to look out for the interests of the people in their areas, and help them develop and maintain competencies necessary to perform the function.

In most project organizations, functional managers retain much the same authority and responsibility as in non-project environments. Nevertheless, some believe that the project manager role undercuts their authority and that they could handle each project better if it were exclusively within their domain. Project managers who try to undermine the authority of functional managers will have difficulty obtaining the support and resources they need (see, for example, Case 15.3 in Chapter 15).

Before a project begins, the responsibilities and contributions to technical content for each functional manager should be clearly delineated.²³ This will ensure a continued strong technical base for all projects and alleviate potential animosity between functional and project managers.

In some projects each functional manager selects a **project functional leader** to serve as liaison between himself and the project manager. This person prepares his department’s portion of the project plan and supervises project work performed by the department.

When a large amount of work is assigned to a given department, that work is divided into multiple work packages, and responsibility for each is delegated to a **work package supervisor**. The supervisor prepares the plan for the work package and supervises the work.



See Chapter 15

Manager of projects or PMO director

The **manager of projects**, also called the **PMO (project management office) director**, or *director of projects* or *vice president of projects* (Figure 15.8 in Chapter 15) helps manage and oversee multiple projects. This person:²⁴

- Ensures projects fit with the organization’s resource limitations and strategic objectives.
- Works with functional heads to allocate resources and resolve priority conflicts between projects.
- Assists in developing project management policies and techniques and systems for project planning and control.
- Ensures consistency among projects and that changes to one project’s cost, schedule, or performance objectives are integrated with those of other projects.
- In some cases, directs and evaluates all the project managers.



See Chapter 15

Chapter 18 describes the PMO and the director role more fully.



See Chapter 18

Senior management governance and steering committees



See Chapter 19

Senior management makes all major decisions about project selection and prioritization. It approves the project feasibility study, selects the project manager, and authorizes project startup. In organizations that practice project portfolio management wherein projects are managed in groups for better alignment with organizational strategies and resource allocation, responsibility is headed by the project portfolio manager, discussed in Chapter 19.

Top management establishes the rules and policies that govern the organization's project management. Either directly or through the PMO, it:²⁵

- Defines the project manager's responsibility and authority relative to other managers.
- Defines the scope of and limitations on the project manager's responsibilities.
- Establishes policies for resolving project conflicts and setting priorities.
- Specifies criteria for evaluating the project manager's performance.
- Supports the project management methodology.

A project manager's authority is as conferred by the manager of projects and stated in the organization's charter, or as agreed in the contract with the customer. In situations involving critical negotiations or irresolvable conflicts, top management may preempt the authority of the project manager.



See Chapter 20

For large or geographically dispersed projects, a **steering committee** composed of senior managers or other influential stakeholders might be formed to guide the project or provide decisions and resources beyond the project manager's authority. This is discussed more in Chapter 20.

Program manager



See Chapter 18

When the project is part of a larger effort called a **program**, the project manager works with and sometimes reports to the **program manager**, who is responsible for coordinating the project with other projects to achieve program goals. Programs and program managers are discussed in Chapter 18.

Project supporters: sponsor and champion

Every project ideally has the support of two key outsiders, the project sponsor and the project champion. The **project sponsor** is someone who works to ensure that the project gets the necessary priority and resources, and, thus, usually has a disproportionately high impact on the project's success. She has the formal authority necessary to clear roadblocks and the leadership ability to influence top management. In many cases, the sponsor is the party that funds or makes funding available—investors, banks, or a senior manager with budget authority. To be effective, she must be able to influence key decision makers and able to act in the long-term interests of the customer organization. When the project is part of a program, the program manager sometimes serves as the project sponsor.

During project conception, the sponsor becomes the “owner” of the project business case, and thereafter is responsible for keeping the project on track and moving toward the goals and benefits as specified in the case. During project conception, estimates of costs and benefits are prepared by cost engineers and subject matter experts, but it is the sponsor who

coordinates their efforts. During project execution and after project closure, the sponsor monitors the actual value produced by the project end-item and its actual costs. In long-duration and **megaprojects** (over a \$1 billion US), the sponsor is a full-time job and is occupied by a senior executive with many years of experience and, often, the final stage of her career before retirement.

Depending on the project, the sponsor might devote little time or much time to the role; in big projects, the role requires a dedicated, full-time person. In all cases, the sponsor is the interface between the customer and project organizations and is available to assist the project manager, particularly for decisions and issues beyond the project manager's authority or control.

The **project champion** is someone who firmly believes in the project and argues in its favor, both at its inception and thereafter. The primary role of the champion is to sell the project, especially to external stakeholders—those not involved in the project but who will be affected by it. To these stakeholders, the champion is the project's most visible and outspoken spokesperson. The champion, like the sponsor, must be respected and the sort of person able to convince stakeholders of the project's value and benefits; unlike the sponsor, however, the champion often lacks the formal authority to remove barriers and command resources. When the project does not have a champion, the project manager must put on his "evangelist" hat and go scout for one or take on the role himself.

Occasionally the champion and sponsor are the same person.

16.6 Project Stakeholders²⁶

The term **stakeholders** appears almost everywhere in this book, in many cases referring to groups and individuals not involved in the project team. That's because such stakeholders play such a critical role in project success. By definition, a project stakeholder is any group or individual who is *affected by, interested in, or potentially can influence* the project. Among the most important stakeholders are project customers and users, but there are others too, including partners, investors, lenders, governments, the press, trade groups, and prospective customers/users. Recognizing that projects have widespread—even global—ecological impacts, we can also include as project stakeholders everyone concerned about or affected by those impacts, including everyone in the larger community and society as whole. For that matter, we might include all of Earth's living organisms and ask, for example, how a project might contribute to species extinctions, now or later, and what could be done to mitigate it? That's an existential question, but one worth pondering as much by project managers as anyone.

Some stakeholders support the project and want it to succeed; others resist and want to kill it. The latter might include societal, environmental, and political interest groups or lobbies, and anyone who competes with the project or perceives it as detrimental to their or society's interests.

16.7 Stakeholder Engagement

As a rule of thumb, the bigger the project, the greater the variety and number of stakeholders. Infrastructure projects like the Big Dig and UK High-speed Railway can have many thousands and, ultimately, millions of stakeholders. For the Big Dig, some of these are shown in Figure 16.6, but those are just the tip of the iceberg. Beyond contractors, consultants, sponsors, and owners are the local businesses and citizens who are inconvenienced, drivers who

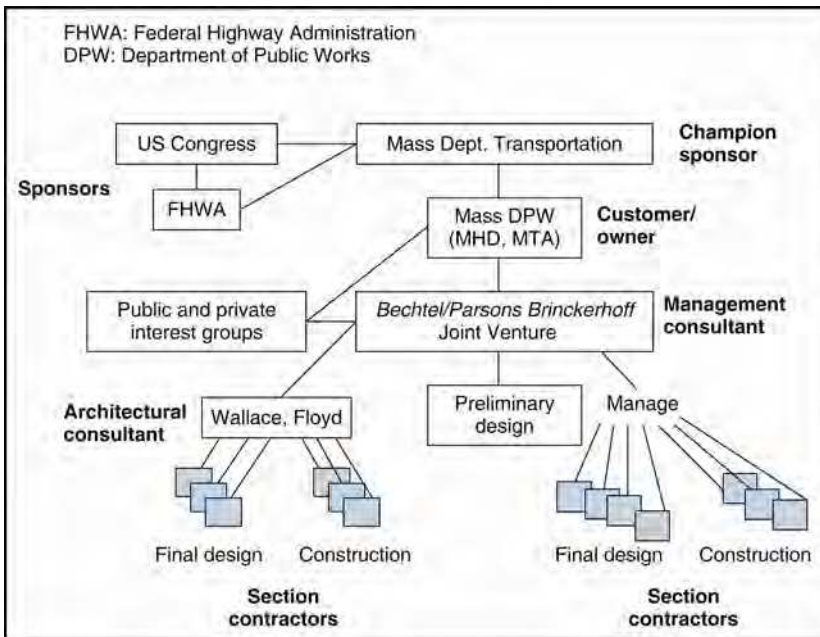


Figure 16.6

Key stakeholders in the Big Dig Project prior to 1992.

will traverse the roadways, and everyone who must adjust to long-term changes the project will bring.

At the start of the project, no one provides the project manager with a comprehensive list of project stakeholders. Users, customers, or sponsors might be known, but not the others. Most of these stakeholders will not be aware of each other. Before a project starts, the project manager needs to discover who the stakeholders are and learn their needs and interests in the project. This begins the process of **stakeholder engagement**, which follows a sequence of steps similar to risk management and other realms of project management, namely: identify, assess and plan, act, and monitor.

Identify

In essence, the project manager should prepare a list of all individuals, organizations, and groups influenced by or able to influence the project and try to determine possible relationships or lines of influence among them. Some stakeholders will be identified during the conception phase of the project (e.g., customer, users, sponsor), others during the planning process (contractors, suppliers, agencies and governing bodies, and involved departments) and execution phase (surrounding communities and private interest groups).

Stakeholders' interests, needs, and attitudes regarding the project vary widely, and the project manager needs to understand them. This requires learning their views and opinions, what they hope for from the project (and project manager), what they need (explicit requirements), what they expect (unstated requirements), and how they might be influenced. Example 16.4 illustrates the range of stakeholders in a big project.

Example 16.4: The Big Dig²⁷

Boston's Central Artery/Tunnel Project (CAT) —known locally as the Big Dig —is an example of a complex project that must accommodate the interests of many stakeholders, including federal, state, and local governments, contractors, and many interest groups (Figure 16.6).²⁸

The central artery portion of the project replaced an elevated interstate highway that ran through downtown Boston with a tunnel. The elevated highway (derisively called the “green snake”) was an eyesore that separated Boston's North End and waterfront from the rest of downtown. Besides replacing the central artery, the CAT project included a tunnel under Boston Harbor to Logan Airport and new bridges across the Charles River to Cambridge—over 160 lane miles over 3.7 miles of tunnels, 2.3 miles of bridges, and 1.5 miles of surface streets. Celebrated as “the largest, most complex highway project ever undertaken in the US” its original price tag was \$5 billion, though its eventual cost was over four times that.

Project supporters faced daunting problems. The Massachusetts congressional delegation had to drive bills through the US Congress that would provide most of the funding; this required accounting for the interests of—and making promises to—a large host of ad hoc congressional allies. Upon gaining funding authorization from the Federal Highway Administration (FHWA), supporters turned to the issue of who should oversee the project, the Massachusetts Bay Area Transportation Authority (MBTA) or the Massachusetts Department of Public Works (DPW). Although MBTA had a better construction management reputation, the job was given instead to DPW on the rationale that MBTA is a transit, not a highway, agency. To manage the project, DPW hired the experienced contractor team of Bechtel/Parson Brinkerhoff, a joint venture formed by two of the world's largest consulting and management engineering firms—Bechtel Corporation and Parsons Brinkerhoff Quade & Douglas. The two (called Joint Venture) had partnered before as contractors for the San Francisco BART System, and Bechtel had worked on the English Channel Tunnel and Disney's theme park in Florida.

The CAT project was broken down into the phases of conceptual design, preliminary design, final design, and construction. Joint Venture created the preliminary design but hired contractors for final design and construction. Initially the project consisted of 56 design and 132 construction work packages, each with a prime contractor. Managing the contractors responsible for the artery and tunnel design packages required especially close coordination since these packages produced contiguous road and tunnel sections that had to dovetail.

In accordance with the law, Joint Venture placed a draft of the project in public libraries and provided public hearings. These resulted in DPW and Joint Venture engineers having to negotiate with hundreds of business, neighborhood, church, and environmental groups, developers, and individuals to mitigate countless issues regarding community and environmental impacts. These ultimately contributed to large escalations in project scope, costs, and schedules.

Opposition or resistance to a project often stems from misunderstanding or unawareness, which can be addressed through better communication. It also comes from stakeholders' concerns over negative consequences, real or perceived, which the project manager might ameliorate through better informing the stakeholders or altering project requirements and plans to satisfy them. The most important thing is to listen to customers and understand their needs and concerns.

Assess and plan



See Chapters
4 and 6

The list of stakeholders might be long, and not all stakeholders can, should, want to, or need to be engaged in the project. For those that should or must be engaged, the *form of engagement*—communication, decision making, and interaction—should be determined and included in the project **responsibility matrix** or **RACI chart** (described in Chapter 6). Also, for each stakeholder, the *form of communication*—reports or updates, frequency, and preferred modes (in person, email, website, etc.)—should be determined and specified in the **communication plan** (see Chapter 4). Project managers who promote participation and input from stakeholders, especially business partners, employees, and customers, potentially expand the innovative and creative abilities available to the project team.

Act and engage

Table 16.1 shows roughly how to prioritize stakeholders and choose appropriate strategies for engaging or otherwise dealing with stakeholders, depending on their level of interest in and ability to influence the project.

The chosen strategy should address whether the stakeholder’s “interest” is supportive, opposing, or neutral, and how the project manager will engage the stakeholder throughout the project. For example, stakeholders with high influence but low interest need to be “kept satisfied,” lest they become unsatisfied and then opposed to the project. Keeping stakeholders with high interest and high influence updated and contented is the most important.



See Chapter 11

For stakeholders opposing the project, the strategy should specify how to gain their support or mitigate or accommodate their opposition. Influential stakeholders can alter the course of the project—providing resources or opportunities on the one hand, or resisting or blocking the project on the other. They might pose potential risks that would be included in the **risk management plan**, discussed in Chapter 11.

Some stakeholders have no interest in the project and no influence over it, although that does not necessarily mean they should be ignored. Most species of plants and animals are not protected by laws from the harmful or destructive impacts of projects, although their need to survive and prosper should be respected (which begins when they are formally acknowledged as project stakeholders).

Table 16.1 Forms of engagement according to stakeholder interest and influence.

	<i>Stakeholders with low influence on the project</i>	<i>Stakeholders with high influence on the project</i>
<i>Stakeholders with high interest in the project</i>	Keep informed	Focus on these: manage closely; keep informed and satisfied
<i>Stakeholders with low interest in project</i>	Possibly ignore but monitor for changes in interest/influence	Keep satisfied

Monitor

As the project progresses, the stakeholders and their stakes and involvement in the project change. Internal stakeholders and members of the project team arrive and depart, depending on their required contributions to the project. As portions of the project are completed, the composition of external stakeholders changes too. Throughout, the project manager tries to keep atop these changes and keep stakeholders engaged by seeking their input through forums, surveys, or project reviews. Sometimes the strategy is to strive to satisfy all high-interest stakeholders, even when they have low influence (Example 16.5). Sometimes it is to establish good relations with the community, to make the project a “good citizen,” and build and retain support from local citizens and the parties who represent them (Example 16.6).

Example 16.5: Disgruntled Stakeholders

Chris is the project manager for a 54-story office tower rising next to the Chicago River. The tower overshadows a 12-story loft building next door, and as it rises, it blocks stunning views the loft residents once had of the river and skyline—a common problem in cities whenever one building goes up next to another. To acknowledge their annoyance, Chris arranged every morning for coffee, rolls and donuts from a popular coffee shop to be served in the loft-building lobby. When some residents complained they didn’t like the coffee, he switched to another shop. One weekend after the lower floors of the new building had been erected and the site cleaned up, he organized a day-long picnic with activities for the loft residents and families. When one resident, Hilda, complained that her small unit would be in full view to anyone in the new building. Chris offered to buy her blinds and drapes, but she refused.

Early every morning, construction on the new building resumes and, with it, bright lights and cacophony. To remind him of her irritation, every morning Hilda leaves a message on Chris’s mobile, something like, “Here it is 6 am and I’m wide awake because of all the commotion you are causing.” Several times a week Chris calls her back. Ever courteous, he apologizes that there isn’t more he can do to make things better for her.

Given limited resources, technical capability, or the demands of other stakeholders, not every need and expectation can be met, in which case the project manager does the best she can, sometimes doing things because they are the “right” or ethical thing to do.

Example 16.6: McCormick Place West²⁹

McCormick Place West is part of a major multi-year, multi-phase project to expand Chicago’s McCormick Place convention complex. The group of companies that teamed up to design and build the structure (another “joint venture”) worked to establish relations with nearby residents and businesses. Project managers and staff visited local high schools to educate students about practices and careers in construction, engineering, and architecture. They offered a program to hire local workers, teach them trade skills through hands-on experience, and the opportunity to become union certified; about 20 people a year be-

came certified in this way. The contractor donated old computers to local schools and cars for their shop classes. Copying a popular reality-TV series, the company remodeled the home of a local family in need. These and other charitable programs benefited the local community and helped the contractor to gain the community's support.

16.8 Stakeholder Engagement Plan

As part of the project execution plan, the **stakeholder engagement plan** specifies the hows and whys of stakeholder engagement. The plan outlines steps to be taken in the engagement process—identify, assess, engage, and monitor, and includes a **stakeholder register** that lists the stakeholders, their importance, and forms of involvement. The modes, methods, and timing of communication with stakeholders is reflected in the project communication plan.

Responsibility for stakeholder engagement extends beyond the project manager and can include the rest of the project team and broader project community. Their involvement and responsibilities should also be outlined in the engagement plan.

Getting a project off the ground involves negotiating hoops and hurdles posed by stakeholders; the project manager is always mindful of this and works to gain and retain their support in ways big and small. For some stakeholders, active involvement at stages of the project is necessary. Examples of involvement are approving project scope, requirements, and plans at the start; staying apprised of progress and setbacks, inspecting work, approving changes during the project; and approving everything at closeout. For others, communicating project status and encouraging feedback is adequate. For influential stakeholders, engagement is equivalent to building relationships and gaining their trust and confidence. Many serious problems, even among potential advisories, can be overcome when parties work together and trust each other.

Whatever their involvement, the project manager knows that to some or large extent it is stakeholders who determine the project's shape and direction. Given that the project manager works with or among stakeholders over which she has no authority to command action or give orders, the interpersonal and emotional intelligence skills mentioned earlier come into play. Whether dealing with internal or external stakeholders, those skills help the project manager to incorporate stakeholders as part of the broader project team, and to resolve conflicts and build enthusiasm for project outcomes.

16.9 Summary

Project managers work at the project–functional–customer interface, integrating project elements to achieve project objectives. They are responsible for project success yet often work outside the traditional hierarchy and have little formal authority. To influence decisions and behavior they tend to rely on negotiation, alliances, favors, and reciprocal agreements. Their strongest source of influence is the respect they gain through skillful and competent administration, technical competency, and charisma.

Successful project managers are perceived as both technically and administratively competent. They have both business and domain competency—broad knowledge encompassing the full scope of the project, with strong competency in the main technology or focus of the project. They also have strong interpersonal and communication skills and can function effectively in uncertain, fluid conditions.

The project manager role is best filled by one person involved in the project, start-to-finish. Sharing or rotating the role among several people is usually less effective, although appointing different project managers for different project phases is sometimes necessary to meet technical, administration, or political considerations.

Project managers get work done through a team composed of people from various functional and support groups scattered throughout the parent company and from outside contractors. Providing administrative assistance to the project manager is the project office. Functional managers contribute to the technical content of the project and share responsibility for developing plans, schedules, and budgets for tasks performed by their areas. They maintain the technical base from which projects draw.

Top management, the manager of projects or PMO director, the program manager, and the project champion and project sponsors all play key roles in the project. Senior managers or steering committees establish the policies, responsibilities, and authority relationships through which project management is conducted. The manager of projects or PMO director ensures that projects are consistent with organizational goals and receive the necessary resources. The champion rallies support for the project and convinces others of its benefits and value. The sponsor supports the project and through organizational clout helps the project get the needed priority and resources.

Numerous other stakeholders support or resist the project and potentially have a big impact on its success or failure. The project manager needs to know who they are, their interests in and influence on the project, and develop strategies for the most important stakeholders to gain their project support or mitigate or accommodate their opposition. How this will be done is summarized in the project stakeholder engagement plan.

People find project work challenging, rewarding, and exhilarating, but without question they often also find it taxing and stressful. Maximizing the chances of project success—and minimizing human casualties along the way—requires special skills for leading groups and individuals, building teams, and resolving conflicts, topics of the next chapter.



Review Questions

1. What is the project manager's primary role?
2. What is meant by "the project manager is an evangelist and entrepreneur"?
3. Describe the responsibilities of a project manager. In what ways are budgeting, scheduling, and controlling considered as integration and coordination responsibilities?
4. Discuss the relative need for both technical and managerial competence in project management.
5. Why is a broad background essential for the project manager? What is a broad background?
6. What is legal authority? How does it differ from charismatic authority?
7. Describe how and in what ways people in organizations, regardless of hierarchical position, influence others.
8. How does the authority of the typical project manager differ from authority of other managers?
9. What is meant by the "authority gap"?
10. What is the most common source of influence used by project managers? How does the project manager use this influence to induce functional managers to assign personnel to the project?

11. List the ideal qualifications—personal, behavioral, technical—for project managers. How do they differ from the qualifications for functional managers? How do these vary depending on the project?
12. Discuss the considerations in selecting a project manager from among each of the following groups: experienced project managers, functional managers, and functional specialists.
13. Discuss the pros and cons in the various ways of filling the role of project manager (e.g. part-time, multiple project managers for one project, one manager for multiple projects, etc.).
14. How are project managers trained on the job? What are the advantages and drawbacks of relying upon on-the-job training as a source for project managers?
15. Describe the responsibilities of key members of the project office for a large-scale project.
16. Describe the responsibilities of the manager of projects or PMO director.
17. Describe the project-related responsibilities of top management.
18. Describe the responsibilities of the functional manager, the project leader, and the work-package supervisor in project management and their interfaces with one another.
19. Who is the project champion and who is the project sponsor?
20. Who are the stakeholders? What influence do they have on a project and why is it important to consider them?
21. What should the project manager do to “engage” stakeholders?



Questions About the Study Project

1. In your project, what is the formal title given to the role of project manager?
2. Where in the organization structure is the project manager? Show this on an organization chart.
3. Describe in one sentence the overall role for the project manager of your project. Now, list his or her *specific* responsibilities.
4. In your opinion, is the so-called project manager the *real* project manager or is someone else controlling the project? If the latter, what effect does this have on the project manager's ability to influence the project?
5. Would you describe the project manager's orientation as being more technical or more managerial? Explain.
6. Describe the project manager's professional background. Has it helped or hindered her ability to be a project manager? (You might pose this question to the project manager.)
7. Describe the project manager's authority. How much legal authority does the project manager have? Is the project manager's authority specified in the organization charter?
8. How big would you say is the project manager's authority gap? Explain. Does the project manager have any complaints about it?
9. From where does this organization get its project managers? Does it have a procedure or seminars for training and selecting project managers? Where did the manager of your project come from?
10. How does this project manager fill the role: part- or full-time, shared or rotated with other managers, manager of several projects at once? Explain. Does the project

manager have enough time to do an effective job? Would another way of filling the position be more effective?

11. Is there a project office? If not, how are the responsibilities (e.g. for contract administration) handled? If so, who is in the project office (a project engineer, contract administrator, field representative, etc.)? Are they on loan, full-time, or part-time? Describe their responsibilities.
12. What functional managers are involved in this project? Describe their responsibilities in the project and decisions they make unilaterally or share with the project manager.
13. Is there a manager of projects or PMO director? Project champion? Sponsor? Describe their responsibilities and influence on the project.
14. What has been the role of top management in your project? What, in general, is the involvement of top management in projects in this organization?
15. Who are the other key stakeholders? How has the project manager communicated or worked with, engaged, or otherwise accommodated them in planning and executing the project?

CASE 16.1 THE LOGON PROJECT

Top management of the Iron Butterfly Company (IBC) has decided to adopt a project-management form of organization for the LOGON project. As a consultant to top management, you have been given two tasks to help implement this. First, you must develop a project management policy statement and a project manager job description. The policy statement should define the project manager's role with respect to functional managers and clarify the role of functional managers in the project. The job description must define the specific responsibilities and legal authority of the project manager. You should consider the functional managers' reactions to the policy statement and job description and how best to get their "buy in." How can the project manager have sufficient authority to manage the LOGON project without usurping the authority of the other managers whose support is necessary? You should also suggest to top management what forms of incentives can be used to get team members to work together toward project goals. Remember, the functional departments are also currently involved in their own work and work in other project activities.

Your second task is to specify and document the qualifications for the position of LOGON project manager. After considering the nature of the project (technical scope, risks, complexity, etc.) as described in Case 15.1, prepare a list of qualifications—general background and experience, personal characteristics, managerial, technical, and interpersonal skills—for screening candidates and making the final selection. IBC has some employees who have worked as project coordinators and expeditors, but none are experienced as a pure project or matrix manager. Consider the assumptions and pros and cons of selecting a functional manager or technical specialist from inside IBC or an experienced project manager from outside the company. A contract has been signed and LOGON is to begin in 4 months.



See Chapter 15

CASE 16.2 SELECTING A PROJECT MANAGER AT NUWAVE PRODUCTS COMPANY

Nuwave Products Company, a medium-sized manufacturer of small motors and motor parts, recently contracted with a software consulting firm, Noware, Inc., to design software for a new integrated manufacturing system to be installed in the near future. The software design is part of a much larger project that also involves procurement and installation of new manufacturing equipment, a new production process, and retraining of workers. The new production process will involve “lean production” concepts that are very different from Nuwave’s current process; it will engage workers in improvement efforts and ultimately require no less than a *cultural change* among Nuwave’s managers, supervisors, and line workers.

Ordinarily the manufacturing department assigns a project manager to projects that involve new processes. However, no one in the department has any experience with a project of this scope, the new software and equipment, or with lean production concepts and cultural change. Some Nuwave managers think that besides designing the software, Noware should oversee the entire project—equipment installation, the “lean transition,” and worker training. In contrast, the manufacturing department manager thinks that one of his senior engineers, Roberta Withers, should handle the project. She has a thorough knowledge of the current manufacturing process and is the department’s expert in mechanical systems. She is a degreed mechanical engineer and has been with Nuwave manufacturing department for 6 years. She knows nothing about lean production or integrated manufacturing systems, but her boss thinks the project would provide a good opportunity for her to learn.

QUESTION

Assume that you must act on the information provided above: if it were up to you, who would you select to manage the project: Noware, Roberta, or someone else? Explain.

CASE 16.3 STAKEHOLDERS IN BOSTON’S BIG DIG³⁰

(Refer to Example 16.4, above) Before the Massachusetts congressional delegation could seek federal funding for the Big Dig project, it first had to poll constituents about sensitive transportation issues. Then-Speaker of the House Philip “Tip” O’Neal wanted to know where his supporters—voters of East Boston—stood. When first told about the project, he said, “We’re not building any tunnel.” He changed his mind when supporters predicted that “the trade unions are going to be marching on you (if you veto the tunnel)” and assured him that “no homes would be lost” in East Boston. The delegation then faced opposition from the Reagan administration and FHWA, both of which initially argued that the project was ineligible for federal funding.

An early responsibility of Joint Venture/DPW management team was to prepare an environmental impact statement, the draft of which consisted of several thick volumes. Part I described impacts in 17 categories, including “transportation,” “air quality,” “noise and vibration,” “economic aspects,” “visual characteristics,” “historic resources,” “water quality,” “wetlands and waterways,” and “vegetation and wildlife.” Under “eco-

conomic aspects” it described commercial and industrial activity, tourism, and employment patterns in the affected areas. The report claimed the project would not displace any residences but would relocate 134 businesses with 4,100 employees.

At the first public hearing 175 people spoke, including some from the EPA and the Sierra Club, and 99 provided written commentary. The project’s magnitude and complexity is reflected in a sampling of the public interest groups represented: The 1000 Friends of Massachusetts, American Automobile Association, Archdiocese of Boston/Can-Do Alliance, Beacon Hill Civic Association, Bikes Not Bombs, Boston Building Trades Association, Boston Society of Architects, Charles River Watershed Association, Conservation Law Foundation of New England, and Haymarket Pushcart Association.

The Massachusetts Secretary of the Environment issued a certificate of approval, allowing construction to proceed only after certain measures had been implemented to mitigate environmental impacts. The certificate recommended planning for utilization of 27 acres of downtown Boston that would be newly created by the removal of the elevated Central Artery and urged formulating “creative strategies” for integrating the new highway system with mass transit, limiting downtown parking, and reserving highway lanes for high-occupancy vehicles.

Beyond environmental matters, the project had to respond to issues raised by hundreds of groups, businesses, and agencies; officials put the number of early mitigation commitments at 1,100 for an added project cost of \$2.8 billion, including \$450 million for temporary lanes, curbs, and sidewalks that would enable businesses to continue during construction, and \$230 million for the City of Cambridge to build a park along the Charles River.

QUESTIONS

1. From information provided here and in Example 16.4, create a list of the project’s stakeholders. Expand Figure 16.6 to include them and show possible links (relations or influences) between them. For each stakeholder, state its likely interests in the project and ways it could influence the conduct of the project and its outcomes.
2. Considering the project’s technical aspects (building tunnels, roadways, and bridges; demolishing the elevated structure and replacing it with parks) and its political, economic, environmental, and social impacts (and stakeholders for each), what characteristics (skills, background, competencies) would the “ideal” manager need to oversee a project of such scope and magnitude?

Notes

1. Sayles L. and Chandler M. *Managing Large Systems: Organizations for the Future*. New York: Harper & Row; 1971, p. 204.
2. Sayles and Chandler, *Managing Large Systems*, p. 212.
3. Portions adapted from Smith R. *The Carving of Mount Rushmore*. New York: Abbeville Press; 1985.
4. Smith, *The Carving of Mount Rushmore*, pp. 17–18.
5. Smith, *The Carving of Mount Rushmore*, p. 164.
6. Archibald R. *Managing High-Technology Programs and Projects*. New York: Wiley-Interscience; 1976, p. 35; Atkins W. Selecting a project manager. *Journal of Systems Management* October; 1980: 34; and Roman D. *Managing Projects: A Systems Approach*. New York: Elsevier; 1986, p. 419.

7. Lawrence P. and Lorsch J. *Organization and Environment: Managing Differentiation and Integration*. Boston: Graduate School of Business, Harvard University; 1967, Chapter 3.
8. These bases of interpersonal power were first described by French J. and Raven B. The bases of social power. Reprinted in Cartwright D. and Zander A. (eds.). *Group Dynamics*, 3rd edn. New York: Harper & Row; 1968, pp. 259–269.
9. Hodgetts R. Leadership techniques in the project organization. *Academy of Management Journal* 11; 1968: 211–219.
10. Rich B. and Janos L. *Skunk Works*. Boston: Little, Brown & Co; 1994.
11. Katz R. and Allen T. Project performance and the locus of influence in the R&D matrix. *Academy of Management Journal* 28(1); 1985: 67–87.
12. Archibald, Managing High-Technology Programs, p. 55.
13. Adams J., Barndt S., and Martin M. *Managing by Project Management*. Dayton: Universal Technology; 1979, p. 137.
14. Gaddis P. The project manager. *Harvard Business Review* May–June; 1959: 89–97.
15. Gaddis, The project manager, 95.
16. Roman, *Managing Projects*, pp. 439–440.
17. Cusumano M. and Selby R. *Microsoft Secrets*. New York: Free Press; 1995, pp. 105–106.
18. Kerzner H. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. New York: Van Nostrand Reinhold; 1979, p. 99.
19. An example is the movie *Heaven's Gate* where the director was allowed to virtually dominate the movie's producers. Scheduled for completion in 6 months at a cost of \$7.5 million, the production was released a year late and \$28 million over budget. The movie was a box office flop and helped clinch the demise of the film's underwriter, United Artists. From Bach S. *Final Cut*. New York: William Morrow; 1985.
20. Responsibilities for project engineers are described in Chase W. *Management of Systems Engineering*. New York: Wiley-Interscience; 1974, pp. 25–29.
21. According to Archibald, *Managing High-Technology Programs*, pp.124–128, 199.
22. Archibald, *Managing High-Technology Programs*, pp. 128–131.
23. Katz and Allen, Project performance and the locus of influence, pp. 83–84.
24. Cleland D. and King W. *Systems Analysis and Project Management*, 3rd edn. New York: McGraw-Hill; 1983, p. 358.
25. Cleland and King, *Systems Analysis and Project Management*, pp. 362–363.
26. See Schibi O. *Managing Stakeholder Expectations for Project Success*. Plantation, FL: J Ross Publishing; 2013; Roeder T. *Managing Project Stakeholders*. New York: Wiley; 2013.
27. Hughes T. *Rescuing Prometheus*. New York: Vintage Books; 1998, Chapter 5; Luberoff D., Altshuler A., and Baxter C. *Mega-Project: A Political History of Boston's Multibillion Dollar Artery/Tunnel Project*. Cambridge, MA: Taubman Center, John F. Kennedy School, Harvard University; 1993; http://www.bigdig.com/:http://lfmsdm.mit.edu/news_articles/sdm_business_trip_fall03/sdm_business_trip_fall03.html.
28. Figure 16.6 shows stakeholders prior to 1992. After that, Joint Venture accountability shifted from the Massachusetts DPW to the Massachusetts Highway Department; after 1997 is shifted to the Massachusetts Turnpike Authority (MTA). In 1998, Joint Venture and MTA formed an "integrated project office" to combine Joint Venture's management-consultant expertise with MTA's long-term dedication and specialized experience; Source: *Completing the "Big Dig": Managing the Final Stages of Boston's Central Artery/Tunnel Project*. National Academies Press, 2003; Chapter 5; http://books.nap.edu/openbook.php?record_id=10629&page=31; accessed May 8, 2007.
29. Klinger A., Belmonte D., Chou E., Phares C., Volman N., and Pina R. *The McCormick Place West Expansion Project*, Loyola University Chicago Report, February 2005.
30. Source: Hughes T. *Rescuing Prometheus*.

Chapter 17

Leadership, Teamwork, and Conflict

During the manned landings on the moon, researcher Richard Chapman conducted a study of NASA project management¹ This was during NASA's heyday—a period marked by extraordinary achievements and a time when NASA was upheld as exemplar of a large public agency that actually worked. It is instructive to begin this chapter with some of Chapman's observations about the project managers of that era, paraphrasing his comments:

Besides technical competency and management capacity, all agree that the project manager must have the ability to build a cohesive project team.

(p.93)

Those project managers who developed the most closely-knit project teams placed an emphasis on decentralized decision making and technical problem-solving at the level where both the problem and the most experience reside. They encouraged project members to feel a sense of responsibility for problem-solving at their respective levels, within the assigned guidelines.

(p.83)

Most project staff believed they received generous support and attention from the project manager, and most acknowledge that the project manager is vigorous and fair in bestowing recognition on team members and in rewarding them to the best of his capability.

(p.82)

In another study of NASA, E. H. Kroman compared the performance of two large projects, Lunar Orbiter and Surveyor. Lunar Orbiter was a success and fulfilled objectives within time and resource limits; Surveyor was less successful and experienced cost and schedule overruns. The study characterized customer/contractor organizations in Lunar Orbiter as tightly knit cohesive units, with good teamwork and mutual respect and trust among project counterparts. In contrast, teamwork in Surveyor was characterized as “slow and fitful” to grow and “spurred by a sense of anxiety and concern.”² Kroman concluded:

What emerges perhaps most forcefully from a broad retrospective view is the importance of the human aspects of organization and management. Both projects demonstrated the critical nature of human skills, interpersonal relations, compatibility between individual managers, and teamwork.

The fact that these studies were conducted some years ago shouldn't matter because numerous studies since then of project leadership and teams have concluded the same things: that behavioral issues such as leadership style, decentralized decision making, and teamwork are all crucial to project performance.³ Unfortunately, such matters are often overlooked in project management practice and education, possibly because inexperienced managers and specialists in the "hard" disciplines (technicians, engineers, and businesspeople) see them as "soft" and relatively inconsequential. But in reality, these are not soft; they are hard as nails and can profoundly impact project performance.

This chapter discusses issues broached by these studies: leadership, participative decision making, teamwork, and the related topics of conflict and emotional stress in work.

17.1 Leadership in Project Management

Leadership style

Chapter 15 described organizational forms suitable for different purposes and types of projects. Likewise, there are a variety of suitable leadership styles, depending on the situation. Leadership is the ability to influence the behavior of others to accomplish something desired; *leadership style* is the way a leader achieves that influence.

Leadership style can be categorized between the two extremes of **task-** or **goal-oriented** and **relations-oriented**. Task-oriented managers show higher concern for the goal and the work and tend to behave in a more dictatorial, bureaucratic fashion. The style is also referred to as *vertical leadership*, implying top-down influence whereby people simply do whatever they are told to do. Relations-oriented managers show greater concern for people and tend to behave more democratically. Close variants of this style are **shared leadership** and **distributed leadership**, meaning that team members are allowed to participate as needed in decision making, sharing accountability, helping each other, and performing duties ordinarily expected of the project manager.⁴ The project manager shares or distributes the leadership role among team members, seeking their advice and opinions, and, sometimes or always, delegating decision-making responsibility.

Numerous studies have attempted to discern the most effective leadership style. Most conclude that no one leadership style is best for all situations. Effectiveness of style depends upon characteristics of the leader, the followers, the leader's interpersonal relationship with followers, the type of project, and the nature and environment of the work. This perspective, called the **contingency** or situational approach to leadership, suggests that the leader should apply the style that best fits the situation and not use the same style for all employees and situations. The following section briefly describes two such approaches as conceived by researchers Fred Fiedler and Hersey and Blanchard.

Contingency and situational leadership

According to Fiedler, the three variables that most affect a leader's influence are whether (1) the work group accepts or rejects the leader, (2) the task is routine or complex, and (3) the

leader has high or low formal authority.⁵ Fiedler's research suggests that in most technical project situations a **relations-oriented** style is the most effective. The most prominent behavior in this style is the leader's positive emotional ties with and showing concern for his subordinates.

Hersey and Blanchard⁶ developed a model called **situational leadership** that weighs the interplay of three variables: (1) the amount of direction and guidance a leader gives (task behavior), (2) the amount of socio-emotional support he gives (relations behavior), and (3) the readiness of followers to perform the task (maturity). The last variable, "maturity," has two aspects: the followers' *skill or ability* to do something, and their *motivation or willingness* to do it. According to the model, the most effective leader behavior depends upon the maturity level of the followers. Project managers seldom manage unskilled laborers; more often they deal with technical specialists, managers, professionals, tradespeople, and other skilled people. Thus, they usually work with people who are either (1) able but perhaps unwilling to do what the manager wants, or (2) both able and willing to do what he wants. For Group (1) the model recommends a **participative style**, i.e., the leader facilitates, supports, and communicates with followers; and the leader shares decision making with them. For Group (2), the model recommends a **delegating style**, i.e., the leader identifies the problem or goal, then gives followers the responsibility for solving the problem and determining how to implement it.

Occasionally project managers encounter a Group (3)—people willing to work but relatively unable or unskilled (e.g. recent college graduates). For this group the model recommends the leader provide instruction and close supervision. This situation is a special case, however, for even when the project manager does provide instructions, he encourages followers to develop the capabilities necessary to enter the ranks of Group (2).

In researching the management of scientific and technical personnel, Hersey and Blanchard found that people with high-level education and experience respond well to participating and delegating leadership, and *do not* respond well to detailed directions and close supervision. Of course, this is not to say that project managers never face workers who are unwilling to follow instructions or take initiative. In cases where participation or delegation fails, a project manager with legal authority (e.g., a pure project manager) may need to cajole, give orders, and even terminate workers.

Other studies support these findings and conclude that in most project environments an engaging, participative style works best. Even in engineering and constructions where the leader needs to provide a strong sense of direction to the team, project managers tend to involve members in helping set that direction and in determining how best to achieve project goals.⁷

Project circumstances

Effective leadership style also depends on project circumstances. For example, a more directive style may be appropriate when there is pressure to complete the work quickly; in other words, sometimes the *pace* of work calls for a more directive leadership style, and the intensity of work serves as the motivator. Also, in a high-paced project there might be little time to build the trust necessary for a more participative style; this is sometimes the case when the project team involves subcontractors or a workforce that the project manager is unfamiliar or unaccustomed to working with. In such situations the project manager may need to be more directive and assertive. As in other regards, the project manager must be adaptable—able to wear different leadership-style hats and change them quickly.

17.2 Participative Management and Shared Leadership

The models of both Fiedler and Hersey and Blanchard offer similar conclusions about project leadership: the most effective style for project managers is a relations-oriented style—engaging, sharing, supportive, facilitative, and encouraging. Sometimes project managers must give orders or tell people what to do, but in most project situations participation and delegation work best, even when combined with task- or goal-oriented behavior. This means that usually project managers involve others in decision making, are supportive, and avoid dogmatic or intolerant behavior. Especially in projects with high potential for conflict, they invest considerable emotional energy in developing trust and interpersonal relationships.

This conclusion is further supported by research from large technical projects showing that the most effective leadership style is **participative management** (also called **shared** or **distributed leadership**). Managers in those projects seldom give orders to those they must influence, partly because most of these individuals are not subordinate to the project manager, partly because giving orders induces a negative “I won’t do it” or “Do it yourself” reaction, and partly because the “subordinates” are, after all, the experts in their fields. Project managers use participative management because, to an extent, they must. Although they have a good purview of the total project—its goals and constraints, they are usually farther removed from technical problems and less qualified to resolve those problems than the people who report to them.⁸

But participative project managers do not relinquish responsibility; depending on the situation, they delegate it and share it. Even when they have legal authority, they involve others by acquainting them with problems, consulting them for their opinions, and giving frequent feedback. This participation does not imply that everyone on the team participates equally and in everything (although sometimes, in agile teams for example, most everyone *does* participate in most everything). Throughout the project, the project manager must decide who is the most qualified and should be delegated authority; those individuals and their level of participation will sometimes change as needed, depending on the project phase or particular situation. People and situations vary, so the project manager must determine for each worker how much responsibility she can handle, and how much she needs to be monitored and directed.

Regardless of their level of participation in planning and delegated responsibility, project individuals and teams must still be directed to ensure their decisions and efforts keep moving the project toward its goals and within budget and schedule constraints. Lacking direction, groups and individuals might do things or make decisions that are detrimental to the project as a whole. The project manager, being the “big picture” person, must *steer* teams and individuals to meet project goals—even in participative, shared leadership situations.

Despite the evidence about effective leadership styles in projects, simply *telling* a project manager she needs to develop a participative, shared, relations- and task-oriented style is not enough. Old behavioral patterns remain, and new ones are hard to develop. Often, companies provide training in interpersonal skills and team building to help managers make the transition. Even with training, however, not everyone is able to change leadership style. The hope in training is that each leader, if so motivated, will at least know which way to try to steer his behavior.

A recent development in participative management is **collaborative project management**, CPM, which refers to the active involvement of dispersed project stakeholders as the source of innovation and creativity—a form of **stakeholder engagement**. CPM takes advantage of **social project management** software, **SPM**, discussed in Chapter 6, which is a form of social media to promote collaboration and eliminate silos throughout the project community. In theory, with increased collaboration, project managers can get better ideas and



See Chapter 6

more realistic insights from stakeholders. CPM and SPM give project managers a social reach beyond the project and provide new perspectives and ideas that stimulate team innovation and creativity.

In the words of Bennis and Nanus, the most effective leaders are able to “align” the energies of people and groups behind the goal. They lead by “pulling rather than by pushing; by inspiring rather than by ordering” and by creating achievable, challenging expectations and rewarding progress rather than by manipulating.⁹ The ample evidence, anecdotal and empirical, is that effective project managers are strong leaders who utilize participative management.

One function of leadership is to motivate behavior. Frederick Herzberg, a motivation researcher, theorized that work behavior rests on two factors called motivation and hygiene. The first includes responsibility, achievement, and recognition; these drive motivation and lead to satisfaction. The second includes salary, supervision, and work environment; these, *per se*, are not motivators but if inadequate can cause dissatisfaction.

The work in projects can itself be motivating—it can be stimulating, satisfying, and provide a great sense of achievement. Elements of project management are motivating too: contracts, project goals, schedules, and budgets, provide clear targets plus constant pressure that motivate people, especially when combined with financial and career rewards.

But project work includes de-motivators as well. Too much pressure leads to stress, tension, and conflict. On large jobs, people can lose sight of project goals. One advantage of participative management and shared leadership is that through engagement in planning and decision making, workers gain appreciation for plans and decisions, and feel more closely associated with the project and dedicated to its success.

17.3 Project Teams

Project organizations are comprised of groups. In a large project some of these groups are comprised of people from within one organization (shown in Figure 17.1, the project office, midlevel management team, and functional and cross-functional work package teams), while others come from multiple organizations (the project management or PM team and cross-organization functional team). Membership in many of these groups overlaps and people serve multiple roles that link the groups together.

The term **project team** as used here refers to any group or collection of groups that work in the project. The difference between a “group” and a “team” is that the former is simply a collection of people, whereas the latter is a collection working toward a *common goal*. Ideally, all work done in a project, mental and physical, is the product of teams; in other words, to be successful a project needs **teamwork**.

The trouble with teams

Project failure often can be traced to the inability of a team to make the right decisions, perform the right tasks, or perform the tasks right. These failures often stem from the maladies of teams: internal conflict; time wasted on irrelevant issues; and haphazardly made decisions. Teams often are more concerned with getting things *done* than with doing them *right*. Many teams never know what their *purpose* or goal is, so they never know when or if they have achieved it.

In projects with multiple teams, each might be oriented toward different goals. They might be in separate offices and physically isolated, which creates and reinforces perceived

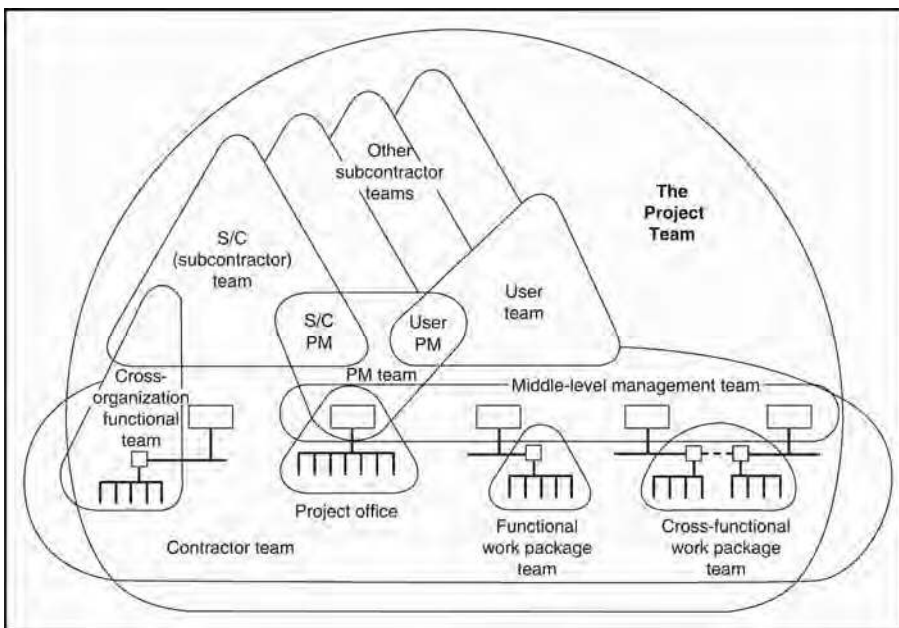


Figure 17.1
Groups comprising the project team.

boundaries and an “us versus them” attitude. These make for a portentous project environment that bodes ill for project success.

High-performing teams

In contrast, successful projects result from the efforts of *effective* teams—teams that succeed in achieving whatever they set out to do. What makes a team effective? Peter Vaill studied a large number of highly effective teams, teams that “perform at levels of excellence far beyond those of comparable systems.”¹⁰ The prominent feature he found among all effective teams is that they know and are committed to team goals. Members are never confused about why the team exists or what their individual roles are. Leaders inculcate belief in the team’s purpose, eliminate doubts, and embody a team spirit. He also found:

- Shared understanding and commitment to the team purpose.
- Teamwork focused on the task. Distinctions between functions dissolve and members work together to do whatever they must.
- Leadership is strong, clear, and never ambivalent. Leaders are reliable and predictable, regardless of style.
- The team views itself as distinct from others; members feel “we are different.”

Vaill found three characteristics *always* present in high-performing teams, which he calls **time**, **feeling** and **focus**. First, leaders and members are fully committed to the project and devote extraordinary amounts of time to it. They work at home, in the office, in taxis and Ubers—anywhere. Second, they feel very strongly about attaining the goal. They care deeply

about the team's purpose, history, future, and members. And third, they focus on key issues; they have a clear list of priorities in mind. Time, feeling, and focus are always found together. Vaill encourages would-be leaders to "Seek constantly to do what is right and what is needed (focus); do it in terms of your energy (time); put your whole psyche into it (feeling)."¹¹

For project managers, these findings underscore the importance of clear definition of and strong commitment to achieving project objectives, clarification of team members' roles and tasks, and a "project spirit" that bonds everyone together.

Example 17.1: Time, Feeling, and Focus in Project Management: Renovating the Statue of Liberty

The renovation of the Statue of Liberty is a good example of the kind of commitment and effort required to successfully manage a big project.¹² Over 25 firms submitted proposals for the task of leading the team of 500 engineers, architects, artisans, and craftsmen that would do the renovation. Selected for the job was the small construction management firm of Lehrer/McGovern, Inc.

Hofer describes the firm's two partners as being very different—Lehrer is soft-spoken and conservative in appearance, McGovern clean-shaves his head, and wears cowboy boots—but sharing similar goals and broad experience as civil engineers and construction managers.

Did they devote a lot of time to the project? To coordinate the more than 50 businesses doing the job, Lehrer and McGovern often worked 16-hour days. They handled everything from helping architects and craftsmen implement plans, to arranging with subcontractors and ensuring materials were ordered and delivered on time.

Did they instill feeling for the project? Said Lehrer, "this project is a labor of love. The spirit and pride of hundreds of men and women involved bring out the best of us as Americans."¹³ They expected and inspired feelings like that from everyone else, too. They only hired people who had "the same commitment and dedication as we do, who are aggressive and ambitious and understand that virtually nothing is impossible."¹⁴ Before beginning the job they lectured each subcontractor that nothing be allowed to damage the "crown jewel of the United States."

Did they maintain focus? Their major emphasis was on top-quality work. The two partners believed that management's close and personal involvement was crucial to quality, so they made frequent visits to the site and personally supervised or handled thousands of details.

This was an exceptional project, highly publicized and faced with considerable political pressure; but many projects bomb, despite high pressure and publicity. In this case, management's time, feeling, and focus helped the project succeed.

17.4 Team Building

Project work requires close collaboration. People in project teams must rely on and accept one another's judgments and support each other—in other words, they trust each other. Managers and team members must share information and consult with each other to make decisions. Every person and group must be committed to project objectives, not just their own, and feel a shared ownership for the outcomes. Most studies show a correlation between teamwork and

project success. In general, greater team collaboration, cohesion, and communication correspond to better project outcomes.¹⁵

One way to increase collaboration and commitment is by locating everyone in the project team in the same office quarters. Frequent daily contact increases the tendency for people to identify with the team and project goals.

But even if co-locating team members is possible, which often it isn't, proximity alone will not guarantee an effective team. Vaill's findings show that effective teams are clear about their purpose, committed to it, know their individual roles, and understand how to work together. In many projects however, especially where people have not previously worked together, team members do not know the team's goals or their own responsibilities, and they never learn to work together. The purpose of team building is to ensure that doesn't happen.

Team-building procedure

In a study of two NASA research centers, 36 project managers were asked to rank the most important functions of their job. Ranked as either first or second by all managers were the functions of organizing, directing, and motivating the project team and supporting groups.¹⁶ In another study involving 32 research and product development projects, group cohesiveness was identified as the single most important factor to achieving project goals.¹⁷

Group cohesiveness and effectiveness do not just happen. Like any purposeful system, a team or organization must be developed. This is the purpose of **team building**, a procedure whereby a team formally addresses how it should work or has been working, with the goal of improving its effectiveness. Team building considers group process issues, which are the processes or methods by which it gets things done. The issues relate to decision making, problem solving, team objectives, internal conflict, and communication. Effective groups recognize and monitor these issues. During team building, the team explores such issues and then plans how it will address them and perform its work.

When it is needed

The need for team building depends on the team and the nature of the task. Generally, the more varied the backgrounds and responsibilities of team members, the greater the need. For example, members of multidisciplinary teams have different work backgrounds and outlooks on planning and doing work; some take a wider perspective, others are detail oriented. Team building can help both types accept their differences and work toward common goals.

Projects involving innovation, new technology, high risks, and tight schedules place teams under heavy stress. Some stress will motivate a team, but too much is detrimental. Team building can help the team to deal with stress and to resolve problems as they occur, before they escalate and interfere with team performance.

Aspects of team-building efforts

The purpose of team building is to improve a group's ability to work together. To this end, the approach strives to build norms such as:

1. Effective communication among members.
2. Effective resolution of group process issues.

3. Constructive resolution of conflict.
4. High-level collaboration among team members.
5. A trusting, supportive atmosphere within the group.
6. Clarification of the team's purpose and the role of each member.

Three features common to any team-building effort are:

- It is carefully planned and facilitated, often by an outside party—a consultant or staff person from human relations or the PMO.
- The outside party collects data about the team's process functioning in advance, then helps the team “work through” the data during a diagnostic/problem-solving workshop.
- The team plans for later self-evaluation and follow-up.

Following are examples of team building as applied to three situations: an experienced work team, a new team, and multiple teams that must work together.

17.5 Improving Ongoing Work Teams

Consider team building as applied to an existing team such as a cross-functional management team, design-build team, Scrum team, or team of clients, contractors, and subcontractors. Problems typical to such teams include inability to reach agreement, lack of innovative ideas, too much conflict, or complacency of team members. A formal teambuilding process such as summarized in Figure 17.2 can help avoid or overcome these problems.

Initially a human relations consultant or other person with facilitation skills is called in by the project manager or PMO director to facilitate the effort. Her function is to help the group solve its own problems by drawing attention to the way the group's behavior is affecting its decisions and performance.

The consultant collects data from members using personal interviews or questionnaires. She then summarizes the data, keeping the sources of comments anonymous. This summary will later be presented to the entire team at an upcoming workshop.

The consultant first shares the results with the team leader (project or department manager, work package supervisor, etc.) and coaches him on how to prepare for the workshop. The consultant remains impartial: the *entire team* is her client.

At the workshop, members review the summary and analyze the group's problems. This workshop differs from ordinary staff meetings in many ways. It convenes at an off-site location away from interruptions, can last up to several days, and includes all team members.

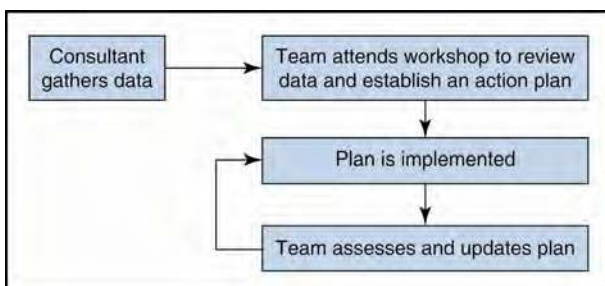


Figure 17.2
The team-building process.

The atmosphere is open and candid, without the usual superior–subordinate restrictions. The consultant facilitates the workshop.

The workshop specifics vary. One common format is this:¹⁸

1. The workshop begins with a discussion of the agenda. Team members describe what they would like and do not want to happen.
2. The consultant posts a summary of the information collected on the wall for easy reference. Discussion may be necessary to make sure everyone understands the issues. The consultant may also post anonymous quotes from the interviews, for example:

“Even though the team leader asks for our opinions, I know she ignores them.”

“Our team has no scheme for how to fit new tasks into the existing workload.”

“There is nothing to distinguish the roles of engineers and researchers in this project.”

3. The team prioritizes the problems it wants to resolve within the time constraint of the workshop.
4. The team works to resolve the priority issues. In the meantime:
 - a. The consultant monitors the session and points out dysfunctional behaviors of the group, encourages members to express their feelings, confronts behaviors that lead to defensiveness or distrust, and reinforces effective behavior.
 - b. The group periodically critiques itself. After working through a problem, it pauses to evaluate what helped or hindered the process.
 - c. The group prepares a formal action plan with solutions, target dates, and people responsible. The plan may include “operating guidelines” specifying how the group will function. (Typical guidelines are described in the next section.)

One of the authors has observed the success of workshops like this in effectively resolving team problems ranging from technical issues to interpersonal conflict.

To ensure that action steps are implemented, follow-up work is scheduled formally in a session to be convened 2 to 3 months later or less formally during regular meetings. The team takes stock of its functioning, improvements it has made, and what is still needed. The group itself takes over the consultant’s role; in case new problems emerge, it repeats the process.

Two conditions are necessary for team-building success. First, the team leader and upper managers must accept the issues uncovered and assist in (or provide resources for) working toward solutions. Second, team members must want to resolve the group’s problems. They must be open and honest in providing information, willing to share in the responsibility for having caused problems, and willing to work toward solutions.

17.6 Building New Teams

Commonly, people in new teams quickly develop interpersonal bonds based on attributes such as similar age, gender, or nationality. Unfortunately, such bonds can be superficial and harmful to team unity and performance; what is better for team cohesion and performance is that they develop bonds around shared skills, competencies, and tasks. Thus, early team-building efforts should provide members an opportunity to work together on tasks related to project goals, and to develop competency- or task-oriented relationships.¹⁹

The purpose of team building for a newly formed team is for the team to reach agreement on its purpose, how it will achieve that purpose, and the roles of its members. The team also addresses how its members will work together in a manner so as to effectively accomplish its purpose and leave everyone feeling good about it.

A team-building workshop is convened by a facilitator during which the members become acquainted, reach agreement on objectives, and decide how they will function as a team. In *Team Building: Issues and Alternatives*, William Dyer describes the agenda of such a workshop, as follows:²⁰

Step 1: Develop a priority level

Members of a team sometimes differ in the priority they place on the project goal or work tasks. Especially in ad hoc teams or task forces with part-time members, some members give the project high priority, others low. One way to acknowledge these differences is for each member to indicate on a scale of 0 to 10 the priority of the project compared to her other work. Another way is to ask each one to indicate the amount of time they can devote to the project each day or week. The information is tallied and posted on a chart similar to Figure 17.3. The team members discuss the differences in their commitments to the project and individuals are invited to explain their positions on the chart.

Step 2: Share expectations

Each person is asked to think about the following: (1) What would this team be like if everything worked ideally? (2) What would it be like if everything went wrong? (3) In general, what kinds of problems occur in work groups? and (4) What actions should be taken to make this an effective team? The responses are shared verbally and then posted. Concerns are discussed. These will be worked through later in Step 4.

Step 3: Clarify purpose and objectives

The team discusses and records its purpose and objectives. The purpose and objectives should be clearly defined and accepted by everyone. The team then develops sub-objectives so that members can be given specific assignments. The objectives should complement any user and system objectives and requirements as defined in the SOW or charter. In fact, a session like this can be used to create the SOW or charter.

Step 4: Formulate operating guidelines

Group dysfunction often arises over mixed expectations about how the group ought to work. This can be avoided by the team establishing **operating guidelines** that address, for example:

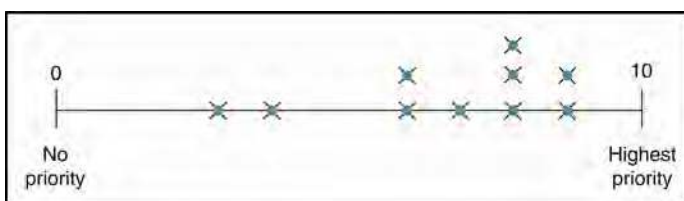


Figure 17.3
Priority ranking of
project for 10 team
members.

1. *How will the team make decisions*—by dictate of the leader, by vote, by consensus, or by other means? Who should be involved in making decisions? This will depend on factors such as size of project, number of team members, complexity of tasks, and so on.
2. *How will the team resolve differences among members and subgroups?* Disagreements waste a lot of time, so guidelines should address the kinds of conflicts likely to arise and options for resolving them—consensus, vote, or calling in a mediator.
3. *How will work be assigned?* Which tasks should be handled by the whole group, which by subgroups or individuals? Should tasks be assigned according to expertise, position of authority, or personal preference? If several people want to do a task, how should they be chosen—by skill, experience, or volunteer?
4. *How will the team ensure that work is completed?* One person falling behind can delay the work of others. How will the team ensure that assignments and completion dates are clear, and that corrective action is taken when efforts lag or are out of control? Who will assist if someone falls behind? How will the team handle slackers?
5. *How will the team ensure open discussion?* The team must ensure that members are able to openly discuss issues so that ideas are not ignored or suppressed, and everyone is heard. How will people less inclined to speak up because of personality, language, or culture be kept engaged? How will loquacious people be quieted?
6. *How frequently and where will the team meet?* Who will be expected to attend? How will absent people be informed about what happened at meetings?
7. *How will the team evaluate its process and any needed changes?* The team specifies a procedure for periodically reviewing whether the above guidelines are working or should be changed. Some teams appoint one member each meeting the role of making sure the team conforms to this guideline.

The teams might also discuss the roles and responsibilities of its members and identify any ambiguity, overlap, or conflict. The results of the team building session—the team's purpose, expectations, and guidelines, are summarized in a **project team charter**.

A new team does not have to wait for problems to arise before it takes action. Through team building, it can develop expectations and guidelines to prevent common group problems.

Disbanding teams

Once the team has completed its work, a formal process should be followed to disband the team. Successful teams generate close ties and strong relationships; when the project ends, people are usually reluctant to abandon relationships and may actually suffer feelings of loss. These feelings should be acknowledged and shared. The closeout of the project may be followed by a ceremony—a banquet, party, or informal get-together—to recognize the team for its accomplishments and say goodbye.

17.7 Intergroup Problem Solving

When several teams must work together, issues among them arise such as communicating or withholding information, competition among them, or coordinating their efforts. **Intergroup problem solving (IGPS)** is a technique for improving working relationships among multiple teams; following is an example of how it works.²¹

The two groups meet together for a day. At that time:

1. Each group compiles four lists: (1) what they believe the other group is responsible for; (2) what they feel are the other group's strengths and weaknesses; (3) what the group thinks are its own responsibilities; and (4) what the group anticipates the other group thinks about them (strengths, weaknesses, responsibilities).
2. The groups meet to share their lists. The only discussion allowed is to clarify points of disagreement.
3. The groups separate, this time to address what they learned from the other group's lists and to list and prioritize the issues they think need to be resolved.
4. Finally, the groups meet together again to discuss their differences and develop a mutual plan to resolve them.

The groups meet again a few weeks later at a follow-up session to assess how well their plan is working. The result is usually a much-improved understanding of each group's expectations about the other and a better working relationship.

IGPS is applied whenever groups interface or must work together. Examples are project teams comprised of groups from different contractors. Without IGPS, each group often tries to optimize its own goals rather than the overall goals of the project or program. IGPS is useful whenever there are interdependencies, deadlines, or situations that induce intergroup conflict and stress.

Participants in an intergroup session are likely to have a "gee whiz" experience. Each group may discover that its expectations differ significantly from (and conflict with) those of other groups. This realization is a first and necessary step to aligning expectations and planning to resolve differences.

One caveat is that groups should not participate in IGPS until they first resolve any serious internal problems. In other words, a group must first have its own house in order (team-build itself) before it attempts to resolve its issues with other groups.

17.8 Virtual Teams

Sometimes everyone on the project team is located in a different place. Such teams, called **virtual** or **distributed teams**, are common in design and development projects where specialists are located around the globe. On occasion the project manager or team members might meet with each other face-to-face, but often that never happens and their interaction is solely via communication technology.

Although many of the leadership and team-building principles previously described still apply, managing virtual teams requires special consideration. People cannot walk across the hall to ask questions or call a meeting on a whim; they must rely upon technology, which makes it more difficult to make decisions, follow up on commitments, monitor results, and build team relationships. Everything seems to get worse when the team is spread among different time zones, languages, and cultures.

Communication technology²²

When travel budgets are meager, most communication happens by virtue of technology. Among available technology options, all fall under the heading of "groupware," which is software to facilitate people working together on a common task.

Groupware that emulates face-to-face meetings and enables people to talk continuously and simultaneously is called **synchronous**; it includes:

- Desktop, real-time data conferencing
- Electronic meeting systems
- Videoconferencing
- Audioconferencing
- Instant messaging (IM).

Groupware that permits only intermittent, back-and-forth communication is called **asynchronous**; this includes:

- Email
- Personal computing devices
- Group calendars and schedules
- Bulletin boards
- Team websites.

The appropriate technology depends on the task. Synchronous technologies such as desktop conferencing are “media-rich” and mimic normal conversation, and they should be used in addressing ambiguous or challenging tasks and making decisions. Asynchronous technologies such as email are not media rich and their use should be confined to sharing information and documentation. In general, virtual teams share lots of documentation because, in the absence of face-to-face meetings, writing replaces conversation and virtual teams *need to write more*. Even audio conferences and old-fashioned phone calls should be followed up in writing to ensure clarity.

The technology to be used must be compatible with the hardware/software at the different team members’ sites. Also, members must be comfortable with using the technology and have access to training.

Cohesive teams²³

In a cohesive team, everyone shares a vision and trusts one another. Among ways the project manager builds a shared vision are to:

- Explain to the team the importance of the project and each member’s contribution. In an international team, the project manager might have to travel to every site to do this.
- Negotiate and clarify everyone’s roles, responsibilities, and accountabilities.
- Identify specific results-oriented performance measures for each member; the measures must be specific enough so the project manager can gauge each member’s performance.
- Develop methods to review progress. This might require weekly audio-conference reviews.
- Establish communication protocols regarding:
 - preferred communication modes—email, voice, IM, texting, etc.
 - acceptable elapsed time in responding to messages
 - best time of day to call or schedule meetings
 - times when people are in the office
 - non-office times acceptable to call or meet.

- Create team operating guidelines for decision making, conflict resolution, etc., and include them in the team charter. If the team can meet face-to-face at least once, this can be done using the procedure in Section 17.6. The leader must expect and reinforce compliance with the guidelines.

Trust²⁴

Team cohesion depends on trust among the project manager and team members. The best all-around way to build trust is through *face-to-face contact*. A virtual team cannot do that regularly, although it might do it occasionally. For example:

- Team members are allowed and encouraged to visit each other's sites on occasion.
- Representatives from sub-teams at different sites *float* among and spend brief periods at other project sites.
- The entire team meets once for a project kickoff meeting.

Regardless of the alternative, the visits/meetings must be long enough (several days or weeks) for people to get to "know" each other and develop personal bonds; this is more the purpose of the visits than doing work. Ideally, face-to-face visits/meetings happen at the start of the project; if that isn't feasible, they should happen whenever it is. At minimum, the project manager should try to meet with everyone at least once in person and, ideally, more often. Cohen calls this "management by flying around"—the virtual-team equivalent of management by walking around. He says to expect the travel budget to increase with virtual teams, not decrease!²⁵

In general, trust develops when people see others performing competently, acting with integrity, and showing concern for others' well-being; it erodes when they doubt each other or the leader. It is important that everyone receives critical information at the same time, else individuals might perceive they are being excluded or forgotten by the leader or others.

In virtual teams, it is difficult to observe physical performance, so even a little negative information can destroy an individual's or team's reputation. Information suggesting poor performance should never be accepted without first investigating it.

Virtual meetings²⁶

Managing meetings of a virtual team raises special problems. Duarte and Snyder recommend the following.

- **Participation.** Not everyone needs or has time to attend all meetings. The project manager must decide for each meeting which team member's attendance is mandatory and which is optional. Explain to people why they were not invited, and when or if they will get the results of the meeting. Store important documents in a web folder so people not at meetings can stay up to date.
- **Preparation.** Distribute the meeting agenda beforehand so people can prepare. Explain which people will be expected to contribute, how much (a little or a lot), and in what ways. Try to get people's reactions on issues and answers to questions in advance of the meeting.
- **During the meeting.** Allow time at the start for small talk and chit-chat. At voice conferences always preface talk by announcing who will be talking. Cohen suggests at voice

conferences appointing one person at each location with an ear for voices to hold up a photo of whoever is speaking on the other end. People get accustomed to this and actually look at the photo as if the photo were doing the talking!²⁷

- **Be inclusive and enforce participation.** Greet each member or ask him to introduce himself. Make sure everyone is heard; ask everyone to participate and call on people who have not spoken. Be culturally sensitive so as not to put anyone on the spot. Seek diverse points of view. Practice communicating in ways that lead to trust: show respect, use names people prefer to be called, create dialogue and listen attentively. Be forgiving when someone makes a mistake. Provide breaks for members to organize and collect their thoughts.
- **Pace the meeting.** Guide discussion toward a resolution or postponement; remind people of remaining time. Set a time allotment for each item; ask if the team wants to extend the meeting.
- **Summarize.** At the end, summarize the discussion and make sure decisions or actions are recorded. Get commitments about who will do what. Try to make the meeting minutes available to everyone within a few days after the meeting. Be careful to make sure the minutes (and interpretations of what was said) are correct.

17.9 Conflict

In all organizations, differences in objectives, interests, expectations, and values lead to conflict. Projects are no exception and, if anything, are predisposed to conflict. Conflicts arise between customers and contractors, project and functional groups, and subcontractors and contractor. Conflict occurs between people on the same team, different teams in the same organization, and teams in different organizations. And it is common in virtual teams where electronic communication media can amplify misunderstandings and make it more difficult to build trust. Some conflict is natural and beneficial; too much is destructive.

Between user and contractor

Seeds of customer–contractor conflict are sown early during contract negotiations. People representing the two parties are usually less concerned with developing trust than with driving a hard bargain for their own best interests. The customer wants to minimize cost, the contractor wants to maximize profit. One’s gain is the other’s loss. In the extreme, each side strives for an agreement that provides an “out” in case it cannot keep its part of the bargain; each tries to make the other responsible in case of failure. Says one manager,

You start with science and engineering, but the project, once it’s decided on, has to be costed. You have to select contractors and get budgets approved. Then you turn to the contractors working with you and write contracts that say you don’t trust them. What starts as a fine scientific dream ends up being a mass of slippery eels.²⁸

The contract itself becomes a source of conflict. A cost-plus agreement might provide little incentive for the contractor to control expenses, and the customer must closely supervise everything, which the contractor finds irritating. In a fixed-price contract, the contractor might request periodic upward revisions, to which the customer might disagree. Any contract wherein the terms for cost, schedule, and performance are poorly-specified will have multiple interpretations and lead to disagreements.

Within the project organization

High-level interdependency in projects between functional areas increases the amount of contact between them and the chances of conflict. Different areas have different ideas, goals, and solutions for similar problems—differences that sometimes must be resolved without the benefit of a common superior.

In addition, the functional areas' needs are often incompatible with the project's needs, and functional areas often request changes to the project plan that the project manager must refuse. The project manager might have to balance the technical standards of the functional departments with project time and cost considerations. Even when project managers agree with the technical judgment of specialists, they sometimes disagree over the means of implementation.

In matrix organizations, functional managers sometimes see project managers as impinging on their territory, and they resent having to share planning and control with them. They might refuse to assign certain personnel to projects or try to retain authority over personnel they do assign. Workers with dual-reporting relationships often feel conflicted over priorities and loyalties.

Finally, projects inherit feuds that have nothing to do with them. Regardless of the setting, clashes arise from differences in attitudes, personal goals, and individual traits, and from people trying to advance their careers. These create a history of antagonisms that set the stage for conflict even before a project begins.

Project life cycle and common conflicts

Thamhain and Wilemon²⁹ investigated sources of conflict in a study that involved 100 project managers. They determined that the three greatest sources of conflict are schedules, project priorities, and the workforce—all areas over which project managers generally have only limited control. Other sources of conflict identified are technical opinions and performance tradeoffs, administrative and organizational issues, interpersonal differences, and costs. Costs are a relatively minor cause of conflict, the authors surmise, not because they are unimportant but because they are difficult to control and usually dealt with incrementally over a project's life.

They also found that the sources of conflict change from one phase to the next, as summarized in Figure 17.4.

Project life cycle			
Start			Finish
<i>Project conception</i>	<i>Project definition</i>	<i>Project execution</i>	<i>Project closeout</i>
Priorities	Priorities	Schedules	Schedules
Procedures	Schedules	Technical	Personality
Schedules	Procedures	Manpower	Manpower
Manpower	Technical	Priorities	Priorities

Figure 17.4

Major sources of conflict during the project life cycle.

Source: Adapted from H. Thamhain and D. Wilemon, *Conflict Management in Project Life Cycles*, *Sloan Management Review* (Spring 1975): 31–50.

During project conception, the most significant sources of conflict are priorities, administrative procedures, schedules, and labor. Disputes between project and functional areas arise over the relative importance of the project compared to other activities, the amount of control the project manager should have, the personnel to be assigned, and scheduling the project into existing workloads.

During project definition, the chief source of conflict remains priorities, followed by schedules, procedures, and technical issues. Priority conflicts carry over from the previous phase, but new disputes arise over the enforcement of schedules and functional departments' efforts to meet technical requirements.

During execution, friction arises over schedule slippages, technical problems, and labor issues. Deadlines may become difficult to meet because of accumulating schedule slippages. Efforts aimed at system integration, technical performance of subsystems, quality control, and reliability also encounter problems. Manpower requirements grow to a maximum and strain the available pool of workers.

During closeout, schedules remain the biggest source of conflict as accumulated slippages make it difficult to meet target completion date. Pressures to meet objectives and anxiety over future projects increase tensions and personality-related conflicts. The phasing in of new projects and the absorption of personnel back into functional areas create further conflicts.

Conflict consequences

Conflict is inevitable in human endeavors and is not always detrimental. Properly managed, a certain amount of conflict is good because it: ³⁰

1. Compels people to search for new approaches.
2. Causes persistent problems to surface and be dealt with.
3. Forces people to clarify their views.
4. Stimulates interest and creativity.
5. Gives people the opportunity to test their capacities.

In fact, total absence of conflict is unhealthy. Called **groupthink**, wherein everyone goes along with the group, it is a sign of over-conformity. It causes dullness and sameness and results in poor or mediocre judgment. In contrast, some amount of conflict over differences in opinion stimulates discussion and can enhance problem solving. In project groups charged with exploring new ideas or solving complex problems, some conflict is essential.

Conflict between groups that are in competition can be beneficial because it increases group cohesion, spirit, loyalty, and the intensity of competition. However, conflict between teams that should be cooperating can be devastating. Each group develops an "us versus them" attitude and selfishly strives to achieve its own objectives. Left uncontrolled and unresolved, conflict spirals upward and creates hostility. Within a project, conflict fosters lack of respect and trust, and destroys communication between groups and individuals. Ideas, opinions, or suggestions of others are rejected or discredited. Project spirit breaks down and the project organization splinters apart.

Example 17.2: Conflict in Product Development Teams³¹

Microsoft forms small teams around products and then allows them to organize and work as they wish. It hires bright, aggressive people right out of school, then pushes them hard to get the most and best out of them.

As author Fred Moody describes, each product team consists of designers whose assignment is to try to add features to the product; developers whose partial role is to resist the features for the sake of meeting deadlines; and a program manager whose role is to mediate and render verdicts. Besides having different assignments and goals, there is a big chasm between developers and designers in terms of temperament, interests, and styles. Developers often feel it is impossible to make designers understand even the simplest elements of a programming problem. Designers might spend weeks on some aspect of a product, only to be rudely told by a developer that it will be impossible to implement. Says Moody, designers are from the arts, developers from math and science. Designers tend to be female, vegetarians, and talkative; developers tend to be male, eat fast food, and not say much. The way they deal with conflict also differs. Developers are given to bursts of mischievous play and will pepper a designer's door with shots from a Nerf-ball gun. Designers merely complain to their supervisor.

This adversarial relationship levies a toll on the team, the product, the customers, and the company. Moody quotes the lead developer on one project, who said, "I've never been through anything like this. We made the same mistakes before, and now we're making them again. Every project is like this. We keep saying that we learn from our mistakes, but we keep going through the same [expletive] over and over again."

17.10 Conflict Management

Conflict management refers to making conscious choices for dealing with conflicts. In general, there are six ways:

1. Withdraw or retreat from the conflict.
2. Smooth over or de-emphasize the importance of the conflict (pretend it does not exist).
3. Force the issue by exerting power.
4. Compromise or bargain to bring at least some degree of satisfaction to all parties.
5. Collaborate to accommodate different viewpoints and reach consensus.
6. Confront the conflict directly; work through the conflict with problem solving.

All of these are at times appropriate. In a heated argument it may be best to withdraw until emotions have calmed down, or to de-emphasize the disagreement before it gets distorted out of proportion. But neither of these resolves the problem, which will likely arise again. A manager might force the issue by using authority; this gets the action done but risks creating hostility. As discussed earlier, if authority must be used, it is better that it is based upon knowledge or expertise. To bargain or compromise, both sides must be willing to give up something

to get something and, ultimately, they may feel they lost more than they gained. Of the six approaches, the only one that works at resolving the underlying issues is *confrontation*.

Confrontation

Confrontation involves identifying potential or existing problems, then facing up to them. At the organization level this happens by all areas involved in the project agreeing on project objectives, plans, labor requirements, and priorities. It requires careful monitoring of schedules, close contact between project groups, and prompt resolution of technical problems.³²

At the individual level, a project manager confronts conflicts by raising questions and challenges such as:³³

“How do you know this redesign will solve the problem? Prove it to me.”

“What have you done to correct the malfunctions that showed up on the test we agreed to?”

“How do you expect to catch up on lost time when you haven’t scheduled overtime?”

Questions like these demonstrate that the project manager is vitally interested and alert, and that everything is subject to question. It is a crucial part of effective project management.

However, there is a catch: The very *process* of being confrontational is itself a source of conflict, but at the *interpersonal* level. Frequently, what begins as a conflict of schedules, priorities, or technical matters degenerates into a conflict over “personalities.”

Successful confrontation assumes a lot about the individuals and groups involved. It assumes that they are willing to reveal why they favor a given course of action, and that they are open to and not hostile toward differing opinions. It assumes that they are all working toward a common goal and are willing to abandon one position in favor of another.

The simple fact is, many groups and managers are highly critical of others’ opinions. Faced with differences, they tend to operate emotionally, not analytically. For individuals to use confrontation as a way to resolve conflict, they must first be able to manage their emotions, an aspect of emotional intelligence as discussed in Chapter 16.



See Chapter 16

Role clarification technique³⁴

Conflict in projects often arises because people have mixed expectations about work plans, roles, and responsibilities. Disagreements arise because:

- Everyone thinks someone else is handling a situation that, really, no one is.
- The project is new, and people are not clear about what they are supposed to do and what others expect of them.
- Changes in projects and work reassignments have made it unclear how individuals in the team should interact.
- People get requests they do not understand or hear about things on the grapevine that they think they should already know.
- People do not understand what their group or other groups are doing.



See Chapter 6

While the **responsibility matrix**, or RACI, described in Chapter 6, addresses the first and, possibly, second source of conflict, the **role clarification technique (RCT)** can help resolve the others. As the term “role clarification” suggests, the goal is that everyone understands their own and other’s responsibilities and duties, and that everyone knows what others expect of them.

RCT is like team building. It includes data collection, a day-long meeting, and a consultant who serves as facilitator. When incorporated as part of team building for a new team, it allows the project manager and team to negotiate team member roles. It is especially useful in cases where responsibilities are somewhat ambiguous.

The technique as applied to an existing team begins with each person answering a questionnaire prior to a meeting:³⁵

1. What are the expectations for your position?
2. Do your actions and responsibilities meet the expectations for your position?
3. What should others know about your job that would help them?
4. What do you need to know about others' jobs that would help you?
5. What difficulties do you experience with others?
6. What changes in the organization or activities would improve the group's work?

At the start of the meeting, ground rules are announced: people must be candid, give honest responses, and express their concerns, and everyone must agree to decisions. The meeting begins with each person reading the answers to the first three questions. As each person reads, others are given the chance to respond. It is important that each person hears how others see their job and what they expect of them.

Each person then reads the answer to Question 4 and hears responses from the people she identified. Issues in Question 5 that have not already been resolved are addressed next. Throughout the process, the emphasis is on solving problems, not placing blame. The group then discusses Question 6 and tries to reach consensus about needed changes.

17.11 Change Management

It is human nature that people have trouble adapting to change, which is another source of conflict in projects. Every project represents a change in that it (1) creates an end-item that did not exist before (e.g., new highway, vaccine, app), (2) replaces or modifies something that did (electric versus gas vehicles), or (3) creates a new process (new ways to purchase products). Often such changes are welcomed, but just as often they are unwanted or resisted, which causes conflict, stress, and sometimes project failure. While many people leap at opportunities to embrace recent technology and engage in new ways to do things, others see problems or drawbacks. People feel their lives, livelihoods, relationships, and values are under attack. It is estimated that over half of all organizational change efforts result in failure due to cultural resistance to change. Thus, in no small ways does project success depend on reducing resistance and gaining **buy in** from stakeholders. This is the purpose of **change management**, which is the process of transitioning organizations and individuals from one situation to another. Important to note is that change management is not the same as **change control**: whereas the purpose of the former is to facilitate acceptance of and support for changes that will result from a project, the purpose of the latter is to minimize unnecessary or undesirable changes made to the project or its outcomes. Both contribute to project success.

There are many models for change management, starting with Kurt Lewin's **three-step model** of unfreezing, changing, and refreezing.³⁶ The model suggests that people are accustomed to the status quo and that their attitudes and perceptions are "frozen." Change begins with unfreezing—when their attitudes are acknowledged and challenged, and false perceptions are corrected. In a project, these attitudes and perceptions are exposed through talks with individual stakeholders or surveys, focus groups, team-building sessions, public forums, or training seminars, and when stakeholders learn compelling reasons for the change and the

benefits it will provide. Generally, support for a change rises and resistance diminishes when people become involved in the change, and their needs and worries are acknowledged. The process is facilitated by change leaders—well-respected figures, “project champions,” or coalitions of influential people—who convincingly argue the merits of the change and can remove obstacles.

Refreezing indicates that the change is sustained—it is accepted and supported. In an organization, change is incorporated into the culture. People receive ongoing encouragement and rewards for adapting to the change, and communication about its benefits.

Lewin’s three-step process can be incorporated into the systems development life cycle:

- **Conception and definition (unfreeze):** Formulate the need for the change, understand stakeholders’ concerns about the change, and prepare a plan to transition to the change.
- **Execution (change):** Implement the planned changes; involve stakeholders and adjust the implementation process to address their suggestions and worries.
- **Operation (refreeze):** Integrate the change into everyday life; reinforce behavior that supports the change. This happens after the project is completed, but measures to sustain the change should be incorporated in the project plan and addressed throughout the project.

Another change management model is Jeffrey Hiatt’s ADKAR, which represents the steps in the process of aiding individuals in adapting to change.³⁷

- **Awareness:** Create awareness of the change by announcing it and the reasons for it.
- **Desire:** Build desire for the change by convincingly explaining its benefits and the negative consequences of not making the change; acknowledge people’s worries and try to mitigate them.
- **Knowledge:** Provide the requisite knowledge by training people to be able to make the change and to use or take advantage of it; identify the necessary behaviors and provide the necessary skills through training, coaching, mentoring, and workshops.
- **Ability:** Provide the ability to make the change through ongoing assistance track and control the change.
- **Reinforcement:** Support the change through encouragement and recognize and reward behavior that supports the change. Provide feedback and positive reinforcement.

Although ADKAR focuses on individual change, it is relevant to projects given that groups and organizations will not change unless the individuals within them change.

17.12 Managing Emotional Stress³⁸

Project work can be stressful. Long hours, tight schedules, high risks, and high stakes take a toll on social relationships and individuals’ mental and physical health. Projects achieve great things, but they also instigate ulcers, divorces, mental breakdowns, and heart attacks. Emotional stress can affect everyone in the project and is a problem that at one time or another most project managers face.

Factors influencing stress

How much emotional stress a person experiences and how well he deals with it depends on the fit between two factors, the demands of the environment and the adaptive capabilities

of the individual. In other words, work-related stress depends upon a person's perception of the demands or opportunities of the job and his self-perceived abilities, confidence, and motivation to perform. A manager faced with impending failure to meet a deadline might feel stressed if he believes the deadline must be met at all costs, but feel no stress if he simply accepts that meeting the deadline is impossible. Stress is a reaction to prolonged internal and environmental conditions that overtax a person's adaptive capabilities. To feel distressed (negative stress), an individual's capabilities must be overtaxed. Even when a person is able to handle a situation, he will still feel distressed if he lacks self-confidence or cannot make a decision.

Stress in projects

Among numerous causes of stress in projects are rapid pace, transient workforce, anxiety over discrepancies between performance and goals, and impending failure to meet cost, schedule, or contract requirements. In construction, for example, say Bryman et al.:

[The project manager] is in the front line controlling the labor force; he's answerable to the client, to his organization at a high level; he's responsible for millions of pounds [or \$] worth of work. . . . In a very fragile environment he is at the mercy of the weather, material deliveries, problems with labor, and problems with getting information.³⁹

We will restrict discussion to four main causes of stress in projects: work overload, role conflict, role ambiguity, and interpersonal relations.

Work overload is experienced in two ways. One is having too much work or doing too many things at once, with time pressures, long hours, and no letup. The other is taking on work that exceeds one's ability and knowledge. Overload can be self-induced by an individual's need to achieve, or it can be imposed by the responsibilities of the job. It is prevalent during crash efforts to recover lost ground and to rush projects toward completion. When overload is balanced with abilities, it can be positive and motivating; when it exceeds abilities, it causes distress.

Role conflict happens, for instance, when a person reports to a functional manager and a project manager, and the two managers impose contradictory or incompatible demands. It also happens when one person takes on multiple, incompatible roles. For example, a project manager might discover that to be a good administrator requires doing things that conflict with her values as a professional engineer.

Role ambiguity results from inadequate or confusing information about what a person needs to do to fulfill his job, or the consequences of not meeting job requirements. The person knows neither where he stands nor what to do. Role conflict and role ambiguity are common in projects where workers try to satisfy the expectations of many people. Project managers might feel frustrated because they have limited authority to satisfy the requirements of many stakeholders.

Stress also develops from the demands and pressures of **interpersonal relations**. Managers who are self-centered and dictatorial create stress for their workers. Irritable, abrasive, or condescending personalities make others feel unimportant and provoke anxiety.

In short, the typical project is a haven of work-related stressors—stress is inevitable.

Stress management

Most people accept stress as the price of success; however, although stress might be inevitable, distress (negative stress) is not. Project managers should be able to anticipate which

work demands are most stressful and try to ameliorate the negative effects through **stress management**.

In general, ways to reduce negative stress at work are aimed either at changing the organizational conditions that cause stress or helping people better cope with stress; because stress results from the interaction of people with their environment, both are necessary. Organizational means to reduce stress are aimed at task, role, physical, and interpersonal stressors; individual means are aimed at peoples' ability to manage and respond to stressful demands. We will focus on organizational means—methods applied by managers to reduce stress in projects.⁴⁰

Set reasonable plans and schedules

One way to reduce stress is planning and scheduling projects so as to allow for reasonable work hours and time off. Well-conceived plans and schedules prepared in advance help balance the workload; workers know what is expected and when, which helps avoid ambiguity, work overload, and the “crunch” that precedes milestones and project closeout.

Modify work demands through participation

Dictatorial, self-centered leaders (the too-bossy boss) cause stress, and so does the opposite, the do-nothing, under-demanding boss. In contrast, there is supporting research that the least-stressful style of leadership is participative—allowing workers decision-latitude and autonomy commensurate with their ability. Participative leaders set goals and define task limits, but allow workers flexibility as to how to achieve those goals and limits.

Social support

One way to reduce stress arising from work roles and relationships is to increase **social support** within project teams. Social support is the assistance provided through interpersonal relationships. Generally, people are better able to cope when they feel others care about and are willing to help them.

Vital sources of social support are family, close friends, and a supportive boss, coworkers, and subordinates. Social support from managers and coworkers does not necessarily alter the stressor but it does help people to cope better. A supportive project manager helps buffer against destructive stress; her subordinates are less likely to suffer harmful consequences than those with unsupportive managers. Coworker social support is equally important; caught between the conflicting expectations of a functional manager and project manager, a person with supportive coworkers will be better able to deal with the conflict.

How do people become supportive? Simply telling someone to be supportive does not work. Even when managers try to be supportive by giving advice, they often leave the distressed person worse off. Giving physical assistance is easy, but giving true emotional support is difficult and subtler. Empathic listening, understanding, and real concern are essential parts of support often missing in naive efforts to help. Thus, usually, it is necessary to provide some training in social support skills and reinforce and reward the usage of these skills. Unfortunately, as with many other behavioral aspects of management, training in empathy and sensitivity are considered “soft” issues and are devalued as “not productive.”

17.13 Summary

Contingency theories of leadership suggest that the most effective leadership style in most project situations is relations-oriented and participative; this is because project managers must rely upon the opinions of knowledgeable members of the project team and others.

A significant factor affecting project performance is team cohesiveness and teamwork. Teamwork must be developed and nurtured. But groups need help in developing effective teamwork, especially when the team comprises members from different backgrounds or exposes members to high stress. Methods for team building apply to a variety of situations, such as for resolving problems in an experienced team, building teamwork in a new group, or resolving issues between two or more groups. With variation these methods can be adapted to bring customers, subcontractors, and suppliers together at the start of a project. Many project teams rarely or never meet face-to-face. Virtual teams, a feature of the modern project landscape, rely on technology to communicate, and require special skills to manage and lead.

Conflict is inevitable in projects and when properly managed is beneficial. The primary conflict sources in projects are schedules, costs, priorities, manpower levels, technical opinions, administrative issues, and interpersonal conflicts; these vary in relative importance depending on stages of the project life cycle. Conflict is generally best dealt with through confrontation, i.e., examining the issues and attempting to resolve the conflict at its source.

All projects involve change and, ultimately, require people to adapt to change. In many cases, project success depends on how well stakeholders embrace and support the change; in this sense, the words change and project are synonymous. Change management is the process of building support for and reducing resistance to change through communication, project participation, and training.

Stress in projects is inevitable. Stress induces energy and increases vitality but in excess can be debilitating. The main sources of stress in projects are demanding goals and schedules, work tasks, roles, and social relations. Advance planning of workloads and deadlines can reduce many of the technical sources of stress. Participative management and social support help workers cope with stress; the former gives workers latitude in meeting requirements, the latter shows workers that others care about them and are willing to assist or provide support.



Review Questions

1. Explain the difference between task-oriented and relations-oriented leadership styles.
2. Describe the contingency approach to leadership. According to this approach, what is the best way to lead?
3. Discuss the differences between the leadership models of Fiedler and Hersey and Blanchard. What do these models say about leadership in the situations faced by project managers?
4. How is participative management and shared leadership useful for motivating and gaining commitment?
5. Why is teamwork important in projects? Isn't it enough that individual workers are highly skilled and motivated?
6. What characteristics are common to Vaill's high-performing systems?
7. What is meant by group process issues? What kinds of issues do they include?
8. What is the purpose of team building? Where is team building needed?
9. Outline the steps in a team-building session for a group that has been working together. Outline the steps for building a new project team.
10. Outline the steps in the IGPS process.
11. What conditions of management and the team members are necessary for team-building interventions to succeed?
12. Describe some situations that you know of where team building could be used.

13. What do you think are the reasons why team building is not used more often? What barriers are there to applying team building?
14. List the technologies available for virtual teams. For what tasks/decisions do each apply?
15. How is trust and cohesion developed in virtual teams?
16. List some special considerations in managing virtual meetings.
17. What are the sources of conflict between the user and the contractor? How do contracts lead to conflict?
18. What are the sources of conflict between parties in the project organization?
19. Describe how the sources of conflict vary with the phases of the project life cycle.
20. Why is some conflict natural and beneficial?
21. Describe four ways of dealing with conflict.
22. Explain how the project manager uses confrontation to resolve conflict.
23. What conditions must exist for confrontation to be successful?
24. Describe the role clarification technique. What sources of conflict does it resolve?
25. Describe these sources of stress in projects: project goals and schedules, work overload, role conflict and ambiguity, and social/interpersonal relations. Describe your work experiences with these sources of stress.
26. Describe the means by which participative management helps reduce work stress.
27. What is “social support”? What are the sources of social support? How does social support reduce job stress?
28. Describe Lewin’s three-step change process. How does it apply to projects? What are the five steps in the ADKAR change model and how does it relates to projects?



Questions About the Study Project

1. How would you characterize the leadership style of the project manager in your project? Is it too authoritarian, laissez faire (do nothing), or participative? Is the project manager task-oriented, relations-oriented, or both?
2. What kind of people must the project manager influence? Given the theories of this chapter, is the project manager’s leadership style appropriate? Despite the theories, does the style used by the project manager seem to be effective?
3. What do you think are the primary work motivators for people in this project? Discuss the relative importance of salary, career potential, incentives, and participation in decision making.
4. Describe the different groups (management teams, project office, functional groups) that comprise the project team in this project.
5. What mechanisms are used to link these teams—for example, coordinators, frequent meetings, or close proximity of workers?
6. What kinds of formal and informal activities are used to increase the cohesiveness of the project team? Can any of these be termed as team building?
7. Is the project team a virtual team? If so, what special provisions does the manager take to lead and manage the team?
8. Are steps taken to resolve problems involving multiple groups?
9. How would you characterize the level of teamwork in this project?
10. Ask if the project manager knows about formal team building and intergroup problem-solving procedures like those described in this book.

11. At the end of this (or other projects), how does the organization disband a team? Are there procedures for recognizing members or dealing with their feelings about disbanding?
12. How prevalent is conflict and what effect does it have on individual and project performance?
13. How does the project manager resolve conflict? Is confrontation used?
14. Are formal procedures used, such as RCT or IGPS, to resolve conflicts?
15. Emotional stress is a personal issue and most people are hesitant to speak about it other than on a general level. Still, you might ask the project manager or other team members about stresses they personally feel or perceive in the project.
16. Is this a high-stress or low-stress project? Explain. If high stress, is it taken for granted or do people take steps to reduce the stress?
17. Does the project manager try to help team members deal with job stress? Explain.

CASE 17.1 WILMA KEITH

Wilma Keith had worked for over 20 years as a successful project manager. But even with that background she found the Wiseteam project frustrating and overwhelming. Soon after being assigned to the project, she met with Cappun Queeg, the VP of communications. “Wilma,” he said, “the long and short of it is that the Wiseteam project *must* be completed and operational inside six months.” She had already estimated the project would take about a year and protested. Queeg became annoyed and said, “Just do it!” Wilma scoured the company for the best people she could find, settling on four young technical analysts from different departments. None of them were people-oriented or very good at communicating; technically, however, they were the best. Upon reviewing the project requirements, they all agreed: it would take a year. When Wilma reported back to Queeg, he said, simply, “If you don’t finish in six months, you’re fired. That’s a promise!”

So, Wilma set the team to work. Everyone knew Queeg’s deadline. At one point he dropped by to say that if they didn’t succeed, they would *all* be fired. This unnerved the analysts, but Wilma promised that if anyone were to be fired, it would be her, not them. She also promised that she would handle all dealings with Queeg, buffer them from his abuse, and take responsibility for any delays or problems. The team warmed to Wilma and set out to work—on average 6 days a week, 15–20 hours a day. Wilma never left them; if they were working, so was she. She started bringing brownies—lots of brownies, acting like a “den mother,” and treating the team like they were family. Indeed, given the long hours, the team seldom saw their real families and Wilma’s maternal care seemed to fill a void.

Several months into the project Queeg stormed in and asked Wilma why she had requested help from two outside consultants. She said despite the long work hours, the team was still behind and needed additional resources to meet the deadline. Queeg fumed that he was not about to hire any consultants. Wilma looked him straight in the eyes. “You don’t, and I quit!” Queeg knew she was serious. “Alright,” he said, “but that’s all you’ll get.” The team was amazed: Wilma had stood up to the vice president. This bonded them even closer and united them against the common “enemy.”

The intense pressure, long hours, strong competency of the team, and Wilma's nurturing worked: the team finished the project two weeks early and under budget—even with the expense of the two consultants. But ultimately the project failed because the Wiseteam system that Queeg had demanded did not provide anything new to its users. Queeg had never talked to the users, and Wiseteam was his own "pet" project. A year later he was gone from the company.

QUESTIONS

1. What do you think about Wilma's leadership style? What aspects of her style motivated the team? Would you say Wilma's style is more task oriented or relations oriented?
2. What aspects of Wilma's style do you think are typical of good project managers?
3. This was a stressful project. What did Wilma do that helped the team manage stress?
4. Is this case realistic? Are unrealistic demands like this actually put on project managers?

CASE 17.2 MARS CLIMATE ORBITER SPACECRAFT⁴¹

NASA designed the Mars Climate Orbiter spacecraft to collect data about Mars' atmospheric conditions and serve as a data relay station. Instruments aboard the Orbiter would provide detailed information about the temperature, dust, water vapor, and carbon dioxide in Mars' atmosphere for approximately 2 Earth years. The Orbiter would also provide a relay point for data transmissions to and from spacecraft on the surface of Mars for up to 5 years.

Nine months after launch the Orbiter arrived in the vicinity of Mars and fired its main engine to go into orbit around the planet. Everything looked normal as it passed behind Mars as seen from the Earth. After that, the Orbiter was never heard from again; presumably it had crashed into the planet. Paraphrasing project manager Richard Cook, "We had planned to approach the planet at an altitude of about 150 kilometers, but upon review of data leading up to the arrival, we saw indications that the approach altitude was much lower, about 60 kilometers. We believe the minimum survivable altitude for the spacecraft would have been 85 kilometers."

Later, an internal peer review attributed the \$280 million mission loss to an error in the information passed between the two teams responsible for the Orbiter's operations, the spacecraft team in Colorado and the mission navigation team in California. In communicating back and forth, one team had used imperial units (feet, pounds), the other had used metric units (meters, grams). Without knowing it, the two teams were using different measurement systems for information critical for maneuvering the spacecraft into proper Mars orbit.

QUESTIONS

1. How could such a mistake have occurred between the two teams?
2. What does the mistake suggest about the degree of interaction and coordination between the teams?
3. How might this problem have been prevented?

Notes

1. Chapman R. *Project Management in NASA: The System and the Men*. Washington, DC: NASA SP-324, NTIS No. N75-15692; 1973. The project team referred to is the project office staff, which numbered from a few members on small matrix projects to 70 in large pure projects.
2. Kloman E. *Unmanned Space Project Management*. Washington, DC: NASA SP-4102; 1972, p. 23.
3. Laufer, A. and Hoffman E. *Project Management Success Stories: Lessons of Project Leadership*. New York, NY: Wiley; 2000. [This book provides ample anecdotal evidence provided by managers from many more-recent NASA projects]; Edward J., Kinlaw C., and Kinlaw D. Developing superior project teams: a study of the characteristics of high performance in project teams. Paper presented at PMI Research Conference 2000, Paris, France. Newtown Square, PA: Project Management Institute, 2000; Geoghegan, L. and Dulewicz, V. Do project managers' leadership competencies contribute to project success? *Project Management Journal*, 39(4); 2008: 58–67.
4. Pretorius S., Steyn H., and Bond-Barnard T. Exploring project-related factors that influence leadership styles and their effect on project performance. *South African Journal of Industrial Engineering* 28(4); Dec. 2017: 95–108; Lindgren M. and Packendorff J. Project leadership revisited: towards distributed leadership perspectives in project research. *Int. J. Project Organisation and Management* 1(3); 2009: 285–308.
5. Fiedler F. *A Theory of Leadership Effectiveness*. New York: McGraw-Hill; 1967.
6. Hersey P. and Blanchard K. *Management of Organization Behavior: Utilizing Human Resources*, 4th edn. Upper Saddle River, NJ: Prentice Hall; 1982, pp. 150–173.
7. Müller R. and Turner R. Leadership competency profiles of successful project managers. *International Journal of Project Management* 28(5); July 2010: 437–448.
8. Sayles L. and Chandler M. *Managing Large Systems: Organizations for the Future*. New York: Harper & Row; 1971, p. 219.
9. Bennis W. and Nanus B. *Leadership: Strategies for Taking Charge*. New York: Harper & Row; 1985, pp. 224–225.
10. Vaill P. The purposing of high-performing systems. *Organizational Dynamics*, Autumn; 1982: 23–39.
11. Vaill, The purposing of high-performing systems, 38.
12. This discussion is largely based on Hofer W. Lady Liberty's business army. *Nation's Business*, July; 1983: 18–28; see also Hall A. Liberty lifts her lamp once more. *National Geographic*, July; 1986: 2–19.
13. Hofer, Lady Liberty's business army, 28.
14. Hofer, Lady Liberty's business army, 21.
15. An exception is projects that require high innovation and creativity, and wherein, in general, strong teamwork tends to *reduce* team effectiveness. This suggests in high-innovation projects that teams be allowed for periods of collaborative work as well as *less-collaborative* and *individual* work to facilitate individual creative-thinking skills. Hoegl M. and Parboteeah K. Creativity in innovative projects: How teamwork matters. *Journal of Engineering and Technology Management* 24(1–2); March–June 2007: 148–166.
16. See Chapman, *Project Management in NASA*, pp. 59–62. The other functions of project management were defined to be project planning, information and control, and consultation.
17. Keller R. Predictors of the performance of project groups in R&D organizations. *Academy of Management Journal* 29(4); December 1986: 715–726.
18. This format is based upon Dyer W. *Team Building: Current Issues and New Alternatives*, 3rd edn. Reading MA: Addison-Wesley; 1995. For other approaches, see: Katzenbach J. and Smith D. *The Wisdom of Teams: Creating the High-Performance Organization*. New York, NY: HarperBusiness; 2006; Harvard Business Review, *Harvard Business Review on Building Better Teams*. Brighton, MA: Harvard Business Review Press; 2011.
19. Based on research in Gratton L., Voight A., and Erickson, T. Bridging faultiness in diverse teams. *MIT Sloan Management Review* 48(4); 2007: 3.
20. This discussion is largely based on Dyer, *Team Building*.
21. Blake R., Shepard H., and Mouton J. *Managing Intergroup Conflicts in Industry*. Houston: Gulf Publishing; 1965.
22. This section is largely based upon Duarte D. and Snyder N. *Mastering Virtual Teams*. San Francisco, CA: Jossey-Bass; 2006, pp. 31–80.
23. Portions of this section are from Duarte and Snyder, *Mastering Virtual Teams*, p. 77.

24. Based upon Duarte and Snyder, *Mastering Virtual Teams*, pp. 145–147; and Cohen M. *Succeeding with Agile*. Upper Saddle River, NJ: Addison-Wesley; 2010, pp. 367–370.
25. Cohen, *Succeeding with Agile*, p. 370.
26. Much of this section adapted from Duarte and Snyder, *Mastering Virtual Teams*, pp. 170–183.
27. Cohen, *Succeeding with Agile*, pp. 376–377.
28. Sayles and Chandler, *Managing Large Systems*, p. 278.
29. Thamhain H. and Wilemon D. Conflict management in project life cycles. *Sloan Management Review*, Spring; 1975: 31–50; and Thamhain H. and Wilemon D. Diagnosing conflict determinants in project management. *IEEE Transactions of Engineering Management* 22; February 1975.
30. Schmidt W. Conflict: a powerful process for (good or bad) change. *Management Review* 63; December 1974: 5.
31. Moody F. *I Sing the Body Electronic*. New York: Viking; 1995, pp. 110–115.
32. Thamhain and Wilemon. Conflict management, pp. 46–47.
33. Sayles and Chandler, *Managing Large Systems*, p. 216.
34. Method based upon Dyer, *Team Building*, pp. 109–116.
35. Dyer, *Team Building*, p. 112.
36. Lewin, K. *Field Theory in Social Science: Selected Theoretical Papers*. New York: Harper & Brothers, 1951.
37. Hiatt, J. *Adkar: A Model for Change: How to Implement Successful Change in Our Personal Lives and Professional Careers*. Prosci Research, 2006. <https://store.prosci.com/bookstore/adkar-a-model-for-change-en.html>.
38. Portions of this section adapted from Huse E. and Cummings T. *Organization Development and Change*, 3rd edn. St. Paul: West; 1985, Chapter 12; Quick J. and Quick J. *Organizational Stress and Preventive Management*. New York: McGraw-Hill; 1984; and Williams J. *Human Behavior in Organizations*, 2nd edn. Cincinnati: South-Western; 1982, Chapter 9.
39. Bryman A. et al., The concept of the temporary system: the case of the construction project. *Research in the Sociology and Psychology of Organizations* 5; 1987: 253–283.
40. Portions of this section adapted from Huse and Cummings, *Organization Development and Change*; Quick and Quick, *Organizational Stress and Preventive Management*; Williams, *Human Behavior in Organizations*; House J. *Work Stress and Social Support*. Reading, MA: Addison-Wesley; 1981; and Warshaw L. *Managing Stress*. Reading, MA: Addison-Wesley; 1982.
41. NASA website, December 1999.

Part V

Project Management in the Corporate Context

18	Project Meta-management and Program Management	633
19	Project Selection and Portfolio Management	671
20	Managing International Projects and Megaprojects	701

Beyond project organization, leadership skills, and tools and methods for management, what else is needed to improve project success? One answer is that the organization must provide support to its managers and encourage and enable them to apply project management best practices. Another is that the organization must choose projects that are viable and beneficial to the organization, i.e., projects that meet sound criteria based upon the organization's objectives and available resources. And, still another is that, if the project is but one element of many in a larger endeavor—a program or portfolio—then that larger endeavor must itself be well managed. These are the topics of Chapters 18 and Chapter 19.

In the growing globalization of today's business and technology, projects of international scope are accelerating in number and becoming increasingly influential on commerce, technology, and the environment. Related is the growing number of projects of immense size and scope, costing US \$ billions, lasting for several years, and potentially impacting thousands or millions of stakeholders. The question of how to manage such projects, is the subject of Chapter 20. The topics of this chapter span most everything covered in this book and serve as a fitting summary and review of all that is project management.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Chapter 18

Project Meta-management and Program Management

Meta-management of projects refers to important aspects of project management over which project managers typically have little or no influence. For example, the general approach to be taken and preparedness of an organization to perform projects usually depends on others—senior managers and directors—and, given that, the likelihood of project success depends partly on measures taken by the organization. Such measures, the first topic of the chapter, include project management maturity, project management methodology, knowledge management, and the project management office (PMO). Frequently projects are undertaken as part of a larger agenda—programs—and these require their own kind of management, program management, which is the second topic of the chapter.

18.1 Project Management Maturity and Maturity Models

How good are we really? How well do we measure up to our competitors? In which areas should we improve? These are questions that companies continually ask themselves about their capabilities and competencies. An organization's capability or competency regarding project management is referred to as its "maturity."

Maturity continuum

Just as people mature physically and mentally, organizations mature in project management. Typical levels or phases through which maturity increases are shown in Figure 18.1.

The first level of **project management maturity** is when a few people in the organization learn the principles of good project management and practice them on their own. Of course, for the organization to further develop its capability many people must practice the principles; for that to happen, it requires executive-level awareness about the importance of project management and willingness to support aspects of those principles throughout the company. These aspects include documenting **lessons learned** in every project for the benefit of future projects and developing a common language of project management terms to be used throughout the company. A company with projects around the world might create a glossary



Figure 18.1

Levels of project management maturity/competency.

of terms in multiple languages. Naturally, moving to higher-level maturity also requires that the organization develop a project management methodology. Ultimately the organization will be in the position to benchmark its project management capabilities against organizations that are industry leaders.¹

Maturity models

Project management maturity is gauged according to so-called “maturity models.” There are many recognized models although none has achieved uniform acceptance worldwide.² **Maturity models** fall into three categories:³

- Technical delivery process models
- Project management process models
- Total organization models.

Technical delivery process models originated in the Total Quality Management movement of the 1980s when companies started to measure their quality management capabilities. An example is the Capability Maturity Model (CMM) developed by the Software Engineering Institute of Carnegie-Mellon University during the 1980s and 1990s to help identify competent software contractors. The model, which emphasizes process documentation and is similar to ISO quality standards, has five levels of maturity similar to Figure 18.1 but with a software engineering focus.

Project management process models focus on knowledge areas.⁴ Many of these models are based on the ten knowledge areas of the Project Management Institute’s (PMI) Project Management Body of Knowledge (PMBOK),⁵ where the level of maturity achieved in each knowledge area is determined by comparison to standardized criteria during an audit. Figure 18.2 shows the audit results for the assessed maturity levels for eight knowledge areas.

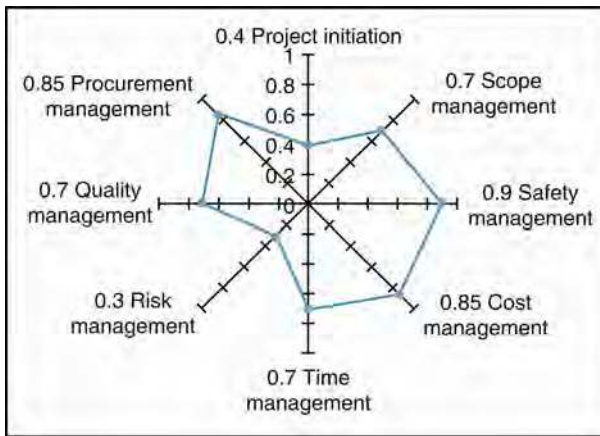


Figure 18.2
Results of maturity
assessment for eight project
management knowledge
areas.

Process models commonly specify five levels of maturity, comparable to those in Figure 18.1:⁶

- Ad hoc: no formal procedures or plans
- Individual project planning
- Systematic project planning and control
- Integrated multi-project and formal planning and control
- Continuous PM improvement.

Following development of the CMM, the PMI sponsored research at the University of California Berkeley and George Washington University that produced the Organizational Project Management Maturity Model (OPM3).⁷ This is an example of a total organizational model, so-called because it addresses the entire organization and how it manages projects, programs (discussed later in this chapter), and portfolios (discussed in Chapter 19).



See Chapter 19

How good should we be?

It would be incorrect to presume that an organization should strive for the highest-level maturity in all aspects as prescribed by these models. Different companies have different needs that require different levels of maturity. For example, whereas a company doing research with limited internal funding needs strong capability in project selection, a construction contractor with capacity to accept whatever work comes along does not. Likewise, a company that develops nuclear reactors needs high maturity in environmental and safety practices, but a company that develops computer games likely does not. One study of project management maturity in product development found that standardized tools and project management processes increase project success up to a point, but beyond that they reduce success; in other words, conformity to industry standards can take you only so far.⁸ No single maturity model enables project success across all types of industries and projects. Each organization must identify which areas of competency are important and avoid wasting resources to achieve high maturity in areas not important or irrelevant.

Benefits and shortcomings of maturity models⁹

The achievement of a high maturity level according to a standard maturity model gives a company bragging rights. For example, the company can point out in its proposals the high ratings it achieved for a recognized maturity model.

By their very nature, however, maturity models emphasize formal processes and procedures and focus only on **explicit knowledge**, which is knowledge that can be documented. A weakness of the models is they ignore **tacit knowledge**, which is knowledge that cannot be easily written or described. Leadership, communication, teamwork, and the knowledge and skills held by project managers and team members play a big role in project success; yet being tacit, they represent knowledge not accounted for by the maturity models.¹⁰

Maturity and project success

Studies indicate that about two-thirds of organizations rate at levels 1 or 2 on the five-level maturity scale. Not very high. Companies in petrochemical and defense industries tend to be more mature; those in insurance, financial and health services, pharmaceutical R&D, and telecommunications tend to be less mature.¹¹

Does higher maturity according to the models correlate to greater project success? The empirical evidence is paltry, but the answer is, “not necessarily.” Project success depends on many things, including the project environment, team, and project manager, none of which the maturity models address. Most senior managers see little association between maturity level and project performance.¹² A few studies claimed a positive correlation between maturity and project success, but they lack a theoretical basis and, not surprisingly, several were conducted by consultants, not researchers.¹³ And it is not obvious that maturity alone offers a competitive advantage. The models measure only explicit knowledge—that which can be standardized and documented and, hence, copied or adopted by every other company. So, an organization that mimics standard practices and ignores developing its own unique strengths can never become better than its competitors.¹⁴

To reach maturity levels 3 and higher in Figure 18.1, processes and standards for managing projects, a *project management methodology* must be created and utilized. Also important are to document and utilize lessons learned, which relates to *knowledge management*, and to create a PMO (project management office). These topics are described next.

18.2 Project Management Methodology¹⁵

A **project management methodology** is a framework or procedure specifying who should do what at each stage of the project life cycle. Standards such as the PMBOK Guide provide processes and tools that, often, are incorporated into the methodology. The methodology used by an organization typically addresses many of the topics of this book, although organized in a way to best suit the organization. It provides a structure so that all projects are managed and performed in a standardized, disciplined, and systematic manner, using common practices to increase the likelihood of project success. An organization creates or adopts a methodology that uniquely fits its business requirements, procedures, and culture, and the size, scope, and technology of its projects. Some methodologies prescribe the technical tasks of a project; our focus is on those that prescribe the *management* tasks of projects.

Why methodology?

By encouraging conformance to a prescribed project management methodology, an organization helps assure that all projects are conducted and managed in a similar manner. Lacking a methodology, individual project managers use their own management practices and tools—some good, some not so good.

The aim of the methodology is to ensure that recognized “good” and “best” practices are applied across all projects, and to elevate the practices of all project managers to those of its best managers. The methodology provides a common way to do things and a common terminology. Everyone doing things in similar ways enhances communication and learning about those ways. Every manager in the organization should follow the methodology. Managers who are accustomed to a structured, documented approach to project management will be more likely to adopt a standardized methodology; those not accustomed will possibly resist it.

What does the methodology mandate?

The methodology specifies the stages of the life cycle for projects in an organization and the roles and management tasks of project managers and stakeholders at each stage. For instance, it specifies who is responsible for initiating, proposing, reviewing, and selecting projects, and roles and responsibilities within the project review board (discussed in the next chapter) and the PMO (discussed later in this chapter). It also specifies the individuals who must review the project at gates and sign off on budgets and schedules.

Phases and gates

Most projects are conducted in phases and stages. The project management methodology defines the phases or stages into which projects are divided—for example, initiation, feasibility, definition, development, and launch—and what should happen in each. At the start of each phase there might be a **gate**, so-called because at that point the status of the project and plans for the remainder of the project are assessed, and a decision is made whether to continue, hold, or cancel the project. The number of phases and gates depends on the methodology, with the minimum usually four or five. For example, the Motorola methodology in Case 18.2 (see end of chapter) had 16 sub-phases within five phases for its Cellular Systems projects. The gates can also represent approval of, say, project initiation, systems requirements, system validation, and system launch. Decisions at each gate are based on specific criteria.

The gating process is common in organizations that conduct concurrent internal projects in product development, IT, infrastructure, product development, or process improvement, and where the projects “compete” for product or market goals and resources. It is one way of culling weaker, less-promising projects so that scarce resources are devoted to stronger, more-promising projects; it also reduces risk in large, stand-alone projects.

Relationship with project life cycle

Figure 18.3 illustrates a project management methodology for a seven-stage project life cycle (Initiation/feasibility through Maintenance). In general, a methodology should conform to

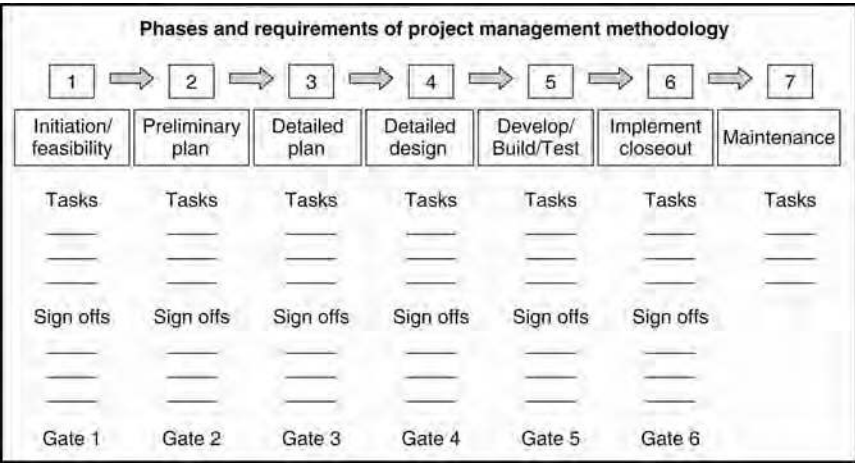


Figure 18.3
Project management methodology: phases and associated tasks, sign offs, and gates.

the technical and business practices of the organization; for example, stages 4 and 5 in the methodology in Figure 18.3 must be compatible with whatever development methodology the organization employs—waterfall, spiral, iterative, Scrum, or hybrid.

Elements of the methodology

The content of a project management methodology has been the subject of most of this book. In fact, one way to create a methodology is to look at the topics and methods in project management books, determine which are applicable to the organization’s projects, and then arrange them into a framework according to the project life cycle.

The actual content and details of the methodology—its tasks and requirements—depend on the scope and scale of the organization’s projects. For large, complex, risky projects, the methodology would specify detailed tasks and methods for analysis, definition, planning, monitoring, control, and closeout. For small, low-risk projects, a somewhat simplistic methodology is adequate. Choices about which aspects of the methodology must be followed and which can be bypassed in a given project should be stated in the methodology and not left to the project manager’s discretion.

This book has used the phases of Conception, Definition, and Execution, each with a series of stages; in general, however, a particular project can be defined using any number of phases or stages. The methodology defines the nominal phases or stages in terms of whatever best represent the “natural” progression of the organization’s projects, from initiation to execution and closeout.

The project phases can be based upon standards. For example, organizations involved in large engineering/construction projects commonly employ a life cycle with phases as defined by the Construction Industries Institute (CII), namely: *Feasibility*, *Concept*, *Detailed Scope*, *Design and Construction*, *Startup and Commissioning*, and *Operations*. Companies that build facilities for the chemical, mineral, and oil and gas industries often use the phases recommended by IPA (Independent Project Analysis), which are: *Generate/Shape Idea*, *Define Opportunity* (FEL-1), *Develop*

Scope (FEL-2), Define the Project (FEL-3), Execute, and Produce. The early phases of this methodology were discussed as “front-end loading” in Chapter 4.

The methodology can also include stages preceding and following the actual project, such as the post-project Maintenance stage in the methodology in Figure 18.3.



See Chapter 4

Required tasks and deliverables

For each phase or stage, the methodology specifies project management tasks and deliverables; for example, Phase 1: Initiation/feasibility in the methodology in Figure 18.3 might specify:

- Assemble team and identify stakeholders.
- Prepare project charter.
- Prepare a preliminary task list.
- Perform risk analysis and prepare key-risk list.
- Develop a requirements list.
- Prepare funding request.
- Prepare resource plan, timeline, spending plan.
- Prepare project proposal.

The methodology will specify tasks and deliverables to cover virtually all of the topics covered in this book, such as those in Table 18.1.

Sign offs and approvals

When the methodology includes gates at which the project must be approved, it also specifies the persons having sign-off authority and the roles of particular stakeholders such as the client, sponsor, champion, and project manager.

The methodology for a large company is shown in Figure 18.4. Interesting in its details, it exemplifies the scope of the tasks, deliverables, and responsibilities covered in a comprehensive project management methodology. Note the required sign offs at each phase.

Table 18.1 Project management tasks and deliverables.

Project initiation/proposal	Recruiting, training, layoffs
Stakeholder identification	Project tracking/review
Project selection	Data entry
Proposal development	Reporting to management
Project planning	Project auditing
Requirements/specification	Quality control/assurance
Work definition	Process control
Resource needs	Change control
Time and cost estimating	Project closeout
Scheduling	Post-project review
Budgeting/accounting	Post-implementation review
Risk analysis	Knowledge management
Procurement/contract management	



Figure 18.4
Comprehensive six-phase project management methodology.

The PRINCE2 methodology

An example of a standard methodology is **PRINCE2**, which the UK Government developed to assist organizations in developing their own unique methodologies. PRINCE2 defines the roles of high-level managers (corporate or program), project board, project manager, and team managers. It prescribes the project stages as *Pre-project*, *Initiation*, *Subsequent-delivery stage(s)*, and *Final delivery*. The methodology includes prescribed documentation: the *Pre-project* stage is initiated by a *mandate document*, the *Initiation* stage by a *project brief*, and the *Subsequent-delivery stage(s)* by *PID(s)* (*Project Initiation Documents*). PRINCE2 also prescribes a stage-management process to be followed upon completion of the *Initiation* stage and *Subsequent-delivery* stages. The process defines the project manager's responsibilities to enable the Project Board to assess the stage, approve the plan for the next stage, update the overall project plan, and manage product delivery in the *Final delivery* stage.

One size fits all?

Most methodologies are flexible. They specify project management requirements for a generic kind of project but allow for exclusion of some requirements and inclusion of others, depending on the unique features of each project. When all projects in an organization are similar in terms of scope, size, and technology, then one methodology might fit all of them.

To accommodate projects of different size and complexity, the methodology can be “scalable” or come in, say, three or four versions, the particular one to be applied depending on the capital resources, duration, number of work packages and contractors, and the risk of the project. A problem with multiple methodologies, however, is deciding which is appropriate for a given project. The decision is usually based upon factors of the project, as discussed in section I.3 of the Introduction: novelty, complexity, technology, and pace.¹⁶

Most organizations have one basic—perhaps scalable—methodology because their projects all tend to be similar. But organizations such as oil and gas companies, which undertake projects in different categories (product development, exploration, construction, applied research, marketing) have multiple, very different methodologies. One methodology would be applied to, say, projects in search of new oil sources, another for projects to construct new refineries or ocean-drilling platforms, another for researching petroleum substitutes. The technical stages, tasks, and life cycles of these projects vary and, thus, require different project management methodologies.



See
Introduction

Developing the methodology

To develop a methodology, an organization can create one from scratch or adopt it from elsewhere. In the first way a small group of the organization’s best project managers meet to create a methodology that incorporates methods they use or recognize as good and believe should be adopted for use in every project. In the second way, managers look at methodologies used by other organizations and that represent industry standards, and adopt portions of the ones they find most suitable. Many companies have developed their own somewhat unique methodologies, some of which can be found online. Many of these methodologies are similar in terms of scope and details, and are a good source for ideas.

When an organization looks at an industry standard or another organization’s methodology, it should use those as baselines from which to create its own methodology and then precisely tailor it to its projects and business practices. Ideally, the tailoring is done by a group of the organization’s best project managers (not senior managers or paid consultants); this helps ensure the methodology is appropriate for the organization’s projects and will be accepted by its project managers.

A project management methodology should not be static; it should be modified and improved based on experience and a changing environment. It should be periodically reviewed to account for changes in projects, technology, and business practices. As new steps and requirements are added, others are pruned to prevent the methodology from becoming unwieldy. Of course, ability to improve the methodology depends on how much the organization is able to learn from its past projects—its knowledge management—covered next.

Perhaps the desideratum for any methodology is that the payoff from using it exceeds the effort in creating and upholding it. The methodology must not become yet more red tape, forcing managers to attend more to the rules than to managing projects. It should not become “Let’s just fill out these forms and ‘tick the boxes’ so we can get on with the work.”

18.3 Managing Project Knowledge

One potential pitfall in managing projects is treating each project as if it were completely distinctive and ignoring lessons from other projects. Solutions to problems are invented . . . and reinvented. Mistakes are repeated . . . and repeated again. Why does that happen? As the saying goes, “Fool me once, shame on you. Fool me twice, shame on me!”

As an example, consider a project that is thought to be truly unique. The project manager must ponder what to expect and how to proceed. He starts with a clean slate and presumes there is no one in the organization to help him because—after all—the project is unique. But rarely can it be said that there is no one in the organization who can help. Usually there is someone, somewhere, with experience and knowledge that is relevant to the project. If only the project manager knew who that someone is!

Authors O’Dell and Grayson describe the problem of wasted knowledge in their book *If Only We Knew What We Know*.¹⁷ The knowledge exists, but people don’t know it exists or how to access it. Often the waste occurs because the organization has no formal process for capturing and disseminating knowledge—it has no **knowledge management**. In a project organization, knowledge management helps ensure that people in every project learn something, and that whatever they learn will be available to others who could use it. Knowledge management can provide project managers with the knowledge they need, even in cases where they themselves don’t know they need it!

Organizational forgetting

According to the classic learning curve, knowledge accumulates with experience: the more of something you do, the more you learn and the better you get, at least up to a point. The same holds for organizational learning, but sometimes with a twist: initially the organization gains knowledge through experience—learning more as it does more—but then it reaches a plateau or starts regressing, knowing less even as it does more. This “organizational forgetting” happens when workers, especially those with tacit knowledge (discussed later), leave the organization, new processes and technologies render old ones obsolete, procedures are not documented, or records are discarded or lost.¹⁸ When teams disband after each project, it is easy for them (and the organization) to forget what they learned or miss opportunities to learn from their experience and share it for future projects.

Capturing knowledge

Knowledge is information put to use. Everything experienced in projects is a source of information, but to learn from each experience managers and teams must reflect on what happened, what they did, and the outcomes, and they need to draw conclusions, otherwise they won’t learn or will forget.

One opportunity to learn from a project is the **post-completion project review** or post-mortem, discussed in Chapters 5 and 10. During the review, the project team carefully looks at what it did and what it learned from doing that. It reflects on significant events, successes, and failures, and the actions that led to them and creates a **lessons learned log**.

Sometimes the post-completion review is not enough; it happens once, at the end of the project, by which time memories of events have faded, recollections of details dimmed, and



See Chapter 5
and 10

information lost. Therefore, especially in long projects, additional, mid-stream, reviews should be held at key milestones and after notable events. Unlike status reviews that measure progress and identify problems, the purpose of these reviews is to reflect on actions taken and to learn from experience.

Common knowledge and knowledge transfer¹⁹

Nancy Dixon defines **organizational common knowledge** as knowledge available and easily accessible to everyone in the organization. It is “how to” knowledge gained through the experiences of the company, largely unique to the company, and generally not available to the public. Because it is gained from experiences inside the company and not known to outsiders, it potentially sets the organization apart. It cannot be measured by maturity models, yet is perhaps the most important kind of knowledge for helping an organization exceed the competition.

But for organizational knowledge to become “common” it must be captured, retained, and shared through a mechanism called **knowledge transfer**. Knowledge transfer can happen broadly throughout the organization or directly between individuals.

Documentation and databases

One way to transfer project-related knowledge is to document learnings from post-completion and mid-stream reviews, and to incorporate those into the project management methodology and checklists for “lessons learned,” “risks and pitfalls,” and “best practices.” Documented knowledge can also be transferred via project report libraries, training seminars, and online knowledge databases. These sources provide information useful for, among other things, “analogy” estimating in project proposals.

Example 18.1: Preparing a Proposal Using Databases and Peer Advice

Jacque has received an RFP from a client to provide engineering consultation for a new process. The client wants an answer soon. Jacques accesses the company knowledge database to see what his company has done and is doing now concerning the process. He also reviews online abstracts and articles to learn about leading industry practices for the process, and then checks the company’s competency tracking system for names of people inside the company who know the process. The name Leslee pops out, someone he met earlier at a companywide networking meeting, which Jacques’s company frequently holds for the express purpose of enabling people to meet each other and share project experiences. Jacques calls Leslee and arranges a phone conference, but before the conference Leslee checks the company database for background about Jacques’s client. During the conference Leslee and Jacques work out the details of the proposal, which Jacques completes and sends to the client—barely a week after receiving the RFP.

Databases play a useful role in knowledge management, but their creation and upkeep is a subject of its own and is beyond the scope of this book. For use in planning, estimating, and prediction, databases must conform to strict requirements for content and formatting. Suffice it to say, a knowledge database requires substantial effort and is ideally managed by a team of knowledge experts who know how to make it useful and user friendly. Ernst & Young, for instance, retains a database of the best-written and most informative proposals, presentations, and plans arranged into topical areas called “Powerpacks,”²⁰ each managed by a team of experts, documented in a standardized form, and targeted to specific user groups.

One problem with information retained in a database is that it is *latent*: it exists but is useful only when the database is accessed. A person needing information has to *initiate* the transfer process by accessing the database—and know where to look in the database and what questions to ask.

Some companies actually *impose* potentially useful information on people who need or could use it. A project support group (PSG) or PMO tracks information that might be of use, and forwards it to those who might need it. If, for example, a project has done an outstanding job at reducing material costs, the PSG will write a brief report about the project and send it to managers in other projects who might be interested. This documenting and distributing of “best practices” reports helps the organization to expand its common knowledge.

Tacit knowledge and personal interaction

But some kinds of knowledge cannot be abridged into written reports and, hence, cannot be transferred via a document or database. Such was the case in Example 18.1, where Jacques relied not only on databases but also on Leslee. Personal interaction like this is necessary for transferring **tacit knowledge**, i.e., knowledge that is difficult to put into written words or even pictures—and that exists only in people’s heads and is sometimes hard to articulate. (For example, although you can easily recognize a person’s face, you might be hard-pressed to describe the person’s facial features that enable you to do that.)

Much of the knowledge required to manage and conduct a project is tacit, which means it cannot be retained or transferred via databases, documents, reports, or checklists, which apply to explicit knowledge. Jacques’s firm encourages exchange of tacit knowledge through networking meetings where people make acquaintances they might one day have to rely on.

After-action reviews

Teams that remain intact from project to project can learn and develop a growing knowledge storehouse through **after-action reviews (AAR)**. The concept is derived from troop teams in the US Army, which use AARs to debrief and learn from the consequences of their actions immediately following an event.²¹ An AAR is a quick meeting immediately after an incident wherein a team looks at what it did, what happened, what was supposed to happen, and what accounted for the difference. Not really a “meeting” but rather a part of the way the team functions, an AAR is quick, to the point, and takes as little as 20 minutes. Everyone involved in the action participates, and one member facilitates. The imperative is that everyone is candid and speaks truthfully without fear of recrimination. AARs are most effective for projects that have clear, specific goals, and where the team has established clear measures to assess the impact of its actions toward reaching the goals.²²

Information from an AAR is usually kept confidential, which encourages candor and reduces fears of the team or individuals getting a bad reputation. Teams wanting to learn must

feel free to try out different actions—some that might not work—and to openly admit mistakes. Whatever the team learns in an AAR remains with the team, unless it decides to share it with outsiders.

Peer consultants and project support groups

AARs apply to intact teams doing repetitive projects. What about newly formed teams just starting out and where much about the project is new to them—its technology, geographic location, culture, and so on? Likely the knowledge they need resides somewhere in the organization, but the trick is to connect those who need it (knowledge receivers) with those who have it (knowledge providers) so the two parties can *interact* personally, one-on-one.

Why interact personally? Because when knowledge providers and receivers get together, amazing things happen, like questions and solutions occurring between them that neither the provider nor receiver would have thought of beforehand. Perhaps you have experienced this: you ask for someone's advice, which leads them to ask you a question, which leads you to ask a question back, and so on. Often this back-and-forth questioning results in going down paths that neither of you anticipated. The knowledge provider sees the situation in a new way, draws parallels, and comes up with insight and new ideas. The question is: what can an organization do to bring knowledge providers and receivers together so this can happen?

Example 18.2: Peer Consultants

A team of spacecraft engineers is preparing a proposal to bid on a satellite for a telecommunications corporation. The team has reviewed the requirements of the customer and prepared a preliminary design but is not able to decide on features of the satellite's configuration because of the project's large risk and investment. For advice the team leader contacts 11 people at different company divisions whom he knows personally or from the grapevine. Six respond that they are willing to help out—four from company divisions in California and Texas, and two working at NASA, and the leader arranges for the six (**peer consultants**) to meet in-person with her team for one day in California. At the meeting, the satellite team presents data it has collected and posts diagrams and charts on the walls. The peer consultants question the team about the implications of the data, and then everyone works together to develop criteria for deciding the final configuration. The satellite team then leaves the room briefly while the peer consultants review everything and prepare recommendations. When the satellite team returns, the consultants summarize their conclusions. No decision is made about the final configuration at the meeting; however, the satellite team has learned much about the issues it still needs to resolve.

For any project manager from the company in Example 18.2 that requests peer consultation, the company will help arrange it and cover the consultants' expenses for travel and time off. The consultation process emphasizes questioning, analysis, and feedback. The peer consultants offer guidance, but the project team makes its own decisions.²³

Some companies use a "locator system," which provides names, addresses, phone numbers, and other pertinent information of people worldwide working in specific knowledge areas; some companies supplement that with their own internal full-time consultants.

Example 18.3: Project Support Group²⁴

The **project support group (PSG)** of a large pharmaceutical corporation includes ten consultants available to provide expert support to any project manager who requests it. Also available are the part-time services of over 50 managers throughout the corporation with experience in project planning and execution. As a profit center, the PSG charges fees to the company units of the project managers it assists. The PSG also sponsors semi-annual forums where project managers meet to share experiences.

The benefits of the PSG are illustrated in the story of Trevor, a typical project manager. Around the time his project was nearing completion, Trevor attended a company forum. Confident that his project had been a big success, he was surprised to learn of two other similar recently-completed projects. One had developed a process that, had he known, could have shaved 3 months off his project; the other had made mistakes similar to ones made in Trevor's project, and had he known, would have saved the project \$50,000. In other words, the cost of Trevor *not* knowing what others in his company already knew was 3 months and \$50,000!

For his next project Trevor contacted the PSG, which assigned Jiang to work with him. Although Trevor's department had to pay for Jiang's services, Trevor felt the advice he would receive could substantially benefit his \$250 million project. The PSG also provided a database of current projects with state-of-the-art practices, which Jiang and Trevor used to develop the project plan. Throughout the 2-year project, Jiang contributed ideas, management tools, benchmarking goals, peer review, and on-call availability for mentoring and coaching.

Although many project managers in the company use knowledge databases, the most important way they gain project-specific knowledge is from the consultants who devote the time to understand a project well enough to draw upon their tacit knowledge for insight and suggestions. They are "living databases" who travel from project to project, tailoring the knowledge they share to the needs of each.

Knowledge transfer from personal interaction works best when the organization supports and facilitates the process. Requests for assistance are viewed as a legitimate business process, not as asking for favors. People freely ask for help without feeling intrusive. Everyone is encouraged or required to take advantage of the process. Knowledge is power, and that is another reason to formalize the process: sad but true, in some organizations, knowledgeable but power-hungry people resist requests for assistance and avoid sharing what they know.

Discussion

Companies with the best knowledge management practices utilize a variety of methods that account for both tacit knowledge and explicit knowledge. For instance, spacecraft designers at Hughes Space & Communication Company are able to reduce development costs by "reusing" designs wherever possible. So as to avoid reinventing anything, they rely on the "Knowledge Highway," a process that includes an intranet database of lessons learned and best practices compiled by an editorial team, and pointers to experts.²⁵ At Microsoft, information sharing is encouraged through monthly informal, cross-group lunch meetings. Managers from Word, Excel, and MS Project product teams meet for two hours to talk about their work, problems,

and thoughts; they are also encouraged to informally meet with or give presentations to other managers company-wide and worldwide.²⁶

Who is responsible for project knowledge management? The project manager is responsible for capturing knowledge in each project and sharing it with his peers, but responsibility for organizational “common knowledge” must fall to the managers or organizational units that oversee projects. In many cases that responsibility resides in a PSG or knowledge management team. Often the team is a part of the PMO.

18.4 Project Management Office²⁷

Think for a moment about everything the project manager does as described in this book and you soon realize that being a project manager is a lot of work! Much of that work involves collecting and processing data and preparing documents, reports, plans, budgets, and presentations. The workload can be overwhelming, and sometimes there is not enough time in a day to do it all.

Example 18.4: Bay Area Medical Center

Gaurav and other project managers from the Bay Area Medical Center attended a series of seminars on project management. At the end of the series everyone agreed on the value of the tools learned, and they returned to work with every intention of putting them into practice. Months later, the reality was that Gaurav and his colleagues had used little of what they had learned, and almost nothing had changed about the way they managed projects. Gaurav had started to create a WBS and Gantt chart for his bigger projects but gave up; already working long hours, he had scant time to devote to them. Besides, BAMC offered neither support nor recognition to use these or any common project management tools.

BAMC is no different from many organizations: they send project managers to seminars to learn new and better ways, but then do nothing to encourage or support usage of those ways. The tools fall by the wayside and nothing changes.

Countering this in other organizations is the **project management office (PMO)**—also called the *project support office* or PSO), a department or unit whose purpose is to assist and support project managers, allocate project resources, and, in general, facilitate good project management practice. The PMO establishes and maintains the project management methodology, instigates initiatives that will increase the organization’s project management maturity, and oversees knowledge management. A PMO formalizes the practice of project management but assists project managers so they are not overwhelmed and can adapt to the formalization.

PMO leadership

Senior managers often do not understand project management; they see it as a role or job, not a profession. To them projects are discrete occurrences that have little in common. They allow project managers to work independently but grant them little formal authority. An early challenge of the PMO is to impress on senior managers the importance of the project manager role

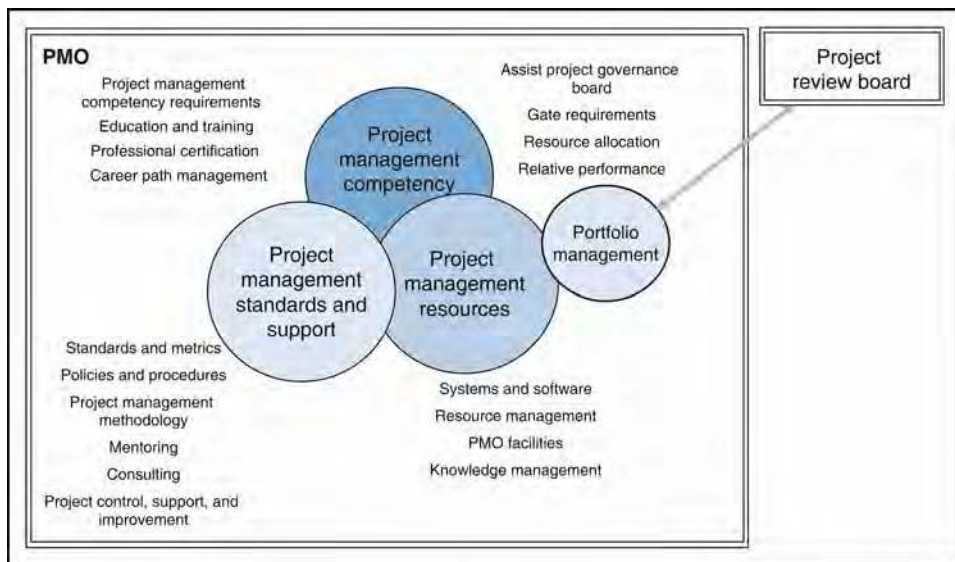


Figure 18.5
Functions and responsibilities of the PMO.

and that of everyone adhering to a prescribed project management methodology. To gain the attention of senior managers, the PMO must be staffed with some of the organization's most experienced and respected project managers.

The PMO can take many forms. Typically, it is a permanent staff that helps guide projects in all or certain departments of the organization; sometimes it is created to serve a single, large project or program and disbands upon project closeout. Some PMOs are client-centered or department-centered, that is, they provide services for projects oriented toward particular clients or for departments where the work is largely project-based, such as IT, research, or product development.

What exactly does a PMO do? The foci and activities of a PMO, shown in Figure 18.5, are described in the following sections, in each case starting with the basic activities of most PMOs and ending with those of only mature project organizations.

1. Project management standards and project management support

Project management methodology

Methodology is the organization's prescribed way to manage a project; if it is the "law," then the PMO is the "law-maker," "law promoter," and "law enforcer." Often the PMO originates the methodology, maintains it, and is responsible for its implementation and improvement.

Project management policies, procedures, standards, and metrics

Application of the methodology requires policies, procedures, standards, and metrics. One example is a policy that requires managers of all projects of a certain size to conform to the methodology. If the methodology includes "Create the Project Plan" it will also specify details

about what constitutes the “plan” and the procedures to create it (e.g. define scope, create WBS, estimate resources, time, and cost, etc.). The PMO sets the policies and defines the procedures.

Ideally the PMO also provides project managers with support and assistance regarding the policies, procedures, or requirements it suggests or mandates. So that the policies and procedures are readily doable and not overly burdensome, the PMO offers various forms of assistance, such as providing clerical, data collection, and data entry support, and easy-to-use standard forms, templates, and checklists.

Mentoring, consulting, and knowledge management

As discussed, the PMO staff can include technical experts and experienced project managers on-call for advice and consultation. Also, the PMO schedules and facilitates team-building sessions, status meetings, and post-completion reviews, and provides facilitators to guide the sessions.

The PMO is the project knowledge management center, not only by virtue of its consulting and mentoring services, but by promoting organizational common knowledge by organizing forums, professional gatherings, and discussion groups where project managers meet and share experiences and lessons learned.

The PMO promotes knowledge management in two ways.²⁸ First, it facilitates knowledge transfer directly between projects by, for example, organizing meetings and forums so managers of different projects can directly share their experiences and learnings with each other. Second, it accumulates, organizes, and retains knowledge gained from multiple past projects (via lessons learned and project post-summary reviews); and transfers that knowledge to new projects via guidelines, project methodology, workshops, and project consultants and support groups (Example 18.3).

2. Project resources

Resource management

A common problem in project-based organizations is that projects simply do not receive adequate resources, which happens when projects are approved and initiated without considering the resources they will require. Short on resources, the organization shifts resources from one project to another; some projects are delayed or deferred so others can be started or finished.

In this capacity, the PMO maintains a record of project resources, such as the number of full-time employees in each job title or skill category. For each resource, the record includes the number allocated to current projects and the number available for new project assignments. This enables the organization to determine for a new project whether sufficient resources are available, additional resources must be acquired, or the project should be postponed or not approved.

In many organizations, the selection and relative priority of projects are set by a project review board (PRB). The PMO provides information about resources so the PRB can determine the feasibility of undertaking each new project and allocate resources to projects with the high-
est priority, using methods as discussed in Chapters 19 and 18, respectively.



See Chapter 18
and 19

Project management software and communication technology

In most project-based companies, all project managers use the same project management software, or set of software, depending on project size. Often this software is integrated with other software for procurement, human resources, and finance, and has Internet/intranet and telecommunication applications. The software comprises an “enterprise project management

system” that is part of the company’s ERP system. Frequently, responsibility for procurement, installation, and upgrade of the software, plus training in its usage and applications, lies with the PMO. For software that requires time-consuming data entry for scheduling, budgeting, planning, and tracking, the PMO provides clerical and data-entry support.

Project facilities

Projects situated at stand-alone sites (e.g., construction projects), away from the home organization (e.g., overseas projects), or involving multiple functions or organizations need a central place—a physical office—for the project staff to meet and work. The PMO arranges for the project office and facilities such as meeting rooms for conferences and forums. For overseas projects, it might also arrange for travel, lodging, and other needs of the project staff.

3. Project manager competency

The PMO oversees most matters pertaining to the skills and abilities of the organization’s project managers; specifically, it:

- Determines skill and competency requirements for project managers.
- Assists in hiring new project managers.
- Arranges for project managers to attend training courses and seminars.
- Prepares career paths for project managers and offers career-path coaching.
- Helps managers in preparing for certification (PMP, CPM, APM, CAPM, RegPM).
- Assists in the assessment and promotion of project managers.
- Offers training in project management methodology, tools, and leadership and communication skills.

4. Liaison with project review board



See Chapter 19

For now, suffice it to say that the **project review board, PRB** is charged with broad oversight of all significant projects, including deciding which to approve, which to defer, and which to kill (discussed Chapter 19). The PMO provides the PRB with the information necessary to make these decisions and serves as its advisor. As each project moves through the gating process, the PRB assesses its performance, partly based on information from website-posted project **dashboards** that compare each project’s performance to that of other projects in terms of a few key metrics. The PMO makes sure that projects arriving at a gate have met the documentation and other gating requirements, and updates posted information on the dashboard for the PRB to review. It could be said that the ability of the PRB to make effective decisions rests largely on the PMO’s ability to provide it with accurate and timely project information. The PMO director sits on the PRB and assists with project selection and priority decisions.

Evolution of the PMO

Creating a PMO is a project in its own right. Sometimes PMOs are established all at once and with the aid of outside consultants, although commonly they are created more slowly and internally. They begin with a small staff and limited purpose, instigated by one or a few veteran

project managers who recognize the need for a standardized approach to project management. Most often this happens in IT or product development (PD) departments where the work is project-based.

The managers who instigated the PMO, with support from a higher-level manager as sponsor, create a project management methodology, plus the procedures, standards, forms, and templates needed to make the methodology workable, and begin to offer training to project managers. Initially, the PMO might consist of one person—the PMO “director,” who (not coincidentally) is often the same person who conceived the PMO and helped create the methodology.

Eventually the director’s role and the PMO staff expand as their responsibilities grow to include counseling, consulting, refining the methodology, and providing clerical and technical support. At first the PMO oversees projects only in the area of the organization wherein it emerged, such as PD or IT. If all goes well, projects in that area will improve and, noticing this, senior management will direct other departments to use the same methodology on their projects. At this point the role of the PMO enlarges to assisting project managers throughout the company in applying the methodology, developing alternative methodologies to better account for the diversity of projects throughout the company, and organizing forums and seminars, accumulating lessons learned, and creating competency lists and knowledge databases (knowledge management). The PMO might also be requested to establish a gating process and assist the PRB in project selection and prioritization. Eventually the PMO might become a full-fledged department wherein all project managers are “based” and from which they are assigned to projects throughout the company.

In response to this section, some readers might react, “That’s not like the PMO in my organization!” Fact is, project managers sometimes view the PMO as being little more than top management’s “project police” whose main purpose is to keep an eye on projects, post red, yellow, or green tickets on the project dashboard, and enforce top-management mandated practices and requirements. Such PMOs are PMOs in name only and are contrary to the intended spirit of the PMO, which is to facilitate project management best practices and enable project managers to do their jobs better.

18.5 Program Management

A **program** (or programme) is a set of projects and other activities organized and coordinated to achieve an *overarching purpose* or goal. Often the projects are interrelated (interdependent or linked in predecessor–successor relationships), share common customers, technology, or resources, and provide a collective capability. An example is an automobile manufacturer’s program to develop “green vehicles.” The program consists of multiple projects, some devoted to developing electric motors, hybrid motors, and alternate-fueled motors, others to developing battery technology and lightweight materials for auto-body components. Like the elements of a system, each serves a function necessary for the success of the larger system—the program.

Program management prioritizes and coordinates the set of projects and other activities to meet program goals and attain benefits that not could not be achieved by any one project alone. Program management is sometimes viewed as an extension of strategic management since it implements strategic initiatives and manages change in the organization or community. The green vehicle program, for instance, can be viewed as the manufacturer’s strategic initiative to move its technology and production toward non-carbon fueled cars.

Project management and program management share common features, but program management is a discipline unto itself. This is necessary because, simply, programs are different than projects.

Programs vis-à-vis projects

The main differences between a project and a program are:²⁹

- A project provides specific deliverables—products, services, or other results—for a specific cost at a specific date. A program provides benefits—e.g., greater revenue, profits, customer satisfaction, or knowledge; better community service; decreased costs or environmental damage; improved business processes and outcomes. The benefits accrue from the deliverables of the projects or other activities and, usually, align with business strategies.
- The program might have no set end date and run indefinitely.
- A project provides outcomes to particular customers and clients. A program provides benefits to multiple stakeholders with differing needs throughout the organization, community, or society.
- A program can evolve over time in response to competitive, technological, or political changes.
- Multiple projects, activities, and resources enable the program to attain *synergy* between projects and benefits that exceed those of the individual projects.

Elaborating on the last point, it says: the benefits of a program go *beyond* those of any of its component projects. For example, when a construction company forms programs based upon the kinds of projects it does, for, say, roadway construction, roadway resurfacing, and water retention systems, each program provides benefits—e.g. consolidation of proposals and work approvals, and streamlining of processes—that could not be attained from the individual projects.

Kinds of programs³⁰

Three common types of programs are goal-oriented, improvement, and portfolio.

A **goal-oriented program** is a group of projects and other activities that, combined, implement an organizational strategy or change, or develop and implement a new application or technology. The program coordinates projects and other activities to achieve overarching benefits tied to business strategies and broad organizational goals. The green vehicle program is one example; the Mercury Exploration program in Case 18.4 is another.

An **improvement and maintenance program** provides regular enhancements to and preservation of existing systems, processes, or infrastructure through advances provided by individual projects. The program serves as the framework for dealing with requests from throughout the organization for added functionality, capacity, performance or maintenance, and does so for the life of the system or process it aims to improve. An example is a hospital adopting methods of “lean production.” The program involves numerous projects and activities—improvement events, training sessions, changes to processes and procedures—that must be coordinated and aligned with hospital goals and ongoing operations.

A **portfolio program** is a group of projects that are otherwise independent but share common resources or technology. The purpose of the program is to coordinate the projects vis-a-vis each other, allocate shared resources, or consolidate procedures so as to improve performance of the overall set of projects. The construction company mentioned above that forms programs based on the same types of projects is an example.

Programs can also be classified by how they are initiated.³¹ Some programs derive from a clear strategy: the program concept is created around a strategic vision, and then projects

and other initiatives are created to achieve the vision. Goal-oriented programs are formed in this way; for example, the NASA Apollo Program goal of a manned lunar landing led to many hundreds of projects that, collectively, fulfilled that goal.

Alternately, a program “emerges” because someone recognizes that pre-existing projects could be better managed if they were organized and coordinated. If the projects are largely independent, they can be grouped into a project portfolio; if they are related and contribute to a greater purpose, they can be grouped into an improvement program.

18.6 Program Life Cycle³²

Programs, per se, have no specific deliverables or end dates, so they do not follow the project life cycle. Rather, those programs that are goal-oriented or improvement-oriented follow a different life cycle—the **program life cycle**. Whereas the *project* life cycle typically ends upon delivery of the end-item, the *program* life cycle can be extended to include the organization’s transition to and initial operation of the end-item. The typical program life cycle is divided into the phases of *Program Initiation*, *Definition*, *Project Execution*, and *Closeout*. At the end of the Project Execution phase, the program might repeat the phases of Definition and Project Execution in cycles (called *Renewal Decision*) until program goals are met, at which point the program moves to Closeout.

Program initiation (or formulation) phase

The **Program Initiation** happens in response to a pressure or need placed on the organization, e.g.: new or changing customers, goals, challenges, strategies, or competitors. An executive (program sponsor) creates a high-level business case that defines the program’s objectives, justifies its feasibility, and aligns it with organizational strategies. Based on the business case, the program governance board either approves or denies the program. If approved, a program manager and the program team are selected.

Definition phase

The Definition phase follows the Program Initiation phase or a **Renewal Decision**, described below. If the phase follows Initiation, it includes creating a more-detailed business case and program objectives, a program plan, and a budget, and defining and sequencing the initial, as-known projects and other work that will comprise the program. It also involves creating an organizational structure for program governance and management, allocating program responsibilities, and setting up operational procedures and systems.

If the phase follows a Renewal Decision, the Definition phase is repeated, this time to revise strategies and goals, establish new or revised projects and other initiatives, and update the program plan and responsibilities. The gate following the Definition phase determines if the program and its component projects are ready to move to the Project Execution phase.

Project execution phase

As the name implies, during this phase the projects and other work activities constituting the program are executed. The program team monitors the projects’ collective performance and

interdependencies, and senior management assesses program benefits and any changes needed to keep the projects or the program aligned with program objectives. The phase includes overseeing delivery of project end-items, preparing the organization for changes, and ensuring support for the installation and operation of end-items. The gate after this phase assesses benefits accrued thus far, program risks, and the projects' alignment with organizational goals, and determines whether the program will be sustained (Renewal Decision) or closed out. With renewal, the program continues as-is or with changes to its direction or composition and begins a new cycle of Definition and Project Execution.

Closeout phase

Program Closeout happens when the objectives are achieved, the program is no longer justifiable, or greater benefits are available in other endeavors. Unfinished projects and other work are terminated or allocated to other programs. A post-program review is conducted for lessons learned, and the program team disbands and is reassigned.

18.7 Program Management Themes³³

Four themes pervade program management: decision management, benefits management, program governance, and managing stakeholder expectations.

Decision management³⁴

How will decisions be made and implemented in the programs? The answer must account for the complexity of program decisions due to the large number of stakeholders, uncertainty about the future, ambiguity over alternatives, and linkages between outcomes and strategic objectives.

Most decision making in programs happens iteratively. Stakeholders discuss matters, agree on a shared vision, and make a series of small decisions aligned with the vision. Decisions and actions are always based on the *benefits* of project deliverables and other activities. Organizations without program management tend to assess projects in terms of ability to meet requirements, but not their longer-term benefits to the organization or its mission and strategies.

Benefits management

Benefits are tangible business improvements that support strategic objectives.³⁵ They can be measured only after the end-items or capabilities of projects or other activities have been implemented and are operational.

Benefits management refers to assessing the organizational impact of the program and managing the interdependent benefits delivered by projects. The benefits as expected during Program Initiation are first defined in the business case. Later, in the Definition phase, a *benefits plan* is created, which specifies how the benefits align with organizational strategies and will derive from program outcomes. This plan also includes a schedule for the realization of benefits, metrics for measuring them, roles and responsibilities for managing them, and means for transferring benefits from the program to the organization. In Project Execution the benefits

are monitored, and in renewal and the Closeout phase responsibility for sustaining the benefits is transferred to customers and other parties.

Program governance

Program governance refers to the way elements of the program are organized and coordinated to meet program objectives. Governance starts with a creating a program vision and objectives based on business strategies and stakeholders' needs, devising ways to monitor the program, and keeping the program aligned with objectives and strategies. It includes the phase-gate reviews mentioned earlier to decide whether the program should be renewed or closed out.

Governance responsibility is shared among the **program governance board** (or steering committee), program director, program manager, and program team. The board—a committee of senior managers—serves many roles: initiate the program, align it with the organization's strategies, approve plans and high-level change requests, review progress and benefits, assist the program manager on difficult issues, assure resource availability, and keep the program in compliance with organizational policies and procedures. The board meets only periodically and thus relies on the program team for program oversight and guidance. In organizations with many programs, governance responsibility is shared with a program management office; this and other program governance roles are discussed later.

Managing stakeholder expectations

Managing stakeholder expectations, discussed in Chapter 16, includes the steps of identifying the stakeholders, their interests, expectations, and influence; communicating with them; and working to increase their acceptance or support of the program and decrease resistance. It is sometimes called **stakeholder engagement** because stakeholders should be engaged in defining and executing the program. Often stakeholders have considerable authority and formal power, which requires a facilitative, shared-leadership style on the part of the program manager. Stakeholders frequently have differing or conflicting expectations, so reaching agreement can involve much negotiation.



See Chapter 16

Key stakeholders are considered program “partners,” and their engagement is a two-way street: the program aims to meet stakeholders' expectations, but at the same time stakeholders must aim to meet program expectations—e.g. cooperate in defining program requirements and expected benefits and, later, supply information and give approvals as requested. Throughout a program, both the stakeholders and their expectations might change, sometimes often.

18.8 Program Organization³⁶

The roles and relationships in a typical program organization are illustrated in Figure 18.6.

Topping the organization are the program director, governance board, sponsor, business change manager, and key stakeholders. The governance board, as mentioned, is responsible for defining the program's relation to organizational strategy and goals, overseeing and guiding the program, and creating an environment that is supportive of the program. The board is led by a **program director**, an executive who “owns” the program and has overall responsibility to ensure it meets objectives and delivers benefits. Often, however, the key decision maker on the board is the **program sponsor**, who champions the program and gives

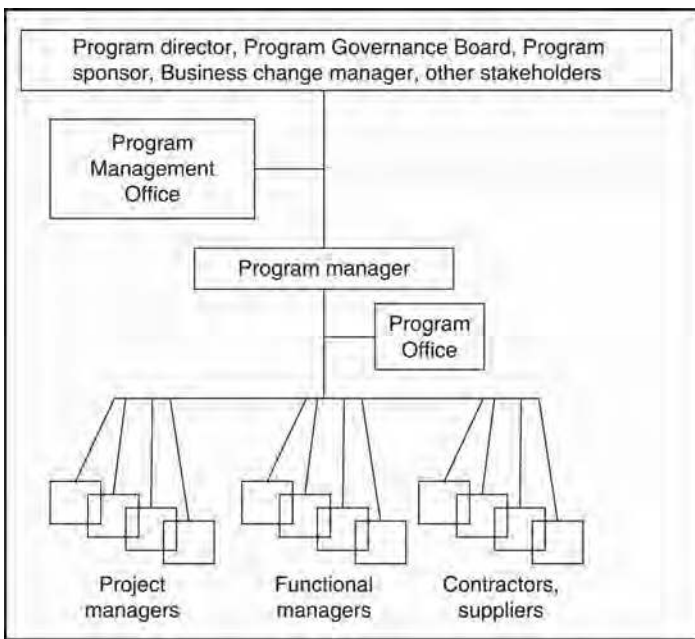


Figure 18.6
Program organization, roles, and relationships.

it top-management’s ratification. She is responsible for the smooth “transitioning” of project end-items into ongoing business operations, and informing the board about resulting benefits. (Sometimes the transitioning responsibility is handled by a special role, the *business change manager*.)

Figure 18.6 also shows a program management office and program office. The two terms are sometimes used interchangeably, but as defined here they are different. The **program office** is the team that assists the program manager in administering the projects or other initiatives within a *single* program. The office handles myriad functions for these projects: contracting, budgeting, training, risk management; eliminating redundancies, tracking outcomes, reporting status; and monitoring and assessing program-level benefits of project outcomes. In large programs, it allocates resources and coordinates efforts among constituent projects. At program closeout, the program office disbands.

In contrast, the **program management office** (PmMO) is a permanent office that provides administrative support to *all* programs. It serves a role similar to the PMO, discussed earlier, except aimed at programs, not projects.

In the middle of the program organization is the **program manager** whose overall responsibility is to achieve the program outcomes as defined in the benefits plan. Specific duties include:³⁷

- Develop program-level plans and schedules.
- Review and approve plans for individual constituent projects for conformity to program strategies, plans, and schedules.
- Be accountable to the governance board for the schedule, budget, and quality of all program activities; provide the board with updates on program progress.

The program manager works with the “program team,” which in Figure 18.6 includes the program office, managers of the constituent projects and other program initiatives, and anyone else involved in administering the program.

18.9 Special Considerations

Most program managers have been project managers and are familiar with project management tools and methods. Upon becoming a program manager, however, they discover managing programs differs from managing projects in many ways, such as the following.

Transition management

Transition management, a central theme in program management, refers to managing the transfer of project outcomes—products, capabilities, knowledge—from program developers to stakeholders (users, operators, customers). It refers to everything associated with the handover of project outcomes to the customer or user, such as taking physical possession, operational testing and training, and monitoring and support. In terms of the system development cycle, it is Phase D, or at least enough of it to determine whether the expected benefits have been realized. Whereas in a project, handover of end-items usually happens just once, at the end of the project, in a program it happens repeatedly, usually at the end of each of a succession of projects or other activities. Consequently, program success hinges on repeated successful implementations and calls for ongoing transition planning and oversight as provided by program management. Transition management utilizes concepts and practices from **change management**, discussed in Chapter 17.



See Chapter 17

Risk and interface management

Program risk management involves identifying, assessing, and managing program-level risks. One such risk is “interface risk,” i.e., risk caused by interdependencies among the program’s component projects. Each project might be the successor or predecessor of at least one other project and, thus, can delay or be delayed by those projects for lack of end-products, information, services, or resources. Addressing such risks is called **interface management** and it involves identifying interfaces (inputs/outputs) among projects and assuring that the inputs necessary for each project will be produced (as outputs) by at least one other project, activity, or external source. It also involves coordinating project schedules so that outputs from one project become available as needed by other projects.

Program managers ordinarily do not meddle with individual project activities. They view each project as an entity with inputs and outputs, and they schedule the projects to account for their interdependencies, when their deliverables are needed, and to allow time for the organization to absorb changes. Appropriate timing of deliverables is often achieved by treating each interface as a contract—a formal agreement between managers of the interfacing projects about what each can expect from the others.³⁸

Work definition

How is the work in a program defined? In a project, work is defined with a WBS. Can a similar approach be used to define program work? The answer depends on the kind of program.

For a typical *improvement program*, the starting point is a long-range (5+ years) plan that specifies the program goals, direction, and priorities. Periodically projects are added to the program based upon their ability to fit the plan and the emerging needs of the organization. Thus, the program work is based on whatever projects are felt necessary to advance program goals; in many cases, this is simply the defined work of the projects selected for the program.

For a *goal-oriented program*, work definition is more challenging since the requirements of the program goal must be defined and then allocated among a set of to-be-determined projects. Often the goal involves innovation, new technology, and uncertain requirements, so program planning happens “incrementally.” Knowledge gained from earlier projects determines next steps and the work of future projects.³⁹ This is similar to the phased-planning process described in Chapter 4. The main work elements in a goal-oriented program can be displayed on a **program breakdown structure** (PBS) that decomposes the program into “program packages.” These packages—the work elements of the program—become the basis for creating the projects that will constitute the program; often, the packages define the top one or two levels of WBSs for the projects.

In a *portfolio program*, the program work is largely “predefined”: it is simply the work of the individual projects that comprise the program. The projects are largely independent, so the program work is merely the sum of the work of the projects and other activities in the portfolio.

Planning and control

Planning for goal-oriented and improvement programs includes identifying the program’s constituent projects. The manager of each project develops a plan that shows his project’s estimated time and resource requirements, and interfaces with other projects. From these the program manager creates a digest that summarizes all the projects’ work, schedules, resource needs, and interfaces. This enables her to create a program-level schedule with milestone dates for program outcomes, and to sequence projects to account for project interdependencies and resource constraints.

Once the program is underway, the program manager tracks expenditures, buffer consumption, and other performance measures for each project, looks for potential problems, and assesses the impact of each project on the others and the overall program. Methods for doing this are covered by sources in the end notes.⁴⁰ Often, the plans and outputs of individual projects must be reworked to accommodate changes in program-level resources and milestones.

Change control

Changes within a particular project that do not affect other projects or the program as a whole are handled by individual project managers, however those that do affect other projects or the program must be handled by the program manager. Similar to project change control, program change control is a process for assessing, then approving or denying change requests, and communicating follow-up actions. The program manager must also address required or requested changes emanating from the outside for their impacts on the program and notify managers of the affected projects to take appropriate action.



See Chapter 4

Procurement management

Most programs are somewhat long-term, and correspondingly so are relationships with contractors and suppliers. Contractors often participate in multiple projects within a program, in which case the contracting emphasis switches from meeting immediate requirements to developing long-term relationships and partnerships based upon the program's and contractors' mutual needs.

Program management misconceptions

Program management should not be confused with multi-project management, or megaproject management.

Multi-project management refers to managing a set of projects that draw from common resource pools. The projects might have nothing in common yet must be coordinated and scheduled so as to meet their individual goals yet not exceed resource constraints. A principle purpose of multi-project management is to enable individual projects to meet their own deadlines and goals, which is unlike program management where the overriding emphasis is on meeting program goals. Unless managing sub-projects provides benefits (other than ease of management) in excess of the sum of the benefits of all the sub-projects, managing such a thing is still multi-project management, not program management.

Normally a very large project—a **megaproject** or super project—is more-easily managed by breaking it down into multiple sub-projects. In a megaproject, however, the sub-projects deliver a single product or system that integrates the elements of all the projects, whereas in a program the individual projects, while interrelated, can each deliver standalone products. Megaproject management is discussed in Chapter 20.



See Chapter 20

18.10 Summary

The first four topics of this chapter—maturity, project management methodology, knowledge management, and the PMO—lie largely or wholly beyond the project manager's responsibility and capability, yet are critical or at least relevant to project success.

The project management methodology provides a framework and set of structured tasks, tools, and techniques to conceive, define, plan, schedule, budget, track, control, and close out projects. The methodology defines the phases or stages of the project and what should happen during each, including the roles and tasks of the project manager and of other project stakeholders. It is the means by which projects in an organization are managed and performed in a standardized, disciplined, and systematic manner, using recognized best practices.

Project management maturity refers to an organization's capability or competency in managing projects, including the extent to which it employs a methodology and formalized methods for planning and control, multi-project integration, and continuous improvement. A high rating on a maturity model indicates the organization has achieved a high level of standardization in its project management practices and processes, but it alone is no guarantee of project success.

Projects are unique and temporary; hence it is easy for individuals and organizations to miss opportunities to learn from project experience, forget what they learned, or not apply learning to new projects. A formal knowledge management process is necessary to learn from

project experience and to retain and share that learning with others. Ways of learning from projects include reviews—mid-stream, post-completion, and after-action. Ways to retain and share explicit knowledge from projects include checklists, databases, and other forms of documentation; tacit knowledge include peer consultation, project resources support groups, and expert knowledge consultants.

The PMO is a unit devoted to supporting project managers and improving project management practice. The PMO establishes and maintains the methodology, instigates initiatives to improve the organization's project management maturity, and manages project knowledge. It develops standards and procedures and manages resources for projects. The PMO provides training, consulting, and mentoring, and assists in integrated multi-project planning and control and portfolio management.

Program management is aimed at managing programs, which are collections of projects and other activities grouped to meet goals and provide a collective capability or benefits that are beyond that of the individual projects. Program management and project management differ in many ways: the life cycle phases, roles and organization structure, themes of management emphasis, and methods for initiation, planning, definition, and control. Consequently, managing programs requires tools and practices beyond project management and uniquely suited for that purpose.



Review Questions

1. What are the benefits of project management methodology? What are the disadvantages of an organization not having one?
2. What does the project management methodology specify? What aspects of project management does the methodology address? Discuss the kinds of tasks and deliverables covered in the methodology.
3. Where does the methodology originate? Who creates and promotes it?
4. What is the purpose of project gates? Describe where the gating process fits into the project management methodology.
5. Why might an organization have more than one methodology? What are the problems with having more than one?
6. Discuss the meaning of the term “project management maturity.”
7. What do project management maturity process models measure or assess?
8. List five levels of project management maturity.
9. Name the benefits of an organization being highly rated on a project management maturity model.
10. What aspects critical to effective project management does the maturity model ignore?
11. In a sentence, what is the purpose of knowledge management in project management?
12. Describe some ways of capturing project knowledge.
13. What is the difference between tacit knowledge and explicit knowledge?
14. Name some difficulties associated with retaining and sharing (transferring) tacit knowledge.
15. What kind of knowledge cannot be retained in a database? Where is that knowledge to be found?
16. What is an after-action review? How does it differ from a post-completion review?
17. How is peer consultation used in knowledge sharing?
18. What responsibility does the project manager have for project knowledge management?

19. What is the overall purpose of the project management office (PMO)?
20. What is the role of the PMO with respect to each of the following:
 - a. project management methodology
 - b. project management policy, procedures, and standards
 - c. project resource management
 - d. project software and communications technology
 - e. mentoring, consulting, and knowledge management
 - f. project manager competency
 - g. project review board (or governance board or project steering committee).
21. How does a typical PMO get started and grow? Describe the role of project managers in initiating and managing the PMO.
22. How do programs and projects differ?
23. Explain the program phases of Program Initiation, Definition, Project Execution, Renewal Decision, and Closeout.
24. Explain the four themes of program management.
25. Explain the following roles within the program structure: sponsor, governance board, director, business change manager, program management office, program manager, and program office.



Questions About the Study Project

1. Did the project follow an established, formal methodology? If so, describe it. What is the opinion of the project manager and project staff as to the effectiveness of the methodology? Where did the methodology originate?
2. If no formal methodology existed, did the project manager use her own, informal methodology? If so, what was it? Was it effective?
3. What is your opinion about the project management maturity of this organization? Is the organization mature or somewhat immature?
4. Was anything done to capture knowledge in this project? Were measures taken to retain this knowledge for application and transfer to other projects?
5. Among the knowledge management methods described in this chapter, which were practiced in this project? How is knowledge shared in the organization?
6. Does the organization have a PMO? If so, what are its functions? How was the role of the PMO visible in this project? In your opinion, did the PMO help or hinder the project manager? Explain.
7. Was the project part of a larger program? If so, try to answer questions 24 and 25 above regarding the program.

CASE 18.1 MAXIM CORPORATION AMERICA (MCA)

Maxim Corporation is a leading provider of risk management services, insurance brokerage, and specialty insurance underwriting. With an employee base of over 50,000 people at 600 offices in more than 100 countries, the corporation has a broad view of the insurance industry and leverages its expertise across hundreds of disciplines worldwide.

IT operations and PMO

The IT operations department for the US division is located at Maxim Corporation America corporate headquarters. Previously the department had over 1,200 employees and was responsible for 80 percent of all MCA IT projects (the other 20 percent going to consultants); it handled three kinds of projects: strategic, infrastructure, and client applications.

In 2009 the ITO department established a PMO to oversee infrastructure projects. The office consisted of a director, support staff, and ten project managers. The director reported to the chief technology officer (CTO), who reported to the global chief information officer (CIO). The PMO's primary role was assigning managers to infrastructure projects and tracking the projects. At any given time about 30 infrastructure projects were underway and many more under consideration.

PM methodology

One of the PMO director's first initiatives was to develop a project management methodology with the assistance of his most-experienced project managers. The methodology, called Project Management Framework (PMF), specifies prescribed project phases, documentation, and gates covering all aspects of the project life cycle from project initiation to completion sign off and postmortem review. It is thought to be quite good: rigorous but not bureaucratically cumbersome.

PMO services

The PMO enforced the methodology and assisted project managers in its usage through training and coaching. Besides this, it conducted courses on topics such as project communication and leadership skills, convened meetings for the project managers to discuss their projects, and sponsored seminars. It created templates and forms to reflect the lessons learned from completed projects and arranged for project managers to be coached and mentored by experienced project managers.

Portfolio management

Also, the PMO assisted the project review board in selection of proposed projects and assessment of underway projects. The PRB is a committee of 10–12 managers that includes the Global CIO, CTO, director of the PMO, VP of Finance, and senior managers with budget responsibility for proposed and current projects. The PMO insured that documentation specified in the PMF for each project was completed and signatures obtained prior to each gate. It also assessed the relative performance of projects to enable the PRB to decide which to approve, hold, or kill at gates.

Reassessment of IT operations

In late 2010 consultants concluded that MCA could save \$30–50 million annually if it outsourced all IT infrastructure operations. MCA responded rapidly and by June 2011 had outsourced all of its IT infrastructure operations to CorCom, a large IT contractor. CorCom had a reputation for operational discipline, solid project management, and good reporting. It had an internal PMO to oversee projects, including those it had acquired from MCA. Of the 600 people in IT Operations at MCA originally working on infrastructure projects, CorCom hired 480.

IT PMO today

The director of MCA's PMO was retained but his unit reduced to four project managers and one support specialist—mostly to oversee tasks associated with the outsourcing of IT projects and the initiation and feasibility of IT projects. The educational role of the PMO has been diminished and its course offerings greatly reduced. The PMO conducts courses to familiarize CorCom staff with MCA's PMF but has ceased providing mentoring and coaching services.

One problem observed since outsourcing IT infrastructure is that CorCom does not become involved in a project until after it has been defined. Whereas in the past MCA project managers were involved during project conception and requirements definition, CorCom project managers are not involved until after project approval and definition. The stance of CorCom managers is, "Tell us exactly what you want and we'll deliver it." This contrasts to the old way of IT telling its customers "Let us help you define your needs and requirements, and suggest the best alternatives." CorCom project managers have no say in defining the requirements they must meet. The concern of some units at MCA is that this lack of early user-developer interaction precludes thorough identification of customer needs. But it is too early to tell if this concern is more than just a perception.

The director is convinced about the continued importance of the IT PMO. He has scheduled a meeting with the Global CIO to discuss the PMO's future.

QUESTIONS

1. Does the IT PMO at MCA have a future? What, if any, role can it retain? Can it assist in user-developer interaction?
2. How does the PMO director's role compare to the VP of projects illustrated in Figure 15.8?



See Chapter 15

CASE 18.2 MOTOROLA'S M-GATE METHODOLOGY AND THE RAZR PROJECT⁴¹

Motorola employs the following 16-stage project methodology called M-Gate:

Note, the M-numbering methodology represents a "countdown" to the end of the product life cycle and roughly corresponds to a five-phase product life cycle:

M15 Idea Concept	M7 Contract Book
M14 Concept Accept	M6 Design Readiness
M13 Solution Select	M5 System Test Readiness
M12 Portfolio Accept	M4 Ready for Field Test
M11 Solution Lock	M3 Ready for Controlled Intro
M10 Project Initiation	M2 Volume Deployment
M9 System Requirements Baseline	M1 Retirement Plan Approved
M8 System Requirements Allocated	M0 End of Life

M15 M14 M13 Business Case	M12 M11 Portfolio Planning	M10 M9 M8 M7 M6 Project Definition	M5 M4 M3 Implementation	M2 M1 M0 Launch and Closeout

Each stage specifies entrance and exit criteria, management and task requirements, and key participants and stakeholders. The full process includes five “go/no-go” gates at which a product’s viability must be proved in order for the project to survive.

The M-Gate methodology emphasizes product quality and customer needs, but it was created before the era of ubiquitous mobile cell phones. It produced some well-known successes—but at the snail’s pace of one every 3 to 4 years. The stages and gates reduce risk and increase quality, but they also discourage new ideas and hold up product launch—big drawbacks in the fiercely competitive handheld phone market.

In fact, the lengthy process had initially killed the RAZR concept that was to become Motorola’s hottest phone in a decade. It imposed cumbersome iterations of market research and mandated requirements that conflicted with RAZR’s design goals. Motorola’s marketing research showed phone sizes increasing, but RAZR aimed for the opposite—to be the thinnest possible (razor thin). As a rule, product designers were required to incorporate whatever features its wireless company customers desired, though for RAZR they thought it better to exclude customers in the interest of secrecy. Only through the persistence of a dedicated cadre of engineers was the project approved. Thanks to high-placed supporters, management allowed RAZR the freedom to operate Skunk Works-like—a small tight-knit team, working in top secrecy and largely by its own rules.

For the RAZR project, stages M15 and M14 were supplanted with a process better suited for break-the-mold products. In terms of the funnel selection method described in Chapter 19 (Figure 19.3b), the process starts with selected and prioritized product concepts streaming out of the narrow end of the funnel. The concepts then go through five stages:

- Stage 1: Prepare a short technical proposal for each product concept.
- Stage 2: Categorize the proposals.
- Stage 3: Develop a resource plan to convert each concept into a prototype.
- Stage 4: Build a prototype to demonstrate the concept to managers and product groups; kill the poorly-received concepts.
- Stage 5: Transfer surviving concepts to the portfolio planning team for entry into a multi-year product portfolio (enter M-Gate process at stage M12).

As soon as the RAZR phone was launched it became an immediate hit, selling more than 110 million units in 4 years and boosting Motorola to second in the cell phone market after Nokia. In 2008 PC World ranked the RAZR as #12 in *The 50 Greatest Gadgets of the Past 50 Years*.

QUESTIONS

1. Why was it necessary for the RAZR team to work outside of the M-Gate methodology? In what situations might it be necessary to work around or modify an existing methodology?
2. What are the potential drawbacks of allowing projects to deviate from the methodology?



See Chapter 19

CASE 18.3 TECKNOKRAT COMPANY

Tecknokrat Company, a software consulting firm, has 18 project managers, many who started as systems analysts and developers when the company was founded in 1977.

Tecknokrat has a good reputation in terms of quality products and services but has recently seen its business and profits fall because many of its projects are completed late and over budget. To reverse the trend the firm hired Drago Kovacic, a project manager who had been PMO director of IT at a bank. Drago's mandate is to assist project managers so as to improve project schedule and budget performance.

In his first two weeks at Tecknokrat, Drago interviewed the project managers and observed them in practice. He noted the following:

- They all have their own way of doing things. There are no prescriptions about how to manage projects.
- They all work in the same office but seldom interact. No one knows much about what the others are doing.
- Some of the managers seem antagonistic toward each other.
- Some seem to be competing with each other.
- There is no mentoring. Old-timers feel: whatever I know I had to learn through experience; new-timers have to do the same.

Digging further, he discovered some curious company policies:

- At year-end the "best" project managers in terms of meeting schedules and budgets get awards: best gets \$20,000, second best gets \$10,000, third best gets \$5,000. Every year for as long as anybody can remember the same four or five people have won the awards; all of them have been with the company over 20 years.
- The company uses education as an incentive. For each project that exceeds goals or receives praise from the customer, the manager can attend a local business seminar of his choice. The incentive tends to go to a small group of managers that, not coincidentally, includes the same group who gets the year-end dollar awards.

The ostensible purpose of the awards and incentives is to spur managers to do a better job in terms of meeting project goals.

QUESTIONS

1. Based on Drago's observations, what do you think are the main issues in Tecknokrat's project management?
2. What do you think about the awards and incentives? Why haven't they had the desired effect?
3. What should Drago do? What difficulties is he likely to encounter?

CASE 18.4 MERCURY EXPLORATION PROGRAM⁴²

An example of program and project management at NASA is the hypothetical Mercury Exploration Program (MEP) for sending a series of three flyby space probes to the planet Mercury. All large space programs at NASA are divided into two phases, Formulation and Implementation, though details of the phases vary depending on the kind of program.⁴³ Projects are also divided into phases, which also differ depending on the kind of project. The MEP consisted of several projects, including one for each of the three “Cosmic” space probes. These projects were divided into four phases: A, conceptualization; B, preliminary design; C, final design and fabrication; and D, launch and operation (Figure 18.7).

One of NASA’s long-range goals is to collect and analyze data about planets in the solar system. The MEP was initiated when NASA scientists urged the director of Goddard Space Flight Center and director of Planetary Programs at NASA headquarters to approve a feasibility study for sending probes to Mercury to conduct geophysical measurements. The study, which initiated the Formulation phase of MEP, would be directed at determining whether the program should be undertaken and, if so, determining the technical course of action and preparing a program plan. A study team of NASA scientists and engineers was appointed by the Goddard director. Besides investigating potential technical approaches for the mission and selecting one, the team identified technological developments and support activities necessary for the program to succeed, created a program management structure, developed high-level program requirements, and prepared preliminary cost and schedule estimates.

NASA management had selected a team leader to oversee the feasibility study, and a liaison officer to coordinate the study with NASA headquarters. Among the requirements for the liaison officer were appropriate technical background and a personality to assure a smooth working relationships with the Goddard director and the study team leader. If the program was approved, this person would become the *program manager* and the study team leader would become the *project manager*.

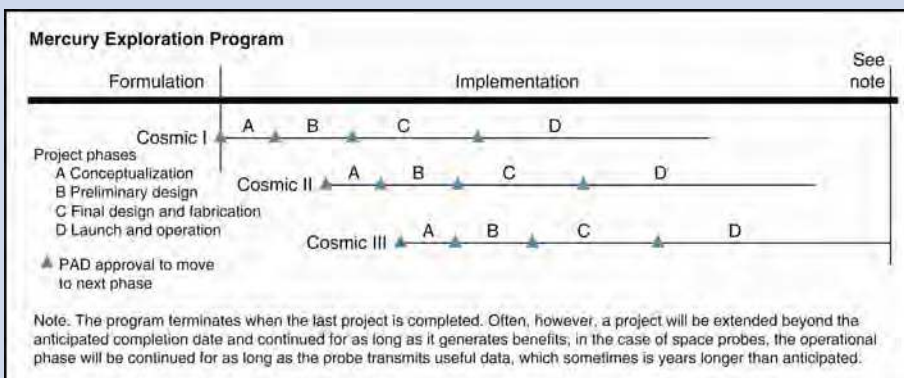


Figure 18.7

Mercury Exploration Program and Cosmic Projects I–III; not shown are various other support projects and activities in the program. Program timescale is 10–15 years.

Based upon the feasibility study, the study team leader and the liaison officer drafted a project proposal and approval document (PAD) for Phase A, conceptualization, of the space probe—to be called “Cosmic.” The document outlined project resources and constraints, estimated the number of space probes and type of launch vehicle needed, and allocated funds and labor. The liaison officer coordinated obtaining all necessary approvals from several NASA divisions and headquarters.⁴⁴

Upon approval of the PAD, MEP was authorized to move into the Implementation phase and the Cosmic I project, the first space probe, into Phase A. The liaison officer was formally named program manager and the study team leader was named project manager of Cosmic I. The program manager’s principal responsibility would be to facilitate all MEP and Cosmic reviews and decisions at NASA headquarters, coordinate work with other governmental agencies, and promote program interests inside and outside of NASA. The program manager position freed the project manager of most liaison work with headquarters and other agencies and provided a “friend” at headquarters to clear away obstacles and provide needed resources and leverage when dealing with other NASA units.

The project manager assembled a team to develop specifications for contractors. (At NASA most actual project “work”—e.g. design, building, and launch of spacecraft—is done by contractors. For example, at its peak, the Apollo lunar program required the support of some 20,000 contractors.) The team prepared schedules and estimated resource requirements, and established relationships with NASA and its contractor teams responsible for major areas of the project such as launch vehicle, reliability, data acquisition, and launch operations. It chose the experiments to be conducted on the missions and determined that a total of three spacecraft missions would be required; besides Cosmic (now called Cosmic I) there would be Cosmic II and Cosmic III; each would constitute a project. Phase A of Cosmic I concluded with a preliminary project plan that specified project technical requirements, launch and tracking requirements, needed manpower and funding, and schedules and milestones to meet project objectives.

The plan was approved by management at Goddard and NASA headquarters; in effect, it became the contract between the program office and Goddard. Thereafter the Cosmic I project manager sent weekly reports to the program manager, and the program manager worked quickly to resolve any snags that required headquarters or other NASA units. For example, whenever a problem arose that required research, the program manager would initiate funds approval to support the research.

The PAD document for Phase A was continuously updated and became the PAD documents for Phases B, C, and D. Upon moving into Phase B, the Cosmic I project appeared on NASA’s information system for reporting and control of financial, schedule, and technical progress.

In Phase B, preliminary design, the project team completed technology development and engineering prototyping, and finalized the preliminary design for the Cosmic I space probe and supporting systems. It selected the launch vehicle to be used for all three missions, the Atlas IX rocket, and a common platform for all three probes. Each probe would be unique, but all of them would be built on this platform and use common navigation, communication, and control equipment; this would help save program costs.

The team also revised the baseline project plan and validated that: all project budgets and schedules were complete and adequate for the anticipated risk; the preliminary design complied with requirements; and the project was mature enough to move into Phase C. Upon headquarters’ approval of the plan the project entered Phase C.



See Chapter 3

During Phase C, final design and fabrication, contractors created detailed engineering designs, mockups, and specifications for all major subsystems on the Cosmic I probe. The project team selected contractors (through an RFP-proposal process similar to the one described in Example 3.8 in Chapter 3) to design, fabricate, and test the space probe, Atlas rocket, and operational systems. The team worked with scientists whose experiments would be on the probes, engineers preparing rockets for the three missions, engineers at Kennedy Space Center where the launches would occur, and scientists at Jet Propulsion Laboratory where data from the space probes would be acquired after launch.

Throughout Phase C the project and program managers participated in numerous formal project reviews. They visited the contractors' plants and participated in meetings for design and test reviews, quality assurance, and system integration. The program manager monitored the project's progress, wrote reports supporting the program's annual budget, and kept the program "sold" at NASA headquarters and to Congressional committees.

Phase D, launch and operation, nominally began when the Cosmic I spacecraft was launched. The project manager oversaw everyone working in this phase, including the launch team, managers of associated NASA projects and programs, scientists whose instruments were on the space probe, contractors that built the probe and Atlas rocket, and the Air Force team that controls the missile range. During countdown before launch, only the project manager had authority to make the final, irrevocable, "go" decision.

Data were recorded between rocket lift-off and successful placement of the space probe in a trajectory toward Mercury, and problems were analyzed so as to avoid repetition on the next probe, Cosmic II. Once communication and instrumentation on Cosmic I were verifiably working and returning usable data, the program manager turned attention to Cosmic II — now in Phase C stage. He continued to monitor Cosmic I's operation so lessons learned from it would be applied to improve the design of Cosmic III, which by then was in Phase B.

QUESTIONS

1. Why was MEP a "program" and not a "project"?
2. What are the distinguishable projects in MEP? Some are named in this case, but what others might there be?
3. Who are the parties/stakeholders in the "project/program team"?
4. What must be "coordinated" among the projects and stakeholders?
5. The program includes three missions to Mercury. What was common to all of them? What was unique to each? In what ways were the projects interdependent?
6. Describe the project manager's role. Describe the program manager's role. Why couldn't one person serve both roles in this program?
7. The case illustrates the phase-gate process. What are the phases and gates? Why do you think the program is managed in this way?

Notes

1. Common language, processes, methodology, benchmarking, and continuous improvement are the five levels of maturity defined by Kerzner H. in *Strategic Planning for Project Management Using a Project Management Maturity Model*. New York, NY: Wiley; 2001.
2. Jugdev K. and Thomas J. Project management maturity models: the silver bullets of competitive advantage? *Project Management Journal* 33(4); 2002: 4–14.

3. Cooke-Davies T., Schlichter J., and Bredillet C. Beyond the PMBOK guide. *Proceedings of the 32nd Annual Project Management Institute, 2001 Seminars & Symposium*. Newton Square, PA: Project Management Institute.
4. Jugdev and Thomas, Project management maturity models.
5. *A Guide to the Project Management Body of Knowledge (PMBOK Guides)*, 6th edn. Newton Square, PA: Project Management Institute; 2017.
6. Kwak Y. and Ibbs C. Project Management Process Maturity (PM)2 Model (Berkeley PM Model), www.ceberkeley.edu/~ibbs/yhkwak/pmmatruty.html; see also Crawford J. *Project Management Maturity Model—Providing a Proven Path to Project Management Excellence*. New York, NY: Marcel Dekker; 2002.
7. Ibbs C. and Kwak Y. Assessing project management maturity. *Project Management Journal* 31(1); 2000: 32–43; PMI. *Organizational Project Management Model*. Newton Square, PA: Project Management Institute; 2003.
8. Milosevic D. and Patanakul P. Standardized project management may increase development project success. *International Journal of Project Management* 23; 2005: 181–192.
9. Pennypacker J. and Grant K. Project management maturity: an industry benchmark. *Project Management Journal* 34(1); 2003: 4–11.
10. Skulmoski G. Project maturity and cost interface. *Cost Engineering* 43(6); 2001: 1–18.
11. Cooke-Davies T. and Arzymanow A. The maturity of project management in different industries: an investigation into variations between project management models. *International Journal of Project Management* 21; 2003: 471–478; Pennypacker and Grant, Project management maturity: An industry benchmark; Crawford L. Senior management perceptions of project management competence. *International Journal of Project Management* 23; 2005: 7–16.
12. Crawford L. Senior management perceptions of project management competence.
13. Jugdev and Thomas, Project management maturity models. An example of an industry study is in Nieto Rodriguez A. and Evrard D. *A First Global Survey on the Current State of Project Management Maturity in Organizations Across the World*. Price Waterhouse Coopers; 2004; Pretorius S., Steyn H., and Bond-Barnard T. The relationship between project management maturity and project success. *Journal of Modern Project Management* 10(1); May/August 2022: 218–231.
14. Jugdev and Thomas, Project management maturity models.
15. This section is based on information from interviews with project managers and PMO directors at 11 organizations: Doug Brandt (Abbott Laboratories); Jacki Koehler (ABN-Amro); Ruta Kulbis (Accenture); Holly Wells (Aon); Carson Neally, Jim Yeck, Robert Wunderlick, and Jeff Roberts (Argonne National Laboratory); Douglas Gilman, Joe Wolke, and Eileen Will (Chicago Board of Trade); Martin Wills (Information Resources, Inc.); Cynthia Reyes (Nicor); Thomas Foley (Sears Roebuck); Gurran Gopal (Tellabs); and Carol Bobbe (TransUnion).
16. Shenhar A. and Dvir D. *Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation*. Boston, MA: Harvard Business School Press; 2007.
17. O'Dell C. and Jackson Grayson C. *If Only We Knew What We Know*. New York, NY: Free Press; 1998.
18. Argote L. *Organizational Learning: Creating, Retaining, and Transferring Knowledge*. Boston, MA: Kluwer Academic Publishers; 1999.
19. Dixon N. *Common Knowledge: How Companies Thrive by Sharing What They Know*. Boston, MA: Harvard Business School Press; 2000.
20. Ernst & Young. *Knowledge Management*. <https://dl.acm.org/doi/pdf/10.5555/359640.359945> downloaded July 25, 2024.
21. *A Leader's Guide to After Action Review* (TC 25–20), US Department of the Army; 1993.
22. Dixon, *Common Knowledge*, pp. 77–79, 81–82.
23. Ernst & Young. *Knowledge Management*; also NHS England, Knowledge and Library Services, *Peer Assist*; <https://library.hee.nhs.uk/knowledge-mobilisation/knowledge-mobilisation-toolkit/peer-assist>, downloaded July 24, 2024.
24. Other examples provided by Ernst & Young. *Knowledge Management*.
25. O'Dell and Grayson, *If Only We Knew What We Know*, pp. 51–52.
26. Cumsumano M. and Selby R. *Microsoft Secrets*. New York, NY: Free Press; 1995, p. 243.
27. Sources: see note 15.
28. Tshuma R., Steyn H., Van Waveren C. The role played by PMOs in the transfer of knowledge between projects: a conceptual framework. *South African Journal of Industrial Engineering*. 29(2), 2018, pp 127–140; Tshuma B, van Waveren C., Steyn H. The supporting role of the project management

- office in the transfer of knowledge between projects: a study of five cases. *International Journal of Project Organisation and Management*, 16(1), Jan 2024; Tshuma B., Steyn H., van Waveren C. The mediation role of the PMO in the transfer of knowledge between projects: a case study of five PMOs. *International Journal of Managing Projects in Business*, 15(1), 2022, pp. 150–174.
29. Pellegrinelli S. Programme management: Organizing project-based change. *International Journal of Project Management* 15(2), 1997, 141–149; PMI. *The Standard for Program Management*, 3rd edn., Newton Square, PA: Project Management Institute, 2013.
 30. Pellegrinelli, Programme management.
 31. Thiry M. *Program Management*. Dorchester, UK: Gower; 2010, p. 39.
 32. Pellegrinelli, Programme management.
 33. PMI, The Standard for Program Management.
 34. Thiry, *Program Management*, pp. 59–63.
 35. Thiry, *Program Management*, p. 77.
 36. PMI, The Standard for Program Management; Thiry, pp. 83–88.
 37. Hanford M. Program management: Different from project management. *DeveloperWorks*, <http://www.ibm.com/developerworks/rational/library/4751.html> Downloaded Dec. 22, 2014.
 38. Kendrick T. *Identifying and Managing Project Risk*. New York: Amacom; 2003, pp. 99–100.
 39. Pellegrinelli, Programme management.
 40. Milosevic D. and Martinelli R. *Program Management for Improved Business Results*. New York: Wiley; 2007; Thiry, Program Management.
 41. Case derived from: McQuellon R., Pini L., Benedicata M., Mjavanadze D., Madura S., Grzybowski M. and Velasco M. *Razr Project*. Graduate School of Business, Loyola University Chicago; February 2007; Lashinsky A. RAZR's edge: How a team of engineers and designers defied Motorola's own rules to create a cell phone that revived the company. *Fortune*, June 1, 2006; Koerner B. Motorola insider blame game: engineers shoved designers aside, *Gizmodo* website, <http://gizmodo.com/5038839/motorola-insider-blame-game-engineers-shoved-designers-aside>, accessed August 24, 2010.
 42. Case loosely based upon Chapman R. *Project Management in NASA: The System and the Men*. NASA, SP-324 Washington, D.C.; 1973, pp. 13–19.
 43. *NASA Space Flight Program and Project Management Handbook*. NASA/SP-2014–3705. Washington, D.C.: NASA Headquarters, September 2014. <http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20150000400.pdf>. Downloaded July 20, 2015.
 44. Program and project approvals occur throughout and at the end of NASA phases and involve a rather complicated review process; the “PAD” described here is a much simplified version of the process.

Chapter 19

Project Selection and Portfolio Management

Many organizations set strategic objectives, and projects are the way they pursue those objectives; hence choosing the *right* projects is critical to their business success. If an organization's objective is to "be the low-cost leader," "expand market share in India," or "adapt to and mitigate climate change" then you would expect that many or most of its projects would be directed at those objectives. But often that is not the case. In many companies, projects have little to do with strategic objectives and, instead, represent short-term interests, easily seized opportunities, or the agendas of a few people. "Pet" projects of senior executives get sacred status despite questionable benefits and hog resources from projects of obvious greater business value.

A study of 35 predominantly North American firms revealed relatively little spending on projects that contribute to company goals.¹ In general, project resources were spread thinly because companies had too many projects and no systematic way to prioritize them. Most projects were "low-hanging fruit"—relatively easy to do but offering few business benefits; such projects waste resources and deprive a company of business opportunities.

Project portfolio management (PPM) includes processes and techniques to support decision making in the selection and prioritization of projects and provides an effective way to implement strategy and maximize the business value of investments in projects. By providing objective, transparent and consistent decision-making processes, it facilitates and depoliticizes project decisions that often are politically charged—at least in theory. Limited funds and other scarce resources are allocated in rational ways to maximize business value.

19.1 Project Portfolio Management

A **project portfolio** is a group of projects and programs aimed at achieving strategic objectives that share resources and compete for funding. Each portfolio supports a theme—for example, a strategic objective, product line, business unit, market, or geographical area of operation. Any program or project that contributes to or falls within a particular theme is added to the portfolio; most organizations would have several portfolios.

As shown in Figure 19.1, a portfolio can consist of programs, projects, other work activities, and even other portfolios. Unlike projects in a program, projects and other elements in a

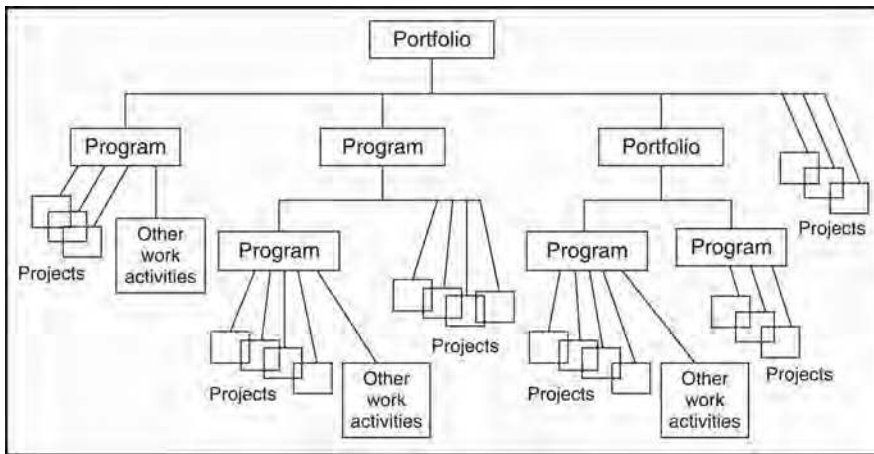


Figure 19.1
Portfolios, programs, and projects.

portfolio are not necessarily interdependent. They are grouped in the portfolio for the purpose of better managing them as a whole and for prioritizing and allocating resources among them to best achieve organizational goals.

By definition, any organization that manages and allocates resources to more than one project has a *project portfolio*—whether or not managers recognize it.² Organizations familiar with the concept of a project portfolio commonly conform to a process called **project portfolio management (PPM)**, which, briefly, involves two steps:

1. Creating portfolios—defining “themes” around which to form them and criteria for including projects and programs in each portfolio; categorizing current and proposed projects and programs into particular portfolios.
2. Managing portfolios—within the portfolio, assessing proposed projects and programs to decide whether each should be approved, put on hold, or rejected, prioritizing approved projects based on criteria such as risk and contribution to strategic objectives, allocating resources so priority projects get adequate funding; and tracking and managing projects and programs collectively in order to maximize the returns on the investments made.

Example 19.1: Portfolios of a Brazilian Mining Company

A mining company in Brazil classifies each of its projects into one of four portfolios:

- *Compliance* projects—projects required to conform to legislation. These projects *must* be done and, thus, have high priority and enjoy preference in the allocation of funds and other resources.
- *Environmental and social responsibility* projects—projects to, for example, treat mine effluent water and projects to provide roads and supply fresh water to local communities and construct sewerage treatment plants.

- *Stay-in-Business (SIB)* projects—projects to (a) *maintain* equipment and processes on current operations, and (b) *improve* current operations, processes and equipment.
- *Growth* projects—projects to expand the business; includes projects to develop new mines and develop new processes and products.

Senior management distributes available funds among the portfolios, while portfolio managers advise senior management on how to allocate those funds.

The most rudimentary form of portfolio management is to simply track projects underway and under consideration. The organization has two lists: one for “active” projects, another for “potential” or on-hold projects. Simple as it might appear, creating and maintaining such lists in the absence of a formal portfolio management process is not trivial since managers routinely start projects without registering them and don’t keep lists of all current and proposed projects.

Academics, consultants, and software firms have proposed many approaches to project portfolio management and both the PMI and the UK Office of Government Commerce have developed portfolio management standards and certifications.³ Software is often used to manage data, facilitate analyses, and to streamline reporting. The breadth of the subject fills books; hence, treatment here is limited to a survey of common principles and approaches.

Process for successful projects⁴

Successful projects and programs depend upon two things: doing the *right projects*, and doing those *projects the right way*. The two happen in a four-phase process that involves senior managers, business unit managers, and project managers:

- **Strategic management:** focus the organization. Senior managers articulate the vision and mission of the organization, define strategies, set budgets, and allocate resources to business units. Some examples of contrasting strategies are to be the low-cost or technology leader, be innovative or imitative, or pursue mass or niche markets.
- **Portfolio management:** choose the right projects. Business unit managers develop strategies, goals, and initiatives consistent with the corporate mission and strategies. Each goal or initiative becomes the theme for creating a portfolio and setting specific criteria, which become the basis for selecting projects from proposals generated internally or by customers.
- **Gating methodology:** nurture or get rid of projects. Business unit managers assess each project as it moves through gates by comparing its performance to other projects’ performance and gating criteria. Important but struggling projects are allocated more resources; poorly performing or mediocre projects are put on hold or cancelled.
- **Project management:** do projects the right way. Project managers guide projects using principles and practices of project management as described throughout this book.

Project portfolio manager

Portfolio management requires roles for decision making and a clear structure. The **project portfolio manager (PPM)** is charged with oversight of the project portfolio. Her aim

is to achieve organizational objectives through the portfolio's investment in programs and projects; thus, she has a key role in guiding the organization in the right direction and, consequently, has a much broader and longer-term business perspective than project and program managers.

Working with the project review board (described below), the portfolio manager's role is to:

- Ensure projects and programs align with strategic objectives and initiatives.
- Assess proposed projects and programs in terms of potential benefits, required resources, and risk.
- Approve or recommend projects that can achieve strategic objectives within acceptable risk and resource constraints.
- Cluster related or interdependent projects and programs into portfolios.
- Prioritize programs and projects based upon contribution to strategic objectives, resource requirements, and risks.
- Develop a resource plan and allocate resources to programs and projects.
- Seek an investment optimum by designing a mix of programs and projects to exploit synergies among them and provide balance in terms of risk, size, duration, etc.
- Monitor, review, and assess projects at phase gates for business impact and business justification.
- Report assessment results and recommendations to senior executives.
- After projects are completed, assess and report the extent to which they delivered benefits as claimed in the business case.

Aspects of the PPM's role may appear similar to that of a project or program manager. While each project manager champions and may fight for his project's existence, the PPM looks at the project's overall *benefits to the organization* and may recommend scaling back or terminating a project or even a program.

Ideally the PPM is experienced in project management and program management; more important, however, is that she understands the organization's business environment (e.g. markets and competitors), capabilities, competitive edge, and strategies, and interacts well with executives and senior stakeholders.

In some cases, the portfolio management function becomes a responsibility of the PMO, and when that happens the office is referred to as a **PPMO (project portfolio management office)** or PPO (project portfolio office). Usually, the PPM is physically located in a PPMO. The role of the PPM and PPMO is to provide advice and support to senior-management decision makers. Final decisions are made by a senior executive or, in the case of major projects, the company board. PMOs are discussed in Chapter 18.



See Chapter 18

Project review board

The portfolio manager shares responsibility for project selection and portfolio management with the **project review board** or **PRB** (aka portfolio management team, project investment committee, project council or governance board). PRB members typically include the

- Portfolio manager
- Chief financial officer (CFO)
- Chief risk manager (CRO)
- Chief human resource officer (CHRO)

- Director of the project management office (PMO) or project portfolio management office (PPMO)
- Chief technical officer (CTO)—someone from IT, engineering, or product development.

For each project proposal, the CFO weighs the costs and financial benefits, the CRO assesses the risks, the CHRO assesses the human resource requirements, the CTO assesses the technical benefits and difficulties, and the PMO director compiles documentation required for selection and gating decisions.⁵ The portfolio manager typically chairs the PRB and may have final say over project additions or deletions in the portfolio.

For research and engineering projects the PRB will include a group of technical “peer reviewers” who independently appraise and rate proposals according to scientific or technical merit, success likelihood, and competency or capability of the proposal originators. If they all assign low scores to a proposal, the project is rejected. If all assign high scores, the project is approved—provided others on the PRB also approve it and funds are available. When funding is tight, few projects are approved, regardless of high scores. When it is abundant, even mediocre-rated projects might be approved.⁶

19.2 Framework for Project Selection and Portfolio Management

In organizations where projects are generated internally, the portfolio management process is used to evaluate proposals and approve projects; in those where projects are generated externally by potential customers, it is used to determine to which RFPs the company should respond.

Projects differ regarding resource requirements, risk, cost, and strategic value, so choosing the right projects is not easy. Since most projects represent investments, many of the methods used in project portfolio management derive from methods in investment management. Just as an investment portfolio can reduce monetary risk by, say, investing in multiple currencies (pound, euro, yen, or dollar), a project portfolio can reduce risk by incorporating projects from multiple business sectors.

Selection process

An organization that routinely faces project selection decisions should follow a consistent and objective process for assessing and comparing project proposals. The process should use a set of *measurable* criteria that reflects the organization’s strategic goals and initiatives.

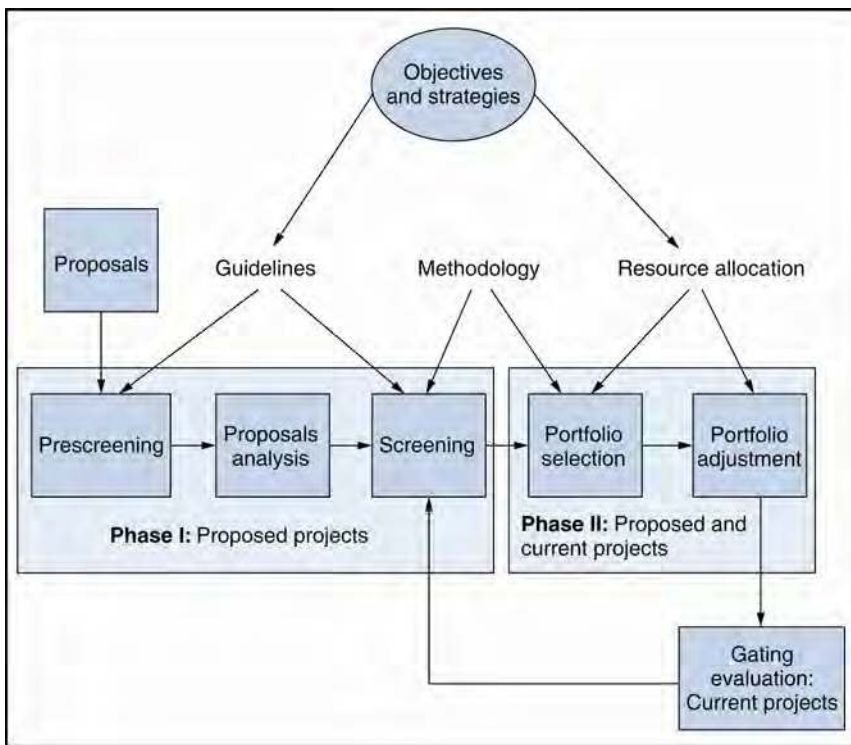
The project evaluation and selection process and its relation to other aspects of portfolio management are shown in Figure 19.2. In Phase I each project is independently evaluated and screened; in Phase II, all projects are compared, and a subset is approved.⁷ The framework largely also applies to the selection of programs and to the projects that comprise them.

Phase I

Phase I starts with a **prescreening stage** to eliminate clearly deficient project proposals; this corresponds to the “pre-project” or FEL-1 stage discussed in Chapter 4. To pass this step a project must be justified in terms of organizational survival or growth.⁸ Survival projects are necessary for the health and continued viability of the organization. Growth projects, though not essential for survival, expand or take advantage of opportunities for the organization. The



See Chapter 4

**Figure 19.2**

Project analysis, selection, and portfolio management methodology.

Source: Concept adapted from Archer N. and Ghasemzadeh F. An integrated framework for project portfolio selection. *International Journal of Project Management*, 17(4); 1999: 207–216.

justification is documented in a business case report that confirms the project is compatible with organizational strategies, worthwhile, and viable (expected benefits exceed expected costs). A project also needs a champion and sponsor who support it. Projects lacking justification, support, or sufficient information upon which to decide are rejected. Sometimes a simple checklist with a small number of criteria is employed and each proposal is rated as excellent, good, or poor. Proposals falling below a “threshold” composite rating are automatically rejected.

Proposals that pass prescreening are subjected to **analysis** using quantitative and qualitative models and scoring methods. The analysis might rate or value the proposal in terms of diverse criteria such as “link to strategic objectives,” “financial value,” or “compliance to constraints,” which leads to a more detailed, verified business case (sometimes called a “bankable business case.”) This analysis is included in the FEL-2 process discussed in Chapter 4.

In order to be considered and approved for Phase II and funding, a proposal must exceed a minimum cutoff value or score; such is the purpose of **screening**—to determine which projects meet minimal requirements for benefits, risk, or other specific criteria.⁹

Phase I restricts the pool of projects entering Phase II to only the “right” projects and generates information for portfolio selection decisions in Phase II. To discourage project proposals that are frivolous or clearly sub-par, RFPs and project initiation procedures should clearly specify the minimal requirements for project approval.



See Chapter 4

Phase II

The first step of Phase II is **portfolio selection** wherein projects approved in Phase I are reviewed together with existing projects to determine which combination of them would constitute the “best” portfolio. Projects are compared in terms of scores from the proposal analysis or, for existing projects, measures of their current status and performance. All projects, proposed and existing, are compared against each other using the same criteria.

Phase II is not a single event but an ongoing process. Existing projects are evaluated, periodically or at the end of each stage of the execution phase for expected business impacts, risks, performance, and costs using the **gating process**. Those in trouble and not meeting minimal requirements are terminated outright. The remaining ones are pooled with new projects, and all are rank ordered for likelihood of continuation, reduced or increased support, or cancellation. This **portfolio balancing** step helps ensure that high-priority projects receive resources and funding. Because objectives, opportunities (new strategies, RFPs, and proposals), threats, resources, and the external environment periodically change, so will the portfolio: new projects are added while some current projects are accelerated, delayed, or cancelled. The gating process is discussed more fully later in the chapter.

Funnel and filter

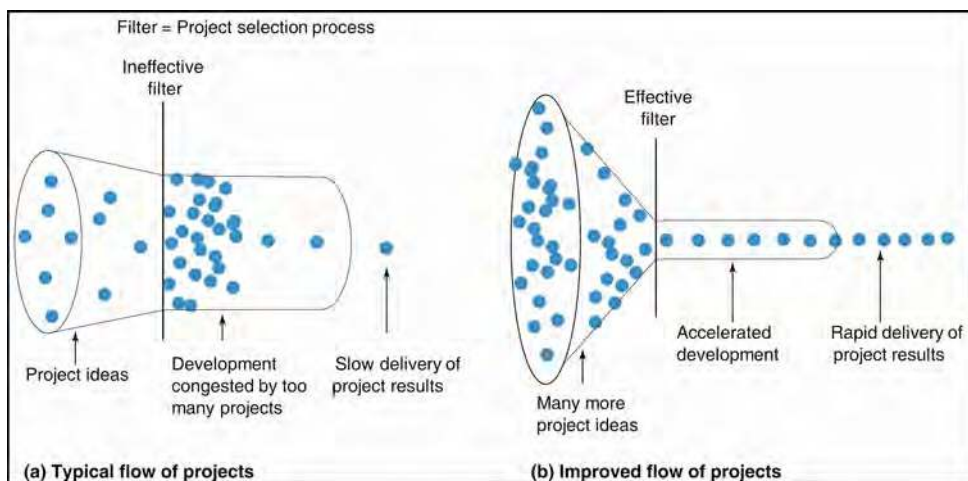
Whether doing work externally under contract or internally to develop new products or processes, every company needs projects for survival and growth; ideally, it encourages proposals for as many innovative projects as possible, and has many proposals, RFPs, and initiation requests to choose from. But a frequent problem is that companies approve and then attempt to execute *too many projects at once* and beyond the capability of available resources. For example, a key resource (such as an engineer) might be able to work on just two or three projects concurrently, yet companies might try to stretch that resource to over twice that number. To avoid this problem, a good selection process should make sure that only the “right” projects are selected and pursued at the right time. Hence, the selection process can be likened to a **funnel** into which many proposals flow, and a **filter** through which only the best emerge. This is illustrated in Figure 19.3. The trick is to design the process with a funnel mouth wide enough to take in many proposals, but a filter fine enough to screen out bad proposals yet provide a constant flow of high-quality projects.¹⁰

19.3 Methods for Assessing Individual Projects

Project selection is based upon analysis of individual projects. Each analysis incorporates assumptions, some that later might prove to be wrong; for this reason the analysis should always be thoroughly documented and include the stated assumptions. Among the most common analysis methods are **financial models** and **scoring models**.

Financial models

Financial models measure project proposals in terms of economic or financial criteria. One example is **net present value (NPV)**. Stakeholder investors expect value for the money invested in a project. Because of the time value of money, values and costs that materialize at various times can be compared only if they are discounted to the same point in time. NPV is

**Figure 19.3**

Project selection process as funnel and filter.

Source: Concept adapted from S. Wheelwright and K. Clark, *Revolutionizing Product Development*. New York: Free Press, 1992.



See Chapter 11

the present value of all future benefits (discounted at a specific rate) minus the present value of all costs incurred (discounted at the same rate).

Another example is **expected commercial value (ECV)**, an application of decision-tree analysis described in the Appendix to Chapter 11. The ECV model, illustrated in Figure 19.4, considers the costs, earnings, and success likelihood of the development and launch of a new product.¹¹ Suppose the product's development cost is \$10M, launch cost is \$1.5M, NPV for the future stream of earnings is \$50M, and probabilities for success are 80 percent in development and 60 percent in the market. Then,

$$ECV = [(\$50)0.6 - \$1.5M] 0.80 - \$10M = \$12.8M$$

Generally, the higher the ECV, the more preferred the project.

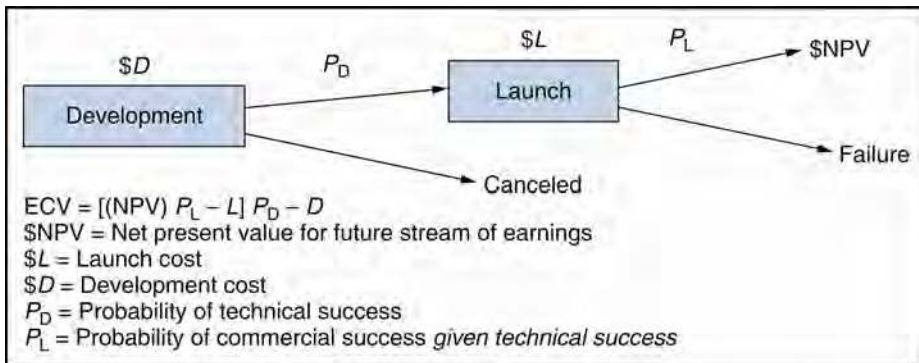
Another financial model, the **benefit/cost ratio**, or **B/C**, weighs the benefits of a project against its costs. A simple example is:

$$B / C = \frac{\text{Estimated revenues} \times \text{probability of success}}{\text{Estimated cost}}$$

Values in the numerator and denominator are expressed in the same form, either as annualized or present value amounts. For example, if the estimated annual revenue of the project is \$100,000, estimated annual cost is \$25,000, and probability of success is 50 percent,

$$B / C = \frac{\$100,000 \times 0.50}{\$25,000} = 2.0$$

Thus, for each dollar spent on the project, two dollars in benefit would be expected in return. B/C can also be computed for other forms of benefits.¹² For instance, in the ratio

**Figure 19.4**

Model for computing expected commercial value, ECV.

$$B / C = \frac{\text{Value of benefits}}{\text{Capital recovery cost} + (\text{operating cost} + \text{maintenance cost})}$$

the “value” can be cost savings. Suppose, for example, renovation of a factory and installation of new equipment will provide an expected present worth savings of \$6M for a present worth cost (facility renovation, equipment installation, and annual operating and maintenance expenses) of \$3M. The B/C ratio for the project is 2.0.

Beyond those mentioned, some other financial models are **internal rate of return (IRR)**, **return on original investment (ROI)**, **payback period**, and **life cycle cost (LCC)**, the last of which is discussed in Chapter 9.



See Chapter 9

Caveats of financial models

The accuracy of financial models such as B/C depends on the accuracy of values for all relevant costs and benefits, including “hidden” or external ones such as impacts on the society, economy, and environment.¹³ Hidden costs and benefits can be difficult to identify and measure and often exceed by far the more obvious costs and benefits. In a renovation project, for example, suppose after the project begins the factory electrical system is discovered to be out-of-code and need replacement, and the flooring is determined to be unsound and in need of reinforcement; or, suppose environmental regulations change, requiring new equipment to clean up smoke and liquids discharged from the factory. Not anticipating these costs would result in an inaccurate and misleading B/C ratio.

The main weaknesses of financial models are overreliance on *estimates* for costs, savings, future streams of earnings, probabilities, etc.; lack of data to estimate these values during project conception; and project supporters’ tendency to understate costs and overstate benefits.

Another weakness of financial criterion is neglect of non-financial criteria of equal or more importance. NPV, for example, does not measure the strategic value of a project or the extent to which a project would contribute to, for example, expanding into new markets, preparing the organization for future technologies and processes, gaining compliance with standards or regulation, improving the organization’s image, developing new business partnerships, or capturing new customer relationships. Some projects exist to gain *new knowledge*, yet the value of such projects can be enormously hard to express monetarily.

Example 19.2: Value of Knowledge

At this moment, a semi-autonomous rover is on the surface of Mars collecting soil samples and storing them in little capsules. As part of the Mars Sample Return mission (MSR), scientists and engineers around the world are working to develop a small fleet of spacecraft to travel to Mars, land, retrieve the capsules, and rocket them back to Earth for analysis—expected to happen 20, maybe 30, years from now. The estimated cost of MSR, once set at \$4 billion, is now \$8–10 billion.

Currently orbiting the sun about 1 million miles from Earth is the James Web Space Telescope (JWST). Since early 2022 it has been sending back data and stunning images of the cosmos and is helping astronomers and cosmologists to understand the origins of the universe, but the project to develop and launch it took 20 years and cost \$10 billion. Since, ideally, the benefits of a project outweigh the costs, the amounts spent on MSR and JWST must represent the *minimum value* being placed on what will be *learned* from these programs—whether there was ever life on Mars (MSR—\$8–10 billion) and how the universe began (JWST—\$10 billion).

Many companies focus primarily on financial models, even though the models on their own are insufficient for assessing project success.¹⁴ Methods that supplement financial models are discussed next.

Scoring models

Scoring models rate projects in terms of multiple criteria that, besides quantifiable measures, include non-quantifiables such as market risk, customer enthusiasm, or fit with company goals—whatever criteria are thought important and discriminate between projects.

In the simplest models, a project is rated on each criterion according to a scale (say, 5 = excellent, 4 = good, 3 = adequate, 2 = poor, 1 = bad), and scores for all criteria are summed to yield an overall project score. The ratings are weighted when the criteria are of different importance.

Table 19.1 illustrates a scoring method that includes probabilities and weights. On the left are the scoring criteria, followed by five columns (“Very Good” through “Very Poor”) that give the *expected probability* that the project will fit the criteria. For example, the probability that the long-range outlook for the product of the project will be “Very Good” is 80 percent. The way these probabilities are obtained depends on the information available and can range from gut-feel to sophisticated quantitative analysis. For example, the score in the table for “Risk level acceptability” can be opinion-based or derived from analysis of risk impacts and probabilities, explained in Chapter 11. As with all analyses, the more data available and more experienced the scoring team, the more accurate the estimates.

Numbers in the Expected Rating column in Table 19.1 are calculated in each row as the sum of the probability times the score. The Expected Rating for long-range outlook for the product, for instance, is $0.8(4) + 0.2(3) = 3.8$.

The next column, Weight, reflects the relative importance of the criteria (a criterion weighted 10 is considered twice as important as one weighted 5); sometimes the weights are set to total to 100, as shown. The next column, Weighted Expected Score, is the Weight multiplied by the Expected Rating; for the long-range outlook of the product, it is $3.8 \times 10 = 38$.



See Chapter 11

Table 19.1 Project weighted scoring model.

Criteria		Scores					Expected Rating	Weight	Weighted Expected Score
		Very Good 4	Good 3	Fair 2	Poor 1	Very Poor 0			
Long-range outlook	1. Product	0.8	0.2				3.8	10	38
	2. Market	1.0					4.0	10	40
Meets objectives	1. ECV	0.8	0.2				3.8	5	19
	2. ROI		1.0				3.0	6	19.0
	3. Image		0.6	0.4			2.6	4	10.4
Fits strategy	Phase I	0.8	0.2				3.8	10	38
	Phase II	1.0					1.0	5	5
	Phase III	0.6	0.2	0.2			3.4	5	17
Goal contribution	Goal A	0.2	0.8				3.2	10	32
	Goal B	1.0					4.0	5	20
	Goal C		0.2	0.2	0.6		1.6	4	6.4
Risk level acceptability	0.7	0.3				3.7	10	37	
Competitive advantage	0.9	0.1				3.9	8	31.2	
Compatibility with other	0.2	0.7	0.1			3.1	8	24.8	
Total								100	336.8/400

Sources: Adapted from Cleland D. in *Project Management: Strategic Design and Implementation*, 3rd edn. New York, NY: McGraw-Hill; 1999; reprinted in Dye L. and Pennypacker J. (eds.). *Project Portfolio Management*. West Chester, PA: Center for Business Practices; 1999, pp. 3–22.

The bottom of Table 19.1 shows the Total Weighted Expected Score (sum over all criteria), 336.8 out of a possible maximum 400. This score is used to screen the proposal in Phase I or rank order it with other projects in Phase II.

One limitation of scoring methods is that they ignore the resources needed to implement projects. Big projects tend to get more attention and score higher than small projects, but they consume more resources and shut out other projects, even important ones. This limitation can be offset by simultaneously considering both a project's required funds or resources and its score or rating, as in the cost-effectiveness method, described later.

19.4 Methods for Comparing and Selecting Projects

Proposed projects that have survived Phase I are compared with current projects in Phase II to determine which combination of projects constitutes the best portfolio. The result will be to add some new projects to the portfolio and to drop some current projects.

In their review of project portfolios in product development, Cooper et al. found that project selection approaches tend to aim at the following goals.¹⁵

- Maximize the value or utility of the portfolio.
- Achieve balance in the portfolio.
- Fit the portfolio to the organization's objectives and strategic initiatives.

Value or utility

Value or utility methods select projects with the highest "value" or usefulness as determined from financial models or scoring methods.

Single-criterion methods

These methods rank order projects according to a single value or utility measure (e.g. ECV from model in Figure 19.4 or score in Table 19.1), and the highest-ranked ones are selected subject to resource availability. A minimum-value threshold can be applied for screening proposals, such as rejecting those with a B/C ratio of less than 1.5, score of less than 50 percent maximum (200/400 in Table 19.1), or IRR of less than 8 percent (called the *hurdle rate*).

Other valuation methods, beyond our scope, include mathematical programming techniques to select the combination of projects that maximizes the portfolio value subject to project dependencies, limited resources, and other constraints.¹⁶

Of course, computed estimates of financial value are based upon *assumptions* about the values of the input variables, the validity of which is always open to question. In a **sensitivity analysis**, the values of input (independent) variables are altered to determine the effect on the project's estimated financial value (dependent variable); in other words, the analysis tests how sensitive the estimated financial value is to changes in the input variables (what happens to ECV if, e.g., costs rise 30 percent and the exchange rate increases 10 percent). By measuring the effect of changes in each or a combination of input variables on the calculated financial value, the *range* of values for input variables that yield an "acceptable" project financial value is determined. A project whose financial value is sensitive to even small changes in input values is considered risky.

The obvious drawback of single-criterion methods is reliance on a single value to rank order projects, which can be risky because underlying estimates of costs, benefits, probabilities,

Table 19.2 Multiple criteria rank-ordered list.

Project	Strategic Fit		ROI		Risk		Ranking	
	Rating	Rank order	%	Rank order	Risk	Rank order	Weighted rank	Rank order
A	4 (best)	1	5 (worst)	4	4 (best)	1	12	1
B	1 (worst)	4	30 (best)	1	2	2	16	3
C	2	2	25	2	2	2	12	1
D	2	2	12	3	1 (worst)	3	15	2
Weight	3		2		1			

etc. are often fraught with inaccuracies. Also, the methods tend to be laden with assumptions that, if incorrect or overlooked, can lead to erroneous conclusions. Rank ordering of projects according to B/C, for instance, assumes that all the projects are comparable, even though often they are not. Project A with a B/C of 3.0 would be ranked ahead Project B with a B/C of 2.0 even if Project B had a benefit of \$2 million and Project A had a benefit of only \$200,000.

Multiple-criteria methods

A project can be valued in many ways, although it might be valued high on one criterion but low on another. To overcome the conundrum of which criterion to use, some methods employ multiple criteria.¹⁷ For example, Table 19.2 rates each project for three criteria: Strategic Fit (subjective rating 4–1, where 4 is perfect fit and 1 is poor fit); ROI; and Risk (subjective rating 4–1, where 4 is no risk and 1 is high risk). Project scores for each criterion are compared and the projects ranked. For example, in Table 19.2 Project A is ranked 1 for Strategic Fit because it scored highest of the four projects, (note, projects are ranked the same when their scores are tied). Project A is ranked 4 for ROI because of the lowest ROI value; and it is ranked 1 for Risk because it scored worst in risk. The Weighted rank is computed by using the following weights: 3 for Strategic Fit (most important), 2 for ROI and 1 for Risk (least important). For example, for Project B the weighted rank is $3 \times 4 + 2 \times 1 + 1 \times 2 = 16$. Since this is the largest weighted rank value, Project B ranks third and last and will receive the lowest priority.

By accounting for multiple criteria and allowing additional criteria to be added as desired, this method somewhat assures that “good” projects (in terms of the financial or scoring criteria used) are retained as candidates for selection. A limitation of this and all value methods is that they alone do not guarantee that the projects constituting the portfolio will be “balanced” or aligned with organizational objectives and strategies.

Example 19.3 Assessing Climate Change Adaptation Options Using Multi-criteria Analysis¹⁸

Glaciers everywhere are melting, including in mountainous regions where the melt threatens mountain villages with landslides, and lowland cities and crops with flooding. As part of its National Adaptation Programme of Action, the Himalayan country of Bhutan assessed

its vulnerability to these climate-change threats and ways to adapt to them. An expert task force team representing agriculture, biodiversity, forestry, natural disasters, infrastructure, health, and water resources identified and ranked possible adaptation projects using multiple criteria.

The team began by identifying the most likely and severe climate-related hazards and sectors at highest risk; these included:

- Increased glacial lake outburst floods, landslides, and flash-floods
- Most vulnerable sectors: agriculture and hydropower
- Most vulnerable communities: the rural poor.

The team considered 17 options (potential projects), then shortened the list by assessing each option for:

- Severity of climate change effects addressed by the option
- Cost effectiveness
- Level of risk (by not adapting this option)
- Fit with country goals such as overcoming poverty and enhancing adaptive capacity.

Table 19.3 shows five shortlisted options, ranked 1 to 5 according to the following criteria:

- Estimated project cost
- Human life and health saved or protected
- Arable land with water supply (for agriculture/livestock) and productive forest (for forestry) saved
- Infrastructure (hydropower plants, communication systems, industrial complexes) and monuments (cultural sites, religious sites, main tourist attractions) saved.

To arrive at the ranking, the task force team divided into three sub-teams, each assigning scores to the criteria for each project. Those scores were combined and standardized on a scale from 0 (no impact) to 1.00 (high impact), as shown in Table 19.3. The team then computed a summary weighted score based on the four criteria (for example, for Project 1, Disaster Management: $0.71 \times 0.2 + 1.00 \times 0.33 + 0.75 \times 0.27 + 0.25 \times 0.2 = 0.7245$). It then adjusted the weighted score for geographical impact of the projects (15 percent increase for projects with national impact; 15 percent decrease for those with only local impact). The initial ranking, adjusted score, and final ranking are shown in the last three columns of Table 19.3.

Based upon the study results, Bhutan received funding from international sources for the two highest-ranked projects: disaster management and lowering of Thorthomi Glacier Lake.

Table 19.3 Results of the ranking of prioritized adaptation options.

Criteria	Estimated cost	Human life/health saved	Arable land, water supply etc. saved	Essential infrastructure and monuments saved	Summary weighted score	Initial rank	National (N) Regional (R) Local (L) Multiplier	Adjusted ranking
Weight Project	0.20	0.33	0.27	0.20			N +15% R +/- 0% L -15%	
(1) Disaster Management Strategy (Food Security and Emergency Medicine)	0.71	1.00	0.75	0.25	0.7245	2	N 0.833175	1
(2) Landslide Management & Flood Prevention	0.56	0.75	0.75	0.50	0.662	4	R 0.662	4
(3) Weather Forecasting System to Serve Farmers and Agriculture	0.81	0.75	0.50	0.25	0.5945	5	N 0.683675	3
(4) Artificial Lowering of Thorthomi Glacier Lake	0.26	1.00	0.75	1.00	0.7845	1	R 0.7845	2
(5) Flood Protection of Downstream Industrial and Agricultural Area	1.00	0.75	0.25	1.00	0.715	3	L 0.60775	5

Portfolio balance¹⁹

Wise investors avoid taking on too much risk. Rather than put all their eggs into one basket, they diversify and try to balance investments, for example, to pair projects that are high-gain, high-risk with ones that are low-gain but low-risk. Despite enticing opportunities for large profits or other rewards, few real estate developers, pharmaceutical companies, software developers, or others put all their resources into projects, markets, or products where outcomes are highly uncertain. They seek **portfolio balance**, i.e., to balance projects that are perceived as gambles with ones that are safe bets.

A way to display this balance is with a **bubble chart**. In Figure 19.5 each “bubble” represents a project; the x-axis represents the project reward or expected benefits; the y-axis, the likelihood of project success. The reward axis can be an interval scale (e.g. values for ECV, NPV, etc.) or ordinal scale (e.g. high, medium, low); similarly, the likelihood axis can be interval (0–100 probability) or ordinal (low, medium, high). The sizes of the bubbles represent the relative sizes of the projects based on, say, funding or resources.

Product-development organizations label the four quadrants in the chart according to the kinds of projects one finds—*pearls*, *oysters*, *bread and butter*, and *white elephants*.

Pearls are the projects that every company wants—high likelihood of success and high reward; but in reality, all companies are strapped with projects in the other quadrants as well. Oysters have lower likelihood of success because of technical or other risks but are worth pursuing because of the high potential reward. The aim is to find pearls in the oysters; most oysters do not contain pearls, but you don’t know that until you look.

Bread and butter projects are the most common: rewards are low to moderate, but the likelihood of success is high. Too many bread and butter projects, however, detract resources from the pearls and oysters and reduce future business opportunities.

White elephants are projects with low likelihood of success and low payoff. You must wonder why a company would retain any projects in this category. Fact is, having spent money and effort

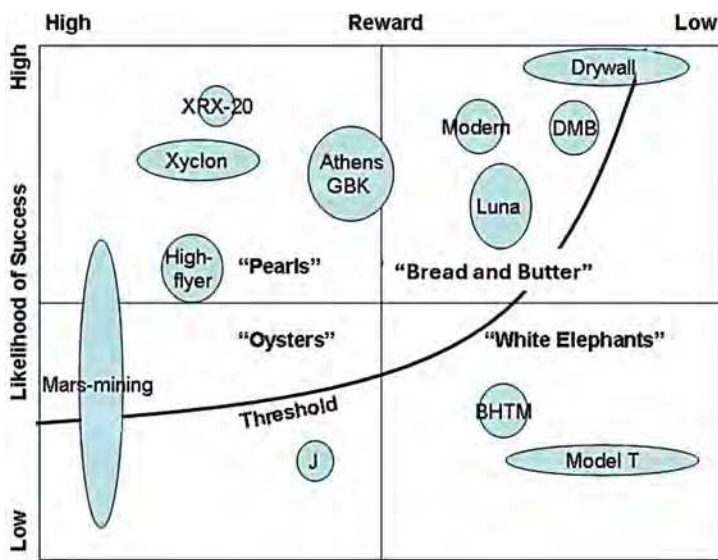


Figure 19.5

Bubble chart for likelihood of success and reward, showing threshold.

on a project, companies feel they *should* continue; they stay committed to the projects because of funds already spent.²⁰ A more prudent approach is to consider funds spent as “sunk costs” and irrelevant to deciding each project’s future. It sometimes takes courage to cull a white-elephant project from the portfolio, especially when it was the idea of an influential manager.

The bubble chart can be modified to reveal other kinds of information. In Figure 19.5, for example, the size and shape of each bubble reflects uncertainty about the likelihood of project success and the reward—the longer the bubble, the greater the uncertainty.²¹

Like other assessment methods, the drawback of bubble charts is heavy reliance on estimates or guesses of likelihoods, rewards, costs, etc. Also, they do not show the projects’ rank ordering or priority using criteria other than reward and success likelihood (e.g. which is better, “Highflyer,” “Xyclon” or “Mars-mining”?) or how projects *should* be distributed across the quadrants. Nonetheless, assuming the project selection team knows the balance it is seeking, such charts can be useful for deciding which projects to analyze more carefully and which to ignore. Conceptually, at least, every organization has a “threshold” line (Figure 19.5) above which projects are accepted, below which they are rejected.

Project categories

Another way to select projects is according to how well they fit organizational goals and strategies. Starting with the organization’s mission, strategic initiatives, and objectives, top management decides on the categories (themes) of projects that best align with them.

Projects are typically categorized according to one of the following:²²

- Strategic goal (e.g. defending the product base, growing the base, diversifying products, etc.)
- Product line (product A, B, C, etc.)
- Project type (R&D, capital improvement, process improvement, etc.)
- Geography (Toronto, California, Panama, Central America, etc.)
- Business unit (marketing, manufacturing, product development, etc.).

Further examples are the five headings in Table 19.4. Associated with each category is an allocated funding amount (for example \$12.5M, \$8.5M, \$10M, etc.), which is the total budget available to all projects in a category. In a small company the categories might be consolidated into one portfolio and managed as a single portfolio group, or into a program and overseen by a program manager. In a large company, *each category* would be a *separate project portfolio* overseen by its own portfolio manager and PRB.

Companies routinely undertake more projects than they can handle. For example, in Table 19.4 the totals at the bottom of the columns indicate that projects in all but the second category require funding in excess of the allocated funding. To decide which projects to include in the portfolio, the PRB rank orders the list of projects (using methods described earlier), and, starting at the top, selects projects until funds run out. Supposing the projects in Table 19.4 are rank ordered, the underlined projects represent the cutoff. In the last category, for instance, Project S is the cutoff; and Projects AI and EI will not be funded.

A project that has been approved is admitted to the portfolio, but its ultimate execution depends on the availability of key, limited resources. Someone, somewhere (perhaps the PMO) keeps track of the allocation of key limited resources, and only when resources become available can a project be scheduled to begin. A systematic procedure for allocating resources to multiple, approved projects based on the Theory of Constraints is described in Chapter 8.



Table 19.4 Projects rank ordered within categories.

Asian Operations \$12.5M		European Operations \$8.5M		OEM Product Line Development \$10M		Domestic Product Line Development \$8M		Process Improvement \$7.2M	
Project E	3.2	Project B	0.2	Project A	3.4	Project D	2.2	Project C	2.2
Project G	1.4	Project F	2.2	Project H	0.8	Project J	1.2	Project I	0.8
Project O	0.6	Project N	0.4	Project L	1.7	Project M	0.1	Project K	1.2
Project Q	3.7	Project P	1.5	<u>Project R</u>	3.1*	Project T	1.3	<u>Project S</u>	2.7*
<u>Project W</u>	2.3*	Project U	1.3	Project CI	1.6	Project V	0.2	Project AI	0.7
Project BI	1.8	Project X	0.6	Project GI	1.1	Project Y	0.8	Project EI	1.2
Total	13.0**	<u>Project FI</u>	<u>1.9*</u>	Total	1.1 y**	<u>Project Z</u>	1.2*	Total	8.8**
		Total	8.1			Project DI	2.2		
						Project HI	0.2		
						Total	9.4**		

* Cutoff project.

** Required funding for all projects exceeds allocation.

Deciding on the categories and the funding for each is top management's responsibility. Such decisions presumably are based upon consideration of the organization's mission, strategies, and objectives, although sometimes the allocation is debatable. The mission of NASA, for instance, is to support research and development in aeronautics, manned spaceflight, and unmanned space exploration, although at times the overwhelming share of NASA funding goes to manned spaceflight programs, which leaves little remaining for unmanned space exploration and even less for aeronautics research. This has led critics to charge that NASA's skewed funding allocation does not support the agency's full range of purported objectives.

Cost–benefit grid

A method well suited for prioritizing and selecting projects according to several criteria is Buss's **cost–benefit grid**.²³ Suppose two important criteria are financial benefits and project cost. The PRB reviews each project's proposal and rates it (high, medium, or low) according to financial benefits and cost. The outcome is displayed on a three-by-three grid. When several projects are rated with these criteria, the result looks like Grid A in Figure 19.6, which shows the ratings for 12 projects.

After reviewing the grid and reaching agreement on the relative positioning of the displayed projects, the team repeats the procedure for additional criteria such as technical benefits,

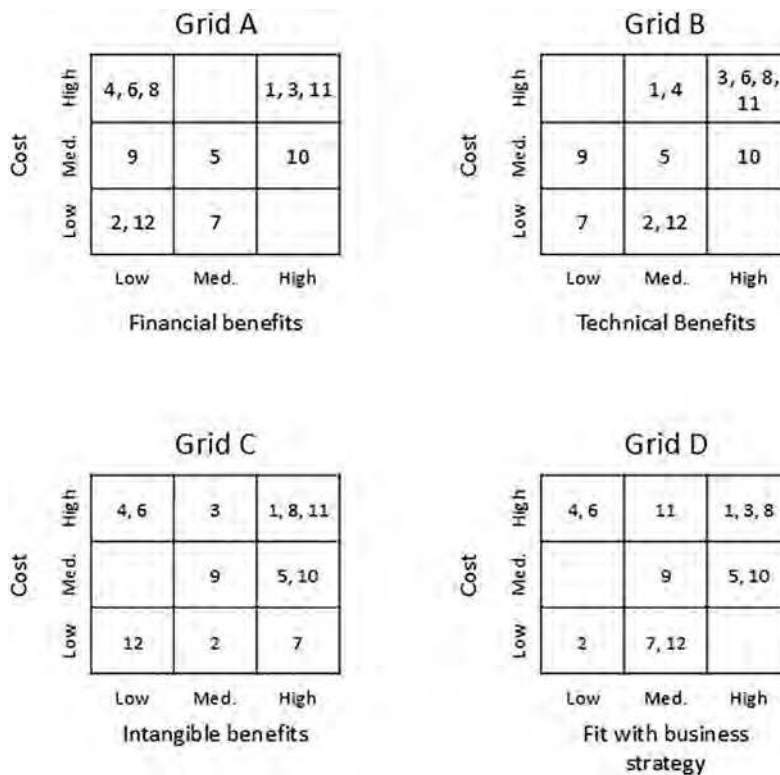


Figure 19.6
Buss's cost–benefit grids; ratings for twelve projects.

intangible benefits, fit with company business strategy, and so on, and plots the results on other grids (Figure 19.6).

How are intangible benefits assessed? First, the team agrees on the intangible benefits it wants to consider, such as company image, customer satisfaction, or strategic fit. Teams having members with different perspectives—i.e., some who see projects in terms of financial return, others who see them in terms of technical capabilities or strategic benefits—are usually better at identifying intangibles than teams where everyone thinks alike. Given the list of intangibles, the team chooses a scoring method. If, for example, there are six intangible benefits and each is scored 1–5, then a project’s maximum possible score for intangibles will be 30. To locate a project in the grid, scores are converted into simple categories, e.g., ≥ 20 is High, ≤ 10 is Low, in between is Medium.²⁴

A rank-ordered list is created from the completed grids. Projects in the lower right cells would be placed at the top of the list; those in the upper left, at the bottom. But besides location in the grids, the rank order also depends on organizational priorities. In Figure 19.6 projects 1, 3, 8, and 11 are each rated High-High in three of the grids, yet if the organization’s top priority is financial benefit, then project 8 would be ranked lower and might even be rejected. Final selection will also depend on each project’s size and available funding and resources, as described earlier.

The main advantage of the grid method is clear exposition of the comparative benefits of projects as determined by the collective judgment of the team. For this and all team assessment and selection methods, ideally team members represent a broad range of perspectives (technical, product/market, financial, environmental, etc.).²⁵

Although the grid method might seem to rely too much on subjective judgment and too little on formal analysis, the team might in fact use formal analysis methods and quantitative models to arrive at their ratings. (As mentioned, however, quantitative methods often rely on estimates that are little more than guesses, making them no more accurate than subjective methods—despite creating false perceptions to the contrary.)

Cost-effectiveness analysis²⁶

Cost-effectiveness analysis is similar to the cost-benefit grid method but uses numerical values for costs and benefits. The term “effectiveness” refers to the degree to which a project is expected to fulfill project requirements; it is interchangeable with terms such as *benefit*, *value*, *utility*, and *performance*. As with those terms, assessing effectiveness involves consideration of multiple factors. In assessing commercial aircraft projects, for instance, effectiveness would account for some combination of aircraft passenger capacity, weight, range, speed, fuel efficiency, and maintainability, which are interrelated in complex ways. One method for deriving a single measure incorporating multiple factors is to rate the factors subjectively (but using results from quantitative analysis and advice from technical experts), weigh the ratings, and add them up—similar to the weighted scoring model illustrated in Table 19.1. The factors chosen for the analysis represent significant ways to distinguish between projects, and the projects are assumed identical in all other important respects.

The example in Table 19.5 shows three proposed projects and seven factors.²⁷ Each factor for each project is scored 0–100 for effectiveness (E) and for weighted effectiveness (WE = weight $\times$ E). Total WE, the sum of the weighted effectiveness across all seven factors, is the project effectiveness.

The method does not rank the projects but suggests which ones should be dropped from consideration and allows tradeoff analysis of the remaining projects. For example, Figure 19.7 shows the three projects from Table 19.5 and five other projects. Projects j, h, and m (in the

Table 19.5 Cost-effectiveness data analysis.

Factors	W(weight %)	Project A		Project B		Project C	
		E	WE	E	WE	E	WE
Speed	10	95	9.5	80	8	85	8.5
Range	15	70	10.5	80	12	75	11.25
Efficiency	20	75	15	75	15	85	17
Comfort	15	70	10.5	85	12.75	85	12.75
Capacity	20	70	14	90	18	95	19
Loaded mass	15	90	13.5	60	9	70	10.5
Maintainability	5	75	3.75	85	4.25	80	4
Total WE			76.75		79		83
Cost			\$1.8B		\$2.0B		\$3.0B

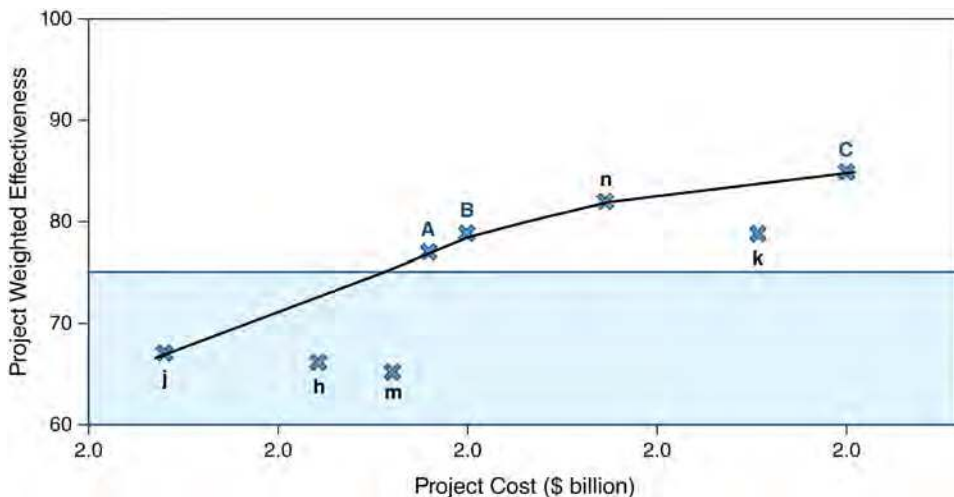


Figure 19.7
Effectiveness versus development cost for eight projects.

shaded area) fall below the minimal effectiveness threshold of 75 and would be dropped from consideration. The line connecting the uppermost points (j, A, B, n, C)—called the “efficient frontier”—represents the *maximum effectiveness level* attainable for a given cost (or minimum cost for a given effectiveness level). Project k is below this line, which means it is inferior to at least one other project in terms of both cost and effectiveness and should also be dropped. Only projects A, B, n, and C are worthy of further consideration.

A maximum effectiveness line with a positive slope indicates that increasing project cost is justified by increasing effectiveness. But the *degree* of slope matters too: Project C is only marginally more effective than Project n but costs a lot more, suggesting that it is probably not worth pursuing.

As mentioned, project selection relies on imperfect information about costs and benefits. While the models in this chapter provide “objective” ways to sort through the maze of facts, figures, and issues associated with project selection, rarely are final decisions based solely

upon them; fact is, human instincts, emotions, and ulterior motives also play a role in project selection.

19.5 Integrating the Gating Process and Portfolio Management²⁸

Portfolio management includes selection of new projects and periodic review of current projects. The portfolio manager and PRB must decide when to start each newly approved project—immediately or later—and whether each current project should be sustained, changed, or terminated. Projects exceeding deadlines or expected costs, not meeting requirements, or no longer suited to changing company objectives or the environment are reconsidered. In order to reallocate funds and other resources appropriately, underperforming or no-longer necessary projects are cancelled to make way for essential or more-promising projects. In many companies, periodic project review happens through a **gating** or **stage-gate process**.

As mentioned, money and effort already spent on a project should be considered as a **sunk cost** and not considered when deciding how to proceed. It is, however, often difficult to cull a white elephant—often for fear of being criticized for having wasted the money on the project. A sign of an effective portfolio management process is that periodically projects *do* get cancelled.

Some projects, however, e.g. in construction, reach a point of no return where the cancellation costs are too high. This emphasizes that projects should not be allowed to proceed to execution without a compelling case, based on reliable estimates.

Gating and portfolio management augment each other, but the two processes are quite different. In gating, at the end of each project phase, sub-phase, or milestone, the project is assessed based on its performance, estimated business impact as of that stage, and external factors. The assessment does not consider the project's impact on organizational resources or objectives, or other projects.

In contrast, portfolio management looks at all projects in the portfolio and compares them in terms of criteria such as benefits, costs, and resources. This involves considerable effort and, consequently, might happen only three or four times a year, or less. Since companies are usually involved in many projects and proposals at any given time, each arriving at decision gates at various times, it is not feasible to compare all projects in a portfolio every time one of them reaches a gate.

Also, the two processes tend to use different decision criteria: whereas gating typically permits a project to continue as long as it conforms to plans, expectations, and the business environment, portfolio management allows it to continue only if it compares favorably to other projects. In addition, the two processes usually involve different teams: decisions in the gating process are made by middle-level managers and customers; in portfolio management they are made by the portfolio manager and PRB.

Nonetheless, ideally the two processes and teams assist each other: gating weeds out marginal projects so the portfolio has none underperforming, and portfolio management weeds out projects that do not contribute to company objectives. Further, the PRB assists managers in the gating process by sharing its rank-ordered listings and noting any changes in company strategy and objectives. Gating managers consider this information and sometimes kill projects that ultimately would have been killed by the PRB anyway.

The gating process as described essentially halts the project at each stage for review and authorization. It is an example of a large-batch, long-queue process as described in Chapter 14. Thus, despite its purported advantages, the gating process *holds up* work and information flow at each stage and does not release it until everything is approved. If speed-to-completion is a goal



See Chapter 14

of the project, the gating process is not the way to achieve it. Also, by its very nature, the gating process does not allow for experimentation and tends to suppress or kill innovation. Thus, you would not find the gating process in the APM projects described in Chapter 14.

19.6 Summary and Discussion

Portfolio management is the process of choosing and managing those projects that best achieve organizational objectives subject to human and financial resource constraints. Portfolio management in combination with strategic management, the gating process, and project management help ensure that the organization does the right projects and does those projects right.

Project selection and portfolio management happen through a multi-stage process of prescreening, analysis, and screening of new-project proposals, and then ranking, selecting, and ongoing reviews of approved new and existing projects. Top management establishes high-level criteria for project selection decisions, but actual project selection and portfolio management rests with the portfolio manager and PRB.

This chapter reviewed a variety of methods for rating, screening, and comparing projects in terms of benefits, costs, risk, resource requirements, and strategic objectives. Yet the methods covered do not account for everything. Project dependency is an example: when Project B depends upon Project A, then Project A's approval might depend on the importance of Project B, and, of course, Project B's approval will depend on whether Project A has been approved.²⁹ A separate but related matter is selection of parallel projects—such as in new-technology development—so as to increase the likelihood that at least one will achieve a breakthrough.

Given the variety of methods to analyze, rate, and select projects, the question is: which is best? In practice, no one method stands far above the rest, but it is not necessary to choose just one method. In fact, the methods described should be used in combination. For example, projects first divided according to strategic categories can then be judged by the cost-benefit grid method; the best of these can then be ranked and selected subject to available resources. Or projects prioritized using financial, scoring, or cost-effectiveness approaches can be checked for portfolio balance on bubble charts, and then judged with the grid method. Using multiple selection methods helps ensure that the projects selected are the “right” ones.



Review Questions and Problems

1. What are four or five main features of project portfolio management?
2. What is a project portfolio? How do project portfolios differ from programs?
3. Is it poor practice to do the easiest projects first (“the low-hanging fruit”)?
4. Compare the following, and for each state the focus and how it relates to projects:
 - a. Strategic management
 - b. Portfolio management
 - c. Gating methodology.
5. What are the responsibilities of the project portfolio manager?
6. What are the responsibilities of the project review board? Who are the members of the PRB?
7. “Some projects you simply have to do. You have no choice.” Give examples of projects where you have no choice.
8. What is the purpose of *prescreening* in the project selection process? How does it differ from *project screening*?

9. Explain portfolio selection. What kinds of projects does it consider—current, proposed, or both?
10. How would spare capacity influence project selection decisions? What should you do with spare capacity?
11. Projects W, X, Y, and Z are each being screened according to four criteria: Potential Return on Investment, Lack of Technological Risk, Environmental “Friendliness,” and Service to Community:
 - a. Project W: Return, high; Risk, medium; Environment, medium; Service, low.
 - b. Project X: Return, medium; Risk, high; Environment, medium; Service, low.
 - c. Project Y: Return, medium; Risk, medium; Environment, high; Service, high.
 - d. Project Z: Return, medium; Risk, medium; Environment, high; Service, low.
 Create a scheme (or multiple schemes) for screening the projects, assuming equal weight for all criteria. Which project comes out best; which is worst?
12. For the above four projects, assign scores of high = 3, medium = 2, and low = 1. Assume the criteria are weighted: Potential Return on Investment = 0.3, Lack of Technological Risk = 0.3, Environmental “Friendliness” = 0.3, and Service to Community = 0.1. Now which projects come out best, and worst?
13. Compare the ECV and B/C methods for evaluating projects.
14. What is the expected commercial value (ECV) of a project involving the launch of a new product with an estimated development cost of \$15M, launch cost of \$0.8M and NPV for the future stream of earnings of \$45M if the probabilities for success are 70 percent in development and 50 percent in the market?
15. A project has three phases—concept, development, and launch—that are expected to cost \$5M, \$15M, and \$4M, respectively. The likelihoods of success for the three phases are 0.5, 0.8, and 0.7, respectively. If the estimated NPV of future earnings is \$90M, what is the ECV for the project? (Answer: \$11.1M.)
16. In the previous example, what else must be considered if the stream of earnings were in euros instead of dollars?
17. In problem 15, suppose the likelihood of project success is $0.5 \times 0.8 \times 0.7 = 0.28$.
 - a. What is the B/C ratio for the project?
 - b. Which measure makes the project look more attractive, ECV or B/C? In your opinion, does the project merit approval?
18. Project A and Project B have the same overall cost of \$4M. Project A’s likelihood of success is 95 percent; Project B’s chances are 50 percent. Project A is expected to generate \$11M revenue but will incur \$5M maintenance costs. Project B is expected to generate \$8M revenue and efficiencies that would save \$5M in expenses. Applying the B/C ratio, which project would you recommend?
19. What advantage do scoring models have over financial models in terms of assessing the value or utility of projects?
20. What are the drawbacks of financial models? Of scoring models?
21. What are the three main approaches to comparing and selecting projects?
22. What is the drawback of ranking projects using single-criterion methods?
23. Top management has decided to reallocate funds among the five categories of projects listed in Table 19.4 as follows: \$13M, \$8M, \$7.5M, \$10M, and \$7.8M, respectively. What are the cutoff projects in each of the five categories?
24. Explain how cost–benefit grids can be used to rank order projects.
25. Discuss similarities and differences between bubble charts and cost–benefit grids.
26. Suppose Project D is added to the projects in Table 19.5 and has been rated for effectiveness as shown in the table below:

Project D	
Factors	E
Speed	80
Range	90
Efficiency	95
Comfort	85
Capacity	95
Loaded mass	90
Maintainability	80
Cost	\$2.5B

Compute the total weighted effectiveness using the weights in Table 19.5. How does Project D compare to the others in Figure 19.7?

27. Once a project has been approved and admitted to the project portfolio, how is it monitored thereafter? Under what circumstances might the project be cancelled?
28. Describe the differences between the gating process and portfolio management. What are the difficulties in integrating the two processes? How might the difficulties be overcome so that portfolio projects can also be gated projects?
29. Draw a bubble chart for Risk versus ROI percent (similar in appearance to Figure 19.5) for the four projects in Table 19.2. Divide the chart into four quadrants: ROI < 20% vs. ROI > 20%, and High Risk vs. Low Risk. Use the following to “size” the project bubbles: projects D and C are the largest and the same size, A is half that size, and B is half the size of A. Based on the bubble chart, which projects are pearls? Which are white elephants? Which are bread and butter?



Questions About the Study Project

1. Does the organization have a portfolio management process? If so, describe the key steps in the process and the managers and others who participate in the process. In your opinion, is the process effective? What are its strengths and weaknesses?
2. Does the organization have portfolio managers or PRBs (governance boards, project steering committees, etc.)? Describe their roles and modus operandi.
3. Describe the organization’s project analysis and selection process. What kind of analysis and selection models and methods are used?
4. How are projects compared and rank ordered? Who makes approval and funding decisions?
5. Does the organization have a gating process? Describe the gates, assessment criteria, and list who participates at each gate. In your opinion, is the process effective?
6. If the organization has both portfolio management and gating processes, discuss the relationship between the two and the manner in which they are integrated.
7. If the organization has a PMO, discuss the PMO’s role in portfolio management and the gating process.

CASE 19.1 PROPOSED CEMENT FACTORY FOR PCS COMPANY

Note, for this case you might review the front-end loading (FEL) process in Chapter 4.

The first sentence in the executive summary of the draft business case reads, “The proposed cement factory would be the perfect answer to top management’s goal to grow PCS’s product base while remaining a low-cost producer of building materials.” Geological work performed as part of the feasibility study had pointed to a site containing high-quality limestone—a key resource in the production of cement—that could be exploited very economically. The site, already owned by PCS, was under-utilized and could suitably house a new plant located right next to the proposed limestone pit. It afforded ample and cost-effective access to all essential logistics and resources, including other materials needed for the manufacture of high-quality cement.

The business case and feasibility study also indicated a high return on investment for the project—well above the hurdle rate set by the company for a new project to be considered for inclusion in its project portfolio. Given the company’s competency in cement production, the technical and production risks were assessed as low.

Everything seemed in order, and the team proposing the project was optimistic that approval of the next phase of the proposed project (FEL-2) would be a mere formality. They presented the draft business case to the PMO for finalization, and to top management for authorization of FEL-2. To their surprise, the PMO director insisted that the team must make all assumptions explicit and do a sensitivity analysis before he would even consider carefully reading the business case. He pointed out that the current high cement price as assumed in the feasibility study was the result of a booming Chinese construction industry that could decline significantly in the foreseeable future. Another member of the PMO added: “And what if assumed costs rise and we can’t stick to the proposed budget, as was the case with several of our recent projects?”

QUESTIONS

1. Explain why a business case should take into account alternative scenarios for the important variables.
2. List topics or issues that are not mentioned in the case but should be considered before the project gets the go-ahead.

CASE 19.2 SELECTING PROJECTS AT GALACTIC MINING, A

Jim LaPlas, the CEO of Galactic Mining, compiles a list of 7 projects to investigate the feasibility of mining seven of Jupiter’s known 79 moons, and all the projects pass the initial screening assessment. He gives the list to Judy Patel, manager of the project portfolio office, and requests her to rank the projects according to priority. As Judy leaves Jim’s office, he specifically comments on the importance of profitability. The following information is available for the seven projects:

Project Europa

Although low risk, this project provides an excellent ECV (expected commercial value, which takes into account net present value, development cost, and probability of technical and commercial success). Europa, however, does not fit the current strategy well.

Project Adrastea

Both ECV and risk are moderate, but compared to the other six projects it fits poorly with the current strategy.

Project Himalia

This project fits the current strategy well, is not risky, but does not provide an exciting ECV.

Project Thebe

With a good strategic fit and moderate risk, the ECV of Thebe is better than that of Adrastea and Project Himalia.

Project Metis

With an excellent strategic fit and relatively low risk, Metis unfortunately promises a poor ECV.

Project Ganymede

Ganymede offers a reasonable ECV with excellent strategic fit and low risk.

Project Callisto

Callisto's ECV is slightly better than Ganymede but worse than Europa; it has very low risk and excellent strategic fit.

ASSIGNMENT AND QUESTIONS

1. Compile a list, ranking projects in priority order, from No. 1 to No. 7. If you do this with a group of people, first do the ranking individually, then discuss people's ranking with the group.
2. Discuss the possibility of prioritizing the projects solely on financial criteria.
3. What role do you think bias and internal politics would play in selecting projects?
4. Discuss the role that resource availability should play in deciding the sequence in which the projects should be commenced. (Consider money, key personnel, and other important resources.)
5. Suggest a way to proceed to reach consensus on the priorities. Indicate what stakeholders should be involved.

CASE 19.3 SELECTING PROJECTS AT GALACTIC MINING, B

Judy sees this assignment (Case 19.2) as an opportunity to test a scoring technique that she recently proposed. The technique requires that for each project a score be assigned for the Strategic Fit and the Risk, and that the specific ECV be calculated.

The following score values are to be used:

- For Risk: Moderate risk = 2; low risk = 3; very low risk = 4
- For Strategic Fit: Excellent = 4; Fits well = 3; Good fit = 2; Does not fit well = 1; Bad fit = 0.

The calculated ECVs for the projects are as follows (the *higher* the ECV, the more desirable the project):

Project	ECV
Metis	2.3
Adrastea	3.5
Thebe	3.1
Himalia	2.6
Europa	6.4
Ganymede	4.6
Callisto	5.3

The scoring technique first ranks the project according to each criterion and then calculates an aggregate ranking.

ASSIGNMENT AND QUESTIONS

1. Use Judy's technique with information in Case 19.2 to numerically score each of the projects according to Strategic Fit, ECV, and Risk. Then numerically rank the projects based on the scores, assigning projects with the same scores the same rank. (For example, on ECV, Europa is ranked 1, Callisto 2, and so on; on Strategic Fit, Metis, Ganymede, and Callisto are all ranked 1, Himalia is ranked 2, etc.) Then, for each project, compute the average weighted rank based on the ranks for the three criteria. Finally, based on the average ranking scores, rank order the projects.
2. Discuss the relative results.
3. Discuss the role the scoring technique could play regarding biases and internal politics.
4. How should preferences and biases of executives be handled?
5. How should resource availability be considered?
6. A turbulent business or technical environment can lead to a change in strategy. How should such a change be handled?
7. Discuss any possibilities you see of further refinement or optimization of the ranking list (consider quantitative techniques, etc.).
8. Repeat the ranking, but besides Strategic Fit, Risk, and ECV include a fourth criterion: Public Image of Project (4 = high, 0 = none). Assume the projects are rated as follows on Public Image (numbers in parentheses are rankings on list:

Project	Public image
Project Metis	0 (5)
Project Adrastea	1 (4)
Project Thebe	4 (1)
Project Himalia	3 (2)
Project Europa	4 (1)
Project Ganymede	2 (3)
Project Callisto	3 (2)

Include this ranking with the rankings for Strategic Fit, Risk, and ECV, and recompute the final rankings for the projects based on all four criteria.

Notes

1. Cooper R., Edgett S., and Kleinschmidt E. Portfolio management in new product development: lessons from leaders, phase II. *Research Technology Management* Nov–Dec; 1997: 43–52, reprinted in Dye L. and Pennypacker J. (eds.). *Project Portfolio Management*. West Chester, PA: Center for Business Practices; 1999, pp. 27–33.
2. Sommer R. Portfolio management for projects: a new paradigm. 1998. *Proceedings of the Annual Project Management Institute Seminars & Symposiums*. Newtown Square, PA: Project Management Institute; reprinted in Dye and Pennypacker, *Project Portfolio Management*, pp. 55–59.
3. Project Management Institute. *The Standard for Portfolio Management*, 4th edn. Newton Square, Pennsylvania: PMI; 2017 and Office of Government Commerce. *Management of Portfolios*. Norwich, UK: The Stationary Office; 2011.
4. This section adopted from Cooper et al. Portfolio management in new product development: lessons from leaders, phase II, pp. 34–35; and Bridges D. Project portfolio management: ideas and practices. In Dye and Pennypacker, *Project Portfolio Management*, pp. 97–116.
5. Levine H. Project portfolio management: a song without words? *PM Network* July; 1999.
6. Davidson Frame J. *The New Project Management*. San Francisco, CA: Jossey-Bass Publishers; 1994, p. 190.
7. Archer N. and Ghasemzadeh F. An integrated framework for project portfolio selection. *International Journal of Project Management* 17(4); 1999: 207–216.
8. Sommer, Portfolio management for projects: a new paradigm; and Bridges, Project portfolio management: ideas and practices.
9. Bridges, Project portfolio management: ideas and practices.
10. The concept of project selection as a funnel is analogous to *product* selection as described in Wheelwright S. and Clark K. *Revolutionizing Product Development*. New York, NY: Free Press; 1992.
11. Method described in Cooper et al. Portfolio management in new product development: lessons from leaders, phase I. *Research Technology Management* Sept–Oct; 1997: 16–19, reprinted in Dye and Pennypacker, *Project Portfolio Management*, pp. 97–116.
12. Shtub A., Bard J.F., and Globerson, S. *Project Management: Processes, Methodologies, and Economics*, 2nd edn. Englewood Cliffs, NJ: Prentice Hall; 2005, pp. 182–183.
13. Shtub et al., *Project Management*, 186–187
14. See for example Voss, M. and Kock, A. Impact of relationship value on project portfolio success: investigating the moderating effects of portfolio characteristics and external turbulence. *International Journal of Project Management* 31; 2013: 847–861; also, Killen C., Hunt R. and Kleinschmidt E. Project portfolio management for product innovation. *International Journal of Quality and Reliability Management* 25(1); 2008; 24–38.
15. Cooper et al., Portfolio management in . . . : lessons from leaders, phase I.
16. Mathematical programming approaches for project portfolio optimization are described in the literature in operations research. For example see Dickinson M., Thornton A., and Graves S. Technology portfolio management: optimizing interdependent projects over multiple time periods. *IEEE Transactions of Engineering Management* 48(4); 2001, 518–527.
17. Method described in Cooper et al., Portfolio management in . . . : lessons from leaders, phase I.
18. Source: National Environment Commission, Royal Government of Bhutan. 2006. Bhutan National Adaptation Programme of Action. <http://unfccc.int/resource/docs/napa/btn01.pdf>. Accessed June 17, 2024.
19. Cooper et al., Portfolio management in . . . : lessons from leaders, phase I. Bubble charts are easily created with commercial software [Google the term “bubble chart” for examples of methods and products].
20. Meyer W.G. The effect of optimism bias on the decision to terminate failing projects. *Project Management Journal* 45(4) Aug./Sept; 2014: 7–20.
21. Cooper et al., Portfolio management in . . . : lessons from leaders, phase I.
22. Method described in Cooper et al., Portfolio management in new product development: lessons from leaders, phase II.
23. Frame, *The New Project Management*, pp. 181–185; Buss M. How to rank computer projects. *Harvard Business Review* Jan–Feb; 1983.
24. Buss, How to rank computer projects.

25. Frame, *The New Project Management*, p. 185.
26. Overview of method as presented in Shtub et al., *Project Management: Engineering, Technology, and Implementation*, pp. 127–130.
27. This example, like others in this chapter, is a much-simplified illustration. Assessing options for development of large-scale systems such as aircraft involves engineering studies to assess alternative configurations and design details, plus economic analysis of development, procurement, and operating costs, and projections of unit sales. See, e.g., Jenkinson L., Simpkin P., and Rhodes D. *Civil Aircraft Design*. London: Arnold; 1999.
28. This section is adapted from Cooper et al., Portfolio management in new product development: lessons from leaders, phase II.
29. On selection of dependent projects, see Dickinson et al., Technology portfolio management: optimizing interdependent projects over multiple time periods.

Chapter 20

Managing International Projects and Megaprojects

This chapter covers two topics of special interest: international projects and megaprojects. Although they are independent topics, they are treated together here because some of the world's biggest projects are also international and they involve stakeholders and venues around world. This connection will become clear after you have read this chapter.

20.1 International Projects

An **international** or **global** project is one where project participants and stakeholders are cross-national and cross-cultural, and project work happens in different countries. An example is the Japan Aerospace Exploration Agency's (JAXA) Martian Moons Exploration program, which will send a probe to the Martian moon Phobos and include a small, autonomous rover called IDEFIX. The mission aims to collect about ten grams of Phobos' material and send it to Earth. Most of the Martian probe and the rocket to send it to Phobos are being built by JAXA. The IDEFIX rover is jointly built by the German Aerospace Center (DLR) and the French space agency Centre National d'Etudes Spatiales (CNES). During the mission, the European Space Agency (ESA) will supply deep-space communication equipment, NASA will supply instrumentation and the collection device, and both will provide ground station support.

International projects have become ubiquitous as companies establish divisions, seek customers, and outsource work to suppliers and contractors around the world. Thanks largely to lower costs and increased capacity of global air and sea transportation, enhanced communication fueled by the Internet, and emerging business and technological capabilities in nations like China, India, and Brazil, companies seek out and execute projects everywhere.

While these projects offer enticing benefits and opportunities, simply being "international" makes them more vulnerable to risk. Regardless of the scope or end-item, a project that is "global," "international," or "overseas" automatically inherits more issues and greater risk than one that is not. And no matter what issues and problems the manager of a domestic, one-country project faces, the manager of an international project automatically faces an "extra layer" of issues that touches leadership, interpersonal relations, stakeholders, procurement, communication, work definition, estimating, risk management, and work tracking and

Table 20.1 Unknowns in an international project.

-
1. Local institutions and culture
 - a) Language (explicit)
 - b) Norms, social customs, attitudes, traditions (tacit)
 - c) Laws, rules, rights, sanctions (explicit)
 2. Local stakeholders—laborers, managers, consultants, suppliers (tacit)
 - a) Skill, experience, motivation
 - b) Reputation, honesty, integrity
 - c) Who knows who; who has knowledge, resources, and connections
 3. Local natural environment (explicit)
 - a) Site environment—soil, ground slope, vegetation
 - b) Regional environment—climate—weather, geography, seismic activity
 4. Local technology (explicit)
 - a) Infrastructure—roads, buildings, communication
 - b) Available tools and systems—GPS, equipment, hardware, software, materials
-

Note: “Local” refers to people and factors situated at the site or region of the project, or that become activated in the local context, including international NGOs, associations, and other organizations that play a role in “promulgating environmental, technological, job-related, and legal” regulations to the local level.

control. Language, local customs, transportation, and infrastructure—all of little or no concern in a home-country project—become potential showstoppers in an international project.

20.2 Problems Managing International Projects¹

Table 20.1 lists aspects of an international project that make planning and execution more difficult; some are “explicit”—somewhat easy to pinpoint and account for in project plans and estimates, others are “tacit”—more difficult to isolate and address. In general, the more “unknowns” the host country and its people present to the project manager and team, the harder it will be for them to plan and execute the project. Hereon, “host” refers to the place where the project is executed, “home” to the native country of the contractor, developer, or project manager.

20.3 Culture and Local Customs and Institutions

Stakeholders in international projects encompass diverse cultures and languages that influence communication, attitudes, behavior, work practices, decision patterns, and, ultimately, project performance. Additionally, their behavior is guided or restricted by regional or national laws, regulations, and rules.

Culture. Culture refers to the set of values, beliefs, behaviors, and attitudes that members of a group, organization, country, or region share. Among ways to measure culture is a study by Hofstede of IBM employees in over 50 countries. The study identified five dimensions of culture.²

- **Power distribution (PD).** The extent to which less powerful members of a culture accept or expect that power should be *unequally* distributed, versus that power should be

equally distributed. People from high PD countries tend to hold superiors or leaders in high regard and not question their directives or actions.

- **Individualism (IND).** The extent to which members of a culture believe they are expected to look out for themselves versus believing they are part of and looked after by the group to which they are loyal.
- **Achievement orientation (ACH).** The extent to which roles are distributed along gender lines: “masculine” implies an assertive, tough, and achievement orientation; “feminine” implies a more relationship, helping, or quality-of-life orientation. People from high ACH countries care more about earnings and “success”; those from lower ACH countries care more about sharing, cooperating, and caring for others.
- **Uncertainty avoidance (UNA).** The extent to which members feel uncomfortable with uncertain or ambiguous situations and need to take steps to impose order and structure, versus accepting uncertainty or ambiguity, “going with the flow.” People from low-UNA countries are more comfortable with ambiguity; they feel less uncomfortable in the absence of detailed plans and formalized team roles than people in high-UNA countries.
- **Long-term orientation (LTO).** The extent to which members look to long-term benefits and deferred results or gratification, versus seeking immediate or short-term results and gratification.

Hofstede’s research showed considerable differences in these dimensions by country: Table 20.2 gives some results. Larger values imply the tendency to accept unequal power distribution, be individualistic, seek achievement, be more masculine and more comfortable with uncertainty, and long-term oriented; smaller values imply the opposite.

The results on Table 20.2 suggest potential challenges when people from different regions and countries must work together. For example, according to the table, a project team with members located in UK, US, and China might expect significant differences in terms of PD, IND, UNA, and LTO. And while such challenges might be significant even among workers from developed nations, they are exacerbated when the workforce combines members from developed countries and developing countries (aka emerging economies). One way to handle such differences is to air them, and that might happen during a team-building session as described in section 20.9 and Chapter 17. One danger with findings like this is the temptation to generalize even though people are unique and do not necessarily fit the average.³



See Chapter 17

Table 20.2 Hofstede’s cultural dimensions: representative results.

Country	PD	IND	ACH	UNA	LTO
Brazil	69	38	49	76	65
China	80	20	66	30	118
Denmark	18	74	16	23	*
India	77	48	56	40	61
Israel	13	54	47	81	*
Japan	54	46	95	92	80
Poland	68	60	64	93	32
Russia	93	39	36	95	*
Spain	57	51	42	86	*
United Kingdom	35	89	66	35	25
United States	40	91	62	46	29

* No data

Of note is that professional or personal interests sometimes matter more to people than their national culture. This says, for example, that an Indian software engineer might feel more in common with an American software engineer than with her average fellow Indians.⁴ A project manager might take advantage of that by developing “communities of practice” to overcome cultural differences.

Language. In projects where stakeholders speak different languages, conversations and project documents like scope, requirements, budgets, and contracts must be translated. The challenge is to ensure that every translation reflects the content and intention of the original message.

Even projects wherein ostensibly everyone uses the same language face difficulties. For example, the same English words when used in America, the UK, South Africa, Australia, and India often have different meanings; add in slang, vernacular, idiomatic terms, and poor diction, and the result is the message gets “lost in translation.” For example, “tell the English to walk on the pavement and they will walk on what Americans call the sidewalk; tell Americans to walk on the pavement, they will walk down the middle of what the English call the tarmac.”⁵ US managers sometimes say it is more difficult to communicate with the British than with the French.

The best practice in international communication is to *always* use the simplest, most concise wording and phrasing. Before sending important messages and documents, ask several people to interpret them. Napoleon did something like this: before issuing military orders, he would ask a corporal to read them, reasoning that if someone of low rank could understand them, then so would his officers.⁶

The manager of an overseas project should learn enough of the local language to at least conduct simple daily transactions. This facilitates communication but also shows respect for people of the host country, who always appreciate foreigners’ attempts at their language.

Managers sometimes create a **glossary** of project terms, which can be extensive and even include pictures. For the project to develop the Anglo-French Concorde supersonic airplane, a special English–French project dictionary was created.

Formality. Whereas business associates in North America tend to address one another—subordinates, superiors, even senior managers—by first name, in most places in the world some variant of sir, mister, or madam would be used. Such formality extends to the way they introduce themselves, communicate ideas, make commitments, and give and receive business cards. The workplace code of behavior may discourage kidding around and other forms of informality. Formality pertains to documents too: while in-country proposals and contracts are commonly shared verbally or via fax or email, doing so internationally is problematic since it poses questions about the country whose contract law and court of law applies.

In some countries, practices of little import elsewhere are raised to a high art. In Japan, the exchange of business cards is a part of business etiquette and involves what amounts to as a business card “ceremony.”

Attitudes about age. In many cultures, in east Asia for example, wisdom is associated with age, and older people automatically garner greater respect, reverence, and credibility than younger people. Managers in senior positions are always older (and usually male) and they tend to ignore or avoid anyone much younger than them. In meetings, older managers do most of the speaking and younger managers avoid contradicting them—even when they disagree. In such cases, younger managers from elsewhere, say, the US, might feel disrespected.

Social behavior. In Middle- and Far-Eastern countries, most relationship-building and even formal business happens after-hours at social gatherings. What is considered proper conduct is dictated by local norms although, generally, any sign of inebriation, fraternization, careless or too-casual dress, or sharing of personal details about family or friends is considered inappropriate. Behavior that would be considered acceptable elsewhere—like bringing

a spouse to a gathering or talking to another's spouse—could potentially harm a business relationship.

Although offensive behavior and dress should always be avoided, what is considered “offensive” varies by country. In the Middle East, a woman's head should be covered in public, and men and women are not supposed to greet each other by shaking hands. When working in Dubai (or Beijing or Mumbai), a good rule of thumb is to adopt some of the local customs of dress and behavior (assuming they do not violate a personal or universal code of ethics).

This applies to all kinds of behavior, including gift giving, which in many countries is considered a suitable way to show gratitude, but in other countries is prohibited and where discretion is necessary to avoid violating laws or local customs of etiquette.

Attitudes about time and punctuality. In some Western countries punctuality is everything. Time is viewed as a limited resource, and being punctual assures it is never wasted. People who dither or are late are considered rude and inconsiderate of others' time! But in the Middle- and Far-East and most of Africa the concept of time is viewed differently: more important than doing things punctually is to make sure they are done right. If it takes time to prepare a plan and then revise it, and revise it again, so be it, even if the schedule slips.

Holidays, weekends, vacations. Every country has its own non-work holidays. The US has seven national public holidays. Most European countries have nine to eleven national holidays, with many additional regional holidays. A project that involves participants from, say, four countries with five different national holidays could conceivably face 20 days of holiday downtime. The Ramadan and Chinese New Year holidays affect the schedules of many projects. In the Southern Hemisphere, the summer holidays fall in December and sometimes halt project work for most of the month. The “weekend” in many parts of the world is Saturday and Sunday, but in the Middle East it is Friday and Saturday.

Vacation time off also varies by country and region. Whereas in the US two or three weeks of vacations are standard, Australian law prescribes four weeks, as does the European Union—usually the whole month of August.

Labor regulations. What constitutes a “usual” workday and workweek also varies. French law mandates a not-to-exceed 35-hour workweek, and Chinese law specifies a five 8-hour day workweek. Labor laws are not always enforced, but no project manager in any country should gamble on violating them. Social norms also matter. If the local culture dictates a 6 am to 2 pm “workday,” the local workforce of a 9-to-5 project will probably start dozing off before 3 pm.

Layoffs. In the US, when a project ends and there is no follow-up work, the employees are commonly terminated. But in some countries, termination is not automatic. In many European countries labor laws dictate who an employer can lay off and how to go about it. According to David Pringle of the *Career Journal Europe*, layoff decisions by German employers must conform to social criteria that sometimes force them “to retain staff that is older, have large families, and might find it difficult to get new jobs.”⁷ French employers must “give detailed reports on the progress of staff-cutting programs to state authorities.”

Laws, contracts, rights. The law in effect for a project is the law of the host country, not of the developer's or contractor's home country—although, confusingly, US contractors working overseas must also comply with US law, and the trick is to not violate the laws of either country. The Foreign Corrupt Practices Act, for example, prohibits US contractors from participating in bribery, even though the practice is rather common in many parts of the world.

In countries like China rules are not always enforced and local contractors and customers might tell you to ignore them (of course, risking the possibility that at any time they could be enforced). A safe practice is to verify whatever the locals say about the law and never do anything illegal.

Because of differences in language, terminology, and laws, international contracts take longer to finalize than domestic contracts. Getting the wording and terminology right on

contracts is extremely important, and even minute details (like initialing changes and pages) matter. The project manager should be involved in contract negotiations from the beginning and (this is essential) have access in the host country to her own legal counsel or sound legal advice.

To minimize confusion about contract terminology, the International Chamber of Commerce has created a list of International Commercial Terms, or **Incoterms**, described on its website as “standard trade definitions most commonly used in international sales contracts . . . (and) at the heart of world trade.” Usage of Incoterms in contracts helps clarify expectations and “goes a long way to providing the legal certainty upon which mutual confidence between business partners must be based.”⁸ The contractor must be sure to include stipulations and actions in the contract to protect its intellectual property and be prepared to act should it discover that its ideas, products, or technology are being pirated.

Litigation, payment, meeting contract terms



See Chapter 12

Contracting in international projects is a topic on its own and beyond the scope of this book. While some companies employ standard-format contracts (discussed in see Chapter 12), some large companies prepare their own contracts. In general, contracts should be designed to avoid legal disputes, which in the international arena can be a nightmare—messy, slow, expensive, and sometimes corrupt. They should specify that legal disputes would be litigated in a neutral country, i.e., in neither the host nor the contractor’s home country. US contractors often specify England.

Each contract should provide stipulations to assure that the customer will receive its deliverables and the contractor its payment. This would seem customary in any project, yet because of the extreme difficulties of litigation in international projects the stipulations must be such as to remove even the slightest chance of problems. The contract might impose severe penalties for failure to meet schedules or requirements and offer strong incentives to exceed them (such incentives assume that the contractor is in the position to perform work to meet requirements—which is not always the case in developing countries).

To protect itself, the contractor might specify a large first payment followed by payments upon meeting frequent time-phased targets. Sometimes customer payments are delayed because international funds transfer must be approved by an agency of the host country, which can take 60 days. Sometimes payments to foreigners must be made via tax agents, which takes more time.

Ordinarily, contractors should never perform work for unsecured payment after project completion. In many countries, including China, the system for managing credit and receivables is not particularly good and customer creditworthiness is difficult to ascertain.

Politics. National and local political stability and the government’s position on the project are potential risk factors. Labor strikes, political reform, overthrow of the government, local military intervention, and terrorism are clearly situations that threaten a project. While phenomena such as labor strikes are rare in countries like the US, they are common elsewhere, although seldom do they materialize on short notice and without warning signs. The contractor on an international project should have reliable people in the host region to monitor these signs and keep the project manager informed.

It should be obvious from this discussion that international projects are fraught with problems absent in single-nation projects. The following example illustrates additional problems and what happens when cross-cultural teams ignore the mixing of their norms and customs.

Example 20.1: The Channel Tunnel Project⁹

The initial construction phase of the 32-mile (51-km) Channel Tunnel between Britain and France was managed almost as two separate projects—one starting from Britain, the other from France, both racing to see which would reach the halfway mark first. Competition, it was felt, would speed things up. But the project teams represented two cultures, and the competition only aggravated differences between them and exacerbated problems.

For starters, ideally contracts are written in one language and governed by one legal system, but the Channel Tunnel project had *two* contracts, one in English, one in French, and neither had precedence over the other. Although the contracts were purportedly based on principles common to the two legal systems, legal approaches to health, safety, trade unions, and taxation differed significantly, and a panel appointed to resolve disputes often faced the situation of having to make tough decisions.

The two countries also differed regarding standards concerning, e.g., train engines and cars, railway width, voltages, and signaling systems, although, clearly, in every case there would have to be only one standard in each instance. It was decided that where differences in standards existed between the two countries, the higher standard should prevail—though it was not always obvious which standard that was (e.g., the best way to pour concrete).

Decisions by a democratic government can require substantial deliberation, but decisions by *two* democratic governments require even more deliberation. Simply deciding whether to increase the railcar door width from 600 mm to 700 mm took 9 months.

20.4 Local Stakeholders

Contractors¹⁰

Project teams in foreign countries often must hire local contractors. Although subcontracting to local contractors can reduce costs for labor and relocation, it can increase costs for training and supervision. Sometimes lower labor cost equates to lower productivity, which translates into needing more workers and erasing any potential savings. (Countries such as India, however, have low labor costs yet productivity as high as Western nations.) A local contractor who is familiar with the local customs and bureaucracy can sometimes cut through red tape and avoid hassles that would stymie an outside contractor.

Considerations for selecting a local contractor go beyond the usual criteria of skill, experience, resources, and financial stability. One is the quality of the contractor's written and spoken communications. Another is the contractor's familiarity with common business practices. Practices that in most countries are taken for granted (e.g., RFPs, proposals, SOWs, change orders, and status reporting) may be unfamiliar to a local contractor and challenging to adopt. Also important is the contractor's ethical reputation ("ethical" as defined according to Western standards, not local standards). Although perhaps difficult to undertake, a due diligence review of the contractor's business history, reputation for honesty, and political connections is nonetheless a necessity.

Customers and Supporters

Good relations with customers and supporters are always important, but even more so in international projects. In general, whereas Westerners tend to first set contractual agreements and then build relationships, Easterners first build relationships and then reach agreements. Regardless of the professional track record of the project manager and his company, local businesses, subcontractors, and potential customers are apt to withhold agreement, collaboration, or support until they feel they know the project manager personally. Building personal relationships and trust with business colleagues and associates is fundamental to the business process.

20.5 Financial and Geographic Issues

Many issues encountered in international projects arise from the simple fact that the stakeholders are dispersed across different nations and geographic regions.

Currency and exchange risk

Economic swings that alter exchange rates and relative currency values put project costs, revenues, and profits at risk. For example, on December 6, 2015, the South African Rand traded at R14.35 per US \$; by December 12 it traded at R15.89; and by January 12, 2016, R16.16. For any South African project that depended on imported items, the consequences of this might have been serious.

To protect the value of its contracted work, a contractor should require payment in terms of its home currency (e.g., US dollars for an American contractor), although it must be said that most all international contracts are concluded in US dollars. Customers are likely to agree to this for short-duration projects, though not necessarily for longer projects because of the greater risk of a significant change in exchange rates. Of course, the matter is moot unless the host government grants the customer the legal right to pay for the project in foreign currency.

Example 20.2: Impact of Change in the Currency Exchange Rate

A French contractor agrees to do a project in France for an American customer. The contractor estimates the project will cost €900,000 and, so as to earn a nice profit, prices the project at €1,000,000. To accommodate the American customer, the contract price is set in dollars. At the time of contract signing the exchange rate is \$1.3 per euro, hence the price specified on the contract is US \$1,300,000.

Many months later the project is completed, and the work ends up costing €900,000 as predicted. The customer pays the agreed price of \$1,300,000, but the exchange rate has changed and is now \$1.4 per euro. That being the case, the payment equates to $\$1,300,000 / 1.40 = \text{€}928,571$. Instead of a tidy €100,000, the contractor profits only $\text{€}(928,571 - 900,000) = \text{€}28,571$. An alternative way of looking at this is to say that the increased \$/€ rate led to an increase in the dollar expense of the project (from $\text{€}900,000(1.3) = \$1,170,000$ to $\text{€}900,000(1.4) = \$1,260,000$). Either way, the contractor made less profit.

One way to reduce the exchange risk is to lock in to the contract today's price for a payment that will not occur until later. Called **forward cover** or **hedging** of expected foreign currency transactions, this protects the future cash flow against negative currency fluctuations. The locked-in forward price reflects the difference in interest rates between the customer's and contractor's countries.¹¹

Financial practices and banking

Although financial practices in most countries conform to common standards for the sake of international financial transactions, there are exceptions as the following example illustrates.

Example 20.3 Islamic Finance¹²

In some Muslim regions of the world, financial practices are governed by Sharia law, and some practices for banking, lending, and investing common in the West are restricted or forbidden. The types of allowable financial vehicles are limited, and investments in activities viewed as forbidden are prohibited. So-called **Islamic project financing** dictates what is acceptable in terms of project financing, forms of contracts and insurance agreements, and relationships between investors, contractors, and borrowers.

Islamic law views lending with interest payments as unfairly favoring the lender, who charges interest at the borrower's expense. In the view of Islam, money is considered a measure of value, not an asset; no one should receive income from money alone, and the practice of charging interest is considered exploitative.

Offsets¹³

Foreign contractors on large government-funded projects are often subject to requirements, specified in the RFP, concerning spending in the host country called **offsets** or **counter trade**. For example, the contractor might be required to spend a percentage of project cost on local labor, locally sourced materials or products, local airlines and transportation services, and local subcontractors. Offsets like these that are tied directly to project activities are called "direct offsets." Another form, called "indirect offsets," requires the contractor to contribute to non-project endeavors such as business enterprises or improvements to roads, communications, or other infrastructure, with the purpose of reducing the net currency payments going outside the country. The value of the offset can range from a few percent to more than the full cost of the project.

Export/import restrictions

The export/import of certain US technology, software, and hardware is regulated by government agencies such as the US Departments of Commerce, State, and Agriculture. Early in the

project, systems designers and project planners must identify items that are essential for the project but are restricted or prohibited from import/export; these items will have to be substituted with non-restricted alternatives.

20.6 Project Manager

Typical problems in an international project:¹⁴

- Team members need travel visas.
- Someone on the project team does not have a valid passport.
- Someone needs health tests and inoculations before heading to the project site.
- Someone gets sick or injured at the project site.
- Someone gets arrested for a local traffic violation.

For predicaments like this, the first person people go to is the project manager, expecting her to manage the situation personally or know where to go for help.

The project manager must be self-sufficient. Faced with unique challenges and without support from nearby associates and family, the project manager must be adaptable to the local environment and able to resolve problematic situations that would perplex or immobilize a lesser person.

Sensitivity and acceptance

The local staff, contractors, and laborers might not know what to expect from, or how to deal with, foreign managers. To gain their trust, the project manager must show respect for and acceptance of their culture. Sometimes she does this in subtle ways, like emulating aspects of their social customs, eating local popular foods, or wearing forms of local dress.

Every culture a new experience

Each project in a new country or region requires new learning and familiarization; experiences from one culture or country cannot be generalized to others. For example, although local laborers might *appear* unmotivated or lacking in creativity, the reality might be that they simply do not know what they are supposed to do and need detailed instructions. The project manager must employ whatever motivational sources work best. Sometimes it is a simple matter of adjusting the workday hours to conform to local biological clocks!

Among the challenges of managing a cross-cultural team is awareness of biases when appraising team members. In general, managers tend to appraise people from their own culture more highly than those from other cultures. The project manager needs to ask: if this person were from my own culture, would I assess her the same?

Ideally everyone in the project, but especially the project manager, has the skills to bridge cultural differences. Such skills include:¹⁵

- Understanding how cultural perspectives influence work and collaboration.
- Understanding how national, functional, and organizational culture affects working style, team interactions, and peoples' expectations.
- Sensitivity to the business practices of different countries and regions.

On hand, fully engaged, fully in charge

Ideally the project manager manages the project at the project site and not from a remote office; she is always on hand to see what is happening on the project and discuss problems with local managers, staff, and workers. She is also available to assist team members with non-project matters like currency, housing, and medical assistance. She remains committed to the project and at the site until the project is completed and the customer has signed off. The project team witnesses her making decisions that affect the project and them personally, and in this way she earns their gratitude, respect, and commitment.

When the project manager cannot be on-site, she must remain fully engaged in other ways—through frequent emails, instant messaging, on online conferences, lest project members perceive that she is out-of-touch.

Local project manager

In situations where the project manager cannot be on-site, day-to-day responsibility for the project should be delegated to a **local project manager** who local workers will see as visibly engaged and fully in charge. Thus, each sub-project in a global project will have two project managers, the global project manager who plans and coordinates from the home office and travels among sites, and the local project manager who is on-site and responsible for detailed planning and daily management. The local manager reports to the global manager, and the responsibilities and authority of the two are clearly delineated and understood by the project team.

Hiring and training a local project manager is not easy, so when a problem arises, she should be given every opportunity to resolve it. If the problem is serious and getting worse, the global project manager should “parachute in” either herself or a trusted person to assess the situation and offer help. Only when the situation is deemed hopeless should the local project manager be replaced; hiring a new manager and allowing settle-in time could delay a project up to 6 months.

20.7 Local Representative¹⁶

Every international project should have someone in the host country to mediate with local laborers, unions, and government officials, keep the project manager informed about local matters, and help resolve cultural and regulatory issues. That person—the **local representative**:

- Represents the project manager and company to the customer, and vice versa.
- Keeps the project sold to customers and supporters.
- Arranges in-country services (hotel and car reservations, local communications, interpreters, office staff and space).
- Arranges meetings with government officials, attaches, and consulates.
- Educates the customer about home-country government requirements, e.g., the transfer of technology and technical knowledge.
- Helps arrange local housing for project personnel.
- Assists in locating local subcontractors.
- Informs the project manager about in-country politics and economy.

Qualifications of the local rep include thorough knowledge of the project—its mission, scope, technology, management, and team, and the contractor company—its officers, products, and services. If the contractor is performing several projects in the host country, the local rep should be familiar with all of them.

The local rep must thoroughly know the culture and social customs of the host country and, ideally, be fluent in the local language. It is not necessary that he is a native of the host country, but it is necessary that he is comfortable with local customs and culture. The local rep must be committed to the project and not eager to race off as it nears completion.

One way to secure a local rep is to partner with a local company for some or all of the project, in which case the partner becomes the local rep. A local project manager who is familiar with the culture, customs, and stakeholders can also serve as the local rep.

20.8 Steering Committee and PMO

Sustained backing and support from top management is crucial, yet when a project is far away, experiencing problems, and taking too long, it is easy for managers back home to lose interest. To avoid that eventuality, top management should create a steering committee to guide the project and assign the PMO a role to help manage it.

Steering committee¹⁷

The steering committee for an international project includes senior managers and sponsors from company headquarters and the project's host country/region. For a global project comprised of multiple project sites, the manager in charge of the overall project (the global project manager) is also on the committee. The purpose of this executive or global steering committee is to establish a governance framework to coordinate and fund the project. If the project comprises sub-projects at multiple sites, the committee also sets global goals and coordinates work and resources among the sub-projects.

Each sub-project should also have its own local steering committee comprised of local sponsors and managers and the local project manager. The committee plans and executes details of the sub-project and manages problems originating at the project site or host country. Serious issues that it cannot resolve are escalated to the executive committee.

Role of PMO¹⁸

Besides those functions described in Chapter 18, the PMO for international projects:

- Identifies people to serve as project managers for international projects.
- Assists senior management in assessing and selecting international projects.
- Accumulates lessons learned from international projects, and incorporates them into templates, check lists, and training materials.
- Follows up on issues and problems identified by management that require coordination among multiple international projects.
- Manages files and documentation for international projects.
- Provides support and mentoring for overseas project managers.
- Schedules forums for managers of international projects to share experiences.
- Provides training and education about language, culture, protocol, laws, etc., pertaining to each international project.



See Chapter 18

In general, project personnel departing for overseas should be well informed about the project and aware of what to expect. After they arrive, they should not have to worry about what to do, where to go or stay, or whom to see; such worries detract from their ability to work on the project. The PMO and executive steering committee share responsibility for these matters, arranging for training and coaching, travel and living arrangements, securing a local project rep, and numerous other matters, big and small.

20.9 Team and Relationship Building

The project manager kicks off the international project with a **team-building** session for key members of the project team, including local managers and staff. The purpose of the session is to develop a common purpose and shared expectations, identify likely or foreseeable problems, and develop project guidelines to avoid problems. The guidelines address familiar matters such as collaboration, conflict management, and role assignments, but also problems unique to international projects such as coordination across countries and time zones, and cross-cultural, language, and social factors that could hamper communication and decision making.

The project manager should also meet with each local subcontractor to discuss potential issues and determine which tasks they will do individually and which together. Many of the project's work packages (about 20–30 percent) should be performed jointly by teams from both the host and home countries. This will encourage local workers to take ownership of the project yet allow the contractor to retain control over the work.

20.10 Project Definition

An international project cannot be approached in the same way as a domestic project. As previously described, many potential issues of little consequence in a domestic project become crucial in international projects and must be dealt with.

Customer requirements

In a multi-language project, the process of creating a list of customer needs and wants can be complicated; The customer's list must be translated into the contractor's language, then that list must be translated back into the customer's language for approval. The process can be lengthy, although, typically, Western managers are eager to get it done as quickly as possible. Non-Western managers might take a different stance, preferring to hold off on defining the details. The attitude is, not to worry, misunderstandings are inevitable but will eventually be worked out. In this regard, a key responsibility for the project manager is to build trust, establish areas of agreement, and to not delegate this to someone in business development, sales, or marketing, as is common in the US.

Scope and SOW in global projects¹⁹

For a global project that consists of multiple international sub-projects, the project global steering committee prepares the scope statement, SOW and a preliminary plan specifying the countries or regions of the sub-projects. The plan identifies goals, strategies, targets, costs, etc., for each sub-project, although only in the form of estimates, proposals, or suggestions.

The *local* project manager, local sponsor, and local steering committee for each sub-project then review the preliminary plan and expand it into greater detail to account for their knowledge of the region and site. Also, they make suggestions to the global steering committee about the sub-project’s purpose, goals, benefits, and costs. The process is repeated for every sub-project, resulting in the information illustrated in Table 20.3.

Owing to differences in culture, norms, and languages, sub-projects that start out nearly the same often end up with different purposes, goals, SOWs, etc. (Table 20.3, rows 1–8). To accommodate this difference the global steering committee adjusts the scope and SOW (rows 9 and 10) for each sub-project. During back-and-forth iterations between the global and local steering committees, the scopes and SOWs of the sub-projects and the global project (row 11) are adjusted for compatibility.

The intended outcomes of the process are that:

- 1. Local project managers and teams become committed to their sub-projects.
- 2. Each local sponsor agrees to the goals and scope of the sub-project and promises support.
- 3. The scope, goals, and SOW of the sub-projects conform to local customs, regulations, and laws.
- 4. Stakeholders at the global and local levels agree.
- 5. Goals, scope, and SOW of the sub-projects align with those of the global project.

Work definition and WBS²⁰

Work definition must account for the many factors that distinguish an international project from a domestic project. One approach is to start with a generic WBS template for the technical part of the project and then expand it to include international factors. The highest levels of the WBS will not be much different than for a domestic project, but as activities are broken down into greater detail, matters relevant to the locale begin to surface. It is at the lower levels of the WBS where an international project becomes unique. Although a generic kind of project repeated in each of several countries might look the same in terms of high-level technical

Table 20.3 Project plan to account for local (sub-project) and global project scope and SOW.

	Sub-project in Country A	Sub-project in Country B	Sub-project in Country C
1. Purposes			
2. Goals			
3. Strategies			
4. Cost			
5. Schedule			
6. Benefits			
7. Issues			
8. Risks			
9. Scope			
10. SOW			
11. Goals, Scope, and SOW of global project			

Table 20.4 Issues in international projects.

-
- Team members speak different languages.
 - Expatriate team members need vaccinations, passports, visas, etc.
 - Expatriate team members need local room, board, and transportation.
 - Local team members lack knowledge and skills about project work.
 - Local communication infrastructure is poor.
 - Project leader lacks prior international experience.
 - Expatriate team members lack knowledge about the local culture and host country.
 - Local team members are unfamiliar with business practices of the contractor.
 - Work status might be difficult to determine.
 - Project will at times require people from the home office with critical skills.
 - Local transportation infrastructure is poor.
 - The business needs of the local office differ from those of the home office.
 - Project will depend on vendors who do not have strong presence in the country.
 - Business processes in the host country differ from those in the home country.
 - Technology or material requires export licenses and import approvals.
 - Project or task startup is dependent on success of another project or task.
 - Team members might be pulled off the project due to other higher-priority needs.
-

activities, sub-projects in different countries look quite different at lower work levels because of differences in culture, institutions, geography, and so on. Local or international issues identified in each work package (e.g., as listed in Table 20.4) must be addressed as tasks within work packages or by additional work packages.

One way to keep track of all the detailed tasks and work packages in a global project is with a summary matrix, shown in Table 20.5. The matrix reveals which tasks are unique to certain sub-projects and countries and which are common among many or all of them (the Xs). It also suggests places where knowledge gained from one sub-project might be used in others and helps ensure that important tasks or issues are not overlooked.

Resources, schedule, and budget²¹

Estimates for resources, time, and cost based on domestic experience must be revised when applied to overseas projects. For example,

- Planned resources must be adjusted for differences in equipment and labor productivity levels.
- Schedules and budgets must be adjusted for extra time and costs for communication (fax, phone, courier, translators), travel (air fares, car rentals, taxi and limo fares) and local arrangements (conferences and services).
- Budgets must include fees and costs for insurance, licenses, governmental reviews, local housing, work salary incentives, automobile, daycare, schooling, security, and medical care.

Expenses and lead times must be adjusted for:

- Obtaining passports and visas, and transporting managers and workers.
- Added time and cost for shipping preparation and customs inspection and clearance.

Table 20.5 Summary matrix of tasks versus sub-projects.

Tasks	Sub-project in Country A	Sub-project in Country B	Sub-project in Country C
Technical Tasks			
Survey		X	X
Site development		X	X
Site construction		X	X
System implementation	X	X	X
System test	X	X	X
Training	X	X	
Local Issues Tasks			
Labor		X	X
Subcontractors		X	X
Permits	X		X
Customs	X		X
Time zone	X	X	
Language	X		X

Approach adapted from Seward J. *Managing a Global Project*, pp. 3–4, ETP The Structure Programme & Project Management Company. <http://www.etpint.com/globalproject.htm>. Accessed September 9, 2005.

- Added transport time in the host country to account for the quality of roads and availability of airport, harbor, trucking and local services.
- Approval by the Department of State for material or equipment brought in from the US and deemed as “transfer of technology.”

Extra considerations like these make international projects, *ceteris paribus*, more costly, lengthy, and risky than domestic projects.

Example 20.4: Added Time and Cost of an International Project

A contractor working on an overseas project encountered severe weather that fouled the equipment and stopped the project. Back home, the contractor simply would have brought in other equipment better suited for the weather, but that equipment was not available in the host country and had to be imported.

Problems associated with international transport of the equipment (export licensing, shipping schedules), local transport (local roads and hauling services), and local bureaucracy (customs inspection, and import regulations on equipment) substantially added to the project’s time, cost, and risk. A solution that would have been straightforward in a domestic project became a lengthy, costly, and risky proposition in the overseas project.

The skills and work ethic of local professionals and laborers must also be factored into time estimates and schedules. Owing to language differences, the productivity of a local engineer might be considered equivalent to only half of, say, an American engineer. On the other hand, lower labor costs of local engineers would allow hiring several of them to replace one American. Seldom, however, are such tradeoffs easy to determine.

Example 20.5: Productivity in International Projects

One of the authors has worked with American, Canadian, and German engineers in projects in South Africa. Despite their professional competency, always these engineers needed considerable time before they became as productive as the South African engineers due to factors such as “settle in” time, lack of personal networks, lack of knowledge about local companies and processes, poor understanding of the culture, and communication problems. These put expatriate (non-South African) engineers at a disadvantage and reduced their productivity, at least initially, and prevented them from working at their full potential. Thus, expatriate engineers were given only technical assignments, whereas South African engineers with similar qualifications and experience were also given management assignments.

20.11 Project Monitoring²²

The project manager should require that the local project manager and team leaders submit weekly task updates, these days using the Internet. If technical work packages have been defined to be of short duration—say 2 or 3 weeks—the project manager will be able to readily discern from weekly reports whether they have been completed or are on schedule or behind.

When a local subcontractor starts to fall behind, the project manager needs to step in and take a direct role in managing the subcontractor’s work; if that is not possible, she should assign a local person to assist the subcontractor. International litigation can be a big hassle, so it is better to first try to coach a subcontractor into getting back on track rather than resort to legal action.

A project manager who cannot be on site must rely on telephone and Internet meetings. Good practice is to precede all such communication with written documentation and directions so local workers will know what to expect and how to prepare. Also good is for the project manager to make frequent unannounced visits to observe actual progress and how well expected procedures are being followed. Nowhere is the value of site visits and visibility more important than in international projects.

20.12 Communication

Communication plan²³

In addition to the contents described in Chapter 4, the **communication plan** for an international project must address the difficulties that arise from differences in languages and time zones. The plan should identify important contact persons (Who’s Who) in the host country, home country, and elsewhere. Everyone—domestic and foreign project staff and



See Chapter 4

subcontractors—must understand the required reports and written communication, and the content and format of each, even for “common” documents like change orders, which they might not be familiar with.

When the project involves multiple languages, a common, “working,” language should be adopted for all or specific portions of the project, and everyone working on the project should be reminded to speak slowly and use simple terms and no slang. The project newsletter should be published in multiple versions for the different languages spoken on the project.

Meetings

The communication plan should include a tentative schedule for all formal reviews and milestone meetings. Since formal meetings in international projects can be difficult to schedule, require much preparation, and expose people to cultural gaffes, the fewer of them, the better. The project manager should meet with local customers or officials before formal meetings to discuss any major problems.

The primary means for tracking status and identifying problems should be one-on-one communication and frequent *informal* meetings, convened as needed, the time and place to be determined by urgency and purpose, e.g., alternate weeks if everything is okay, more often if not, and at the site experiencing problems or issues. Attendance should be restricted to meeting contributors and those who would benefit from being there.

The team in an international project might be dispersed across multiple locations around the globe, so most meetings occur via audio or Internet conferencing. In the interest of building personal relations and team trust, begin every meeting with a few minutes for members to talk about their families, hobbies, and interests, and before a holiday to explain the holiday and its customs. It is not that members need to know much about each other, but to show they *care about each other enough* to want to hear about their personal lives and holidays.

Setting time for virtual meetings can be problematical. Says Cohen, “It’s not the distance, it’s the time difference.”²⁴ Cape Town and San Francisco are roughly equidistant from London (9,700 km versus 8,600 km), but the former is 2 hours ahead of London, while the latter is 8 hours behind. With such time differences, the project manager should try to “share the pain” in scheduling meetings.²⁵ Chicago and Mumbai are 10.5 hours apart and allow no convenient meeting times in both places. The solution is to rotate meeting times—sometimes 8 am Chicago time (6:30 pm in Mumbai), sometimes 8 am Mumbai time (9:30 pm in Chicago). This should hold even if Chicago is the project home office with 30 people and Mumbai is a satellite team with six people. Rotation reduces perceived “power” differences among members. If the project team is dispersed around the world, the number of people involved should be minimized to “minimize the pain” by requiring only a few members from each place to participate.

To raise awareness of differences, the project manager should distribute a guide to all team members showing for every member their country, time zone, and the times, days, mealtimes, and holidays when they say they cannot or prefer not to meet.²⁶

20.13 Risks



See Chapter 11 or country.

Any standing risk policies of the contractor or customer (described in Chapter 11) should be applied in a consistent manner across all projects in all countries. In other words, a company’s risk tolerance as expressed in the risk policy should remain the same, no matter the project

As discussed in Chapter 11, risk analysis begins during project conception and definition by imagining different scenarios about what could go wrong. In an international project much of the uncertainty relates to ignorance about local culture, language, institutions, infrastructure, and so on; thus, an important strategy for reducing the risks is to *learn* about these things: the more you know about these matters, the better you can identify and mitigate the risks.

Another strategy is to decrease the amount of learning necessary, especially regarding local regulations, laws, and resources. This is done in the following ways:²⁷

- **Outsource activities that are heavily restricted by local regulations.** Purchasing land, obtaining permits, hiring locals, and moving materials through customs are risky because they involve local laws and customs. By hiring knowledgeable subcontractors to assume these activities, the burden of risk is shifted largely to them.
- **Perform technology-intensive work at home.** Rather than dealing with the uncertainties of local labor, materials, and infrastructure, do as much as possible of the work for the project at home, reserving on-site work for assembly and installation.
- **Sign contracts under international law or third-country law.** Rather than learn the intricacies of local laws and depend on local lawyers, finalize all contract agreements according to international law or a neutral third country where the laws are familiar.

Most companies employ a mix of the above—they learn about and deal with some aspects of the host country and culture themselves but avoid having to learn about and deal with others. In general, the more a project requires the project manager to be “imbedded” in a foreign country, the more she must learn about that country. Contractors that do large construction projects that take years, are large in scope, and rely on local resources are, *de facto*, heavily imbedded in the local environment, so managers in these firms must *learn* about the country or region of the project. At the same time, they reduce their need to learn about *everything* by, wherever feasible, doing much of the work at home—e.g., prefabricating deliverables, and by outsourcing to local suppliers and hiring local representatives to deal with local issues.

Of course, the project manager who must be on-site is always “imbedded” in the host country—even when the contractor (his employer) is not. Although knowing the local practices, ways, and protocols might not matter to his firm, it does matter to a manager who must live and work in the host country. Of all the ways to reduce the risks in an international project, perhaps the overall best is to learn and adapt to the local customs, laws, infrastructure, and social norms, and build relationships with the leaders, subcontractors, laborers, and officials of the host region.

20.14 Project Failure in Developing Nations²⁸

Development projects in countries with emerging or developing economies are more likely to fail. To use projects in Africa as an example, the World Bank reported that half of its funded projects there failed—where, e.g., libraries were filled with books in English where none of the locals knew the language, and rooms filled with computers were locked to prevent entry. The main reasons for these failures and suggested solution are as follows:

- **Inflexible project management methods** are applied that do not fit the local context. Managers focus on expenditures and accountability and utilize detailed plans and gate reviews, often to the neglect of local environmental, social, and cultural matters. A \$22 million fish processing plant was built in Kenya to provide jobs for the local people. But among the locals, few ate fish or fished commercially, and they looked down on

those who did. Also, operating freezers and obtaining clean water for the plant were too expensive in the desert. The plant ceased operations after only a few days.

Lesson: Project management methods must be **adaptable and compliant to the local context**. Traditional PM practices must be modified to adapt to the locale and actively engage local stakeholders.

- The **political climate** in emergent economies is young, evolving, and can be cruel. The World Bank funded a \$4.2 billion oil pipeline to the Atlantic from Chad, intended to generate income to help develop the country. After two years, Chad's president decided all oil money would be spent on weapons and regime support, otherwise the oil companies would have to cease operations, which they chose not to do. This situation continued for 16 years until the president's death. Meantime the population lived in abject poverty.

Lesson: Infrastructure development projects cannot succeed without the **support of local governments**. International donors and sponsors must work with governments to reach agreements about what happens if a government goes rogue or decides to renege on agreements.

- Projects designed to help the locals are **run by outsiders**; the locals are sidelined, do not feel a part of the project, and their needs are not met. The PlayPump project was intended to solve a drinking water problem for millions of Africans, using thousands of water pumps powered by merry-go-round devices that pumped water whenever children played on them. But villagers needed water even when children were not playing, and not enough was pumped when they were, so they had to rotate the merry-go-rounds themselves, which was too difficult for the older ones. The villagers returned to using traditional hand pumps and stopped using PlayPumps.

Lesson: **Buy-in and involvement from all stakeholders** is essential. Donors, national and community governments, agencies, contractors, but most importantly the **local citizenry**, must agree that the project is technologically feasible, fits with the local culture, is practical, and will work.

As mentioned, international projects are sometimes also very large projects—they are megaprojects, and this raises a host of additional problems that exacerbate the difficulty of managing them. Just as managing international projects involves additional considerations, so does managing megaprojects. These are discussed next.

20.15 Megaprojects²⁹

Megaprojects are large, complex ventures that cost \$1 billion or more, last for years, involve multiple public and private stakeholders, and sometimes impact millions of people. Megaprojects in the \$50–\$100 billion range are common. Examples of megaprojects include:

- high-speed rail lines
- airports and seaports
- Olympic games
- high-energy particle accelerators
- stadiums
- energy projects—oil, gas, wind, dams
- development of products for defense, aerospace, and information–communication technologies.

Not surprisingly, many megaprojects are also international projects. A good example is the \$150 billion International Space Station—designed and built by the space agencies of the US, Russia, Europe, Canada, and Japan. Megaprojects are risky and tend to suffer prolonged delays and big cost overruns and fall short on promised benefits. Despite that, the number and size of megaprojects continue to grow.

Megaproject drivers

The following “drivers” explain the demand for megaprojects:³⁰

- **Technology.** Engineers and technologists aim for projects to be the “first” and to create the “biggest or tallest or most advanced” of anything (e.g., the US F-35 fighter).
- **Politics.** Politicians build monumental infrastructure to benefit themselves and their causes and attract media attention that gives them wide public exposure (UK-French Concorde airliner).
- **Economics and jobs.** Enormous budgets attract a spectrum of stakeholders such as businesses and trade unions in search of profits and jobs (Boston’s Big Dig).
- **Aesthetics.** Stakeholders who admire art and clever design push for projects that create iconic and beautiful structures (Sydney’s Opera House).
- **Community pride.** Overlapping with other drivers, communities see projects as providing economic and aesthetic benefits (stadiums—Los Angeles’ SoFi and London’s Wembley).
- **Altruism and sustainability.** Stakeholders concerned about human welfare and ecology support humanitarian and environmental projects to improve both (North Wales tidal lagoon).

Barriers to success

The barriers to megaproject success are many, although thanks to the drivers listed above, they are often downplayed or overlooked. For example:³¹

- Due to long lead times, enormous size, and complexity, megaprojects are inherently risky and subject to “black swan” (improbable but disastrous) events.
- People in charge often do not have the necessary managerial and technical competency.
- The decision process involves numerous stakeholders with conflicting interests.
- Technology and designs are often non-standard, unique, and immature.
- Commitment to the project concept is locked in early before alternatives are evaluated.
- Large budgets drive behavior that puts profits over value and benefits.
- Complexity and unplanned events are not adequately accounted for in plans and budgets.

The results of these barriers are schedule and cost overruns and benefit shortfalls. But not all megaprojects fail in all these dimensions, and while many fail in terms of time or cost, they succeed in other ways. The Channel Tunnel is an economic disaster but otherwise a success and a delight to travelers going between the UK and France. As is typical of megaprojects, the success of the end-item overshadows financial failure (even though the latter sometimes weakens the economies of companies and countries, and can lead to bankruptcies, unemployment, inability to provide citizens with adequate services, and, in the extreme, political instability). Examples of such projects cited elsewhere in this book are London’s Jubilee Line Extension, Boston’s Big Dig, Denver’s International Airport, Sydney’s Opera House, and the Anglo-French Concorde airliner.

Most of the methods for project planning and control discussed in Chapters 4–13 directly apply to megaprojects, as do the leadership and stakeholder concepts in Chapters 16 and 17. However, due to a megaproject’s size, scope, and duration, the impacts of **front-end planning**, **stakeholders**, and **leadership** are especially significant.

20.16 Megaproject Conception, Planning, and Organization

Conception and front-end loading

The conception phase of megaprojects is the same as in other projects. It involves careful **feasibility studies**, **costbenefit analyses**, **business case studies**, and **environmental impact studies**. As described in Chapters 3, 4 and 19, these require demand forecasts and time and cost estimates. Long-term forecasting is difficult at best, but due to the megaproject drivers described above, forecasts and estimates for megaprojects tend to be biased and optimistic—the concept of *overselling*. Time and costs are underestimated, and benefits are overestimated, resulting in benefit–cost ratios and rates-of-return that are unrealistic and misleading, and the approval of projects that are not economically viable. For example, investors in the Channel Tunnel had been promised a 10 percent rate of return, but the current ROR is closer to minus 14.5 percent thanks to lower-than-forecasted tunnel traffic and greater than expected costs. The Olympic games offer more examples of megaproject losses—\$2 billion for the 2016 Rio de Janeiro games, \$7.2 billion for the 2020 Tokyo games, and \$14.5 billion for the 2004 Athens games. In many cases, the structures and venues constructed for the games thereafter sit vacant or relatively unused.

Chapters 3 and 4 describe **front-end loading—FEL**—as part of the early project-planning process. FEL is especially important in megaprojects, although it is often hurried because project supporters want quick approval, or contractors are not competent planners.

Megaprojects can often be conceived as complex “supersystems,” wherein the systems and subsystems are interdependent and must all be designed, built, and integrated, and for which the **systems engineering** process, described in Chapter 2, is appropriate.³² Unfortunately, however, except for defense and space-related projects, rarely is the process applied in megaprojects. (Case 2.2, London’s Jubilee Line Extension is an example.)

Organization and integration

Megaprojects, as complex systems with numerous components, are managed by decomposing them into smaller projects or areas with clearly-defined interfaces, and relying on sponsoring agencies or contractors, called **system integrators**, to coordinate the efforts of the multiple (often numerous) organizations involved in the projects. The system integrator understands all the project components and their interdependencies, and manages their interaction. The organization and integration of large-scale projects is discussed in Chapter 15.

The biggest projects rely on two levels of integration.³³ The **meta-systems integration** level understands and oversees the entire megaproject—the total system of systems; it manages interfaces with external stakeholders and coordinates efforts of the component projects and other activities that comprise the megaproject. The other, **system integration**, level is responsible for managing each individual project (which itself might be very large) as a somewhat independent subsystem with clearly defined interfaces to allow coordination with other projects within the megaproject.³⁴



See Chapter 3
and 4



See Chapter 2



See Chapter 15

The meta-systems integrator manages whatever influences the overall megaproject—estimates, budgets, schedules, contracts, logistics, and procurement of long-lead-time items. Beyond that, decisions about projects or areas of the megaproject not impacting the others are pushed to the next level; these would be system integrators for very large projects, and project managers for smaller projects. To the extent that the component projects in a megaproject are often somewhat interdependent and happen concurrently, megaprojects are better managed as exceptionally large projects with system integrators than as programs, which are discussed in Chapter 18.



See Chapter 18

Beyond the system integrator is the role of **interface management**. An interface is wherever different functions or organizations come together. Disputes and mistakes are common at interfaces, and a megaproject with hundreds of organizations will have thousands of interfaces. To avoid mistakes, every interface must have a competent person who understands issues at the interface, has skills to communicate them, and clout to resolve them. Unfortunately, interface management is sometimes weak in megaprojects.³⁵

The consequence of poor planning and inadequate integration and interface management is paradoxical: the *biggest, most difficult, and riskiest projects are approved over smaller, more viable and less risky projects*—what Flyvbjerg calls “survival of the unfittest.”³⁶ Consequently, when problems arise during project execution, the project is stopped to search for solutions. Meantime, costs and interest payments continue to mount, and revenue income is pushed farther into the future—a disastrous scenario for debt-financed projects. Flyvbjerg calls this approach to megaprojects the “break-fix” model: due to haste, poor planning, incompetence, or lack of incentives, supporters undertake a project that eventually “breaks”; they then pause it in attempts to reorganize and “fix” the project, and to deliver something that looks like the promised outcome and gives the appearance of success.³⁷ Due to the megaproject drivers, contractual commitments, and regret over sunk costs, megaprojects are rarely cancelled.

The cure to break-fix is better front-end planning and better system integration so projects will not break. This begins with better forecasts and estimates, which, says Flyvbjerg, depend as much on human interests and influence as data and math. Estimates of costs and benefits should reflect the perspectives of a broad **range of stakeholders**—those who not only support and promote the project but are objective, skeptical, or even opposed to it.

20.17 Megaproject Stakeholders³⁸

A distinguishing feature of megaproject stakeholders is that they constitute large groups, which is why most megaprojects have numerous stakeholders and the practice of good **stakeholder management** (described in Chapter 16) is so important. They include, for example:



See Chapter 16

- **Citizen stakeholders.** In, for example, a rail system project, these include people and communities situated in the path of and next to the rail route, travelers who ride the trains, businesses who benefit from its construction and operation, taxpayers who help fund it, workers who build and maintain it, and people and groups who oppose it and think its funding would be better used elsewhere.
- **Organizational private sector stakeholders.** These are organizations involved in the project’s approval, funding, execution, and promotion; they include banks, investment firms, media companies, environmental groups, and contractors—primes, subs, suppliers, etc. Stakeholders within these organizations range from leadership positions to managers and workers at the site.
- **Public sector stakeholders.** These are organizations that provide the approvals, support, and sometimes funding necessary to undertake the project; they include local, regional, and national governmental agencies.

- **Infrastructure stakeholders.** Many megaprojects are infrastructure projects, and their stakeholders include power, water, and telecommunications utility companies, resident and community associations, and national and international interest groups and NGOs.



See Chapter 10 Chapter 10, the FIFA 2010 World Cup in South Africa.

A good example of stakeholders and their involvement in a megaproject is Case 10.2,

Stakeholders and front-end tradeoffs³⁹



See Chapter 3,
12, 19

The FEL process includes selecting the “solution” or end-item that is thought best to provide benefits and meet stakeholders’ needs and requirements. Ideally, the process starts with a range of viable solutions or opportunities, and through rigorous evaluation selects one. This is the **project-selection** and **procurement** process described in Chapters 3, 12, and 19. But in megaprojects the process is seldom straightforward and involves trade-offs and compromises, often because of the range of stakeholders. These include, for example:

- The “**knowing versus exploring**” tradeoff. This tradeoff is the compromise between exploring a wide variety of solutions or opportunities versus generating the information necessary to evaluate any one of them. The fact is, until a solution is already somewhat well-defined, it is hard to probe the details necessary to learn more about it. In other words, you must commit to a solution just to get the information needed to evaluate it! Consequently, decision makers tend to pick solutions about which they know (or think they know) something over others about which they know nothing.
- The “**pluralism versus support**” tradeoff. This is the compromise between encouraging the broadest possible scrutiny of solutions versus emphasizing the one solution that already has some support. Ideally, decision makers seek feedback from a broad range of stakeholders, for example through public hearings, but that invites many stakeholders with different and potentially competing interests, which could delay reaching agreement on any one solution.

Tradeoffs such as this distort the front-end planning process and result in compromised decisions and projects that don’t deliver what they should, or do deliver but at exorbitant cost.

20.18 Megaproject Leadership

Megaproject success is especially dependent on project leadership and the ability of different leaders to work together. A good example is the Rocky Flats megaproject described in Example 20.6.

Example 20.6: Rocky Flats Decommission and Cleanup⁴⁰

The **Rocky Flats project (RF)** was the first cleanup of a major nuclear site in the US. Originally estimated to take 70 years and cost \$36 billion, it was completed in just 10 years, cost \$7 billion, and met or exceeded all state and Environmental Protection Agency (EPA) safety standards. The project site included a 385-acre facility where 3,000–7,000 workers had produced nuclear weapon components over a 40-year span. It was occupied by nearly

a thousand structures, many built of reinforced concrete, with over 21 tons of uranium and plutonium wastes and toxins stored in thousands of containers, tanks, and pipes, and 10,000s of cubic meters of contaminated soil—all that had to be—and ultimately was—successfully removed (Figure 20.1). Project stakeholders included federal agencies—EPA and



Figure 20.1

Rocky Mountain Flats Nuclear Facility, 1995 and 2015.

Sources: Wikimedia and US Fish & Wildlife Service.

Department of Energy (DOE); prime contractor Kaiser-Hill; hundreds of smaller contractors; unions; citizen action groups; environmental groups; and community, city, state, and federal administrators (mayors, governors, and representatives from the state of Colorado and the US Congress). Many of these were initially adversaries (of each other and the project), but almost all came together as partners and project advocates.

The Rocky Flats project illustrates many aspects of leadership crucial to megaproject success, as explained in the following sections.

Shared leadership and leadership continuity

Megaprojects are excessively big, so unlike smaller projects, no one manager can single-handedly drive everything. Whereas a smaller project might have one project manager and a few other managers and leaders, a megaproject has **many of them** dispersed among the many stakeholders. Some are more prominent than others—they make high-level decisions, remove barriers, build support, and drive the project vision. In the RF project, these included company CEOs, senators and congresspeople, secretaries of DOE and EPA, state governors, and senior managers and project directors. A commonality among all of them was shared goals and coordinated effort, which is the next point.

The RF project lasted 10 years, so unlike shorter-duration projects that might have the same project manager start to finish, managers and leader in a megaproject will **change** positions or be replaced. During the RF project, leadership at all levels changed, sometimes often; for example, the CEO of contractor Kaiser-Hill changed three times. In the face of so much change, one thing that must remain constant is the project's direction. At RF, that direction never wavered thanks to **leadership continuity**—the shared, consistent goals and coordinated efforts of all the leaders. Successors picked up where their predecessors left off, and common vision and consistent leadership were maintained everywhere.

Leadership roles

Few projects succeed because of one star or hero; rather, there are *many* of them. The same can be said about a megaproject, although three roles (which might have multiple individuals in each) stand out:

- **Idea champions** are those who *verbalize and motivate the vision of the project and energize people to follow it*. At RF, the role was filled by several people including the CEOs of Hill-Kaiser and senior administrators at DOE and EPA. These leaders originated the vision but, most importantly, the project succeeded because the vision was widely shared by all relevant stakeholders.
- **Sponsors**: those who *acquire the resources and support necessary to accomplish the vision*. On megaprojects, the client organization often appoints an influential executive sponsor to be the client–project organization interface and to support the project director, particularly on issues beyond her control. The sponsor's responsibilities include developing the business case and, ultimately, monitoring the benefits anticipated from the business case. The client of the RF project was the federal government, and the sponsor role was filled by the Governor of Colorado and regulators at EPA and DOE.

- **Orchestrators**, who assemble the resources, integrate groups and organizations, and coordinate the efforts of stakeholders to achieve the vision. At the RF project they included the lieutenant governor of Colorado plus project directors and managers at all levels and areas at the Rocky Flats site.

Leadership emphasis

Rocky Flats is an example of a project that began with every card stacked against it. It was a dirty site with few good solutions for how to clean it, and relations among its stakeholders—citizens, state and federal agencies, labor unions—were adversarial. Everything about it was negative, including that the site, which was a production facility, would be closed and everyone working there would be out of a job.

One reason the project succeeded despite all that, according to Cameron and Lavine, was the project leaders' *emphasis on success*.⁴¹ Rather than focusing solely on solving the project's many problems and achieving only acceptable results, they emphasized searching for opportunities and striving for exceptional and virtuous results; rather than working with what was possible, they emphasized working to everyone's highest potential. They encouraged innovation, creativity, and experimentation. Much of the project involved dismantling, containing, and transporting contaminated wastes, for which preexisting procedures were inadequate. Workers at the site invented innovative procedures that were simpler, more practical, and more efficient than preexisting methods—a main reason the project finished decades earlier and \$29 billion less than estimated.

Beyond this, what else did leaders do to contribute to project success? Notably, they practiced planning methods as described throughout this book, but their most significant actions fell into the "behavioral" category and included creating and maintaining the following:

- **Clear, shared, inspiring vision** that stakeholders found meaningful beyond personal benefits. At RF, the vision was something like "we will shut down and clean up Rocky Flats in 10 years and for \$10 billion, and when it's done you can bring your children to a restored prairie ecosystem."
- **Clear goals, objectives, and targets**, constantly reinforced by senior leaders. This enabled work to be prioritized and helped stakeholders establish partnerships.
- A **project culture** that promoted **collaboration among stakeholders**. This was encouraged by, e.g., collating managers and workers at the site with frequent face-to-face meetings, and actively engaging **external stakeholders** to mitigate antagonisms and foster common ground and shared agendas.
- **Trust and integrity**: contractors delivered on their **commitments** to targets and objectives and were **honest** about problems encountered and actions taken.
- Strong **political relationships**. Like many megaprojects, RF required government approval, funding, and support, so political relationships were cultivated and carefully managed.
- **Shared responsibility and accountability**. Contracts were structured around responsibility and risks, and rewards were shared among multiple stakeholders, including employees. Contracts were **incentive-based** and focused on outcomes to encourage innovation and creativity in solving emerging problems and to combat complacency. (The pros and cons of incentive contracting are beyond the scope of this book. See, e.g., Merrow.⁴²)

Although drawn from the RF project, these points apply to virtually all megaprojects.

20.19 Shortening the Megaproject Life Cycle

Among reasons for megaproject failure, a big one is long duration and trouble in estimating costs and demands far into the future. Forecasting 2–3 years ahead is difficult, and further out is almost impossible. The lengthy timeline often comes from the fact that many things in megaprojects are newly created—and, as a rule—take much time and involve risks that cannot be foreseen or imagined. Assuming that issues relating to front-end planning and reaching agreement among important stakeholders can be satisfactorily addressed, then the likelihood of success can be increased by shortening the project’s duration.

One way to do that is to apply a **hybrid** project management methodology that incorporates concepts from agile project management, APM, described in Chapter 14. Although megaprojects typically deal with big, complex, one-of-a-kind end-items for which traditional project management is appropriate, some concepts from APM that speed up projects are applicable. These include short-cycle iterations and multiple releases, where each release provides new knowledge and a potentially revenue-generating product. An example is Madrid’s Metro subway system megaproject.



See Chapter 14

Example 20.7: Madrid Metro Subway System⁴³

Madrid Metro overhauled and extended its subway system to create 95 miles of new track (many through newly bored tunnels) and 76 new stations. It was completed in two stages of 4 years each, for a total cost of \$6 billion—about half the time and cost of comparable subway systems in Europe and the US.

The megaproject focused on delivering value through short-cycle repetitions. For example, rather than treating every station as new and different, virtually all stations in the Metro system used the same design, repeatedly. This concept, dividing a system into sub-systems that can each be easily replicated, called **modularization**, is discussed in Chapter 2. The same concept of repetition was applied to boring new tunnels for the system. Rather than boring one or two tunnels at a time, as is typical, the project used as many as six boring machines at once, each working on a different segment of the system—doing work in parallel that might have been done in sequence. The boring teams were friendly competitors, meeting after-hours at tapas bars to discuss their progress and share ideas. In this way, knowledge about the best boring techniques spread and was replicated throughout the system.

Beyond the above measures to reduce time and costs, another was the use of tried-and-true, **off-the-shelf** technology. The project made use of readily available products and existing know-how, which generally minimizes project costs and risks. The project also saved time and costs by resisting some of the megaproject drivers listed earlier:

- **Technology.** No attempts were made to be the first to use the newest technology or to be the biggest or best of anything.
- **Politics.** The focus was on serving the community, not on political ambitions.
- **Aesthetics.** No famous-name (and expensive) architects were involved.

Rather than one eight-year project, the two-phased approach enabled the project to begin generating revenues in four years after the first phase was completed. Perhaps most important is that the Madrid Metro megaproject succeeded not only in time and cost, but



See Chapter 2

in functionality. It is efficient, effective, aesthetically pleasing, and does exactly what it was designed to do—provide Madrid’s commuting public with safe, efficient transportation.

20.20 Summary

A project that is international in scope, a megaproject, or both, automatically inherits more issues and greater risk than one that is not. These issues touch most everything about project management—leadership, interpersonal relations, stakeholder involvement, communication, planning, risk management, and monitoring.

In international projects, the project manager must be able to work with local subcontractors, suppliers, customers, business associates, and officials. Often these stakeholders withhold effort, collaboration, or support until they feel they know the project manager personally, so gaining personal familiarity and building relationships is fundamental to managing international projects.

When the project manager cannot always be on-site, she should appoint a local project manager to oversee detailed planning and daily management. Even when she can be on-site, she should appoint a local representative to keep her updated on local matters and to help mediate with local stakeholders and resolve local issues.

A global project should have an executive steering committee to oversee governance and funding, set goals, and coordinate work among sub-projects at different sites. Each sub-project should have a local steering committee to plan and execute local details and manage local problems. This applies to megaprojects too.

Definition and planning for an international project requires identifying the many issues and unknowns associated with culture, country, laws, people, etc., and accounting for them in project plans, schedules, and budgets. Managers and others familiar with the local environment must be consulted and involved in preparing detailed plans. Most everything requires greater effort, time, and cost, which must be factored into plans, schedules, and budgets.

Many risks in international projects stem from ignorance about local and international customs and conditions; thus, one of the best ways to reduce risk is to learn about local customs, laws, infrastructure, and social norms, and to build trusting relationships with local stakeholders.

Megaprojects are common, the result of a combination of technology, politics, economics, competition, aesthetics, and pride. They tend to be risky and suffer prolonged delays and big cost overruns; among reasons are incompetency, numerous stakeholders, immature or unique designs and technologies, and inadequate front-end planning given the projects’ size and complexity. Regarding the last point, careful front-end loading and application of systems engineering are especially important in megaprojects, although both are too often not applied or hurried and inadequate.

A prominent feature of megaprojects is the substantial number and wide range of stakeholders (especially for infrastructure and sports venue projects), which makes it difficult to reach early agreements on major aspects of the project. The result is tradeoffs: decisions that do not necessarily provide the best results or meet everyone’s expectations.

An important feature of successful megaprojects is leadership. Big, long-duration projects have many leaders, but among them one feature stands out: continuity—i.e., agreement on project goals and coordination of efforts to achieve goals. Even as the leaders change, project

goals remain constant. Among attributes of successful megaproject leaders are providing a clear and inspiring vision, clear goals and objectives, collaboration among stakeholders, trust and integrity, strong political relationships, and shared responsibility and accountability.

A big reason for megaproject problems is long duration and trouble in estimating costs and demands far into the future. One way to speed up a project is to apply hybrid project management methodologies that incorporate concepts from agile project management. Other ways to succeed in megaprojects are to be aware of and avoid the reasons for their failure—especially regarding technology, politics, and pride, all which tend to drive up cost, time, and uncertainty in big projects.



Review Questions

1. Consider the analogy of an international project being like a play. In international projects, who are the actors, what are the scripts, what are the sets, and what are the props?
2. What are the four main categories of “unknowns” in an international project?
3. In the above list of unknowns, which are “implicit” and which are “explicit”? Why are implicit unknowns potentially more problematic for the project manager?
4. Describe each of Hofstede’s five cultural dimensions. How is awareness of these dimensions relevant to project management?
5. Consider two countries with which you are familiar. Compare and contrast them in terms of the following: Hofstede’s five cultural dimensions, language, formality, gift giving, attitudes about age and about time, food and drink, holidays and time off, and customary labor time.
6. Why might worker layoffs after the project cause legal problems for the contractor or employer?
7. For an overseas project, whose laws prevail, the host country’s or the home country’s?
8. What are “Incoterms”?
9. What legal problems are associated with contracts in international projects? What steps should be taken to avoid them, or to deal with them should they arise?
10. How can the project manager know in advance of impending political or labor/union problems in the host country?
11. What are some benefits of hiring local contractors in an international project? What are the drawbacks and difficulties?
12. Describe the role of informal gatherings and social events in building trust.
13. Describe ways a contractor can protect against rising costs or falling prices resulting from fluctuating exchange rates.
14. What is an “offset”? What is the difference between indirect and direct offsets?
15. Name some forms of export/import restrictions. In what ways can they impact an international project?
16. A project involves team members in New York and Rome. Discuss how you would accommodate the 6-hour time difference to maximize communication and coordination between them.
17. In global projects that include sub-projects at multiple sites, who is responsible for day-to-day oversight of each sub-project at each site?
18. Can it be assumed that a technology or process that proved successful in a project in one country will automatically be successful in an identical project in another country? Explain.

19. Who should be trained in the cultures, traditions, and regulations of the home or host country, the managers and staff who will be going to the host country to work on the project, or the local managers and subcontractors who will be working on the project for a contractor that is based overseas?
20. What are the responsibilities and qualifications of the local representative?
21. What is the role of the project steering committee (or governance committee or review board) in an international project? What is the difference between the global and local steering committees?
22. What is the role of the PMO in an international project?
23. What are ways to build teamwork and encourage cooperation between members of the project team from the home country and host country?
24. What are ways to build relations with local vendors and officials? Why are these relations so important?
25. How can the project manager learn about the host country and potential risks related to culture and environment in the project?
26. Discuss the process of developing the scope and SOW for sub-projects in a multi-site, multinational global project.
27. Describe the WBS for identifying the unique issues of an international project. How is the technical WBS of an international project similar to or different than the technical WBS for a single-country, domestic project?
28. Name some issues that the WBS in an international project might have to address.
29. Describe the purpose and content of the summary matrix in Table 20.5.
30. Comment on the size of work packages in an international project. How are work packages tracked and controlled?
31. List some factors that must be accounted for in estimating resources, time, and cost for an international project, and in establishing budgets and schedules for the project.
32. What special issues should the communication plan for an international project address?
33. What are strategies for handling risks in international projects?
34. What are the “drivers” in megaprojects?
35. What are barriers to success in megaprojects?
36. What is the role of “front-end loading” in megaprojects?
37. Name some groups to which megaproject stakeholders belong.
38. Discuss the concepts of shared leadership and leadership continuity in megaprojects.
39. Identify a megaproject that made headlines in local or national news sources, then discuss the reported issues or risks. Provide some recommendations based on the material in this chapter.
40. Flyvbjerg says, “. . . just like you would not want someone with only a driver’s license to fly a jumbo, you don’t want conventional project managers to manage megaprojects.”⁴⁴ Explain this statement.
41. Merrow states that, typically, megaprojects either perform better than planned or, very often, far worse than planned.⁴⁵ The Mozal Project (Case 20.1) is an example of a successful megaproject.
 - a. What other megaproject case(s) mentioned in this chapter performed better than planned?
 - b. List some megaprojects mentioned in this book that overspent and delivered late.
 - c. Do the products of those late and overspent projects ever excite end users or deliver excellent value? Explain the difference between project management success and project success and name some megaproject examples cited earlier. (Hint: also see Introduction section of the book.)



Questions About the Study Project

If your investigation project was a global or international project, or involved customers and/or contractors overseas, consider the following questions.

1. What did the contractor and/or project manager have to do in this project that differed from a typical domestic project?
2. Discuss aspects of the country, culture, language, and social behavior of the host country that challenged the project manager.
3. How did the project manager and staff learn about the culture and traditions of the host country? In your opinion, were they knowledgeable and well-prepared to work in the host country?
4. What difficulties were encountered that stemmed from the international nature of the project? Could they have been avoided through better planning?
5. Discuss the following roles, as appropriate: of the local project manager; of the steering committee, of the PMO.
6. How did the project manager identify special issues related to the international nature of the project and account for them in planning the project?
7. What adjustments did the project manager have to make in estimating the resources, time, and cost of the project to account for differences in countries supplying labor and materials to the project?
8. What strategies were employed to identify and reduce project risks?
9. If the project was a megaproject, list the factors that you attribute to its success or failure. In answering this question, be sure to specify what kind of success or failure you are referring to—project success or project management success? The answer to this question might relate to your answers to the previous questions. Comment on the project's leadership, stakeholder involvement, and front-end planning.

CASE 20.1 MOZAL PROJECT: INTERNATIONAL INVESTMENT IN AN UNDERDEVELOPED COUNTRY⁴⁶

Mozal is a \$1.4 billion megaproject to construct a 250,000 tons per annum (tpa) state-of-the-art aluminum smelter in Mozambique (Figure 20.2). The idea of such a project at first seemed preposterous since Mozambique was one of the world's poorest nations with an infrastructure in ruins after two decades of civil war, and the project would require international financing and stable supplies of raw materials and labor. Yet the project was a success, completed months ahead of schedule and well under budget. It is worthwhile to see how that happened.

Mozal's primary promoting and controlling shareholder was Gencor, a large South African (SA) mining firm (later a part of the BHP company) that had recently completed the world's largest (500,000 tpa) smelter, called Hillside, in Republic of South Africa (RSA). In 1995, Gencor sent a multinational team of South African, Canadian, and French specialists from the Hillside project to search for a site for another smelter.

Mozambique

The team chose Mozambique for several reasons (Figure 20.3). Its capital, Maputo, offered a suitable (though run-down) harbor for importing alumina and exporting alu-



Figure 20.2

A typical aluminum smelter.

Source: Ruediger Fessel /iStock/Getty Images ®.

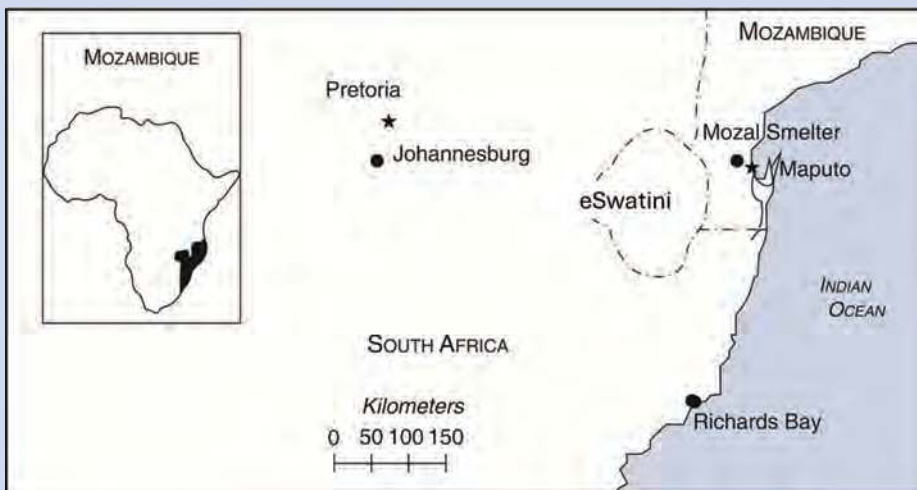


Figure 20.3

Mozal Smelter location and surroundings.

minum, plus abundant low cost (though largely unskilled) labor. Also, Eskom, the SA power utility, saw an opportunity to extend its power grid into Mozambique. The grid would provide the eSwatini region with reliable power and, later, be the conduit to supply hydropower from the Zambesi River in Mozambique to the RSA.

Mozambique's government was receptive to Mozal since the project would provide impetus to its industrialization policy. Mozal would become the first enterprise to qualify as an enterprise in the Industrial Free Zone, giving its supporters tax and duty exemptions. In addition, since Mozambique is an Asian-Pacific-Caribbean country under the Lome Agreement, aluminum produced there would enter the European Union duty free. After a visit to the Hillside smelter, Mozambique's prime minister championed the project and facilitated the regulatory and bureaucratic changes necessary for it to proceed.

The site chosen for the smelter lay in an undeveloped area 17 km from the harbor. To clinch the project, Gencor agreed to finance all related infrastructure work, including developing the harbor facilities, against repayment over time through taxes and harbor revenue offsets. Key members of the Mozal team relocated to Mozambique, enabling them to build relationships with stakeholders throughout local government and the community.

Financing

Another sponsoring shareholder for the project was the Industrial Development Corporation (IDC), a development bank of the RSA government created to identify investment opportunities that promote economic stability. IDC agreed to provide low-cost financing, export credit, and guarantees to SA manufacturers and contractors. The International Finance Corporation (IFC), a member of the World Bank Group that promotes sustainable investment in developing countries, also agreed to finance the project after being convinced that it was commercially viable, environmentally sound, and benefitted the region. All major cash inputs and outputs were set in US dollars to minimize currency exposure.

Risk mitigation and go-ahead

Costs for produced aluminum were anticipated to be in the bottom 5 percent of industry capacity, and its commercial case surpassed Gencor's investment criteria. The only major risk in the project was Mozambique. In May 1997, the governments of Mozambique and the RSA signed an agreement pledging to honor and protect cross-border investments. After private discussions with influential interest groups in Mozambique, the IDC and IFC decided to seek an influential international shareholder to share in the risk. In 1997 Mitsubishi Corporation, the Japanese conglomerate, signed on, and in May 1998 the project was given permission.

Construction

Construction at the Mozambique site posed major challenges. The locals speak Portuguese, but the expatriate managers, supervisors, and the computer software use English. Some basic engineering work was done in Canada and France; some specialized equipment was designed and manufactured in Japan and France. Most of the planning, coordination, detailed design, and preparation of material took place in the RSA.

Road and rail links connected Mozal to Richards Bay, RSA, the port where material and equipment arrived from overseas for transport to the project site. At one stage of the project, it became clear that Mozambique agents were having trouble processing the 60 to

80 trucks of equipment and materials that crossed the border daily. But the project director had built good relationships with key stakeholders, including Mozambique's president, and convinced them to allow the Mozal team to assist in managing the border post.

The project employed many experienced workers from the Hillside project, although thousands more unskilled workers had to be hired. Schools were set up to train them in construction and increase awareness of safety and the risks of HIV infection. To combat malaria, the area surrounding the site was continually sprayed, and full-time, on-site clinics set up that would eventually handle over 6,000 cases. For residents displaced by the project, new farming land was allocated and cultivated, and a development trust established to provide for local schooling and other community needs. Before contractors could access the site and road corridors, land mines from the civil war had to be cleared. Construction of cross-country power supply lines and the smelter were threatened by major cyclonic floods. Heavy lift helicopters were needed to fly in large pylons prefabricated offsite and to string power cables.

One goal of the project was to maximize local content. An estimated \$75M was spent in the local economy. At peak construction, 70 percent of the 9,000 people employed at the site were Mozambicans.

QUESTIONS

1. Summarize the issues and factors that posed risks to the Mozal project. Which of these arose from the international nature of the project?
2. What actions led to successful completion of the project despite the risks?
3. The team began searching for a suitable site in 1995 but the project was not launched until 1998. Discuss the kind of work required during the pre-project phase of a high-risk international project such as Mozal and the importance of that work.
4. Discuss the social responsibilities relating to projects in developing countries such as Mozambique.
5. Discuss (a) the most important megaproject drivers in this case and (b) the roles you think they played in the project's success.
6. Megaprojects typically have numerous stakeholders. (a) List the stakeholders mentioned in the case and (b) discuss how engaging and managing them played a role in the success of this project.
7. (a) What role do you think the existing Hillside smelter played in the success of this project? Hint: consider the NTCP diamond model discussed in the Introduction. (b) What challenges did Mozal experience that differed from the Hillside project?

STUDY ASSIGNMENTS

1. You are the newly appointed director for the proposed Mozal project. The feasibility study is completed, and you must convince the international sponsors and lenders to commit to the project. Prepare a presentation to a special board of stakeholders that asks for their go-ahead to commit \$1.4 billion to the project and addresses their expectations and how you will deal with the perceived risks.
2. The project has received permission, and you now face the reality of mobilizing your team to start work in a foreign country. What special project challenges can you expect and how will you go about laying the foundations for success?
3. What do you see as the criteria for evaluating the success of this international project?

CASE 20.2 SPIRIT ELECTRONICS' PUERTO RICO OFFICE⁴⁷

Spirit Electronics Company, a US firm, is building an office branch in Puerto Rico. Susan Marcie of the construction management firm Weller & Waxhall is managing the project, her first non-US project. She visited the project site and met with the person who would be the local project representative. In preparing the budget she sought bids from vendors in the US and Puerto Rico. Bids from US firms seemed extremely high; this plus the fact that Puerto Rican labor laws require some jobs be performed by locals led Susan to select mostly Puerto Rican vendors.

Spirit wanted the project completed within 30 weeks. Since cost bids from some vendors were slow to arrive, Susan prepared a budget using her firm's cost estimating spreadsheet and standardized costs. Spirit's budget review process takes four weeks and, she thought, the quicker the budget is approved, the sooner the project can begin. The project budget for \$690,457 was approved.

As project planning progressed, issues arose because the project was in Puerto Rico:

- Permits are required from both city and state (the US requires only city permits).
- Labor insurance is required at 5 percent of construction cost (not required in the US).
- Unusually high city taxes for construction work.
- High furniture cost (much higher than in US).
- High security cost due to risk of theft (higher than in US).
- Work shut down due to state holiday (December 22 through January 15).

These plus other smaller issues raised the estimated cost to \$1,250,998. Spirit considered cancelling the project, but Susan was able to negotiate with vendors and reduce the cost to \$987,655, to which Spirit agreed.

Susan knew that in overseas projects extra time should be included in the schedule to account for unknowns. She proposed delaying the target completion by 8 weeks, but Spirit objected. She was able to create a schedule to meet the original target date by paying the government an additional \$20,000 to rush the permits.

As the project progressed Susan had to respond to several issues:

- Long lead times for custom-made fixtures (6–8 weeks). Susan asked contractors to order the needed fixtures as early as possible.
- Millwork for cabinets and shelving, which usually must be done on-site after walls are completed and exact dimensions known. To avoid this, the building was redesigned so millwork could be premade offsite.
- Long lead times on permits (3–16 weeks). She submitted drawings and permit applications far in advance to account for when the permits were needed.
- Disorganized furniture installation vendors. Susan required the vendors create a plan and schedule, then held them to it.
- Local labor pool dichotomy: extremely high-cost (five times more expensive than in US but very reliable) or extremely low-cost (but of uncertain ability to do quality, on-time work). Typically, Susan hired the first.

- Added cost and time for imported materials due to import tax, shipping costs, and six weeks for government inspections. To avoid delays, Susan arranged for local storage space and shipping of materials far in advance of need.
- Language differences between US and local team members. For tasks that required coordination between these members, Susan extended the duration times.

QUESTIONS

1. In managing the project, how did Susan explicitly address the fact that it was an “overseas” project?
2. How might she have pre-identified issues that ultimately required her to redo the budget? How might she have anticipated other issues that emerged later?

Notes

1. Adapted from Orr R. *Strategies to succeed in foreign environments*. Collaboratory for Research on Global Projects, Stanford University. CIB W92 International Symposium *Construction Procurement—The Impact of Cultural Differences and Systems on Construction Performance*, (Las Vegas, Feb 8–10; 2005: 3). http://crgp.stanford.edu/publications/conference_papers/RyanVegas.pdf. Downloaded April 10, 2007.
2. These descriptions are simplified and each dimension has much broader ramifications than implied here. See Hofstede G., Hofstede G.J., and Minkov M. *Cultures and Organizations: Software of the Mind*, 3rd edn. New York, NY: McGraw-Hill Education; 2010.
3. McSweeney B. Hofstede’s model of national cultural differences and their consequences: a triumph of faith—a failure of analysis. *Human Relations* [55]; 2002: 89.
4. Carmel E. *Global Software Teams: Collaborating Across Borders and Time Zones*. Upper Saddle River, NJ: Prentice Hall; 1998.
5. Turner J. *Handbook of Project-Based Management*. New York, NY: McGraw-Hill; 1998, p. 483.
6. Murphy O. *International Project Management*. Mason, OH: Thompson; 2005, p. 23.
7. Pringle D. Finding ways around rigid labour laws. *Career Journal Europe*. http://www.expatica.com/actual/article.asp?subchannel_id=157&story_id=10562. Accessed May 8, 2006.
8. International Chamber of Commerce *Policy and Business Practices, Incoterms 2000*. <http://www.iccwbo.org/policy/law/id315/index.html>. Accessed April 11, 2007.
9. Lemley J. Managing the Channel Tunnel—lessons learned, *Tunneling and Underground Space Technology* 10(2); 1995: 9–11. http://www.ita-aites.org/applications/30th/PDF/TUST_95_v10_n1_5-29.pdf. Downloaded December 2, 2007; Anbari F. (ed.) *Case Studies in Project Management: The Chunnel Project*. Newton Square, PA: Project Management Institute; 2005.
10. Murphy, *International Project Management*, p. 63.
11. Tools for Hedging Foreign Exchange Risk. *Study.com*. <https://study.com/academy/lesson/tools-for-hedging-foreign-exchange-risk.html>. Accessed October 28, 2019.
12. See for example Sadikot R. Islamic project finance: Shari’a compliant financing of large scale infrastructure projects. *Al Nakhlah, Online Journal on Southwest Asia and Islamic Civilization* Spring 2012. https://ciaotest.cc.columbia.edu/journals/aln/aln1007/f_0025194_20573.pdf
13. Murphy, *International Project Management*, p. 63.
14. Lientz B. and Rea K. *International Project Management*. Amsterdam: Academic Press; 2003, p. 49.
15. Duarte D. and Snyder N. *Mastering Virtual Teams*. San Francisco, CA: Jossey-Bass; 2006, p. 134.
16. Murphy, *International Project Management*, pp. 65–66.
17. Lientz and Rea, *International Project Management*, pp. 44–45.
18. Lientz and Rea, *International Project Management*, p. 56.
19. Similar approaches discussed Lientz and Rea, *International Project Management*, Chapter 2.
20. Lientz and Rea, *International Project Management*, Chapter 2.

21. These and other considerations discussed in Murphy, *International Project Management*, pp. 72–73, 88, 91–94, 178.
22. Lientz and Rea, *International Project Management*, pp. 124–128.
23. Lientz and Rea, *International Project Management*, pp. 155–170.
24. Cohen M. *Succeeding with Agile*. Upper Saddle River, NJ: Addison-Wesley; 2010, p. 375.
25. Cohen, *Succeeding with Agile*, p. 377.
26. Duarte and Snyder, *Mastering Virtual Teams*, p. 68.
27. Orr, *Strategies to Succeed in Foreign Environments*, pp. 6–7.
28. Primary source for this section is G. Locatelli, E. Konstantinou, J. Gerald, T. Sainati. The dark side of projects: dimensionality, research methods, and agenda. *Project Management Journal* 53(4); 2022: 367–381. <https://doi.org/10.1177/87569728221103911>. Accessed Feb. 14, 2024. Unless noted otherwise, all cited statistics and their sources come from this article. See also N. Gulali. The World Bank in Africa: An analysis of World Bank aid and programmes in Africa and their impact; *Global Reports*, ROA 2018, <https://realityofaid.org/global-reports/>
29. The topic of megaproject management is a book unto its own. Interested readers are referred to Merrow E. *Industrial Megaprojects*. Hoboken, NJ: John Wiley; 2011.
30. Most of these drivers are discussed in Flyvbjerg B. What you should know about megaprojects and why: an overview. *Project Management Journal* 45(2); 2014: 3. Flyvbjerg is a prolific researcher and writer on megaprojects and author of *The Oxford Handbook of Megaproject Management*. Oxford: Oxford University Press; 2019.
31. Flyvbjerg, What you should know about megaprojects and why, p. 4.
32. Emes M., Smith A., and Marianovic-Halburd L. Systems for construction: lessons for the construction industry from experience in spacecraft systems engineering. *Intelligent Buildings International* April; 2012. DOI 10.1080/17508975.2012.680428.
33. Davies A. and Mackenzie, I. Project complexity and systems integration: constructing the London 2012 Olympics and Paralympics Games. *International Journal of Project Management* 32(5); July 2014: 773–790. <https://doi.org/10.1016/j.ijproman.2013.10.004>. Downloaded July 30, 2024.
34. Davies and Mckenzie, Project complexity and systems integration, p.774.
35. Merrow, *Industrial Megaprojects*, pp. 184, 276.
36. Flyvbjerg, What you should know . . . , p. 14.
37. Flyvbjerg, p. 11.
38. Flyvbjerg, p. 11.
39. Aubry M. and Foricel S. Undertaking the project front-end. In Williams T., Samset K., and Holst Volden G. (eds.). *The Front-end of Large Public Projects: Paradoxes and Ways Ahead*, eBook London: Routledge; 2022. DOI <https://doi.org/10.4324/9781003257172>.
40. This example derived from Cameron K. and Lavine M. *Making the Impossible Possible*. San Francisco: Berrett-Koehler; 2006.
41. Cameron and Lavine, pp. 18–44.
42. Merrow, *Industrial Megaprojects*, pp. 130–131, 286–288.
43. Flyvbjerg, B. Make megaprojects more modular. *Harvard Business Review* Nov–Dec; 2021.
44. Flyvbjerg, What you should know about megaprojects and why.
45. Merrow, *Industrial Megaprojects*.
46. Personal communication, Rob A. Barbour, former chairman, Mozal.
47. Case adapted from Carta K., Cisek A., Cho L., Farr B., Hobbs M., and Kim C. *Sprint Powered Workplace*. Graduate School of Business, Loyola University: Chicago; Feb. 2007.

Appendix A

Request for Proposal from Midwest Parcel Distribution Company

Following is the request for proposal (RFP) for the proposed LOGON system, sent from Midwest Parcel Distribution Company (MPD) to contractors deemed qualified. Only partial entries are shown to minimize the length of the example. (Note: Reference to “Appendix” is for a hypothetical appendix attached to the RFP, not to any appendix in this book.)

Introduction

You have been selected by Midwest Parcel Distribution Company (MPD) as potentially capable of meeting our requirements for a new system. You are invited to submit a proposal to supply the hardware, software, and support services for the system described in this request.

Section 1 Background

MPD seeks to award a contract for the design, fabrication, installation, test, and checkout of a transport, storage, and database system for the automatic placement, storage, and retrieval (PSR) of standardized shipping containers. The system, called the Logistical Online system (LOGON), will be installed at MPD’s Chicago distribution facility . . . (additional information regarding the Chicago distribution facility, projected future needs, and purpose and objective of the LOGON system).

Section 2 Statement of Work

The contractor shall be responsible for furnishing expertise, labor, material, tools, supervision, and services for the complete design, development, installation, checkout, and related services for full turnkey capability for the LOGON system. All necessary testing of systems and subsystems designed and installed by the contractor, as well as of existing facilities to ensure compatibility with the new system and with local, state, and federal requirements, will be performed by the contractor.

The LOGON system must meet performance requirements, be compatible with existing structural and utility limitations of the facility, and be compliant with packaging and logistical

standards and codes as specified in Section 6: Technical Information . . . (additional information regarding the services, equipment, and material to be provided by the contractor, and a list of specific end-items).

Exclusions

Removal of existing PSR equipment will be performed under separate contract and is the responsibility of MPD. Removal will be completed in time for the new system to be installed . . . (information regarding services, equipment, and material provided by MPD or other contractors and for which the contractor is NOT responsible).

Scheduled delivery date

The LOGON system is to be fully operational on or before May 2, 2025. All hardware, software, and support services necessary for full system operation will be supplied and/or completed by that date. Site installation will initiate no later than November 30, 2024.

Subcontractors

Included in the proposal, the contractor will list all subcontractors and the work to be assigned to each. Subcontractors will be subject to MPD approval prior to placement of the contract.

Cost and contract

Price of contract will not exceed \$15 million. Contract will be fixed price with a penalty charge of \$10,000 per day for failure to meet the operational completion date of May 2, 2025.

Section 3 Proposal Content and Format

Proposal will include the following sections and conform to instructions as follows.

Proposal table of contents

1. Cover sheet (use Form I provided in Appendix)
2. Executive summary
3. Statement of work
 - a. Background statement of need
 - b. Technical approach and distinguishing features
 - c. Project plan and schedule (use Forms II through V provided in Appendix)
4. Budget and price (use Form VI provided in Appendix)
5. Project organization and management plan
6. Prior experience and key personnel
7. Attachments
 - a. Signed statement of confidentiality (use Form VII in Appendix)
 - b. MPD supplied confidential information
 - c. Letters of commitment for work contracted to third parties.

Specific instructions

(Details about the purpose, specific content, specific format, and approximate length for each of the sections listed above.)

Section 4 Proposal Submittal

Submittal

Contractor will submit two (2) digital copies of the completed proposal along with all MPD confidential information to:

Lynn Joffrey
Administrative Assistant
Midwest Parcel Distribution Company
ljoffrey@mpd.com
(773) 773-7733

Deadline

Proposal must be received by MPD by 5:00 pm August 15, 2023.

Section 5 Selection Date and Criteria

Selection and Award Date

September 5, 2023.

Selection Criteria

Completed proposals received by the deadline will be evaluated by the following criteria:

1. Technical capability:
 - a. Capability of system to meet performance requirements within limitations of existing facility, standards, and codes. (15%)
 - b. User friendliness of system with respect to operation, reliability, and maintenance. (5%)
 - c. Use of state-of-the-art technology to ensure system remains current into the next decade. (15%)
 - d. System support services during contract period and available afterward. (5%)
2. Contractor's bid price. (25%)
3. Contractor experience and qualifications. (25%)
4. Project organization and management plan. (10%)

Section 6 Technical Information

Confidentiality

The attached technical data and any additional drawings, specifications, requirements, and addenda requested by contractors shall be treated as confidential and the property of MPD. Information provided in this RFP or requested from MPD will not be duplicated beyond that necessary to prepare the proposal. The original and all duplicates will be returned with the proposal. (See Form VII, Appendix.)

(Attached to the RFP in the Appendix are forms, agreements, and supporting technical data, standards, and performance requirements necessary for preparing and submitting a proposal.)

Supporting technical data

1. The following technical data is attached in the Appendix to this RFP:
 - a. Technical performance requirements and standards for LOGON system
 - b. Facility structural and utility specifications
 - c. Facility floor plan
2. For clarification and additional information, contact:

Mr. Ed Demerest
Project Director, Facilities
Midwest Parcel Distribution Company
13257 N. Wavelength Avenue
Chicago, IL 60699, USA
(773) 773-7733
edemerest@mpd.com

Appendix B

Proposal for Logistical Online System Project (LOGON)

*Submitted by Iron Butterfly Corporation to
Midwest Parcel Distribution Company*

1 Cover Sheet

Form I: Cover sheet

1. **Project Name:** Logistical Online System Project (LOGON) for the Midwest Parcel Distribution Company, Chicago distribution center
2. **Ref. Job No. 904-01**
3. **Contractor:** Iron Butterfly Corporation, Goose Rocks, Maine
4. **Name and Address of Contact:** Frank Wesley, Project Manager, Iron Butterfly Corporation, Robotics Applications Division, 150 Seaview Lane, Goose Rocks, Maine 715-332-9132, fwesley@ibuttc.com
5. Proposal Contents Check-off
 - a. Cover Sheet
 - b. Executive Summary X
 - c. Statement of Work
 - i. Background Statement of Need X
 - ii. Technical Approach and Distinguishing Features X
 - iii. Project Plan and Schedule (Forms from RFP: II. Work packages; III. Deliverables; IV. Work schedule; V. Subcontractors) X
 - d. Budget and Price (Project Price: \$14,413,905)
 - i. Budget and Price (Form VI from RFP) X
 - ii. Variations, Changes, Contingencies X
 - iii. Billing and Payments X
 - e. Project Organization and Management Plan X

- f. Qualifications and Key Personnel
 - i. Company and Prior Projects X
 - ii. Resumes of Project Manager and Project Engineer X
- g. Attachments (provide as specified in the RFP or as necessary to substantiate assertions in the proposal) X

2 Executive Summary

Iron Butterfly Corporation of Goose Rocks, MN, is submitting this proposal for the design and installation of the LOGON system at Midwest Parcel Distribution Company's Chicago distribution center. Our proposed system integrates robotic and neural network technology to streamline parcel transport and storage, and will complement MPD's existing distribution information processing system.

The proposed system utilizes robotic drone transporters to place and retrieve stored parcels. The system will utilize neural network technology, and, thus, will actually learn where to place and retrieve parcels and gain efficiency over time.

The significant benefits of the proposed system are:

- It can readily accommodate the expected 20% increase in volume anticipated by MPD.
- It can be operated for about 10% less than the annual operating cost of the current system.
- It can be readily implemented in the existing facility with no structural changes to the building and only minor changes to the electrical utilities.
- It is easily expandable in case the current facility is extended into the adjacent vacant lot.
- It can be designed, installed, and made fully operational within 1 year of contract. Conversion can be done in three 2-month phases, each on only one-third of the facility. Hence, throughout the 6-month conversion the current facility will be able to operate at more than 60% capacity.
- The system hardware and software is durable and easy to maintain as demonstrated by many 1,000s of hours operational usage of current systems by Iron Butterfly Company (IBC) customers.

IBC has 40 years of experience in the design and implementation of large warehousing transporter and storage systems. We have chosen highly experienced professionals as the project manager and the project engineer to oversee the LOGON project management and technology. They will work closely with MPD to assure that the installed system satisfies the MPD needs identified in the RFP and feasibility study and as emerge during the project.

Creative Robotics Company of Newton, MA, is IBC's partner in this project. They will modify the robotic transporter drones for this project. CRC is the industry leader in robotic drone technology and has developed robots for NASA as well as the robotic drone transporters for all of IBC's installed robotic drone transporter systems.

Our price for the system is \$14,413,905; we will hold this price fixed for the next 120 days. Iron Butterfly and Creative Robotics are fully committed to this project and guarantee its benefits. We invite you to contact us for more information and a formal presentation at your convenience.

3 Statement of Work

A. Background statement of need

MPD Company seeks a parcel storage, transport, and tracking system to replace the current system at its main distribution facility system in Chicago. The existing system is operating at

capacity; the new system must be able to accommodate an expected 20 percent increase in parcel shipments over the next seven years. Further, we recognize that the existing system utilizes a process that has become antiquated. MPD's objectives for the new system are to accommodate the expected growth, substantially improve the speed of parcel handling, increase utilization of existing storage facility space, enhance record keeping, and reduce the costs of labor, insurance, and shrinkage. The new system, to be called LOGON, will fully automate the process for placement, storage, and retrieval (PSR) of standardized shipping containers. MPD seeks a contractor to design, fabricate, and install the system, which will include all hardware and software for transport and storage of parcels, and the associated processing and storage of information for inventory and parcel tracking and control. This will be achieved with the deliverables listed in Form III, described in Section B.

We also recognize that removal of existing PSR equipment will be performed under a separate contract, and that during system installation MPD will arrange for alternate storage at other sites.

B. Technical approach and distinguishing features

Based upon analysis of information provided to us by Mr. Ed Demerest about MPD's Tulsa facility, which is considered a model facility, and data included in the RFP package, we conclude that the best approach for meeting MPD's needs and objectives is a system that uses robotic drone transporters, racks with standard-size shipping containers and storage buckets, and a computer database for automatic placement and retrieval of parcels and record keeping. The new system will be derived from a combination of advances in robotic and drone technology, artificial intelligence, as well as application of existing and emerging technology. Our company has 40 years' experience in design and installation of parcel handling and associated information systems, including eight installed robotic drone systems for companies in North America and Europe. This experience is explained in Section 6, Qualifications and Key Personnel. While using advanced technology, the proposed system will incorporate features of MPD existing systems to avoid duplication of effort and provide a fully operational system in less than 12 months from start.

The proposed systems work like this:

Upon a parcel's arrival at the distribution center receiving dock, it is placed into one of three standard-sized parcel "buckets." The buckets are electronically coded as to item and shipping destination. This code is relayed to a master database from any of four terminal workstations located at the dock. The workstations are connected via a DEM-LAN network to a CRC Model 4000 server. The Model 4000 has 10 terabyte storage plus backup for retaining information about parcel description, status, location, and destination. The system tracks available, remaining storage space, and, if needed, reallocates buckets for optimal space utilization. Allocation for space utilization relies on neural network technology, which enables the system to "learn" and improve its reallocations over time. The CRC 4000 will also provide reports about system status and performance as requested by management.

Parcel buckets are attached to a robot drone transporter that carries the bucket to a "suitable" vacant storage slot within a shipping container located on a rack. The computer determines which container has a vacant slot of sufficient size and contains parcels destined for the same or nearby destination as parcels in the transporter's bucket. The robotic drone transporter then conveys the bucket to the appropriate shipping container and drops it into the vacant slot. Each slot in the storage rack has a roller tray that automatically extends to accept/ yield incoming/outgoing buckets from arriving drones, and then retracts. Shipping containers

are stacked in seven rows of racks, three high (Items 2 and 3). The storage capacity of the facility is 400 shipping containers, each with 150 cubic feet storage capacity.

When a truck headed to a specific destination is to be loaded, the destination is entered at the dock terminal workstation, and the database system identifies all containers with buckets and parcels going to the same or nearby destinations. The system routes the robotic drone transporters to the appropriate containers for retrieval of the buckets. The trays with the outgoing parcels are extended for parcel retrieval by the drones. The system uses six robot drone transporters that operate independently and simultaneously. The drone transporters deliver buckets from incoming trucks, and retrieve and transport them to the loading dock for placement into departing trucks. The longest specified delivery/retrieval time is 6 minutes. A seventh drone transporter will be included as backup.

(More detail is provided about features of the robotic drone system and neural network software, including the benefits and advantages over alternative designs.)

C. Project plan and schedule (Forms ii to V from RFP)

Form II: Work packages

1. Perform functional design of overall system.
2. Prepare detailed design specifications for subcontractors of robotic transporter, storage rack systems, and shipping and parcel containers.
3. Prepare specifications for the software for DEM-LAN and CRC 4000 system interface.
4. Prepare detailed modification drawings for robotic drone transporter units and storage rack system.
5. Prepare plan for system installation and test at the site.
6. Fabricate robotic drone transporter units and rack support subassemblies at IBC facility.
7. Perform preliminary functionality tests on seven robotic drone transporter units.
8. Perform structural and functional tests of storage rack systems.
9. Perform installation of all subsystems at MPD Chicago facility site.
10. Perform checkout of subsystems and final checkout of overall system at MPD facility site.
11. Codes and Standards. (List of requirements and standards for local, state, and federal agencies, and measures for compliance.)

Form III: Deliverables

Hardware Group A

7 storage racks, 109 3 159 3 69, installed at site

Final structural, functional checkout of racks

400 shipping containers installed at site

1,000 size D43A parcel buckets

600 size D25B parcel buckets

600 size D12C parcel buckets

Final structural, functional checkout

Hardware Group B

7 robotic transporter units, each 20 pounds maximum load capacity compatible with three-size parcel buckets, 6 minutes retrieval at farthest point, installed at site
Four unit functional checkout
Integration checkout, Groups A and B

Software Group

DEM-LAN network, four CRC 2950 workstation terminals and CRC 4000 server, operating system software and AI algorithm (CRC)
Vista-Robotic software (Creative Robotics)
Triad warehousing system; Mobius transaction processing (CRC)

Support

Two copies, system operation/maintenance manuals
Robotic drone transporter/CRC 4000 integration
User training to competency
Final system checkout, user

Form IV: Work Schedule Target Dates

1.	Commence basic design	May 2024
2.	Basic design review	July 2024
3.	Process/Track design approval	September 2024
4.	Computer system specs review	October 2024
5.	Hardware Groups A and B received	December 2024
6.	Begin installation at site	January 2025
7.	Finish installation of complete system 1/2	March 2025
8.	Final user approval	May 2025

Form V: Subcontractors

1. Creative Robotics, Inc., Newton, MA, will supply the seven robotic drone transporters and necessary software.
2. Steel Enterprises, Inc., West Arroyo, OH, will supply and install the parts for the storage racks.
3. United Plastics Co., Provo, UT, will supply the shipping containers and parcel buckets.
4. CompuResearch Corp. (CRC), Toronto, Ont., will supply terminal workstations, DEM-LAN network, and CRC 4000 computer neural network software, and installation of software and related hardware.

4 Budget and Price (Project Price: \$14,413,905)

A budget and price (form VI from RFP)

Task	Labor Cost (\$)	O/H @0.25 (\$)	Material Cost (\$)	S/C (\$)	G/A @0.10 (\$)	Total (\$)
Project coordination	800,000	20,000	20,000		12,000	852,000
Project design and development	260,000	65,000	51,000		143,000	519,000
Basic hardware	684,000	171,000	54,100		90,910	1,000,010
Hardware design and drawings	1,165,200	291,300	143,400		160,000	1,759,900
Software specs	150,400	37,600	23,300	116,000	32,730	360,030
Parts purchase	10,320	2,490	600	1,477,500	149,100	1,640,010
Drawings	703,000	175,750	121,200	0	100,000	1,099,950
Software purchase	6,080	1,520	2,000	2,550,000	72,720	2,632,320
Assembly	562,800	140,700	151,000	0	85,450	939,950
Test	343,000	85,750	117,000	0	54,580	600,330
Final installation and test	997,600	249,400	133,500	165,000	154,550	1,700,050
Totals	5,682,400	1,240,510	817,100	4,308,500	1,055,040	13,103,550
Price		profit	10%	1,310,355		14,413,905

B. Variations, changes, contingencies

(List conditions under which costs will change, such as change in the scope of work, cost of steel-fabricated materials, work stoppages for labor disputes, etc.)

C. Billing and payments

(Proposes the method for billing and payment.)

5 Project Organization and Management Plan

Our company knows project management and has the experience, skills, procedures, and software to successfully perform this project. The project manager, Frank Wesley (contact at fwesley@ibc.com), will be responsible for managing project work, including all client contact work, reporting of progress, adherence to contractual commitments regarding schedule and technical performance, and monitoring of budgetary expenditures (see Section 6, Qualifications and Key Personnel). The project engineer, Julia Melissa, will be responsible for specification definition and ensuring the system meets technical requirements. She will supervise preparation of design requirements and drawings, and ensure fulfillment of system technical requirements at the site.

Ms. Melissa has worked at IBC for 7 years and on IBC's three most recent robotic drone projects. The fabrication manager, Ira Block, will be responsible for managing materials procurement and assembly and related work at the IBC plant, and coordinate assembly operations and give approval for assemblies prior to shipment to the MPD site. Mr. Block has worked with IBC for 9 years.

Within 1 month of contract signing the project manager will prepare a project execution plan for MPD to review. Thereafter he will present progress reports at monthly meetings with MPD staff. Written documentation will be provided in advance to MPD. The meetings will review expenditures to date, progress on work, and milestones and deliverables attained, all tracked by IBC's IRIS project management planning, tracking, and control system. Other formal meetings include a mid-project review meeting and a project summary meeting, plus others as requested by MPD or IBC.

(Additional sections address reporting and communication structure and risk mitigation.)

6 Qualifications and Key Personnel

A. Company and prior projects

Iron Butterfly Corporation has been designing and installing custom warehousing systems for 35 years. Among our customers are Nalco, Firebrand, Kraft, Abbott Laboratories, Cardinal Health, Swiss Guard, and Boeing. Our company has been ISO 9000 certified since 1996; we have also been certified as a Category A supplier for Grego Systems and a Class IIA supplier for Boeing's Commercial Aircraft Division. *(Author's note: this is a hypothetical example.)* In 2005 we received the Genie Design Award from IAWA. In 1998 we teamed with Creative Robotics to design the first fully-automated robotic warehousing system, and in 2011 we installed the first operational robotic drone transporter system at the 300,000 sq. ft. AIKEN distribution center in Hamilton, Ont. In 2014 we installed a similar system for Genteco Distributors at their 400,000 sq. ft. packaging center in Everett, WA. The robotic drone transporters are based on the standard industrial Model EZ, produced by Fancy Free Aerospace, Inc. The Model EZ has over three years and 350,000 hours of industrial service without major incident. The Model EZ design was modified for warehousing applications by CRC president and MIT professor Dr. Sanjeev Rayu. *(Include a few sentences about Creative Robotics' experience, projects, and achievements.)*

So far, we have installed a total of eight of these systems for satisfied customers.

(Additional paragraphs provide details of these systems: size and applications, cost of projects, names of customers, and information for contacting these customers.)

B. Resumes of project manager and project engineer

(Attach one-page resume each for project manager and project engineer showing experience on related projects and relevant background—degrees, memberships, certifications, contact information and past projects. Also include half-page resumes for one or two other key people in the project.)

7 Attachments

(This section provides attachments as specified in the RFP or as necessary to substantiate assertions in the proposal); e.g.:

- A. Signed statement of confidentiality (use Form VII in RFP)
- B. MPD supplied confidential information
- C. Technical data and analysis to support the proposed system
- D. Letters of commitment for work contracted to third parties.

Appendix C

Project Execution Plan for Logistical Online System

Contents

Cover Letter

- I. Management Summary
- II. Project Description
- III. Organization Section
 - III.1 Project administration
 - III.2 Project organization and responsibility
 - III.3 Subcontractor administration
 - III.4 Client interface
 - III.5 Manpower and training
 - III.6 User training
- IV. Technical Section
 - IV.1 Statement of work and scope
 - IV.2 Schedule and milestone targets
 - IV.3 Budget and cost
 - IV.4 Information requirements
 - IV.5 Documentation and maintenance
 - IV.6 Work review
 - IV.7 Applicable codes and standards
 - IV.8 Variations, changes, contingencies
 - IV.9 Contract deliverables

Memorandum



To: SEE DISTRIBUTION

Ref. Job No.: 904-01

From: Frank Wesley, Project Manager, fwesley@ibc.com

Date: January 3, 2024

Subject: Logistical Online System Project

Project Execution Plan

The Project Execution Plan for the Logistical Online System Project for the Midwest Parcel Distribution Company's Chicago distribution center has been modified to include your suggestions and approved by everyone in distribution. Copies of this document are herewith sent for use in the performance of contract requirements.

FW:es

Enclosure

Distribution:

Julia Melissa, Project Engineer
Sam Block, Fabrication Manager
Noah Errs, Quality Control Supervisor
Larry Fine, Software Manager
Sharry Hyman, Design Manager
Brian Jennings, Assembly Supervisor
Frank Nichol, Site Operations Manager
Emily Nichol, Assembly Supervisor
Robert Powers, Drawing Supervisor
Burton Vance, Purchasing Manager

I. Management Summary

On September 5, 2023, the Midwest Parcel Distribution (MPD) Company awarded the Iron Butterfly Company (IBC) the contract for the Logistical Online (LOGON) System to be installed at MPD's Chicago distribution facility.

The project consists of designing, fabricating, and installing a parcel transport, storage, and database system, for automatic placement, storage, and retrieval of standardized shipping containers. The system uses robotic drone transporter units, a computerized database, and an AI algorithm for automatic placement and retrieval of parcels and record keeping.

Iron Butterfly is the prime contractor and is responsible for the design of hardware and software, fabrication of component parts, system installation, and checkout. The major sub-contractors are Creative Robotics, Inc. (CRI), Steel Enterprises, Inc. (SEI), United Plastics Co. (UPC), and CompuResearch Corp. (CRC). Iron Butterfly will provide overall project management between CRI, SEI, and UPC Corp. and related contract administration. The project manager is Mr. Frank Wesley, and the project engineer is Ms. Julia Melissa.

The project will commence with basic design on or before May 17, 2024 and final system approval by MPD Co. will happen on or before May 2, 2025. The principle subtasks are shown in Item 7.

The price of the contract is \$14,520,000, fixed fee with limited escalation, based on a target final approval date of May 2, 2025. Total expenses, tabulated in Item 8, for labor, overhead, materials, subcontracting, and general/administrative are \$13,140,270. The agreement provides for an escalation clause tied to inflation indices for material expenses for the steel rack support system. A penalty of \$10,000 a day will be imposed on IBC for target completion overruns. Contingency arrangements in the agreement allow for reconsideration of the penalty in event of disruption of work for labor dispute with management.

II. Statement of Work

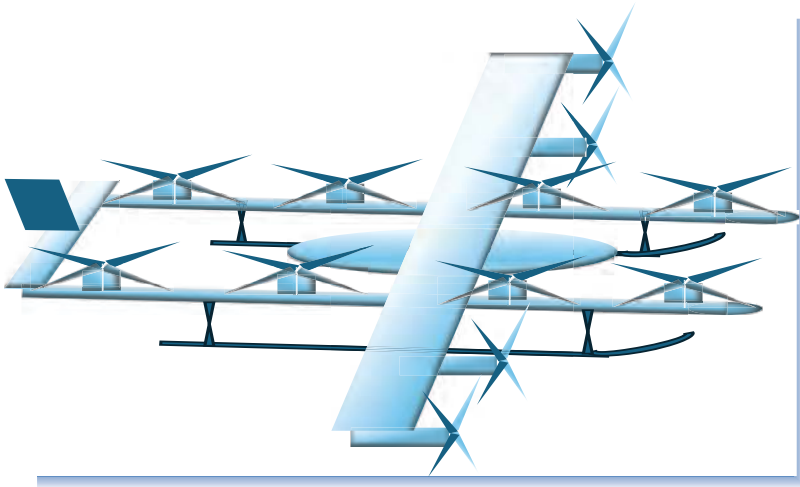
On September 5, 2023, IBC was awarded the contract for the LOGON System Project. The award followed a 2-month competitive bidding review by the MPD Company of New York. The system is to be installed at MPD Co.'s main Chicago distribution facility.

The project consists of designing, fabricating, and installing a parcel transport, storage, and database system (LOGON) for placement, storage, and retrieval of standardized shipping containers. The system will substantially improve the speed of parcel handling, increase the utilization of storage facility space, enhance record keeping, and reduce labor costs at the facility. Anticipated ancillary benefits include reduced insurance premium and shrinkage costs.

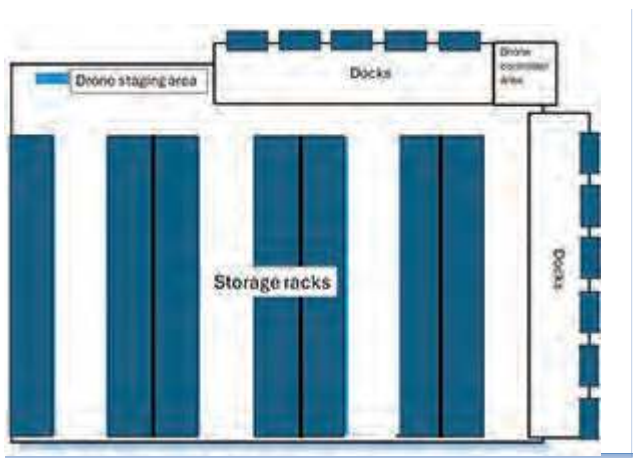
The system uses robotic drone transporter units, racks with standard size shipping containers and storage buckets, and a computerized database and AI for automatic placement and retrieval of parcels and record keeping. The system works as follows:

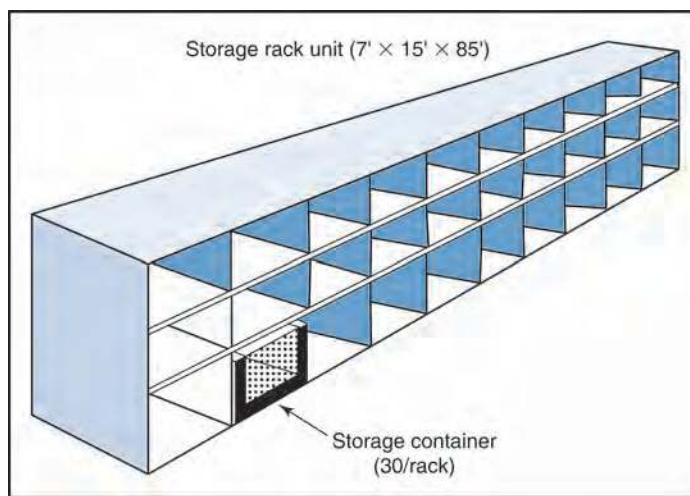
Upon a parcel's arrival at the distribution center receiving dock, it is placed into one of three standard-sized parcel "buckets" that are electronically coded as to parcel item and shipping destination. This code is relayed to a master database from any of four terminal workstations. The workstations are connected via a DEM-LAN network to a CRC Model 4000 server with 4 terabyte storage with backup to retain information about parcel description, status, storage location, and destination. The system keeps track of available storage space, and reallocates buckets for optimal space utilization; upon request it provides reports about system status and performance.

The parcel buckets are attached to a robotic drone transporter (Item 1). The drones are industrial Model EZ, produced by Fancy Free Aerospace, Inc., and modified by CRI for this application. Model EZ has been in use for over three years with 350,000 hours of industrial service and no accidents.

**Item 1****Robotic drone transporter**

The drone transporter carries the bucket to a “suitable” vacant storage slot within a shipping container located on a rack in the facility. The computer determines which shipping container has a vacant slot of sufficient size and contains parcels going to the same or nearby destination as parcels in the drone transporter’s parcel bucket. The transporter then conveys the bucket to the appropriate shipping container and drops it into the vacant slot. Shipping containers are stacked three high in seven rows of racks (Items 2 and 3). The facility holds 400 containers, each with 150 cu. ft. of storage capacity.

**Item 2****MPD site**



Item 3
Storage rack assembly

When a truck going to a specific destination is to be loaded, the destination is keyed in at the dock terminal workstation so the database system can identify all shipping containers with parcels going to the same or nearby destinations. The system then routes the robotic transporters to the appropriate shipping containers for retrieval of parcel buckets. The system has six robotic drone transporters that operate independently and simultaneously. The transporters retrieve the buckets and transport them back to the loading dock for placement of parcels into departing trucks. The longest retrieval time in the system is 6 minutes. The system will employ neural network technology that will enable it to improve its ability to place and retrieve containers. A seventh drone transporter is provided for backup.

IBC is the prime contractor and is responsible for the design of hardware and software, fabrication of components, system installation, and checkout. The major subcontractors are CRI, which will supply the major components for the robotic drone transporters; SEI, which will supply the storage rack system; UPC, which will supply the shipping containers and parcel buckets; and CRC, which will supply the terminal workstations, DEM-LAN network, neural network software, CRC 4000 computer, as well as software development support and installation of computer hardware.

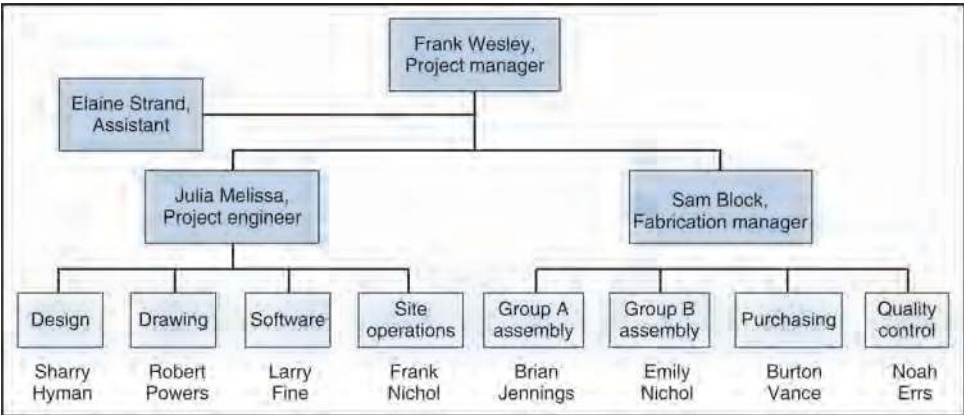
During system installation, MPD has arranged for alternate, temporary storage at another facility and rerouting of most parcel traffic to its other sites.

Design information about MPD's Tulsa facility will be utilized to initially move the project to an advanced stage. Remaining design work will use, as much as possible, work that has been done already, without compromising confidentiality of clients, on previous, similar projects.

III. Organization Section

III.1 Project administration

Correspondence on project matters will be between IBC's project manager and MPD's project director. Project personnel may correspond directly with the client or subcontractors for information, but must provide the project manager and project director with copies of the correspondence.



Item 4
LOGON project organization chart

The account number assigned to the LOGON project is 901-0000. Work packages and tasks will be assigned subaccount numbers at the time when work package instructions and schedules are authorized. A single invoice for the project accounts as a whole is acceptable for billing at monthly intervals.

III.2 Project organization and responsibility

The organization of IBC for the performance of the LOGON project is shown in Item 4. Administrative and managerial responsibilities are summarized in Item 5.

The project manager, Mr. Wesley, is responsible for all client contact, progress reporting, adherence to contractual commitments regarding schedule and technical performance, and monitoring of budgetary expenditures. He and his staff will report directly to Mr. Ed Demerest, vice president and project director for MPD Co.

The project engineer, Ms. Melissa, is responsible for establishing specifications and system delivery to meet technical requirements. She will supervise the preparation of design requirements and drawings, estimate quantities, check drawings and calculations, and ensure that system technical requirements are fulfilled at the site.

The fabrication manager, Mr. Block, is responsible for managing procurement, assembly, and related work at the IBC plant. He will ensure that delivered parts from subcontractors meet requirements, coordinate assembly of robotic drone transporters and storage rack subsystems, and sign off final approval for assemblies prior to shipment to the site.

III.3 Subcontractor administration

Key personnel at the four primary subcontractors CRI, SEI, UPC, and CRC are:

Bill Plante	Project coordinator, CRI
Terry Hemmart	Manager, manufacturing, SEI
Delbert Dillert	Customer representation, UPC
Lynn Duthbart	Systems engineering representative, CRC
Elmer Hyman	Customer representative, CRC

Persons responsible																													
Responsibility code	Project manager																												
	Project engineer																				Fabrication manager								
	Design								Drawing				Software				Site operations				Assembly A			Assembly B					
	FW	JM	SEH	RLQ	PJ	DVR	RIP	OEM	PVPR	DMN	RL	LSF	LLL	JRS	DVQ	FWN	JMMN	LOT	AUA	DAR	SOB	EN	GGF	RTT	BVL	BJ	TTY	HRD	BV-Purchasing
	Project task or Activity																												
	P	S																			S								
	A	P																											
	A	P					A					N			A						N								N
H Basic design	N	A	A	P	S	S	N					N			N														
I Hardware design A			A		P	S																							
J Hardware design B			A	P		S						N																	
K Drawings B							A	S	P												N								
L Software specs	N	A										A	P	S	S														
M Parts purchase B	N																				A								P
N Parts purchase A	N																				A								P
O Drawings A							A			S	P										N								
P Installation drawings							A	P		S					N														
Q Software purchase	N																				A								P
U Assembly A	N																				N	A	P	S	S				
V Assembly B	N																				N				A	P	S		
W Test A	N															N					A	P							
X Test B	N															N					A					P			
Y Final installation	N															A	P	S	S										
Z Final test	N															A				S									

Item 5
Project responsibilities

Changes to the respective agreements requested by a subcontractor or by IBC will be acted upon by the IBC project manager, Mr. Wesley, upon receipt of a written proposal from the subcontractor.

Correspondence to subcontractors concerning technical matters will be directed to the above listed first four parties or their representative. Software specifications-related work with CRC will be coordinated by the CRC customer representative. Project video conferences between IBC and subcontractors shall be noted in handwritten memos and copies sent to the IBC project engineer.

Formal progress reports shall be prepared by the CRI project coordinator, the SEI manufacturing manager, the UPC customer representative, and the CRC systems engineering representative for presentation at weekly meetings to be held at IBC’s Chicago office for the duration of scheduled involvement. Informal meetings will be scheduled as needed and may require attendance by other individuals as requested by the subcontractors or the project manager. The following minimum number of formal meetings are included in the respective subcontractor agreements.

CRI	5 meetings
SEI	3 meetings
UPC	2 meetings
CRC	5 meetings (software development)
CRC	8 meetings (site system integration)

Subcontractors will provide information and perform services as follows:

1. CRI will perform all work associated with procurement, manufacturing, and component functional tests of parts and subassemblies according to specifications, plans, and drawings provided by IBC. Parts and components for seven robotic drone transporters will be delivered to IBC per the criteria and dates specified in the agreement.
2. SEI will perform all work associated with procurement, manufacturing, and functional tests of parts and subassemblies per specifications, plans, and drawings provided by IBC. Parts and components for the seven storage racks will be delivered to IBC per criteria and dates specified in the agreement.
3. UPC will perform all work associated with procurement, manufacturing, and component functional tests of parts and subassemblies per specifications provided by IBC. Plastic containers and parcel buckets will be delivered to the MPD Chicago distribution facility in quantities and according to dates specified in the agreement. One plastic container and one each of three-size parcel buckets will be delivered to the IBC facility for tests per the agreement.
4. CRC will perform all work associated with development, programming, and tests of LOGON system robotic transporter control and neural networking software and system database per specifications provided by IBC. Software will be delivered to the IBC facility per the agreement.
5. CRC will transport, install, and perform component and integration tests for checkout of five terminal workstations, DEM-LAN network, CRC 4000 server, NN software, backup system, and peripheral hardware per criteria and dates specified in the agreement.

IBC will provide overall project management of CRI, SEI, and UPC and related contract administration, and legal, accounting, insurance, auditing, and counselling services as may be required.

III.4 Client interface

Key personnel associated with the project for MPD Company are:

Ed Demerest	Project director, Chicago	edemerest@mpd.com
Lynn Joffrey	Administrative assistant, Chicago	ljoffrey@mpd.com
Cecil Party	Financial manager, Chicago	cparty@mpd.com
Mary Marquart	Operations director, New York	mmarquart@mpd.com

Changes or modifications to the agreement requested by MPD or by IBC will be acted upon by the operations director upon receipt of a written proposal from IBC.

Correspondence with MPD will be directed to the project director. Project telephone conversations between IBC and outside parties shall be noted in handwritten memos and copies sent to Ms. Joffrey.

Progress reports shall be prepared by Mr. Wesley, IBC project manager, for presentation at monthly meetings to be held at MPD Co.'s Chicago office. Other meetings may require attendance by other individuals as required by MPD or requested by Mr. Wesley. Mr. Wesley shall also convene a mid-project review and a project summary at MPD's New York office. Fifteen meetings are included in the agreement. MPD will provide information and services as follows:

1. Perform all elements of work associated with vacating the site prior to the date in the agreement for commencing of system installation.
2. Provide surveys, design criteria, drawings, and preliminary plans prepared under previous agreements or received through requests for proposals for the LOGON system.
3. Provide design criteria, drawings, and plans prepared for the automated parcel storage and retrieval system at MPD Co.'s Tulsa facility.
4. Obtain all internal, municipal, state, and federal approvals as may be necessary to complete the project.
5. Provide overall project management between MPD, IBC, and CRC Corp., and legal, accounting, insurance, auditing, and consulting services as may be required by the project.

The contract administrator is the operations director. Changes or modifications to the agreement with MPD, requested either by MPD or IBC, shall be subject to a written proposal by IBC to MPD's contract administrator through IBC's project manager.

The financial manager is responsible for approvals of monthly expense summaries provided by INC and monthly payment to IBC. MPD is responsible for securing the necessary support from electrical and telephone utilities for system hook up, and for making available to IBC all criteria, drawings, and studies prepared for the Chicago site facility and the Tulsa facility automated system.

III.5 Manpower and training

No additional manpower requirements beyond current staffing levels are envisioned to perform services for this project. Five personnel from IBC's design group have been enrolled in and will have completed a seminar on robotics and AI/neural networks before the project begins.

III.6 User training

Two systems operations manuals and 16 hours of technical assistance will be provided. Thereafter, ongoing operator training will be the responsibility of MPD.

IV. Technical Section

IV.1 Statement of work and scope

The major tasks to be performed are the design, fabrication, installation, and checkout of the LOGON system for the Chicago distribution center of MPD Co. The work will be executed in accordance with the conditions set forth in the specifications in IBC’s proposal and confirmed in the agreement.

Subtasks required to perform the major tasks are shown in Item 5 (letters refer to simplified task designations used elsewhere such as Item 8):

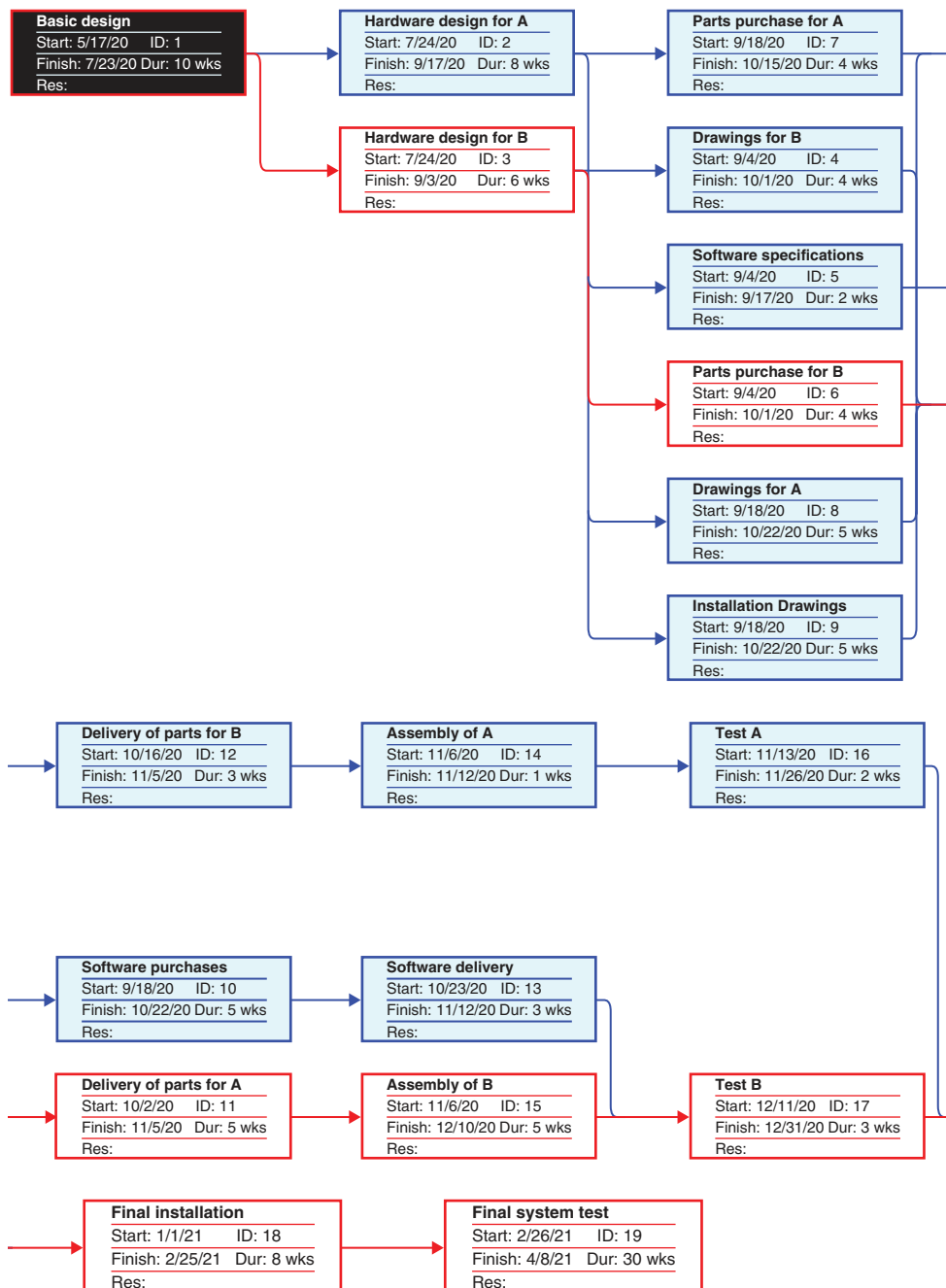
1. Perform basic design of overall system (H).
2. Prepare detailed design specifications for robotic drone transporter, storage rack systems, and shipping and parcel containers to be sent to CRC, SEI, and UPC (J, I, M, N).
3. Prepare specifications for the software and DEM-LAN and CRC 4000 system interface (L).
4. Prepare detailed assembly drawings for robotic drone transporter units and storage rack system (O, K).
5. Prepare drawings and a master plan for system installation and test (P).
6. Fabricate seven robotic drone transporter units and rack support subassemblies at IBC facility (U, V).
7. Perform functionality tests on all transporter units at IBC facility (X).
8. Perform structural and functional tests on rack systems at IBC facility (W).
9. Perform installation of all subsystems at MPD Chicago facility site (Y).
10. Perform subsystems checkout and overall system final checkout at MPD site (Z).

IV.2 Schedule and milestone targets

The project will commence with basic design on or before May 11, 2024; installation at the site will begin on or before January 10, 2025; and final system approval by MPD Co. will be on or before May 2, 2025. The schedule for significant aspects of the project is in Item 7. The indicated milestone targets are:

1. Commence basic design	May 11, 2024
2. Basic design review	July 26, 2024
3. Transporter and conveyor design review	September 6, 2024
4. Computer system specs review	September 20, 2024
5. Hardware group A and B review	November 29, 2024
6. Begin installation at site	January 10, 2025
7. Final user approval	May 2, 2025

Starting dates for activities that are dependent on the results of formal reviews will be adjusted to allow for significant changes in the length of predecessor activities, although no adjustments are anticipated.



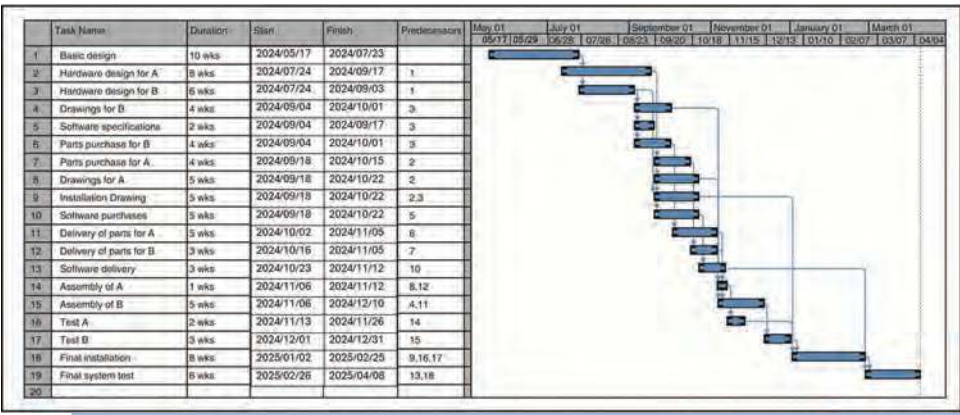
Item 6

Project network.

Note: Shown for illustration purposes. The network diagram is correct, but scheduled start and finish dates are not. Refer to Item 7 for scheduled dates.

Work package instructions and a detailed schedule for basic design have been distributed. Subsequent schedule and work package information will be distributed and discussed at review meetings.

The schedule of contract deliverables is given in Section IV.9.



Item 7

Project schedule

IV.3 Budget and Cost

The price of the contract is \$14,520,000, fixed fee with limited escalation, based on a target final approval date of May 2, 2025. Expenses and fees will be billed and are payable monthly as incurred. The agreement provides for an escalation clause tied to inflation indices for material expenses for the steel rack support system. A penalty of \$10,000 a day will be imposed on IBC for target completion date overruns. Contingency arrangements in the agreement allow for reconsideration of the penalty in event of disruption of work for labor disputes.

Principal tasks, subtasks, man-hours, and dollars to perform them have been estimated. Total expenses, as tabulated in Item 8, for labor, overhead, materials, subcontracting, and general/administrative are \$13,140,270.

Expenditures of direct labor are under immediate control of department heads in design, fabrication, procurement, and customer service departments because they assign personnel to the project.

The project manager is responsible for man-hour and direct expenses, and will receive biweekly reports of time and money expenditures.

IV.4 Information requirements

Most of the information required by IBC to perform under the terms of the agreement has been supplied by MPD Co. A limited amount of site information will be obtained from additional surveys performed by IBC. MPD will assist in survey work to expedite the project.

IV.5 Documentation and maintenance

Functional managers will send biweekly expense and progress reports to the project manager. The project manager will send monthly project summary reports to functional managers and to other managers and supervisors listed in distribution.

Cost, performance, and progress documentation will be maintained and reported through the company project cost accounting system.

The project manager will prepare a final summary report for IBC and MPD company archives.

The project manager is responsible for maintenance of all project files. All copies of project documents sent outside IBC will leave only under his direction.

IV.6 Work review

Internal review of work produced in each of the design, fabrication, procurement, and customer service divisions of MPD is the responsibility of the division head for each of the functional disciplines.

IV.7 Applicable codes and standards

Storage racks and supporting structures, electrical harnesses, and radio transmitters are to be designed to the applicable standards of AATOP, ASMER, OSHA, the Illinois Building Requirements Board, and the City of Chicago.

IV.8 Variations, changes, contingencies

The agreement with MPD defines the conditions for considering a change in compensation or penalties due to a change in the scope of work or cost of steel-fabricated materials, or unanticipated stoppage of work for labor dispute. It describes the procedure whereby authorization for such a change may be obtained from MPD.

The agreement, Paragraph 9.2, under prime compensation, states:

Whenever there is a major change in the scope, character, or complexity of the work, or if extra work is required, or if there is an increase in the expense to the CONTRACTOR for steel-fabricated materials as negotiated in the agreement with the responsible SUBCONTRACTORS, or if there is a stoppage of work resulting from a labor dispute with management, the CONTRACTOR shall, upon request of the CLIENT, submit a cost estimate of CONSULTANT services and expenses for the change, whether it shall involve an increase or a decrease in the Lump Sum. The CLIENT shall request such an estimate using the form provided herein (Attachment F). Changes for reasons of labor dispute with management will be reviewed and determined according to the conditions specified (Attachment G).

During system installation and tests, MPD has made arrangements to reroute 70 percent of its Chicago parcel business to other centers. The remainder will be stored at an alternate facility near Chicago. In the event of a schedule overrun, the reroute plan will remain in effect. MPD requires 30 days' notice of anticipated schedule overrun to extend the agreement with the alternate Chicago storage facility.

Item 8 LOGON project cost estimate

Task	Labor time (h)	Labor rate (\$)	Labor cost (\$)	O/H @ 0.25 (\$)	Materials (\$)	S/C (\$)	G/A @ 0.1 (\$)	Total (\$)
Project coordination	5,000	112	560,000	140,000	20,000		102,00	1,122,000
	5,000	48	240,000	60,000				
		Total	800,000	200,000				
Project development	1,000	112	112,000	28,000	45,000		28,500	313,500
	1,000	80	80,000	20,000				
		Total	192,000	48,000				
System design	125	112	14,000	3,500	6,000	1,550,000	164,100	1,805,100
	375	96	36,000	9,000				
	375	48	18,000	4,500				
		Total	68,000	17,000				
H Basic hardware	750	120	90,000	22,500	54,100		90,910	1,000,010
	4,000	96	384,000	96,000				
	3,500	60	210,000	52,500				
		Total	684,000	171,000				
I Hardware design A	450	104	46,800	11,700	24,500		58,175	639,925
	2,750	96	264,000	66,000				
	2,250	60	135,000	33,750				
		Total	445,800	111,450				
J Hardware design B	625	104	65,000	16,250	61,500		87,275	960,025
	3,375	96	324,000	81,000				
	3,250	80	260,000	65,000				
		Total	649,000	162,250				

(Continued)

Item 8
(CONTINUED)

Task	Labor time (h)	Labor rate (\$)	Labor cost (\$)	O/H @ 0.25 (\$)	Materials (\$)	S/C (\$)	G/A @ 0.1 (\$)	Total (\$)
K Drawings B	400	104	41,600	10,400	57,400		14,540	159,940
	400	72	28,800	7,200				
		Total	70,400	17,600				
L Software specs	400	112	44,800	11,200	23,300	116,000	32,730	360,030
	600	96	57,600	14,400				
	600	80	48,000	12,000				
		Total	150,400	37,600				
M Parts purchase B	5	112	560	140	250	758,000	76,375	840,125
	40	96	3,840	960				
		Total	4,400	1,100				
N Parts purchase A	10	112	1,120	280	350	719,500	72,725	799,975
	50	96	4,800	1,200				
		Total	5,920	1,480				
O Drawings A	1,625	104	169,000	42,250	85,800		45,455	500,005
	1,750	72	126,000	31,500				
		Total	295,000	73,750				
P Installation drawings	1,125	112	126,000	31,500	35,400		54,540	599,940
	1,500	104	156,000	39,000				
	1,750	72	126,000	31,500				
		Total	408,000	102,000				

Q Software purchase	20	112	2,240	560	1,600	717,500	72,670	799,370
	40	96	3,840	960				
		Total	6,080	1,520				
U Assembly A	25	112	2,800	700	64,000		12,750	140,250
	250	96	24,000	6,000				
	300	80	24,000	6,000				
		Total	50,800	12,700				
V Assembly B	250	112	28,000	7,000	87,000		72,700	799,700
	2,750	96	264,000	66,000				
	2,750	80	220,000	55,000				
		Total	512,000	128,000				
W Test A	50	104	5,200	1,300	47,000		21,850	240,350
	750	96	72,000	18,000				
	750	80	60,000	15,000				
		Total	137,200	34,300				
X Test B	75	104	7,800	1,950	70,000		32,725	359,975
	1,125	96	108,000	27,000				
	1,125	80	90,000	22,500				
		Total	205,800	51,450				
Y Final installation	800	112	89,600	22,400	121,000	105,000	94,550	1,040,050
	3,000	96	288,000	72,000				
	2,250	88	198,000	49,500				
		Total	575,600	143,900				
Z Final test	500	112	56,000	14,000	12,500	60,000	60,000	660,000
	2,500	96	240,000	60,000				
	1,500	84	126,000	31,500				
		Total	422,000	105,500				
Totals			5,682,400	1,420,600	816,700	4,026,000	1,194,570	13,140,270

IV.9 Contract deliverables

All items are to be assembled, installed, and in operation at the site in accordance with technical specifications in the agreement.

Subcontractors will transport components and parts to the IBC plant per this schedule:

Item	Date
Parts and components for robot transporters from CRI	November 1, 2024
Parts and components for storage rack systems from SEI	November 4, 2024
One shipping container and one each of three-size parcel buckets from UPC	November 10, 2024
Robotic drone transporter system control software from CRC	October 25, 2024

Following are the items identified in the agreement as deliverable to MPD:

Item	Date
Hardware (group A)	
Seven storage racks, 7' × 15' × 85' (D × H × L) Installed at site	November 15, 2024
Final structural, functional checkout	November 29, 2024
Delivered 400 shipping containers installed at site	December 6, 2024
Delivered 1,000 size D43A parcel buckets	December 13, 2024
Delivered 600 size D25B parcel buckets	December 13, 2024
Delivered 600 size D12C parcel buckets	December 13, 2024
Final structural, functional checkout	November 8, 2024

Hardware (group B)

Seven robot transporter units (each 80 pounds maximum load capacity compatible with three-size parcel buckets)

Installed at site November 8, 2024

Item	Date
Seven unit functional checkout	November 10, 2024
Integration checkout, groups A and B	January 3, 2025
Software group	
Submission of software specifications to CRC	September 19, 2024
(Installation of DEM-LAN network, four CRC 2950 workstation terminals, and CRC 4000 server, performed by CRC)	February 7, 2025
Software-integration checkout, performed by CRC	March 7, 2025
Final checkout	
Two copies, system operation/maintenance manuals	March 7, 2025
Robotic drone transporter/CRC 4000 integration	April 4, 2025
Benchmark systems test, with parcels	April 8, 2025
User training	April 11–12, 2025
Final system checkout, user	Latest, May 2, 2025

Index

Note: Information in figures and tables is indicated by page numbers in *italics* or **bold**.

- AAR. *see* after-action review (AAR)
- AAS. *see* Advanced Automation System (AAS)
- Accent, Inc., 529–530
- acceptance testing, 372
- accountant, project, 586
- accounting. *see* project cost accounting system (PCAS)
- accuracy, in estimates, 195
- achievement orientation, 703
- activity(ies): crashed, 274; critical activity, 233; decoupling of, 404; dummy, 258, 258; parallel activities, 230; sequential activities, 230; splitting of, 251–252, 252; summary activity, 234; time-cost relationship for, 273–275, 274; variability of duration of, 280–282, 281, 282; work, in project planning, 175; in work breakdown structure, 182
- activity contingency, 201
- activity-on-arrow (AOA), 228, 257–259, 258, 259
- activity-on-node (AON), 228, 229–231, **230**, 231, 258, 258, 259, 259
- actual cost (AC), 470
- actual cost of work performed (ACWP), 470
- ACWP. *see* actual cost of work performed (ACWP)
- adaptation measures, 87
- adaptive approach, 123
- adjustments, negotiated, in project termination, 163–164
- administrative expenses, 324–325
- administrator, contract, 163, 437, 468, 488, 586, 758
- Advanced Automation System (AAS), 54
- advertising, in procurement management, 433
- affinity diagrams, in risk identification, 388, 390
- after-action review (AAR), 644–645
- aggregation, principle of, 297
- agile approach, 70
- agile methodology, 123
- agile project management (APM), 173, 500, 502–505, 503, 504, 513–515
- agile projects, 111
- AI. *see* artificial intelligence (AI)
- airbags, 381–382
- alliances, networks of, 578
- allocation matrix, 136, 136
- “all systems go,” 53
- ambiguity, role, 623
- ambition, in cost escalation, 321–323
- analogy, project, 388
- analogy estimate, 196–197
- AOA. *see* activity-on-arrow (AOA)
- AON. *see* activity-on-node (AON)
- APM. *see* agile project management (APM)
- APM. *see* Association for Project Management
- APMP (APM Professional). 11, 538, 650
- Apollo spacecraft, 95, 95–96, 96, 103
- Archdiocese of Boston, 34–35
- architecture, of system, 135
- array complexity, 7
- artifacts, project, 460, 509–510
- artificial intelligence (AI), 214–218; applications of, 217; applied, 214; generative, 214, 216
- as-built, 361
- as late as possible, 293
- assembly complexity, 6
- as soon as possible, 293
- Association for Project Management (APM), 11–12, 16–17
- assumptions, risk, 386
- atoms projects, 513
- attributes, in systems, 45
- audit, quality, 365–366
- audit projects, 34
- authority: charismatic, 576; charter and, 113; in engineering, procurement, and construction, 440–441; gap, 577; influence and, 577; legal, 576; in matrix organizations, 548; multifunctional teams and, 544; project, 577; project-functional shared, 579; project management and, 576–579, 578; in projects,

- 577; reward power and, 576; subunit integration and, 541; traditional, 576
- availability requirements, 119
- backlog: in lean project management, 520; product, 505
- Bandra-Worli Sea Link (BWSL), 322
- bankable business case, 80
- banking, international projects and, 709
- Barrage Construction Company, 222, 223
- baselines, 194, 323, 480, 487
- batch production, 515–516
- Bay Area Medical Center, 647
- B/C. *see* benefit/cost ratio (B/C)
- BCWS. *see* budgeted cost of work scheduled (BCWS)
- benchmarking, 470
- benefit/cost ratio (B/C), 678
- benefits management, 654–655
- benefits realization management (BRM), 74, 76
- Beta probability distribution, 283
- bias: in cost escalation, 321–323; merge-point, 281, 286, 389
- bid(s): solicitation and evaluation, 432–433
- bidders list, 78
- bid documents, 432
- Big Bang, 160, 160, 598–599
- Big Dig, 376–378, 589–590, 590, 591
- big visible charts (BVCs), 484, 525
- billing, total, in budget and estimates, 326
- BIM. *see* building information modeling (BIM)
- bits projects, 513
- blitz pace, 7
- Boeing, 124–125, 212, 369, 370
- boilerplate contract, 441
- bonds, surety, 449
- bonus payments, 279
- Borglum, Gutzon, 573, 573–575
- bottom-up approach, in estimation, 200
- boundary, in systems, 45–46
- brainstorming, in risk identification, 390
- breach of contract, 437, 449
- breadboards, 139, 406
- breakthrough, 6
- Bridgecon Construction Company, 310–312, **311**
- brownfield project, 83, 158
- bubble chart, 686, 686
- budget: administrative expenses in, 324–325; baseline and, 323; control accounts in, 328–329; cost accounts in, 328–329; elements of, 323–326, **326**324, 327; in execution plan, 177; feasible budget region, 337; international projects and, 715–717; labor expenses in, 323; non-labor expenses in, 323–324; overhead expenses in, 324–325; profit in, 326; with project management information system, 211; in project planning, 176; reserve, 318–319, 465–466; time-phased, 328; total billing in, 326
- budgeted cost of work scheduled (BCWS), 470
- buffer: drum, 302; feeding, 295–297, **296**, 296, **299**; management, for project control, 301; milestone, 298, **299**; project, **299**; project time, 295–297, **296**, 296; resource, 298, **299**, 466; sizing of, 299; time, 202, 242, 272–273, 319, 462
- building information modeling (BIM), 219
- build stage, in project execution, 157–158
- burn charts, 482–484, 483
- burn-down chart, 482, 510, 510–511
- burn-up chart, 482
- business case, 74, 80; bankable, 80
- business skills, of project manager, 581
- buyers: in systems development cycle, 73
- buy in, 318, 621
- BVCs. *see* big visible charts (BVCs)
- BWSL. *see* Bandra-Worli Sea Link (BWSL)
- calendar schedule, 241
- Canadian Shield gold mine, 150–151
- Capability Maturity Model (CMM), 634
- capital: as input, 47; as resource, 51; working, 247, 248
- capital expenditure projects, 34
- cash flow, 340
- cause-and-effect (CE) diagram, 373–374, 374, 390, 390
- cause-effect relationships, in systems, 49
- CB. *see* configuration board (CB)
- CCB. *see* configuration control board (CCB)
- CCPM. *see* critical chain project management (CCPM)
- CE. *see* cause-and-effect (CE) diagram; concurrent engineering (CE); Consolidated Energy (CE)
- center, project, 545
- champion, project, 589
- change, 25–26
- change clause, 319–320
- change control, 485, 485–488, 487, 658; artificial intelligence in, 217; board, 488; cost escalation and, 320; plan, 177; process, 487, 487–488, 498; project management information system and, 211–212; requirements management and, 122; scope and, 179; system, 156–157

- change control system, 461
- change log, 488
- change management, 621–622; in procurement management, 437; stakeholder engagement and, 26; transition management and, 657
- change order, 362
- change proposal, 362
- change register, 488
- change request, 362, 559
- Channel Tunnel, 155, 163, 348, 707, 721
- characteristics, classification of, 366–367
- charismatic authority, 576
- chart: big visible, 484, 525; burn, 482–484, 483; control, 372; fever, 462; Gantt, 205–206, 206, 207, 208, 513–514; Pareto, 372–373, 373; RACI, 190; run, 372
- charter, project, 113–114, 154, 177, 227n1, 612
- checklist, 374, **388**, 388–389
- CI. *see* configuration item (CI)
- CIF. *see* composite impact factor (CIF)
- classification of characteristics. *see* characteristics, classification
- CLF. *see* composite likelihood factor (CLF)
- climate change, 87–88, **88**, 400–402, 683–684
- closed systems, 48
- close function, 23, 24
- closeout: meeting, 163; procurement, 431, 436–438, 450; project, 161 (*see also* termination, project)
- closeout phase, in program life cycle, 654
- CMM. *see* Capability Maturity Model (CMM)
- code identifier, unique, 183
- cold turkey, 160, 160
- collaboration, culture of, 561
- collaborative contracting, in procurement management, 435
- collaborative project management (CPM), 604
- collocation, 523
- commitment, estimate vs., 194–195
- commonality requirements, 119
- communication, project: client-contractor, 148–149; cost escalation and, 320; informal, 127, 454; plan, 125–127, 126; in project definition, 125–127, 126; project management office and, 649–650
- communication channels, in risk management, 409
- communication hub, manager as, 572
- communication plan, 177, 460, 592, 717–718
- community, project, 213
- compatibility requirements, 119
- compensation, negotiated, in project termination, 164
- competencies, project management, 10–11, 650
- competitive pace, 7
- complexity, of projects, 6–7
- composite impact factor (CIF), 396
- composite likelihood factor (CLF), **392**, 392–393
- conception phase, in systems development cycle, 68–69, 69, 74–76, 77, 106
- conceptual design, in systems engineering, 128–129
- concurrent engineering (CE): DevOps in, 512, 560, 560–563; life cycle costs and, 343; multifunctional teams and, 544; quality assurance and, 363; in systems engineering, 54; teams in, 124, 561–563
- conditions of satisfaction (COS), 508
- configuration board (CB), 362
- configuration control, 362
- configuration control board (CCB), 362
- configuration identification, 361
- configuration item (CI), 135, 137, 182, 361
- configuration management, 488; plan, 360; project management information system and, 212; quality assurance and, 360; traceability and, 122, 135
- conflict(s): consequences of, 618; groupthink and, 618; life cycle and, **617**, 617–618; within organization, 617; role, 623; in systems, 47; between user and contractor, 616
- conflict management: confrontation in, 620; defined, 619; methods, 619; role clarification technique with, 620–621
- conflicts of interest, 168
- conformance, cost of, 355
- Consolidated Energy (CE), 453
- constraints: elevating, 303–304; exploitation of, 303; feasibility and, 83; resource, 248, 248, 254, 254–255, 255; subordination to decisions regarding, 303; systems, 294; in systems, 47, 50; theory of, 294–301, 295–301, **296**, **299**; time, 248; utilization of, 303
- construction projects, 34; execution phase and, 153; fast tracking and, 155; scope definition and, 179; subcontracting and, 439–440; and methodology, 638–639
- consultants, peer, 645
- contingency, in estimates, 194, 201–202, *see also* buffer
- contingency fund, 318–319
- continuous improvement, 354

- contract(s): boilerplate, 441; breach of, 449; closing of, 163–164; cost escalation and, 321; cost-plus, 403, 442, 445–448; cost-reimbursement, 97, 445–448; customer recourse with, 449; EPC, 577; fixed price, 97, 403, 442–445, 448; fixed-price incentive fee, 443–444; force majeure in, 449; guaranteed maximum price, 447–448; incentive, 97; informal agreements and, 454; international projects and, 705, 706; liquidated damages with, 449; penalty clause in, 449; performance incentive, 447; -related matters, 448–450; release, 468; risk and, 448, 448–449; schedule incentive, 447; subcontract, 450; surety bonds with, 449; time and materials, 448; types of, 441–442
- contract administration: defined, 97, 437; in procurement management, 436–437; project termination and, 162–163; proposals and, 97
- contract administrator, 163, 437, 468, 488, 586, 758
- contract award process, in procurement management, 434–435
- contract choice, as risk transfer, 403
- contracted out, 99
- contracted projects, 181
- contracting: collaborative, in procurement management, 435; environment, 438–439; in procurement management, 438, 438–441, 440
- contractor(s): associate, 555; in conception phase, 68–69; contract administration and, 436, 437; contract choice and, 403; defined, 73; in feasibility, 77; integration, 554, 557; lead, 555; in needs definition, 81; oversight and integration, 555–557; prime, 431, 555; procured item quality and, 359; in project management broadly, 24; retention money and, 340; as stakeholder, 73; system, 73; in systems development cycle, 73
- contract statement of work (CSOW), 97–98, 98
- contractual agreement, in systems development cycle, 97
- contractual incentives, 279
- control: chart, 372; cost escalation and, 320; of design, 156–157; earned value management in, 470–481, 471, 473, **474**, 474, **475**, 477, 478, 481; emphasis, 460–466, 463; external, 457; internal, 457; issue management in, 484; problems with, 488–490, 489; procurement, 431, 436–438, 450; in project planning, 176; schedule, 462–463, 463; of scope, 461; start-stop, 467; visual management in, 482–484, 483; *see also* monitoring
- control accounts, 328–329, 467, 467–470, 469
- control chart, 372
- controller, project, 586
- controlling, in project planning, 176
- controlling function, 23, 24
- control process, 456–457
- convergence points, risk and, 389
- conversion, in project execution, 159–161, 160
- coordinator, project, 544
- corrective action, 457
- corruption, 168
- COS. *see* conditions of satisfaction (COS)
- cost(s): actual, 470; control, 462; direct, 278; of disposal, 341–342; estimated cost at completion, 476–479, 477, 478; in estimates, classification of, 195; expected project completion, 413–414, **414**; indirect, 278–279; international projects and, 716; life cycle, 341–343, 679; in life-cycle costing, 74, 76; lowest total project, 278–280, 279, 280; nonrecurring, 195; normal, 274; project, 341; project cost accounting system, 326–327; of quality, 355–356; recurring, 195; risk, 413; sunk, 692; target, 442; total project, 279; in work packages, 186; *see also* expenses
- cost accounts, in budgeting, 328–329
- cost analysis, 330–337, 331–333, **334–337**, 336
- cost apportionment, **326**
- cost-benefit analysis, 355
- cost-benefit grid, 689, 689–690
- cost control, project management information system in, 211–212
- cost curve, 4–6, 5
- cost dimension, of project goal, 9, 9
- cost-effectiveness analysis, 690–692, **691**, 691
- cost-effectiveness requirements, 119
- cost engineering, in estimation, 198–200
- cost escalation: ambition and, 321–323; biases in, 321–323; communication and, 320; contract and, 321; control and, 320; defined, 318; design changes and, 319–320; ego of estimator in, 321; exchange rates and, 320; inaccuracies in, 321; inefficiency and, 320; lack of information in, 318–319; megaprojects and, 323; requirements in, 319–320; uncertainty in, 318–319
- cost estimates, 317–318
- cost plus award fee contract (CPAF), 447
- cost-plus contract, 403, 442
- cost plus fixed fee contract (CPFF), 445–446, 448

- cost plus incentive fee contract (CPIF), 446–447
- cost plus percent of cost contract (CPPC), 446, 449
- cost-reimbursable contract, 442, 445–448
- cost risk, 405–407
- cost schedule, 335
- cost-sharing ratio (CSR), 443
- cost slope, 274
- cost summaries, 329–330, 330
- cost variance analysis, 461
- counter trade, 709
- CPAF. *see* cost plus award fee contract (CPAF)
- CPAone, 34
- CPFF. *see* cost plus fixed fee contract (CPFF)
- CPM. *see* collaborative project management (CPM); critical path method (CPM)
- CPPC. *see* cost plus percent of cost contract (CPPC)
- crashed activity, 274
- crashing of project, 278
- critical activity, 233
- critical chain, 193, 255, 272–273, 297–298, 298
- critical chain method, 234, 294–301, 295–301, **296, 299**
- critical chain project management (CCPM), 294–295; challenge of, 301; schedule control and, 462; software support for, 301
- critical characteristics, 366
- critical design review, 363, 459, 481
- critical path, 193, 233, 234, 235, 235, 255; near-, 286
- critical path method (CPM), 26; crashing of project in, 278; defined, 273; lowest total project cost in, 278–280, 279, 280; PERT and, 282–292, **289, 290, 291, 291**; reducing project duration with, 273–280, 274, **275, 275, 276, 277, 278–280**; resource allocation and, 290, **291, 291, 292–294, 294**; shortening of critical path in, **275, 275, 275–277, 276**; time-cost relationship and, 273–275, 274; variability of activity duration and, 280–282, 281, 282
- cross-functional, project as, 3
- CSOW. *see* contract statement of work (CSOW)
- CSR. *see* cost-sharing ratio (CSR)
- culture of collaboration, 561
- cumulative flow diagrams, 482
- currency risk, 708–709
- current system, 83
- customer differentiation, 540
- customer liaison, 586
- customers: in systems development cycle, 73
- Cybersonic, 495–496
- cycle time, 520–521
- cycle timing, 521
- daily spring meeting, 509
- daily status notifications, 466
- Dalian Company, 33
- damages, liquidated, 449
- dark side of projects, 167–168
- dashboard, 211, 214
- databases, in knowledge management, 643–644
- date: commitment to due, 295; due, 237, 240–241, 295; expected completion, 240–241; freeze, 157, 488; internal target ship, 466; revision of task completion, 476; target completion, 237, 240–241, 284–286, **285, 285**; visual freeze, 466
- DBT. *see* design-build team (DBT)
- decision maker, project manager as, 572
- decision management, 654
- decision trees, 415–416, 416
- Decline phase, in product life cycle, 72
- decoupling of activities, 404
- deep learning, 216
- defects, absence of, 353
- definition phase: in program life cycle, 653; in systems development cycle, 69, 69, 105–108, 106, 107 (*see also* project definition; system definition)
- deliverables: as feature of project management, 25; in work breakdown structure, 182
- Delphi technique: in estimation, 201; in risk identification, 391
- Denver Airport, 18, 19
- dependencies: discretionary, 229; external, 229; internal, 229; mandatory, 229; in network diagrams, 228, 229
- derivativeness, 6
- design, detailed: controlling design in, 156–157; fast-tracking in, 155; functional design in, 154–155; interaction design in, 156; logical design in, 154–155; physical design in, 155; stage, in project execution, 154–156; in systems engineering, 139–141
- design, organizational, 538–540, 539
- design, preliminary: defined, 133; in systems engineering, 133–139
- design-build team (DBT), 124–125
- design-build-test, 70
- design changes, in cost escalation, 319–320
- design margins, 404, 405, 480
- design output, 172n1

- design reviews, 362–363, 364–365, 459
- design stage, in systems development cycle, 70
- deterministic approach, 280
- developing nations, international projects in, 719–720
- developmental tasks, 195
- deviation, 359
- DevOps, 512, 560, 560–563
- DevOps engineers, 512
- DevOps teams, 124
- diagram: affinity, in risk identification, 390; cause-and-effect, 373–374, 374, 390, 390; cumulative flow, 482; functional flow block, 118–119, 130–131, 131, 134; influence, 393, 394; scatter, 374; *see also* network diagrams
- diamond model, 7
- dictionary, work breakdown structure, 182, 187
- differentiation: customer, 540; functional, 539, 539; geographic, 539; in organizational design, 538–540, 539; process, 540; product, 539–540
- digital twin, 219
- direct costs, 278
- director: program, 655; of project management office, 587
- disaster recovery projects, 35–36, 39–40
- discretionary extensions, 167
- discrimination, gender, 168
- DMADV, 355
- DMAIC, 354–355
- DMS. *see* documentation management system (DMS)
- documentation: inspection, 371; in knowledge management, 643–644; in monitoring, 460; plan, 177; -sharing tools, 212
- documentation management system (DMS), 460
- domain competency, of project manager, 575–576, 581
- downstroke, in V-model, 55
- drum buffer, 302
- drum resource, 302
- due date, 237, 240–241
- dummy activities, 258, 258
- duration: expected project, 233; lean project management and, 516; reducing with critical path method, 273–280, 274, **275**, 275, 276, **277**, 278–280; shortest activity, 293; shortest project, **277**, 277–278, 278; variability of activity, 280–282, 281, 282
- Dynacom Company, 498
- dynamic environment, in appropriateness for project management, 27
- EAC. *see* estimated cost at completion (EAC)
- early finish (EF), 235–237, 236
- early start (ES), 235–237, 236, 330–337, 331–333, **334–337**, 336
- earned value (EV), 470
- earned value management (EVM), 470–481, 471, 473, **474**, 474, **475**, 477, 478, 481
- EC. *see* expected project completion cost (EC)
- ECP. *see* engineering change proposal (ECP)
- ECV. *see* expected commercial value (ECV)
- EF. *see* early finish (EF)
- EIS. *see* Environmental Impact Statement (EIS)
- elements: in systems, 45, 50, 51
- emergent issues, 213
- emotional intelligence (EI), 580
- end-items, 3, 25
- engineer, project, 586
- engineering change proposal (ECP), 362
- environment: external, 46; internal, 46; organizations and, 48, 49; in systems, 45–46, 50
- environmental impact, feasibility and, 84–85
- Environmental Impact Statement (EIS), 84–85
- EPC (engineering, procurement, and construction management), 440, 440–441
- EPC contract, 577
- EPCM (engineering, procurement, and construction management), 440, 440
- epics, 507
- ES. *see* early start (ES)
- escalation: project, 195; in risk response planning, 407; *see also* cost escalation
- estimated cost at completion (EAC), 476–479, 477, 478
- estimated cost to complete (ETC), 476
- estimates and estimation: accuracy in, 195; administrative expenses in, 324–325; analogy, 196–197; base, 202; baselines and, 194, 323; bottom-up approach in, 200, 202–203; classification of activities and costs in, 195; commitment vs., 194–195; contingency in, 194, 201–202; cost, 317–318; Delphi technique in, 201; ego of estimator in, 321; elements of, 323–326, **326324**, 327; expert judgment in, 196, 201; final cost, 202; labor expenses in, 323, 327–328; methods of, 196–203; non-labor expenses in, 323; overhead expenses in, 324–325; parametric, 197–198; PERT and, 195, 282–283; precision in, 195; profit in, 326; project planning and, 190–194, 191–193; reconciling, 203; reserve in, 194; systems development cycle and,

- 190–194, 191–193; target vs., 194–195; three-point, 195; top-down approach in, 202–203; total billing in, 326
- ET. *see* expected project completion time (ET)
- ETC. *see* estimated cost to complete (ETC)
- ethics, 167–168
- evaluation: project summary, 164–166; in systems engineering, 138–139
- evangelist, manager as, 573
- event management, 37–38
- events, in scheduling, 204, 228
- EVM. *see* earned value management (EVM)
- exchange rates, 320, 708–709
- exclusions, in scope, 179
- execute function, 23, 24
- execution, project: controlling design in, 156–157; detail design stage in, 154–156; implementation planning in, 158; implementation stage in, 158–159; integrated plan in, 189; interaction design in, 156; planning for later stages in, 157; production/build stage in, 157–158; project summary evaluation in, 164–166; side items in, 161; stages in, 153; system conversion in, 159–161, 160; system fabrication in, 157–158; system installation in, 159–161, 160; termination and closeout in, 161–164; testing in, 157–158; user acceptance testing in, 159; user training in, 159
- execution phase: in program life cycle, 653–654; in systems development cycle, 69, 70, 106, 153–154
- execution plan: budget in, 177; change control plan in, 177; communication plan in, 177; contents of, 176–178, 177; documentation plan in, 177; health, safety, and environmental plan in, 177; human resource plan in, 177; implementation plan in, 177; management section in, 177; organization section in, 177; procurement plan in, 177; in project definition, 109–110; in project planning, 176–179, 177; risk plan in, 177; scope in, 177; statement of work in, 181; technical section in, 177; testing plan in, 177; work review plan in, 177
- expandability requirements, 119
- expected commercial value (ECV), 678–679, 679
- expected completion date, 240–241
- expected payoff, 417–418
- expected project completion cost (EC), 413–414, **414**
- expected project completion time (ET), 413–414, **414**
- expected project duration, 233
- expected time, 282
- expected value, in risk analysis, 412–415, **414**, 415
- expeditor, project, 543
- expenses: administrative, 324–325; direct labor, 323; direct non-labor, 323–324, 324; general, 324–325; overhead, 324–325; *see also* cost(s)
- expertise, in estimates, 196, 201
- exploitation, worker, 167–168
- exposure, to climate change, 401
- extensions: discretionary, 167; essential, 167
- extranet, 212
- F117 Stealth Fighter, 411
- fabrication, system: in project execution, 157–158; in systems engineering, 141
- facility, workload and, 248–249
- failure mode and effect analysis (FMEA), 367–368, 389
- failure tests, 371
- fast pace, 7
- fast-tracking: in design, 155; project networks and, 231; in systems development cycle, 72–73
- fatigue tests, 371
- feasibility: alternative solutions analysis and, 84; brownfield project status and, 83; business case and, 80; commercial, 76; constraints and, 83; current system and, 83; defined, 76; environmental impact and, 84; greenfield project status and, 83; interface requirements and, 83; life cycle cost and, 76; life cycle requirements and, 82; needs definition and, 80–81; objectives and, 82; operational modes requirements and, 83; request for proposal in, 77–79, 78; sustainability and, 86–89; in systems development cycle, 76–89, 77, 78, 84, **88**; in systems engineering, 130; user requirements definition and, 81–82; value and, 76
- feasibility studies, 74, 76, 79
- feasible budget region, 337
- feedback, in lean project management, 517–518, 518
- feeding buffer, 295–297, **296**, 296, **299**
- FEL. *see* front-end loading (FEL)
- fever chart, 462–464, 463
- FF. *see* finish-to-finish (FF)
- FFBD. *see* functional flow block diagram (FFBD)
- FFP. *see* firm fixed-price contract (FFP)
- field manager, 586

- FIDIC, 455
- financial models, 677–679
- finish-to-finish (FF), 243, 244
- finish-to-start (FS), 242, 244, 244
- firm fixed-price contract (FFP), 442–443, 448
- first come, first served, 293
- first handover at completion, 438
- fitness for purpose, 352–353
- fixed-price contract, 403, 442–445
- fixed-price incentive fee contract (FPIF), 443–444
- fixed price with economic price adjustment (FP-EPA), 444–445
- float, total, 238
- flowchart, process, 389
- FMEA. *see* failure mode and effect analysis (FMEA)
- force majeure, 449
- forgetting, organizational, 642
- formal organization structure, 537–538
- formal reviews, 363–364
- FP-EPA. *see* fixed price with economic price adjustment (FP-EPA)
- FPIF. *see* fixed-price incentive fee contract (FPIF)
- free slack, 240
- freeze date, 157, 488
- front-end loading (FEL), 80, 114–115, 193, 722
- FS. *see* finish-to-start (FS)
- functional areas, 543
- functional baseline, 133
- functional baseline design, 128
- functional design, 154–155
- functional differentiation, 539, 539
- functional flow block diagram (FFBD), 118–119, 130–131, 131, 134
- functional managers, 587
- functional readiness review, 363
- functional requirements, 118–119
- funding plan, 340
- fundraising projects, 34–35
- Galactic Mining, 696–698
- Gantt calendar schedules, 241–242
- Gantt charts, 205–206, 206, 207, 208, 513–514
- gate, in methodology, 637
- gating, artificial intelligence in, 217
- gating methodology, 673
- gating process, 677
- GCSD. *see* Glades County Sanitary District (GCSD)
- gender discrimination, 168
- general and administrative expense, 325
- geographic differentiation, 539
- Glades County Sanitary District (GCSD), 58–59
- globalization, 631
- global projects, defined, 701
- GMP. *see* guaranteed maximum price contract (GMP)
- goal, in systems, 45, 50
- goal-oriented, 3, 24; leadership, 602; program, 652
- Goman Publishing relocation, 29
- good enough quality, 353
- Goods, Work, Services (GWS), 430
- governance, program, 655
- government projects, 35–37, 37
- grade, product, 354
- Grand Entry, 529–530
- Great Pyramid of Khufu, 1–2, 2, 8
- greenfield project, 83
- green projects, 86–89
- greenwashing, 168
- group productivity software, 212
- groupthink, 618
- guaranteed maximum price contract (GMP), 447–448
- hammock, 234
- handoffs, 523
- hand-over points, 251–252, 252
- HAZOP-Hazard and Operability Study, 389
- health, safety, and environmental (HSE): in execution plan, 177; in project definition, 109
- heavyweight teams, 563
- heuristic methods, in resource allocation, 293–294, 294
- hierarchy: horizontal, 25; of schedules, 206–208, 208
- histogram, 373
- Hoover Dam, 447
- horizontal hierarchy, 25
- horizontal integration, 558, 558
- House of Quality, 143, 143–146, 144
- huddles, 525
- human-made systems, 49
- human resource plan, 177
- hybrid methodology, 70–71, 526, 728
- hybrid project management, 173
- idea champions, 726
- IGPS. *see* intergroup problem solving (IGPS)
- immediate replacement, 160, 160
- immediate successors, 238
- implementation plan, 177
- implementation planning, 158

- implementation stage: in project execution, 158–159; in systems development cycle, 70
- improvement and maintenance program, 652
- incentives, contractual, 279
- incidental characteristics, 366
- inclusions, in scope, 179
- indirect costs, 278–279
- individualism, 703
- Infinity & Beyond, Inc., 423–424
- inflexible project management, 719–720
- influence: authority and, 577; outward, 577; sideward, 577; upward, 577
- influence diagram, 393, 394
- informal communication, 127, 454
- informal connections, 578
- information management, 210–214
- information radiators, 484
- initiation phase, in program life cycle, 653
- input-process-output relationship, 47
- inputs: in systems, 47, 47; in work packages, 186
- inspection: in procurement management, 437–438; in quality assurance, 369–371
- installation: in implementation planning, 158; parallel, 160, 160; in project execution, 159–161, 160
- insurance, as risk transfer, 403
- integrated product development (IPD), 560–561
- integrated product development, DevOps in, 512
- integration: horizontal, 558, 558; manager in, 572; of megaprojects, 554–558, 556, 557, 722–723; meta-systems, 722; in organizational design, 538–540, 539; of subunits in projects, 541–543, 542; system, 48; in systems development projects, 558, 558–559, 559; vertical, 558, 558
- integration contractor, 554, 557
- integrative planning and control, 10
- integrators, 541
- intended purpose, 352
- interaction design, 156
- interdependency, in systems, 49
- interface(s): feasibility and, 83; requirements, 83; in systems, 47, 49, 50; in systems engineering, 137–138
- interface event, 204
- interface management, 657, 723
- intergroup problem solving (IGPS), 612–613
- internal rate of return (IRR), 679
- internal target ship dates, 466
- International Project Management Association (IPMA), 11–12, 15
- international projects: acceptance in, 710; achievement orientation and, 703; age attitudes and, 704; banking and, 709; budget and, 715–717; communication in, 717–718; communication plan in, 717–718; contracts and, 705, 706; costs and, 716; counter trade and, 709; culture and, 702–703, **703**, 710; currency risk in, 708–709; customer requirements and, 713; customers and, 708; customs and, 702–703; defined, 701; in developing nations, 719–720; exchange rates and, 320, 708–709; export restrictions and, 709–710; financial issues in, 708–710; financial practices and, 709; formality and, 704; geographic issues in, 708–710; holidays and, 705; import restrictions and, 709–710; individualism and, 703; institutions in, 702–703; labor regulations and, 705; language and, 704; layoffs and, 705; litigation and, 706; local context and, 720; local government support and, 720; local project manager in, 711; local representative in, 711–712; local stakeholders and, 707–708; long-term orientation and, 703; meetings and, 718; monitoring in, 460, 717; offsets and, 709; payment and, 706; political climate and, 720; politics and, 706; power distribution and, 702–703; problems managing, 702; productivity in, 717; project definition with, 713–717, **714**, **716**; project management office and, 712–713; project manager in, 710–711; proposals in, 94; punctuality and, 705; relationship building and, 713; resources and, 715–717; risks with, 718–719; schedule and, 715–717; scope and, 713–714; sensitivity in, 710; social behavior and, 704–705; statement of work and, 713–714; steering committee with, 712; supporters and, 708; teams and, 713; time and, 716; time attitudes and, 705; uncertainty avoidance and, 703; unknowns in, **702**; vacations and, 705; weekends and, 705; work definition and, 714–715
- International Space Station (ISS), 3, 3
- Internet of Things (IoT), 214
- interpersonal skills, of project manager, 580–581
- intranet, 212
- Introduction phase, in product life cycle, 72
- investigation: initial, in systems development cycle, 75–76
- invitation for bid (IFB), 97, 104n7
- IoT. *see* Internet of Things (IoT)

- IPD. *see* integrated product development (IPD)
- IPMA. *see* International Project Management Association
- Iron Butterfly Company (IBC), 178, 566, 743, 751
- Islamic project financing, 709
- ISO 9001, 357
- issue log, 484
- issue management, artificial intelligence in, 217
- issue register, 484
- iterations, in project definition, 123
- iterative analysis-synthesis evaluation cycle, 54–56, 55
- iterative design-testing, 123–124
- Jamal Industries, 71
- joint ventures, 557
- Jubilee Line Extension Project (JLEP), 60–61
- judgment, expert, in estimates, 196, 201
- Kafala system, 167–168
- kanban boards, 482–484, 483, 523, 523, 524
- Kanban project management (KPM), 522–525, 523, 525
- Keith, Wilma, 627–628
- key performance indicators (KPIs), 457
- kickoff, project, 107–108
- kickoff meeting, 107, 154
- knowledge management: after-action reviews and, 644–645; common knowledge in, 643; databases in, 643–644; documentation in, 643–644; knowledge capture in, 642–643; knowledge transfer in, 643; lessons learned and, 642–643; organizational forgetting and, 642; peer consultants in, 645; personal interaction and, 644; post-completion review and, 165; project management office and, 649; in project quality management, 358; project support group in, 644, 645, 646; risk and, 388; tacit knowledge in, 644; wasted knowledge and, 642
- KPIs. *see* key performance indicators (KPIs)
- KPM. *see* Kanban project management (KPM)
- Kulczyński Products, 225
- labeling, in machine learning, 215
- labor category, workload and, 248–249
- labor expenses, 323
- laddering, 252
- large language models, 216
- large scale project (LSP), 1, 26, 99, 554–558, 556, 557; *see also* megaprojects
- late finish (LF), 237–238
- latent defect, 438
- late start (LS), 237–238, 330–337, 331–333, **334–337**, 336, 337–339, 338, 339
- Lavasoft.com, 149–150
- LCC. *see* life cycle costs (LCC)
- leader, project functional, 587
- leadership: contingency perspective in, 602–603; continuity, 726; distributed, 602; function, 24, 24; goal-oriented, 602; in megaprojects, 724–727; participative style in, 603; project circumstances and, 603; project management office, 647–648; shared, 602, 604–605; situational approach to, 602–603; style, 602; task-oriented, 602; vertical, 602
- lean production (LP), 354, 500
- lean project management (LPM), 515–522, 516, 517, 518, 520, 521
- learning from past projects, 178; *see also* knowledge management
- learning curve, 4
- legal authority, 576
- lessons learned: knowledge management and, 642–643; list, 358; log, 165, 642–643; maturity and, 633; meeting, 164–165; in project planning, 178–179; register, 165
- letter of interest, 69
- LF. *see* late finish (LF)
- liaison, customer, 586
- liaison role, 541–542, 542
- life cycle(s): conflict and, **617**, 617–618; in megaprojects, 728–729; methodology and, 637–638, 638; product, 52, 72; program, 653–654; project, 52, 67–68; requirements, 82; systems, 52; *see also* systems development cycle (SDC)
- life cycle costs (LCC), 74, 76, 341–343, 679
- likelihood: probability vs., 427n4; risk, 391–393, **392**, 394
- liquidated damages, 449
- list, work breakdown structure, 182
- local representative, with international projects, 711–712
- Lockheed-Martin, 411, 562–563
- logic, project, 218
- logical design, 154–155
- logistics plan, in procurement management, 436
- LOGON project, 178, 180, 185, 186, 187, **232**, 232, **250**, 313, **313**, 313, **336–337**, 338, 473, 474, 499, 566, 597, 740, 743–749, 750–766
- London Tower Bridge, 351–352, 352
- long-term orientation, 703

- LP. *see* lean production (LP)
 LPM. *see* lean project management (LPM)
 LS. *see* late start (LS)
 LSP. *see* large scale project (LSP)
- machine learning (ML), 214, 215
 Madrid Metro, 728–729
 magnitude of effort, in appropriateness for project management, 27
 maintainability requirements, 119
 maintenance, repair, operation (MRO), 430
 maintenance, repair, overhaul (MRO), 167
 maintenance projects, 37
 major characteristics, 366
 make or buy decision, in procurement management, 432
 management functions, in project management, 23–24, 24
 management reserve, 201, 242
 management roles, 585, 585–589
 management section, in execution plan, 177
 manager, portfolio, 673–674
 manager, program, 656
 manager, project: authority of, 578; business skills of, 581; as communication hub, 572; as decision maker, 572; defined, 10; domain competency of, 575–576; emotional intelligence of, 580; as evangelist, 573; interpersonal skills of, 580–581; matrix, 548; motivation and, 573; orientation of, 575–576; personal characteristics of, 580; portfolio, 31; product, 31; program, 31; project, 10; of projects, 548–549; pure project, 545; qualifications of, 579–584, 582; recruiting of, 582, 582–583; relationship management by, 581; responsibilities in, 575; role of, 572–573, 584–585; selection of, 582, 582–583; self-awareness of, 580; self-management by, 580; as single person, 24; social awareness of, 580; stakeholders and, 572–576; styles of, 578–579; in subcontracting, 440, 440–441; technical skills of, 581; training of, 583
 manager of projects, 587
 managing by project (MBP), 28–29
 margin(s): design, 404, 405, 480; of requirements, 119
 market risk, 386, 680
 Mars Climate Orbiter, 628
 Marshall Field's, 39–40
 Mars Pathfinder Project, 364–365, 465–466
 materials: direct, 324; overhead on, 324
 matrix management, 30
 matrix manager, 548
 matrix organizations, 547, 547–550, 548, 567–569
 maturity: continuum, 633–634, 634; explicit knowledge and, 636; lessons learned and, 633; levels of, 633, 634; models of, 634–635, 635, 636; phase, in product life cycle, 72; project success and, 636; risk and, 387; tacit knowledge and, 636
 maximax decision criterion, 417
 Maxim Corporation America (MCA), 661–663
 maximin decision criterion, 417
 maximum expected payoff, 417–418
 MBP. *see* managing by project (MBP)
 McCormick Place West, 593–594
 meeting(s): closeout, 163; huddles, 525; in international projects, 718; kickoff, 107; lessons learned, 164–165; refinement, 508; risk review, 409; spring planning, 509; sprint review, 512; standup, 458–459; status review, 458; virtual, 615–616
 megaprojects, 423, 589; aesthetics and, 721; altruism and, 721; barriers to success in, 721–722; community pride and, 721; conception of, 722; cost-escalation with, 323; drivers in, 721; front-end loading in, 114–115, 722; idea champions in, 726; integration of, 554–558, 556, 557, 722–723; interface management and, 723; leadership continuity in, 726; leadership emphasis in, 727; leadership in, 724–727; leadership roles in, 726–727; life cycle of, 728–729; off-the-shelf technology in, 728; politics and, 721; program management and, 659; sponsors in, 726; stakeholders in, 723–724; sustainability and, 721; systems engineering and, 722; technology and, 721
 Melbourne Construction Company, 268–271, 349–350
 MEP. *see* Mercury Exploitation Program (MEP)
 Mercury Exploitation Program (MEP), 666–668
 merge-point bias, 281, 286, 389
 meta-systems integration, 722
 methodology, project management, 662; agile methodology, 123; approvals in, 639; defined, 636; deliverables in, 639, **639**; development of, 641; elements of, 638–639; gates in, 637, 673; hybrid, 70–71, 728; life cycle and, 637–638, 638; as mandate, 637; phases of, 637, 638; PRINCE2 methodology, 113–114, 640; project, 70–71; project management office and, 648; reasons for use of, 637; sign

- offs in, 639; tasks in, 639, **639**; traditional, 70; waterfall, 70, 173
- Microsoft, 71–72, 75, 165, 551–552, 619
- Midwest Parcel Distribution (MPD), 93–94, 180, 739–742, 744, 751
- milestone buffers, 298, **299**
- milestone event, 204
- milestones, in scheduling, 204, 228
- milestone schedule, 207, 209
- minimax decision criterion, 417
- minimum viable product (MVP), 508
- minor characteristics, 366
- mitigation measures, 87
- ML. *see* machine learning (ML)
- modularization: defined, 54–55; in megaprojects, 728; in systems engineering, 54–56
- money laundering, 168
- monitoring: artificial intelligence in, 217; data for, 457; informal reviews in, 458; in international projects, 460, 717; key performance indicators in, 457; observation in, 459; problems with, 488–490, 489; process, 456–457; project, 458–460; in project planning, 176; reports in, 460; site visits in, 459; in stakeholder engagement, 593; standup meetings in, 458–459; status review meetings in, 458; technology in, 459–460; *see also* control
- Monte Carlo computer simulation, 288–291, **289**, 290
- most likely time, 283
- motivation, manager and, 573
- Motorola, 562, 663–664
- Mozal, 732–735
- MPD Company. *see* Midwest Parcel Distribution
- MRO. *see* maintenance, repair, operation (MRO); maintenance, repair, overhaul (MRO)
- multidisciplinarity, of systems engineering, 52–54
- multifunctionality, in appropriateness for project management, 27
- multifunctional team, 544, 544
- multiple project scheduling, 290, **291**, 291, 292–294, 294, 302–304
- multiple sourcing, 432
- multi-project organizations, 547
- multi-tasking, 251–252, 252, 300, 300
- MVP. *see* minimum viable product (MVP)
- name, project, 108
- NASA project, 36–37, 37
- natural language processing (NLP), 214, 215
- natural systems, 49
- near-critical paths, 286
- NEC. *see* New Engineering Contract
- needs definition, 80–81
- needs identification, in systems engineering, 128–129
- negotiated adjustments, 163–164
- negotiation: in procurement management, 433–434; in systems development cycle, 97
- Nelson Mandela Bridge, 424–426, 425
- net present value (NPV), 677–678
- network: of alliances, 578; project, 231–233, **232**, 232; risk sources and, 389; time-scaled, 245, 245–247
- network diagrams, 228–233, 229, **230–232**, 231, 232, 266–267; defined, 228; dependencies in, 228; events in, 228; precedence relationships in, 228
- network methods, criticisms of, 255–256
- net worth, project, late start times and, 337–339, 338, 339
- New Engineering Contract (NEC), 455
- new to market (NTM), 6
- new to world (NTW), 6
- NLP. *see* natural language processing (NLP)
- nonconformance, cost of, 355
- nonconformity, 353
- nondestructive tests (NDT), 371
- non-working time, in schedule, 241
- normal cost, 274
- normal distribution, 284–286, **285**, 285, **287**
- normal time, 273–274
- normal work pace, 273
- Normandy invasion, 4
- novelty, 6
- NPV. *see* net present value (NPV)
- NTCP model, 6–7, 7
- NTM. *see* new to market (NTM)
- NTW. *see* new to world (NTW)
- Nuwave Products Company, 598
- objectives: in feasibility, 82; in project planning, 175; in systems, 45, 50
- office: program, 554, 656; program management, 656; project, 458, 552–553, 585–586; project management, 165, 179, 216, 327, 550, 552, 585–586, 647–651; of projects, 552, 553–554
- offsets, international projects and, 709
- off-the-shelf (OTS): designs/products, 133; in megaprojects, 728; in procurement management, 430; systems, 84; in task classification, 195
- one-piece flow, 516

- open systems, 48
- operating guidelines, with teams, 611, 611–612
- operation, system, in systems engineering, 141
- operational modes requirements, 83
- operation phase: in life cycle costs, 341; in systems development cycle, 69, 70–72, 106, 166–167
- operators: in systems development cycle, 73
- OPM3. *see* Organizational Project Management Maturity Model (OPM3)
- optimistic time, 283
- organization: drawbacks of traditional, 540; matrix, 547, 547–550, 548, 567–569; multi-project, 547; pure project, 544–546, 546
- organization, project: in project planning, 176; requirements of, 540–541
- organizational design, 538–540, 539
- organizational forgetting, 642
- organizational form, selection of, 550, 550–552
- Organizational Project Management Maturity Model (OPM3), 635
- organization section, in execution plan, 177
- organizing function, 23, 24
- orientation, of manager, 575–576
- OTS. *see* off-the-shelf (OTS)
- outputs: in systems, 46–47, 47
- outward influence, 577
- overhead: direct, 325; indirect, 325; in lean project management, 519
- overhead expenses, 324–325
- overhead on materials, 324
- overrun allowance, 202
- owners: in systems development cycle, 73

- pace, of project, 7
- padding, 300
- Papua Energy Company, 314, 314–315
- parallel activities, 230
- parallel installation, 160
- parametric estimate, 197–198
- Pareto chart, 372–373, 373
- Parmete Company, 470–472, 471
- partial project, 545
- participative management, 604–605
- participative style, 603
- patent defects, 438
- payback period, 679
- payoff table, 416–417, **417**
- PCAS. *see* project cost accounting system (PCAS)
- PD. *see* power distribution (PD)
- PDM. *see* precedence diagramming method (PDM)
- peer consultants, 645
- peer reviews, 458
- penalty charges, 279
- penalty clause, 449
- percent complete, 468
- performance analysis, 470–481, 471, 473, **474**, 474, **475**, 477, 478, 481
- performance dimension, of project goal, 9, 9
- performance guarantee, 340
- performance incentive contract, 447
- performance indexes, 472–473
- performance standards, 456
- permanent teams, 542–543
- PERT: critical path method and, 282–292, **289**, 290, **291**, 291; criticisms of, 288; estimates and, 195, 282–283; in evolution of project management, 26; Monte Carlo simulation with, 288–291, **289**, 290; near-critical paths in, 286; probability of finishing by target completion dates in, 284–286, **285**, 285; in risk assessment, 397–398, 398
- pessimistic time, 283
- phase gate, 111
- phased project planning. *see* rolling wave project planning
- phase-gate review, 111–112
- physical design, 155
- pilot operation, 160, 160
- Pinhole Camera and Optics, Inc., 567
- planned value (PV), 470
- planning, project: budget in, 176; for build stage, 157; case, 223, 224, 225–226, 226; charts in, 205–209, 206–209; controlling in, 176; detailed, 108–109; estimation and, 190–194, 191–193, 194–203, 199, 201; execution plan in, 176–179, 177; implementation planning in, 158; learning from past projects in, 178–179; objectives in, 175; organization in, 190; phased, 110–113, 111, 112, 150–151, 194, 195; process, 175–176; for production stage, 157; in project definition, 108–113, 111, 112; purpose of, 175; requirements in, 175; resource plan in, 176; responsibilities in, 176, 190; responsibility matrix in, 190; rolling wave, 110–113, 111, 112, 194, 195, 231; schedule in, 176; scheduling in, 203–205, 204; scope definition in, 179–181; scope in, 175, 179–181; systems development cycle and, 190–194, 191–193; work activities in, 175; work definition in, 181–189, 182, 183, 185–187; *see also* risk response planning
- planning function, 23, 24
- pluralism, support vs., 724

- PMBOK. *see* Project Management Body of Knowledge (PMBOK)
- PMI. *see* Project Management Institute
- PMIS. *see* project management information system (PMIS)
- PmMO. *see* program management office (PmMO)
- PMO. *see* project management office (PMO)
- PMP. *see* Project Management Professional
- PO. *see* product owner (PO)
- Polanski Developers Company, 452
- portfolio, defined, 671
- portfolio, project, 31; defined, 89; work breakdown structure in, 90
- portfolio balance, 686, 686–687
- portfolio balancing, 677
- portfolio management, 673; *see also* project portfolio management
- portfolio manager, 31, 673–674
- portfolio program, 652
- portfolio selection, 677
- post-acceptance issues, 164
- post-completion project review, 164–165, 179, 358
- post-installation system review, 166
- postmortem, 164–165, 179
- potentially shippable product (PSP), 508
- power distribution (PD), 702–703
- PPM. *see* project portfolio management (PPM)
- PRB. *see* project review board (PRB)
- precedence diagramming method (PDM), 228, 242–247, 242–247
- precedence relationship, 205
- precision, in estimates, 195
- pre-critical review, 481
- predecessors, in scheduling charts, 205, 229, 230; intermediate, 230
- predictive approach, 122–123
- preliminary design. *see* design, preliminary
- preliminary design review, 363, 459
- pre-review meeting, 364
- prescreening stage, in project selection, 675–676, 676
- prime contractor, 431
- PRINCE2 methodology, 12–13, 17, 113–114, 640
- principle of aggregation, 297
- priority: of requirements, 119; risk, 398–402, **399, 401, 402**
- priority rules, 293
- PRiSM (Projects Integrating Sustainability Measures), 86–87
- process: in systems, 47, 47
- process differentiation, 540
- process flowchart, 389
- process models, 634–635
- procrastination, 300
- procured goods, work, and services, 97
- procured items, quality of, 359–360
- procurement: closeout, 431, 436–438, 450; conducting, 431, 432–436, 436, 450; controlling, 431, 436–438, 450, 462; defined, 429; defining, 430, 450; planning, 430, 450; process, 430–431
- procurement management, 659; acceptance in, 437–438; advertising in, 433; bid solicitation and evaluation in, 432–433; change management in, 437; collaborative contracting in, 435; contract administration in, 436–437; contract award process in, 434–435; contracting environment in, 438–439; contracting in, 438, 438–441, 440; contractors in, 430; contractual agreement and, 97; equipment in, 430; inspection in, 437–438; latent defects in, 438; logistics plan in, 436; make or buy decision in, 432; materials in, 430; misconceptions, 659; negotiation in, 433–434; patent defects in, 438; plan, 431; products in, 430; proposal in, 433–434; purchase in, 429; request for proposal in, 433; requirements in, 431; schedule in, 435–436, 436; services in, 430; subcontracting in, 431, 439–440; suppliers in, 430; tendering in, 433; termination in, 437–438; terminology with, 429–430; vendors in, 430; work breakdown structure and, 184; work definition in, 431; work in, 430
- procurement plan, 177, 462
- product backlog, 505, 507–508
- product development, user requirements in, 115–116
- Product Development phase, in product life cycle, 72
- product development projects, 32–33
- product differentiation, 539–540
- production coordinator, 586
- production stage: in project execution, 157–158; in systems development cycle, 70
- product life cycle, 52, 72
- product management, 31
- product manager, 31
- product owner (PO), 506
- product readiness review, 363
- product release, 508
- products, 25

- profit, in budget and estimates, 326
- program: defined, 588, 651; NASA, 36–37, 37; project vs., 30–31, 652
- program director, 655
- program governance, 655
- program life cycle, 653–654
- program management, 30–31, 651–653
- program management office (PmMO), 656
- program management themes, 654–655
- program manager, 31, 588
- program office, 554, 656
- program organization, 655–657, 656
- program sponsor, 655–656
- progressive elaboration, 233
- project(s): agile, 111; brownfield, 83, 158; complexity of, 6–7; in construction, 34; contracted, 181; as cross-functional, 3; dark side of, 167–168; defined, 3–6; diversity of, 6–7, 7; failure, 9; as goal-oriented, 3, 24; green, 86–89; greenfield, 83; management by, 28–29; NASA, 36–37, 37; novelty of, 6; pace of, 7; partial, 545; programs vs., 30–31, 652; selection, 675–677, 676; stand-alone, 545; technology level of, 6; as temporary, 3, 24; typology of, 5; as unique, 3
- project analogy, 388
- project artifacts, 460, 509–510
- project center, 545
- project charter, *see* charter
- project community, 213
- project contingency, 202
- project control, 359
- project cost accounting system (PCAS), 326–327; cost summaries and, 329–330
- project costs, 341
- project definition: alternative approaches to, 122, 122–125; definition in, 108–110; execution plan in, 109–110; front-end loading in, 114–115; health, safety, and environmental in, 109; iterations in, 123; project charter in, 113–114; project communication in, 125–127, 126; project kickoff in, 107–108; project name in, 108; project planning in, 108–113, 111, 112; rapid prototyping in, 123–124; system definition vs., 106–107, 107; team involvement in, 124; *see also* definition phase
- project failure, 9, 115, 400, 415, 621, 719, 728
- project goal: cost dimension of, 9, 9; performance dimension of, 9, 9; time dimension of, 9, 9
- project initiation: in systems development cycle, 74
- project kickoff, 107–108
- project life cycle, 52, 67–68
- project logic, 218
- project management: agile approach in, 70, 173, 502–505, 503, 504; appropriate situations for, 27–28; basic, 30; competencies, 10–11; critical chain, 294–295; in disaster recovery, 35–36, 39–40; in events, 37–38; evolution of, 26; failure, 9; features of, 24–25; forms of, 30–31; functions of management in, 23–24, 24; in fundraising, 34–35; hybrid, 173; inappropriate situations for, 28; Kanban, 522–525, 523, 525; lean, 515–522, 516, 517, 518, 520, 521; in maintenance projects, 37; matrix, 30; methodology, 10, 662; need for, 8; pure, 30; risk management as, 408–412; in service sector, 34–35; social, 213; standards of knowledge, 10–11; systems approach in, 56; in systems development projects, 32–33, 41; waterfall approach in, 70; web-enabled, 212–214; *see also* traditional project management (TPM)
- Project Management Body of Knowledge (PMBOK), 15, 634–635
- project management failure, 9, 25
- project management information system (PMIS): budgeting with, 211; capabilities of, 211–212; in change control, 211–212; in cost control, 211–212; defined, 210–211; in performance analysis, 211–212; project cost accounting system and, 327; resource management with, 211; scheduling with, 211; software, 211
- Project Management Institute (PMI), 11–12, 13–15
- project management methodology: *see* methodology, project management
- project management office (PMO), 179, 552–554, 647–651; artificial intelligence and, 216; communication technology and, 649–650; consulting and, 649; director, 548–549; evolution of, 650–651; facilities and, 650; functions and responsibilities of, 648; international projects and, 712–713; knowledge management and, 649; leadership, 647–648, 648; in matrix organizations, 550; mentoring and, 649; methodology and, 648; metrics and, 648–649; policies and, 648–649; procedures and, 648–649; project cost accounting system and, 327; project review board and, 650; project summary report and, 165; resources and, 649–650; software and, 649–650; standards and, 648–649

- project management process models, 634–635
- Project Management Professional, 11, 538, 650
- project manager. *see* manager, project
- project methodology, 70–71
- project name, 108
- project office, 552–553
- project plan, example, 750–766; *see also* planning, project
- project portfolio management (PPM), 31, 662, 671–675, 692–693
- project portfolio manager (PPM), 673–674
- project proposal: contents of, 92; cost and payment section in, 92; example, 739–742; executive summary in, 92; management/qualifications section in, 92; preparation of, 89–91; request for proposal and, 69, 77–79, 78, 94; selection of winning, 91–97, 92; simple rating in, 92; in systems development cycle, 89–97, 92, 95, 96; technical section in, 92; weighted rating in, 93
- project quality, 355; *see also* quality
- project quality management: defined, 355; processes, 356, 356–360
- project review board (PRB), 453, 649, 650, 674–675
- project support group (PSG), 644, 645, 646
- project selection, 91, 675–693
- project time buffer, 295–297, **296**, 296
- proposal: in conception phase, 69; *see also* project proposal
- proposal preparation process, 77
- prototype: agile project management and, 503; in quality assurance, **368**, 368–369, **369**; in systems engineering, 139
- prototyping, rapid, 123–124
- PSG. *see* project support group (PSG)
- PSP. *see* potentially shippable product (PSP)
- public-sector projects, 35–37, 37
- punch list, in project termination, 163
- purchase, defined, 429; *see also* procurement
- pure project manager, 545
- pure project organizations, 544–546, 546
- purpose: fitness for, 352–353; intended, 352
- PV. *see* planned value (PV)

- qualifications: of project manager, 579–584, 582
- quality: absence of defects and, 353; concept of, 351–356; costs of, 355–356; defined, 352; by design, 357; fitness for purpose and, 352–353; good enough, 353; lean production and, 354; movements, 354–355; planning, 356; of processes, 355; of procured items, 359–360; progress, 354–355; of project end-item, 355; risk and, 387; total quality management, 354; what it is not, 354; *see also* project quality management
- quality assurance: classification of characteristics in, 366–367; concurrent engineering and, 363; configuration control in, 362; configuration identification in, 361; configuration management in, 360; design reviews in, 362–363, 364–365; documentation inspection in, 371; failure mode and effect analysis in, 367–368; failure tests in, 371; fatigue tests in, 371; formal reviews in, 363–364; inspections in, 369–371; modeling in, **368**, 368–369, **369**; in project quality management processes, 356, 358; prototypes in, **368**, 368–369, **369**; stress testing in, 370–371; in system development, 360–371, **368**, **369**; techniques, 360–371, **368**, **369**; testing in, 369–371; in work packages, 187
- quality assurance supervisor, 586
- quality audit, 365–366
- quality control, 461; acceptance testing in, 372; cause-and-effect diagram in, 373–374, 374; control charts in, 372; Pareto charts in, 372–373, 373; in project quality management, 356, 359; run charts in, 372; scatter diagram in, 374; system fabrication and, 157; techniques, 371–375, 373, 374; tools, 372
- quality function deployment (QFD), 143, 143, 143–146, 144, 147, 374
- quality management plan, 357, 461
- quality measures, 82
- quality plan: in execution phase, 158; in execution plan, 177
- quality surveying, 198–200
- queues, in lean project management, 520, 520, 521

- RACI chart, 190, 592
- rapid prototyping, 123–124
- rating: simple, 92; weighted, 93
- RBL. *see* rules-based learning (RBL)
- RBS. *see* requirements breakdown structure (RBS)
- RC. *see* risk cost (RC)
- RCR. *see* risk consequence rating (RCR)
- RCT. *see* role clarification technique (RCT)
- real estate projects, 90
- recruiting, of project manager, 582, 582–583
- referent power, 577

- refinement meetings, 508
- refreezing, 621
- register: change, 488; issue, 484; lessons learned: register, 165; risk, 398, 407, 409; stakeholder, 594
- regret table, 417, **417**
- reinforcement learning, 215
- relationship management, 581
- release: contract, 468; project, 468; work order, 468
- reliability requirements, 119
- replacement, immediate, in system conversion, 160, 160
- report, project summary, 165
- representatives: office, 586; system, 166–167
- reputation of organization, in appropriateness for project management, 27
- request for proposal (RFP), 69, 77–79, 78, 94, 99, 102, 119, 433, 643, 739–742
- request for qualifications, 78
- requests for information (RFIs), 432–433
- requirement(s): allocation, 136, 136–137; analysis, 118, 130; availability, 119; commonality, 119; compatibility, 119; cost-effectiveness, 119; cost escalation in, 319–320; expandability, 119; functional, 118–119, 130, 131; international projects and, 713; interpretation of customer, 149–150; maintainability, 119; margin of, 119; non-functional, 119; performance, 132; priority of, 119; in procurement management, 431; in project planning, 175; reliability, 119; robustness, 119; in systems, 45; testability, 119; tolerance of, 119; verification, 132; in work packages, 186; *see also* system requirements; user requirements
- requirements breakdown structure (RBS), 119
- requirements definition, in systems engineering, 129–130
- requirements management (RM), 122
- requirements traceability matrix, 122
- reserve: budget, 318–319, 413, 465–466; in estimates, 194; management, 242; risk, 409; schedule, 242, 319, 413, 462; *see also* buffer; *see also* contingency
- resource allocation, scheduling and, 248–249, 290, **291**, 291, 292–294, 294
- resource buffers, 298, **299**, 466
- resource constraints, 254, 254–255, 255; scheduling and, 247–255, 248, 249–255, **250**
- resource leveling, 249, 252–253, 254, 254–255, 255
- resource loading, scheduling and, 248–249
- resource plan, in project planning, 176
- resources: in basic project management, 30; drum, 302; individual, 248; international projects and, 715–717; in management, 23; most, 293; project management office and, 649–650; scheduling, 247; in systems, 50, 51; T-shaped, 522
- resource smoothing, 249
- responsibilities: of manager, 575; of project management office, 648; in project planning, 176, 190; in project termination, 162–163; risk, 403–404; shared, 561; in work packages, 186
- responsibility matrix, 190, 592, 620
- retention money, 340
- return on original investment (ROI), 679
- Revcon Products, 148–149
- review: after-action, 644–645; critical design, 363, 459, 481; design, 362, 364–365; formal, 363–364, 459; functional readiness, 363; informal, 458; peer, 458; phase-gate, 111–112; post-completion project, 164–165, 179; post-installation system, 166; pre-critical, 481; preliminary design, 363, 459; product readiness, 363; status, in project monitoring, 458
- reward power, 576
- RF. *see* Rocky Flats project (RF)
- RFIs. *see* requests for information (RFIs)
- RFP. *see* request for proposal (RFP)
- risk: affinity diagrams with, 390; assumptions, 386; brainstorming with, 390; cause-and-effect diagrams with, 390, 390; checklist, **388**, 388–389; complexity and, 387; concepts, 384–385, 385; concurrency and, 387; contracts and, 448, 448–449; convergence points and, 389; cost, 405–407; currency, 708–709; in definition of project, 3; Delphi technique with, 391; dependency and, 387; exchange rate, 708–709; external sources of, 387; failure mode and effect analysis in, 389; HAZOP in, 389; identification, 385–391, **388**, 390; identification techniques, 388; internal sources of, 386–387; with international projects, 718–719; knowledge management and, 388; in lean project management, 519; likelihood in, 384, 395; market, 386, 680; maturity and, 387; merge-point bias and, 389; opportunities in, 384–385; process flowchart and, 389; project analogy in, 388; project networks and, 389; quality and, 387; schedule,

- 405–407; sources, 386–387; symptoms, 391; technical, 386; threats and, 385; triggers, 391; work breakdown structure and, 389; in work packages, 187
- risk analysis: decision trees in, 415–416, 416; expected value in, 412–415, **414**, 415; methods, 412–418, **414**, 415, 416, **417**; payoff tables in, 416–417, **417**; simulation in, 418
- risk assessment: PERT in, 397–398, 398; risk consequence in, 396–397; risk impact in, 394–396, **395**; risk likelihood in, 391–393, **392**, 394; risk priority in, 398–402, **399**, **401**, **402**; risk tolerance and, 391
- risk consequence, 396–397, 398
- risk consequence rating (RCR), 397
- risk cost (RC), 413
- risk exposure, 396–397
- risk impact, 384, 394–396, **395**
- risk log, 407, 409
- risk management: caveats, 411–412; communication channels in, 409; expecting the unexpected in, 409; principles, 408–409; procedures in, 409; program, 657; project management as, 408–412
- risk management plan, 408, 592
- risk of failure, 385
- risk officer, 409
- risk plan, in execution plan, 177
- risk priority, 398–402, **399**, **401**, **402**
- risk priority number (RPN), 367
- risk profile, 409, 410
- risk register, 398, 407, 409
- risk reserve, 409
- risk response planning: acceptance of risk in, 407; avoiding risk in, 404; choice of contract in, 403; contingency planning in, 407; defined, 402; escalation in, 407; insurance in, 403; reduction of risk in, 404–405; risk mitigation in, 404–405; risk responsibility in, 403–404; risk transfer in, 402–404; subcontracting in, 403
- risk responsibility, 403–404
- risk review meeting, 409
- risk time, 413
- risk transfer: contracts and, 448; in risk response planning, 402–404
- RM. *see* requirements management (RM)
- robustness requirements, 119
- Rocky Flats project (RF), 724–725
- role ambiguity, 623
- role clarification technique (RCT), 620–621
- role conflict, 623
- rolling wave project planning, 110–113, 111, 112, 194, 195, 231
- RPN. *see* risk priority number (RPN)
- RT. *see* risk time
- rules: dispatching, 293; priority, 293
- rules-based learning (RBL), 214, 215–216
- run chart, 372
- Santa Clara County Traffic Operations System and Signal Coordination Project, 61–62, 152
- SBC. *see* Star-Board Construction (SBC)
- scatter diagram, 374
- schedule: activity, 205; control, 462–463, 463; cost, 335; Gantt calendar, 241–242; hierarchy of, 206–208, 208; international projects and, 715–717; milestone, 207, 209; procurement, 435–436, 436; project, 204–205; in project planning, 176; reserve, 242; task, 205
- schedule incentive contract, 447
- schedule performance index, 472
- schedule reserve, 202, 319, 413, 462
- schedule variance (SV), 472
- scheduling: alternative relationships in, 242–247, 242–247; critical path in, 233, 234, 235, 235; due date in, 237; early finish times in, 235–237, 236; early start times in, 235–237, 236; free slack in, 240; hand-over points in, 251–252, 252; late finish times in, 237–238, 238; late start times in, 237–238, 238; multiple project, 290, **291**, 291, 292–294, 294; multi-tasking and, 251–252, 252; non-working time in, 241; precedence diagramming method in, 242–247, 242–247; project duration and, 248; in project planning, 203–205, 204; resource allocation and, 248–249; resource availability and, 248; resource constraints and, 247–255, 248, 249–255, **250**; resource leveling and, 249, 252–253, 254, 254–255, 255; resource loading and, 248–249; resources, 247; resource smoothing and, 249; splitting of activities in, 251–252, 252; target completion dates in, 237; time constraints and, 249–251, **250**, 250; total slack in, 238–240; workload and, 248–249
- scope: change control and, 179; control, 461; creep, 180–181, 319, 461; defined, 179; definition, 179–181; exclusions in, 179; in execution plan, 177; inclusions in, 179; international projects and, 713–714; in project planning, 175, 179–181; statement, 179, 180; validation, 359
- scoring models, 680–682, **681**

- screening, in project selection, 676, 676
- Scrum, 505, 505–512
- Scrumban, 526
- scrum master (SM), 506
- S-curve, 193, 335
- SDC. *see* systems development cycle (SDC)
- SDO. *see* systems development organization (SDO)
- SE. *see* systems engineering (SE)
- second handover after defect liability period, 438
- selection: methods, 682–692, **683**, **685**, 686, **688**, 689, **691**, 691; project, 675–677, 676; of project manager, 582, 582–583; of project proposal, 91–97, 92
- self-awareness, of project manager, 580
- self-managed teams, 507, 561
- sensitivity, to climate change, 400
- sensitivity analysis, 682
- sequential activities, 230
- serial systems development, 559, 559
- servant leaders, 507, 523
- service-sector project management, 34–35
- sexism, 168
- SF. *see* start-to-finish (SF)
- Shah Alam Medical Center, 8, 41
- shared responsibility, 561
- side effects, 49
- side items, in project execution, 161
- sideward influence, 577
- simple rating, 92
- simulation: in risk analysis, 418; of project duration, 288–291
- single sourcing, 432
- site manager, 586
- site visits, in monitoring, 459
- situational leadership, 602–603
- Six Sigma, 354
- slack: free, 240; least, 293; time remaining, 293; total, 238–240
- slavery, modern, 167–168
- slope, cost, 274
- SM. *see* scrum master (SM)
- small batch flow, 515–516
- smooth workflow, 521, 526
- social awareness, of project manager, 580
- social project management (SPM), 213, 604–605
- social support, 580, 624
- software: in critical chain project management, 301; fitting, to project, 218; group productivity, 212; project management information system, 211
- sole sourcing, 432
- solicitation, of bids, 432–433
- solicitation package, 432
- sourcing: multiple, 432; single, 432; sole, 432
- South African gold mine, 497
- South Land University (SLU) Information Central Building, 171–172
- SOW. *see* statement of work (SOW)
- SpaceShipOne, 32, 32–33, 140–141, 142, 152n12
- specifications, system, 120–121
- spiral model, 504, 504
- Spirit Electronics Company, 736–737
- splitting, of activities, 251–252, 252
- SPM. *see* social project management (SPM)
- sponsor: in megaprojects, 726; program, 655–656; project, 588
- sprint backlog, 508
- sprint planning meeting, 509
- sprint review meeting, 512
- sprints, 508–509
- sprint teams, 561
- SRD. *see* stakeholder requirements document (SRD)
- SS. *see* start-to-start (SS)
- stage-gate process, 692
- stakeholder engagement, 589–590; assessment in, 592; defined, 590; as feature of project management, 25–26; forms of, **592**; identification in, 590–591; monitoring in, 593; plan, 594; planning in, 592; social project management and, 213
- stakeholder engagement plan, 126
- stakeholder expectations, 655
- stakeholder identification, in systems engineering, 129
- stakeholder register, 594
- stakeholder requirements document (SRD), 129
- stakeholders, 589; defined, 73, 571; disgruntled, 593; external, 571, 572; internal, 571, 572; international projects and local, 707–708; in megaprojects, 723–724; in project management, 10, 24; project manager and, 572–576; in systems development cycle, 73; in systems engineering, 54
- stand-alone project, 545
- Standard for Portfolio Management, 31
- Standard for Program Management, 31
- standardization, 521
- standards of knowledge, 10–11
- standup meetings, 458–459
- Star-Board Construction (SBC), 147–148
- Star Trek Enterprises, Inc., 223
- start-stop control, 467

- start-to-finish (SF), 243, 244, 244
- start-to-start (SS), 243, 243
- statement of work (SOW), 79, 97–98, 98, 109, 177, 181, 186, 713–714
- states of nature, 416
- Statue of Liberty, 28, 607
- status notifications, daily, 466
- steering committees, 588, 712
- stochastic approach, 280
- stories, user, 507
- strategic management, 673
- strategy: and front-end loading, 114; and project portfolio management, 671
- stress management, 622–624
- stress testing, 370–371
- structure: formal organizational, 537–538; system, 46, 47
- students' syndrome, 300, 301
- subcontracting: contracts in, 450; in procurement management, 431, 439–440; project manager roles in, 440–441; as risk transfer, 403; *see also* contractor(s)
- subsystems: functions of, 134, 134; in systems, 45; in systems engineering, 134, 134
- successors, 229; immediate, 238
- summary activity, 234
- summary evaluation, project, 164–166
- summary report, project, 165
- sunk cost, 692
- Sunrise Beach, 426–427
- supervised learning, 215
- supervisor, work package, 587
- support, system, in systems engineering, 141
- supporters, project, 588–589
- surety bond, 449
- sustainability, 86–89, 721
- SV. *see* schedule variance (SV)
- Sydney Opera House, 423–424
- symptoms, risk, 391
- synthesis, in systems engineering, 132–133, 138–139
- system(s): architecture of, 135; attributes in, 45; boundary in, 45–46; cause-effect relationships in, 49; closed, 48; company as, 44; conflicts in, 47; constraints in, 47, 50, 294; current, 83; defined, 43; elements in, 45, 50, 51; environment in, 45–46, 50; external environment in, 46; features of, 43–44; goals in, 45, 50; human-made, 49; inputs in, 47, 47; integration, 48, 722; interdependency in, 49; interface in, 47, 50; internal environment in, 46; life cycles, 52; models, 51; natural, 49; objective in, 45, 50; off-the-shelf, 84; open, 48; organizations as, 48, 49; outputs in, 46–47, 47; process in, 47, 47; requirements in, 45; resources in, 50, 51; side effects in, 49; specifications, 120–121, 133; structure, 46, 47; subsystems in, 45; termination, 167
- system architecture, 55–56
- system complexity, 6
- system definition: functional requirements in, 118–119; project definition vs., 106–107, 107; system requirements in, **117**, 117–118; system specifications in, 120–121; traceability in, 121–122; user requirements in, 115–117; *see also* definition phase; work definition
- system development, quality assurance in, 360–371, **368**, **369**
- system representatives, 166–167
- system requirements: in system definition, **117**, 117–118; user requirements and, **117**
- systems approach: defined, 43; framework, 49–50; models in, 51; orderly appraisal in, 50–51; in project management, 56; project management as, 24, 38
- systems development cycle (SDC), 154; benefits realization management in, 74, 76; business case in, 74, 80; buyers in, 73; clients in, 73; conception phase of, 68–69, 69, 74–76, 77, 106; contractor in, 73; contractual agreement in, 97; customers in, 73; definition phase in, 69, 69, 105–108, 106, 107; design stage in, 70; execution phase in, 69, 70, 106, 153–154; fast-tracking in, 72–73; feasibility in, 76–89, 77, 78, 84, **88**; feasibility studies and, 74, 79; implementation stage in, 70; initial investigation in, 75–76; iterations in, 72–73; management in, 74; negotiation in, 97; operation phase in, 69, 70–72, 106, 166–167; operators in, 73; owners in, 73; phased project planning in, 72–73; phases of, 68, 69; production stage in, 70; project initiation in, 74; project planning in, 190–194, 191–193; project proposal in, 89–97, 92, 95, 96; stage-gating in, 72–73; stakeholders in, 73; statement of work in, 79, 97–98, 98; users in, 73; value creation in, 74
- systems development organization (SDO), 73
- systems development projects, 32–33, 41, 558, 558–559, 559
- systems engineering (SE): “all systems go” and, 53; architecture in, 135; conceptual design in, 128–129; concurrent engineering in, 54; defined, 52; detailed design and system

- development in, 139–141; dimensions of, 53; evaluation in, 138–139; feasibility and, 130; functional requirements in, 130, 131; interfaces in, 137–138; iterative analysis-synthesis evaluation cycle and, 54–56, 55; megaprojects and, 722; modularization in, 54–56; as multidisciplinary, 52–54; needs identification in, 128–129; overview of, 52–54; performance requirements in, 132; preliminary design in, 133–139; process, 72; requirements allocation in, 136, 136–137; requirements analysis in, 130; requirements definition in, 129–130; stages of, 128–142; stakeholder identification in, 129; stakeholders in, 54; subsystems in, 134, 134; synthesis in, 132–133, 138–139; system construction in, 141; system fabrication in, 141; system operation and support in, 141; system production in, 141; verification requirements in, 132
- systems models: conceptual, 51; physical, 51; in systems approach, 51
- systems thinking, 44
- Takata Corporation, 381–382
- target, estimate vs., 194–195
- target completion date, 237, 240–241
- target cost, 442
- task(s): in estimates, 195; in methodology, 639, **639**; multi-tasking, 251–252, 252; telltale, 470; in work breakdown structure, 182
- task force, 542
- task-oriented leadership, 602
- task schedule, 205
- team(s), project: building, 607–609; charter and, 612; cohesive, 614–615; in concurrent engineering, 561–563; defined, 605; design-build, 124–125; DevOps, 124; disbanding, 612; distributed, 613; expectations with, 611; feeling in, 606–607; focus in, 606–607; heavyweight, 563; high-performing, 606–607; improving ongoing, 609, 609–610; inspection, 371; international projects and, 713; multifunctional, 544, 544; new, 610–612; objectives with, 611; operating guidelines with, 611, 611–612; permanent, 542–543; priority levels with, 611; problem solving with, 612–613; in project definition, 124; purpose with, 611; in Scrum, 507; self-managed, 507, 522–523, 561; sprint, 561; time in, 606–607; troubles with, 605–606, 606; trust in, 615; virtual, 613–616
- team effort, 10
- teamwork, 607–608
- technical delivery process models, 634
- technical performance measurement (TePM), 137, 374, 461, 479–480
- technical risk, 386
- technical section, in execution plan, 177
- technical skills, of project manager, 581
- Technokrat, 665
- technology: communication, 613–614; high-tech, 6; low-tech, 6; medium-tech, 6; megaprojects and, 721; in monitoring, 459–460; project management office and, 649–650; risk likelihood and impact with, 395; super-high-tech, 6
- telltale tasks, 470
- templates, work breakdown structure, 188–189
- temporary, project as, 3, 24
- tendering, in procurement management, 433
- TePM. *see* technical performance measurement (TePM)
- termination, project: contract closing in, 163–164; defined, 161; negotiated adjustments in, 163–164; post-acceptance items in, 164; punch list in, 163; reasons for, 162; responsibilities in, 162–163
- termination, system, 167
- testability requirements, 119
- testing: acceptance, 372; failure, 371; fatigue, 371; nondestructive, 371; in project execution, 157–158, 159; in quality assurance, 369–371; stress, 370–371; user acceptance, 159
- testing plan, 177
- TFI. *see* Track & Found, Inc. (TFI)
- theory of constraints (TOC), 294–301, 295–301, **296**, **299**, 302–304
- threat profile, 409
- three-point estimate, 195
- 3Ps, 87
- time: cycle, 520–521; early start, 235–237, 236, 330–337, 331–333, **334–337**, 336; expected, 282; expected project completion, 413–414, **414**; international projects and, 716; late start, 237–238, 330–337, 331–333, **334–337**, 336, 337–339, 338, 339; most likely, 283; normal, 273–274; optimistic, 283; pessimistic, 283; remaining, 475; risk, 413; shortest processing, 293; shortest task, 293
- time and materials (T&M) contract, 448
- time-boxed, 508
- time buffer, 202, 242, 272–273, 319, 462
- time constraints, 248, 249–251, **250**, **250**

- time-cost relationship, 273–275, 274
- time-critical pace, 7
- time dimension, of project goal, 9, 9
- time-phased budgets, 328
- time-scaled network, 245, 245–247
- time variance (TV), 472
- T&M. *see* time and materials (T&M) contract
- tolerance: of requirements, 119; risk, 391
- top-down approach, 202–203
- total billing, in budget and estimates, 326
- total quality management (TQM), 354
- total slack, 238–240
- Toyota, 515
- TPM. *see* traditional project management (TPM)
- TQM. *see* total quality management (TQM)
- traceability: in configuration identification, 361–362; configuration management and, 135; in system definition, 121–122; in work breakdown structure, 184
- traceability matrix, 136, 136
- Track & Found, Inc. (TFI), 531–532
- traditional project management (TPM), 173, 500; agile project management and, 504; inappropriate use of, 502; incremental, 501, 501; operation phase and, 70; technical performance measurement and, 479
- training: in implementation planning, 158; of project manager, 583; user, in project execution, 159
- transfer, of risk, 402–404
- transition management, 657
- trigger symptoms, risk, 391
- triple bottom line, 87
- T-shaped resources, 522
- TV. *see* time variance (TV)
- twin, digital, 219
- uncertainty, 4; in risk analysis, 416
- uncertainty avoidance, 703
- unfamiliarity: in appropriateness for project management, 27; in definition of project, 3
- unfreezing, 621
- unique, project as, 3
- unique code identifier, 183
- unsolicited proposals, 79
- unsupervised learning, 215
- upstroke, in V-model, 56
- upward influence, 577
- user acceptance testing, 159
- user requirements: in feasibility, 81–82; for product development, 115–116; in system definition, 115–117; system requirements and, 117
- users: in systems development cycle, 73
- user stories, 507
- user training, in execution phase, 159
- VA. *see* virtual assistant (VA)
- value, 3; earned, 470; expected, in risk analysis, 412–415, **414**, 415; expected commercial, 678–679, 679; feasibility and, 76; generation, 25; of knowledge, 680; net present, 677–678; planned, 470
- value creation: in systems development cycle, 74
- value delivery system, 25
- value for money, 352
- Vancouver Airport, 406
- variance limits, 473–475, **475**
- velocity, 521
- verification requirements, 132
- vertical integration, 558, 558
- vertical leadership, 602
- virtual assistant (VA), 217
- virtual teams, 613–616
- vision statement, 75
- visual freeze, 466
- visual management, 482–484, 483, 523–525, 523–525
- V-model, 55
- vulnerability, to climate change, 400–402
- waterfall approach, 70, 173; *see also* traditional project management (TPM)
- waterfall methodology, 122–123
- WBS. *see* work breakdown structure (WBS)
- WCMC. *see* West Coast University Medical Center (WCMC)
- web-enabled project management, 212–214
- West Coast University Medical Center (WCMC), 101
- whiteboard, 482, 523, 523, 524
- “white elephants,” 161
- wideband Delphi, 201
- Wikispeed car project, 514
- WIP. *see* work in progress (WIP)
- work activities, in project planning, 175
- work authorization process, 467, 467–468
- work breakdown structure (WBS), 90, 181–185, 182, 183, 186, 189, 190; case, 222, 223; dictionary, 182, 187; function-oriented, 184; hardware, 183–184; list, 182; product-oriented, 183–184; risk and, 389; templates, 188–189; traceability in, 184

- work definition, 657–658; integrated execution
 - project plan in, 189; international projects and, 714–715; in procurement management, 431; in project planning, 181–189, 182, 183, 185–187; work breakdown structure in, 181–185, 182, 183, 186, 188–189; work packages in, 186–188
- worker exploitation, 167–168
- workflow: in lean project management, 515, 521; smooth, 521, 526
- working capital, 247, 248, 340
- work in progress (WIP), 520, 524–525
- workload, scheduling and, 248–249
- work order, 98, 468
- work order release, 468
- work overload, 623
- work package analysis, 472–473
- work packages, 181, 182, 186–188, 189, 467, 467–470, 469
- work package supervisor, 587
- work progress, 468
- work progress metrics, 469–470
- work requisition, 98
- work review plan, 177
- World Cup (South Africa), 378–381, 379, **380**
- Wutzrite Company, 90
- X-Philes Data Management (XDM) Corporation, 102
- X-Prize, 32–33, 85