PROJECT MANAGEMENT FOR ENGINEERING, BUSINESS AND TECHNOLOGY

FIFTH EDITION

JOHN M. NICHOLAS AND HERMAN STEYN

COMPANION WEBSITE
ROUTLEDGE

Project Management for Engineering, Business and Technology

FIFTH EDITION

Project Management for Engineering, Business and Technology, 5th edition, 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. Coverage includes project initiation and proposals, scope and task definition, scheduling, budgeting, risk analysis, control, project selection and portfolio management, program management, project organization, and all-important "people" aspects—project leadership, team building, conflict resolution and stress management.

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. The authors focus 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.

This new edition features:

- Updates throughout to cover the latest developments in project management methodologies
- New examples and 18 new case studies to help students develop their understanding and put principles into practice
- A new chapter on agile project management and lean
- Expanded coverage of program management, stakeholder engagement, buffer management, and managing virtual teams and cultural differences in international projects.
- Alignment with PMBOK terms and definitions for ease of use alongside PMI certifications

- Cross-reference to IPMA, APM, and PRINCE2 methodologies
- Extensive instructor support materials, including an Instructor's Manual, PowerPoint slides, answers to chapter review questions, problems and cases, and a test bank of questions.

Taking a technical yet accessible approach, *Project Management for Business, Engineering and Technology*, 5th edition, 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 Nicholas, PhD, is Professor of Operations Management at Loyola University, Chicago, USA.

Herman Steyn, PhD, is a Professor in the Graduate School of Technology Management, University of Pretoria, South Africa where he specializes in project management.

"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. And my students like the paperback nature of the book. 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

"The publication of the 5th edition of *Project Management for Engineering, Business and Technology* by John Nicholas and Herman Steyn is an important milestone in a continuing conversation between the authors and the current and future practitioners of project management around the world. This book has long been a comprehensive but accessible publication that provides valuable insights into the strategic and day-today management of projects both large and small. There are numerous publications in this field but Nicholas and Steyn have found the balance between the needs of experienced practitioners looking for ways to improve project outcomes, and the needs of students who are new to the project management field. The concepts are clearly and logically laid out, and the language is appropriate for a wide range of audiences. It continues to be a benchmark in a crowded field of publications offering both practical and strategic insights into the art and craft of project management."

Barrie Todhunter, University of Southern Queensland, Australia

"I have been using the earlier editions of this book in my Project Management teaching to working executives of a major engineering company employing close to 40000 people in various types of projects. I have evaluated the current 5th edition of the book from the perspective of (a) a teaching resource (b) study material and (c) as a resource for case studies and references. I find that the 5th edition has been thoroughly revamped and incorporates several relevant resources and is presented in a very lucid and structured way. I have absolutely no hesitation in recommending this book as a standard resource for teaching students in a university set up and/or for working executives in a project environment. The book is also a good resource as a study material for certification courses."

Krishna Moorthy, Ex-Dean, Larsen & Toubro Institute of Project Management, India

"*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 plays a vital role in achieving project objectives. Projects bring change and project management is recognised as the most effective way to managing such change. This book encourages readers to become interested and involved in the change towards renewed project management and management of projects."

Benita Zulch, University of the Free State, South Africa

"A very comprehensive text. An excellent mix of materials to enable students to learn techniques and engage in discussion of scenarios."

Richard Kamm, University of Bath, UK

Project Management for Engineering, Business and Technology

FIFTH EDITION

John M. Nicholas
Loyola University Chicago

Herman Steyn
University of Pretoria

Routledge
Taylor & Francis Group
LONDON AND NEW YORK

Fifth edition published 2017
by Routledge
2 Park Square, Milton Park, Abingdon, Oxon OX14 4RN

and by Routledge
711 Third Avenue, New York, NY 10017

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

Fourth edition published by Routledge 2012
Third edition published by Elsevier Inc. 2008

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

Library of Congress Cataloging in Publication Data
A catalog record for this book has been requested

ISBN: 978-1-138-93735-2 (hbk)
ISBN: 978-1-138-93734-5 (pbk)
ISBN: 978-1-315-67631-9 (ebk)

Typeset in Joanna by Servis Filmsetting Ltd, Stockport, Cheshire

Visit the companion website: www.routledge.com/cw/nicholas

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

To Karen and Janine
H.S.

Brief Contents

Contents

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 sometimes wonder, "How did they do that?" By *they*, of course, they are referring to the creators, designers, and builders, the people who created—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 astounding things from concept to reality and without whom most neat ideas would never have been achieved. This book is about them—the managers of project managers, the mostly unsung heroes of engineering, business, and technology who stand outside the public eye but ultimately 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, Woody Allen, 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, pollution, global warming, 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 "other" necessary skills. The flaw in this reasoning is that project management encompasses a broad range of skills—managerial, leadership, interpersonal—that are much different from 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" these other 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. As the title says, 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. It describes the usual project management topics of schedules, budgets, and controls, but also the human side of project management, including leadership and conflict.

Why a book on project management in engineering *and* business *and* technology? In our experience, technology specialists such as engineers, programmers, architects, chemists, and so on, involved in "engineering/technology projects" often have little or no 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 variety 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 fifth 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 new additions are a chapter on agile project management and lean production, extended coverage of program management, as well as 18 new end-of-chapter case studies. The Introduction includes tables that relate sections of the book to the most-common project management knowledge areas and methodologies: 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 what it was in the previous edition.

Our goal in writing this book is to provide students and practicing managers the most practical, current, and interesting text possible. We appreciate hearing your comments and suggestions. Please send them to us at jnichol@luc.edu and herman.steyn@up.ac.za.

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. 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 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 Dr. Gezinus Hidding for his enthusiasm and contributions to the field of project management; and to Drs. 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 this and earlier editions. In Pretoria, thanks to Dr. Tinus Pretorius 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 Dr. Giel Bekker, Philip Viljoen, Dr. Taryn Bond-Barnard, Dr. Pieter Pretorius, Dr. Krige Visser, Corro van Waveren, Dr. Michael Carruthers and Dr. Marie-Louise Barry 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, dear friends and project managers extraordinaire, for ongoing sharing of 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 and helped reduce the amount of "techno-jargon" in the book; she managed the home front and freed up time 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 not it been for her support.

Thanks also to Amy Laurens and the folks at Routledge and Taylor and Francis, and special thanks to Holly Davis for her ongoing support throughout preparation of this fifth 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

About the Authors

JOHN NICHOLAS is Professor Operations Management and Project Management in the Quinlan School of Business at Loyola University Chicago. He is an active teacher, writer, and researcher in project management and production management, and conducts executive seminars and consults 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* (2011) and *The Portal to Lean Production* (2006). He has held the positions of team lead and engineer on aircraft development projects at Lockheed-Martin Corporation, team lead and business systems analyst on operations projects at Bank America, and researcher on energy-environmental research projects at Argonne National Laboratory. He has a BS in aeronautical and astronautical 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 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 to his current position 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 graduate diploma in metallurgical engineering, an MBA, and a PhD in engineering management.

Introduction

I.1 In The Beginning…

Sometime during the third millennium BC, workers on the Great Pyramid of Cheops 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, they 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 m^2), deviates less than 1 inch (25 mm) from level (Figure I.1).[1]

Equally as staggering was the number of workers involved. To quarry the stones and transport them down the Nile, about 100,000 laborers were levied. In addition, 40,000 skilled masons and attendants were employed in preparing and laying the blocks and erecting or dismantling the ramps. Public works were essential to keep the working population employed and fed, and it is estimated that no less than 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 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]

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 and managerial competency. 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 also was 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, Solomon 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.

Figure I.1 The Great Pyramid of Cheops, an early (circa 2500BC) large-scale project.

Photo courtesy of iStock.

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 of projects.

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. In recent times, development of large systems for communications, defense, transportation, research, and information technology have spurred different, more complex kinds of project activity.

As long as people do things, there will be projects. Many projects of the future will be similar to 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 two recent projects, the English Channel tunnel (Chunnel) and the International Space Station. The Chunnel required tremendous resources and took a decade to complete. The International Space Station (Figure I.2) required development of new technologies and the efforts of the US, Russian, European, Canadian, and Japanese space agencies.

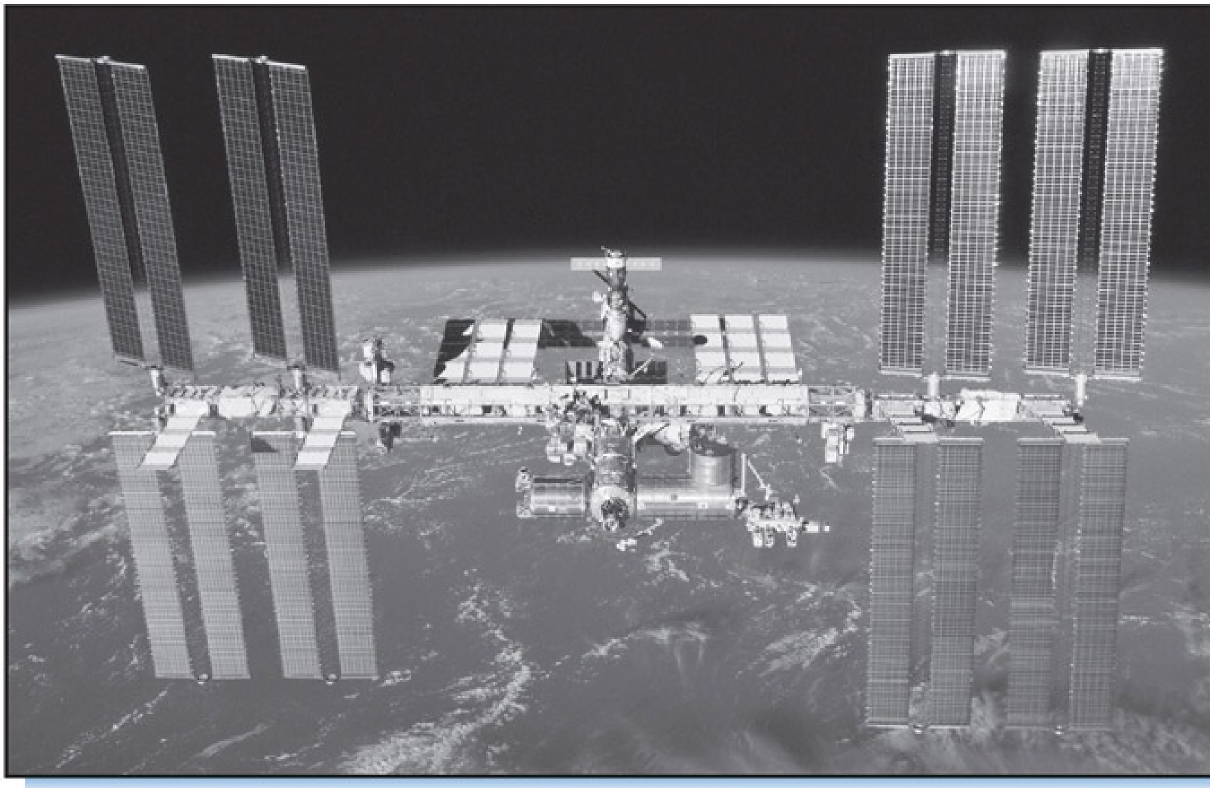


Figure I.2 The International Space Station, a modern large-scale project.

Photo courtesy of NASA.

I.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 human activities, such as 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 has a defined *goal—a purpose* with *well-defined end-items, deliverables, results*, or *products* to achieve specific *benefits*.
2. It is *unique*; it requires doing something different than was done previously. It is a one-time activity, never to be exactly repeated again.
3. It is a *temporary* organization that seeks to accomplish the goal within a scheduled time frame.
4. It utilizes people and other resources *from different organizations and functions*.
5. Given that each project is unique, it carries *unfamiliarity* and *risk*.

The examples described earlier 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 projects; and so are efforts to develop new products and implement new systems. Military campaigns also qualify as projects; they are temporary, unique efforts directed toward a specific 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 1500 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 in Chile, Haiti, China, Pakistan, Mexico, Turkey, and elsewhere, the Indian Ocean tsunami of 2004, and the Ebola outbreak in western Africa in 2014.

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 skills or 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 dimensions (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 a large number of times, then the work is usually no longer considered a project. For example, building a skyscraper is definitely 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. When in the future tourists begin taking chartered excursions to Mars, trips there will not be considered projects either. They will just be ordinary scheduled operations.

The cost curve in Figure I.3 indicates that a project's expense tends to increase roughly in proportion to its complexity and uncertainty. Cost, represented in terms of time or economic value, 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, the Mars exploration team). In contrast, repetitive, high-certainty activities (prefabricated housing, supply flights, and 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 is the reason they must be managed differently.

I.3 All Projects are Not the Same[8]

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

- *Novelty*: This represents how new the project end-item or product is to customers and potential users and how well defined are its initial product requirements. It includes three levels:

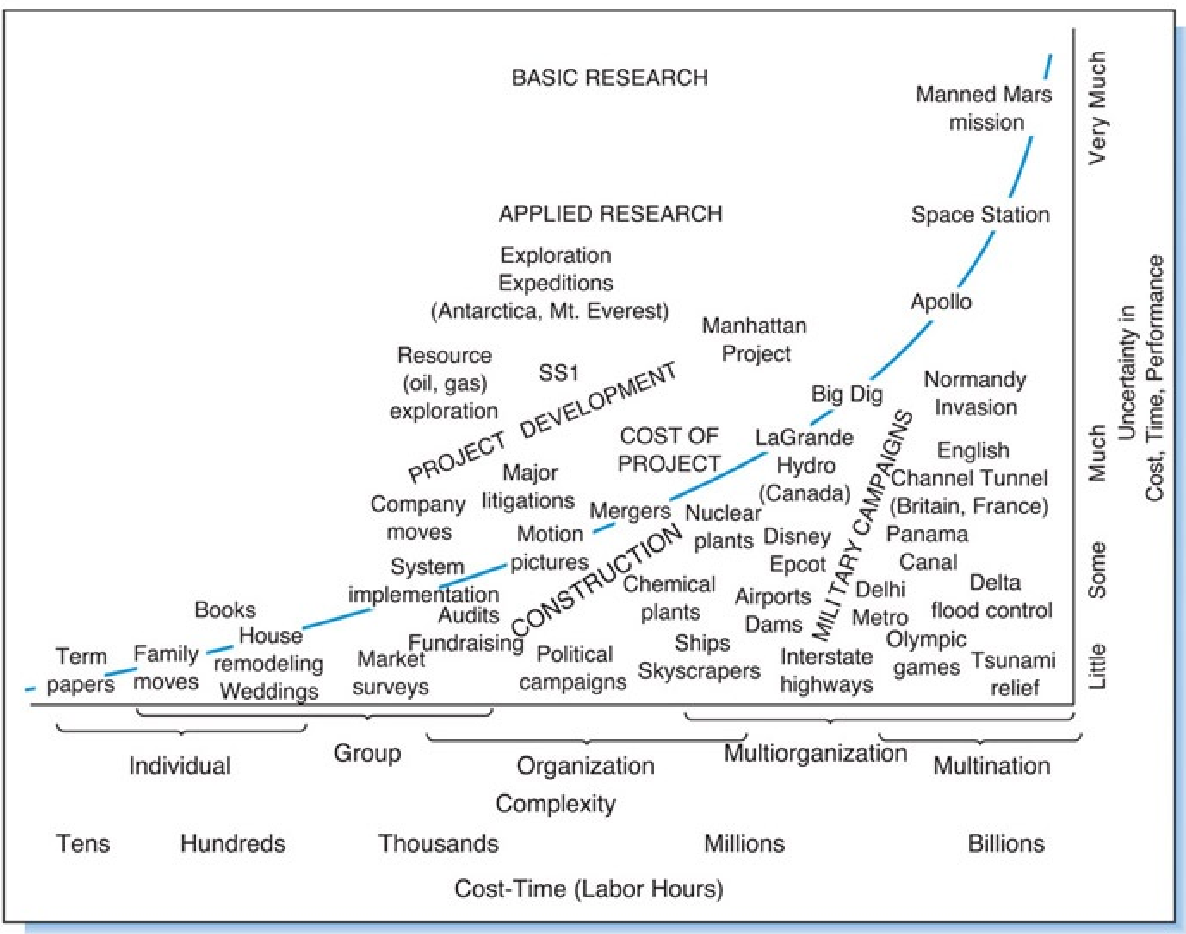


Figure I.3 A typology of projects.

    - *Derivative*—the project end-item or product is an extension or improvement of an existing product or system; e.g. new features 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*—the end-item or product is new to the world; e.g. the first mobile telephone, the first 3M Post-it notes.

- *Technology*: This represents the project's technological uncertainty and whether it is new or mature. It 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. automotive and appliances industries;
    - *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; e.g. landing a man on the moon; is often synonymous with "very high-risk."

- *Complexity*: This measures the complexity of the product and the project organization. There are 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. developing a new coffee machine or creating a department 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. a new car, new computer, 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. national communications network, mass transit infrastructure, regional power generation and distribution network, an entire corporation.

- *Pace*: This refers to time available for the project—the urgency or criticality of meeting the project's time goals. There are 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, introducing a new product line;
  - *Time-critical*—complete project by a specific deadline; missing the deadline means project failure; e.g. Y2K projects; construction of facilities for the Olympic Games; launch of space probe to a comet;
  - *Blitz*—a crisis project; the criterion for success is solving a problem as fast as possible; e.g. save people from a sinking ship.

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.

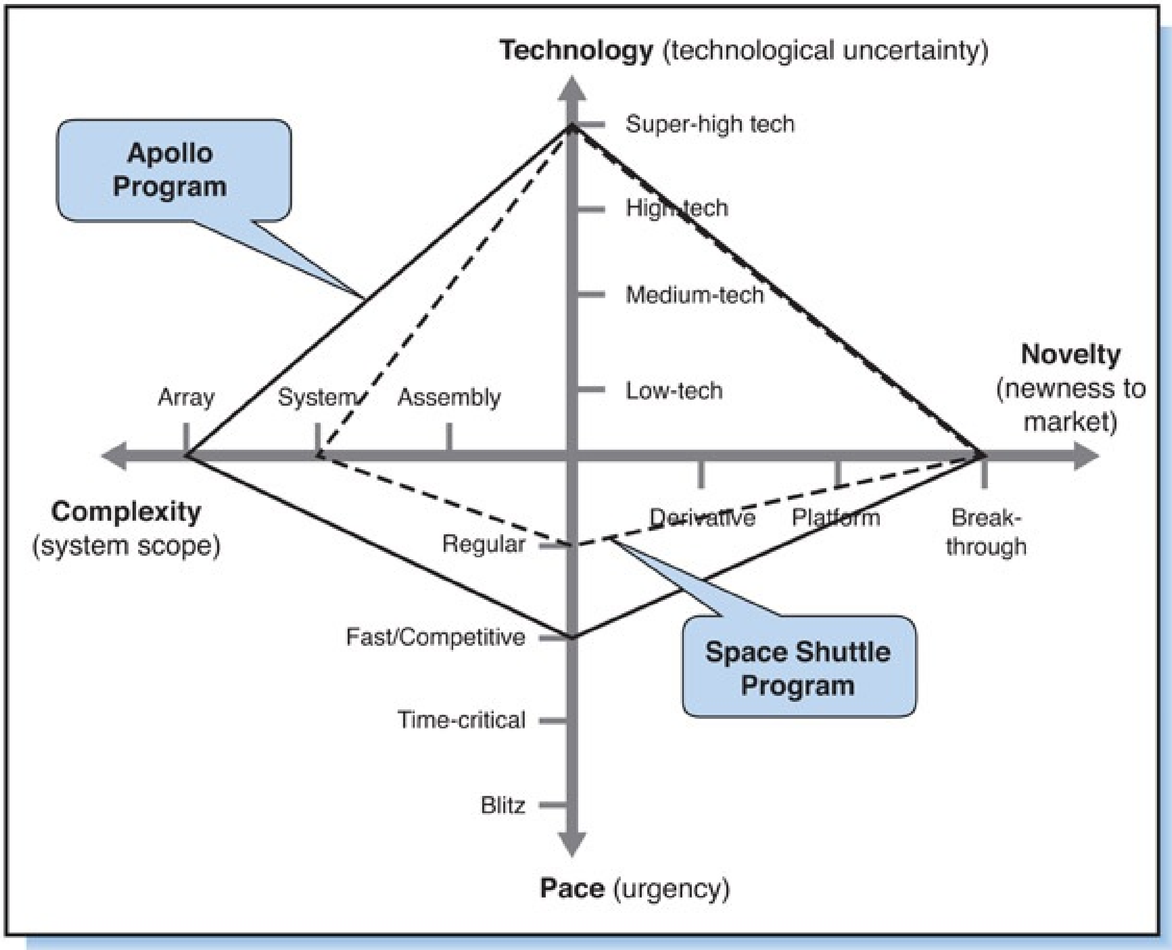


Figure I.4 Shenhar and Dvir's NTCP Diamond model contrasting the Apollo and space shuttle programs.

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

I.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 is unparalleled not only in terms of technical difficulty and organizational complexity, but also in terms of the requirements imposed on it. In ancient times, the requirements were flexible. If the Pharaohs needed more workers, then more slaves or more of the general population were conscripted. If Renaissance builders ran out of funding during construction of a cathedral, the work was stopped until more funds could be raised (one reason why cathedrals took decades or centuries to complete). If a king ran out of money while building a palace, he simply raised taxes. In cases where additional money or workers could not be found or the project delayed, then the scale of effort or quality of workmanship was reduced to accommodate the constraints.

In the Pathfinder project, many of the requirements were inflexible: the mission team was challenged with developing and landing a vehicle on Mars in less than 3 years' time and on a $150 million budget, which was 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 development and explored new areas of science and engineering. Technical performance requirements could not be compromised; to do so would increase the risk to undertakings that were already very risky.

Constraints and uncertainty in project work are not restricted to large-scale governmental science programs. They are common in everyday business and technology where organizations continually 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 product development project that exemplifies what companies everywhere must do to be competitive and survive. Product J is a promising but radically new idea.

To move the idea from a concept to a real product will require the involvement of engineers and technicians from several Dalian divisions and suppliers. Product J will require meeting tough technical challenges, launching the product well ahead of the competition, and doing it for a cost the company can afford.

Another example is Shah Alam Hospital's installation of a new employee benefits plan. The project would involve developing new policies, training staff workers, familiarizing 10,000 employees with the plan, and installing a new computer network and database, and require active participation from personnel in human resources, financial service, and information systems, as well as 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 or will have 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 are also examples of organizations' need to change, in this case to keep pace with advances in information technology and business processes.

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

I.5 Project Goal: Time, Cost, and Performance

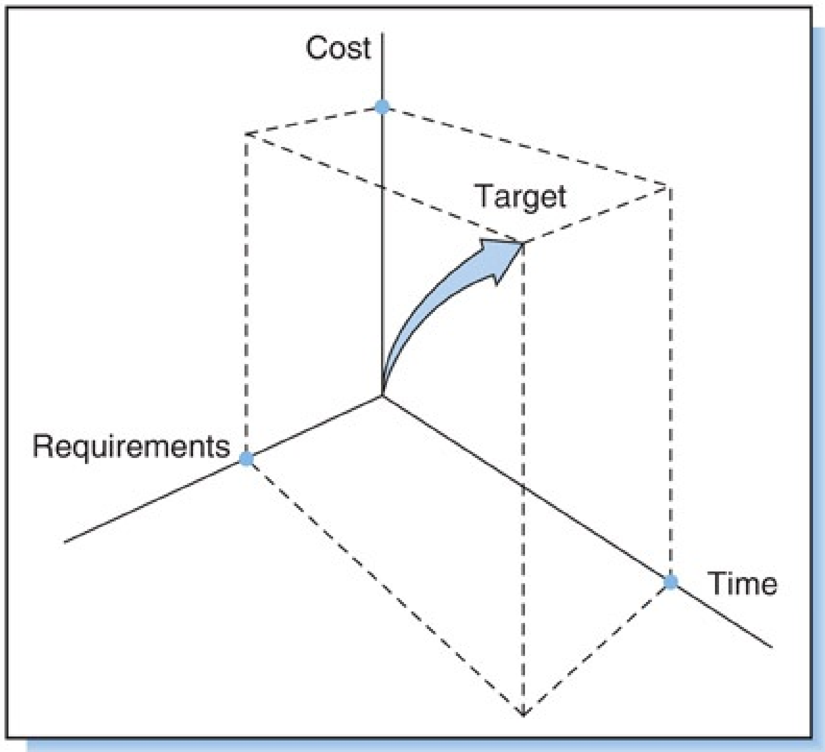


Figure I.5 Three-dimensional project goal.

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

The goal of virtually every project can be conceptualized in terms of hitting a target that floats in three-dimensional space—the dimensions being *cost, time*, and *performance* (Figure I.5). Cost is thespecified 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 final result must do; it includes whatever the project customer, end-user, and other stakeholders consider necessary or important. The target represents a goal to deliver a certain something to somebody by a certain date and for a certain cost. The purpose of project management is to hit the target—i.e., to achieve the goal of the project.[9]

But technological complexity, changing markets, and an uncontrollable environment make it difficult to hit the target. Time, cost, and technical performance are interrelated, and exclusive emphasis on any 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 were simply allowed to slide so that the "most fixed" could be met. Most projects, as the Pathfinder, Dalian Company, and Shah Alam Hospital examples show, do not have this luxury. Project management offers a way to maintain focus on all three dimensions and to control the tradeoffs among them.

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

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

The most prominent feature about project management is the role of the project manager—the individual who has overall responsibility to *plan, direct*, and *integrate* the efforts of everyone involved in the project (stakeholders) to achieve the project goal. In the role of project manager, one person is held accountable for the project and is totally dedicated to achieving its goals. The project manager coordinates the efforts of every functional area and organization in the project and oversees the planning and control of costs, schedules, and work tasks. As we will discuss later, numerous other parties (stakeholders) are involved in and crucial to project management; nonetheless, the role of project manager is a key feature that distinguishes project- from non-project management.

Doing a project is a team effort, and project management means bringing individuals and groups together to form the team 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 according to 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 more well-known of these organizations are IPMA (International Project Management Association), APM Group (Association for Project Management), and PMI (Project Management Institute). The PMI is based in the US and is the largest of these organizations; the IPMA, based in the Netherlands, is an international group of national project management associations in Europe, Africa, Asia and North and South America; the APM is based in the UK.

These professional organizations have gathered the accepted best practices of project management and published them as standards or "bodies of knowledge" (BOKs) and competencies for the profession.[12] Although none of the standards or BOKs 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 "body of knowledge-based"; IPMA's certifications are "competency-based." Another certification popular in Europe and particularly the UK is based upon PRINCE2 (PRojects IN Controlled Environments, Version 2), a methodology for managing projects 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,” “PERT,” 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, and attitudes, and systems of organization.

The intent of this book is to give a balanced view that emphasizes both the behaviorist and quantitativist sides of project management. The philosophy of this book is that for managers to “do” project management, they must gain 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 issues in 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 the skills and talent of a good project manager; hence, a final objective of the book is to cover:

5. Responsibilities of the *organization* for assuring effective project management and successful projects.

Organization of This Book

Beyond this introductory chapter, the book is divided into five main sections. The first section is devoted to the basic concepts of project management. This section describes project management principles, systems methodologies, and the systems approach—the philosophy that underlies project management. Also covered are the origins and concepts of project management, situations where it is needed, and examples of applications. The second section 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 and project management. The third section is devoted to methods and procedures for planning, scheduling, cost estimating, budgeting, resource allocation, controlling, and terminating a project. The topics of resource planning, computer and web-based project management, and project evaluation are also covered. The fourth section 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, styles of leadership, and methods for managing teamwork, conflict, and emotional stress. The last section 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 a topic that spans most other topics in this book but requires special attention, managing projects in different countries.

The five stated objectives of this book are roughly divided among chapters in the book's five sections:

1. Basic concepts and systems philosophy: Chapters 1 and 2.
2. Systems development and project life cycle: Chapters 3 and 4.
3. Methods, procedures, and systems for planning and control: Chapters 5 through 13.
4. Organization, management, and human behavior: Chapters 14 through 16.
5. The corporate context and international project management: Chapters 17 through 19.

Three Appendices provide examples of topics mentioned 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 actually participate in it or, failing that, to witness 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." The purpose of these questions and the study project is to 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 are currently working in projects as managers or project team members, the questions can be related to their current work. The questions 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 are currently 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 study (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 better opportunity 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 (i.e., perform the study using a team). 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 **PMI Project Management Bodies of Knowledge and Process Groups**

PMBOK GUIDE AND TEN KNOWLEDGE AREAS	CHAPTERS ADDRESSING THESE AREAS	
	MOST RELEVANT	RELATED
1. Introduction	0, 1	15, 16
2. Organizational influence & project Life cycle	3, 14, 16	1, 2, 4, 5, 13, 14–17
3. Project management processes		3, 13
4. Project integration management*	4, 11	2, 5, 9, 12, 14, 19
5. Project scope management*	4, 5, 11	2, 13, 19
6. Project schedule management*	6, 7, 11	5, 13, 19
7. Project cost management*	8, 11	19
8. Project quality management*	9	11, 13
9. Project resource management*	6, 16	7, 11, 14, 15, 19
10. Project communications management*	11, 12	13, 19
11. Project risk management*	10	7, 11, 18, 19
12. Project procurement management*	3, 5	11
13. Project stakeholder engagement*	15	1, 2, 3, 19
14. Appendix X3: Interpersonal & behavioral skills	16	
*Knowledge area		
Process Groups		
Initiating Process Group	3, 4	
Planning Process Group	5, 6, 7, 8	9, 10, 13, 19
Executing Process Group	11	13, 19

Monitoring and Controlling Process Group	11	12, 13, 19
Closing Process Group	12	

Table I.2 IPMA Project Management Competencies

ICB - IPMA COMPETENCE BASELINE	CHAPTERS ADDRESSING THESE COMPETENCIES	
	MOST RELEVANT	RELATED
1. Technical competencies		
1.01 Project management success		3, 5, 9
1.02 Interested parties	15	1, 3, 19
1.03 Project requirements & objectives	4, 5	2, 11, 19
1.04 Risk & opportunity	10	7, 11, 18, 19
1.05 Quality	9	11, 13
1.06 Project organization	14, 15	13, 16, 19
1.07 Teamwork	16	13
1.08 Problem resolution	16	2, 9, 10
1.09 Project structures	5, 14	1, 4, 8, 13, 15
1.10 Scope & deliverables	4, 5	2, 3, 13
1.11 Time & project phases	3, 4, 6, 7	3
1.12 Resources	5, 6, 7	8, 11, 12, 14, 16, 18, 19
1.13 Cost & finance	8	-
1.14 Procurement & contract	3, 5	11, 19
1.15 Changes	11	13
1.16 Control & reports	11	13, 19
1.17 Information & documentation	9, 12	
1.18 Communication	11, 12	19
1.19 Startup	3, 4	16
1.20Closeout	12	
2. Behavioral competencies		
2.01 Leadership	16	15, 19

2.02 Engagement		15, 16
2.03 Self-control		16
2.04 Assertiveness		16
2.05 Relaxation		16
2.06 Openness		16
2.07 Creativity		9, 10
2.08 Results orientation		16
2.09 Efficiency		5–9, 11, 16
2.10 Consultation		5, 16
2.11 Negotiation		3, 16
2.12 Conflict & crisis	16	
2.13 Reliability		5–9, 16
2.14 Values appreciation		16
2.15 Ethics		16
3. Contextual competencies		
3.01 Project orientation	I, 1, 17	
3.02 Program orientation	17	1
3.03 Portfolio orientation	18	1
3.04 Project, program & portfolio implementation	18	17
3.05 Permanent organization		4, 14, 17
3.06 Business		14, 17–19
3.07 Systems, products & technology	2, 3, 4	9
3.08 Personnel management		6, 16, 19
3.09 Health, security, safety & environment	3	4, 10
3.10 Finance		8, 11, 18
3.11 Legal		3, 19

Table I.3 APM Project Management Knowledge Areas

APMP QUALIFICATION 37	CHAPTERS ADDRESSING THESE

KNOWLEDGE AREAS	AREAS	
	MOST RELEVANT	RELATED
Project management in context		
1.1 Project management	1	I, 17, 19
1.2 Programme management	17	1
1.3 Portfolio management	18	1
1.4 Project context	1	2
1.5 Project sponsorship	15	19
1.6 Project office	17	14
Planning the strategy		
2.1 Project success & benefits management	3	9
2.2 Stakeholder management	15	1–3, 19
2.4 Project management plan	4	5–10
2.5 Project risk management	10	7, 11, 18, 19
2.6 Project quality management	9	11, 13
2.7 Health and safety	3	4, 10
Executing the strategy		
3.1 Scope management	4, 5, 11	2, 13, 19
3.2 Scheduling	6,7	5, 11, 13, 19
3.3 Resource management	5–7	8, 11, 12, 14, 16, 18, 19
3.4 Budgeting & cost management	8	11, 19
3.5 Change control	11	13
3.6 Earned value analysis	11	
3.7 Information management & reporting	12	19
3.8 Issue management	11	
Techniques		
4.1 Requirements management	4,5	2, 11, 13, 19
4.3 Estimating	8	
[illegible]	[illegible]	[illegible]

4.7 Configuration management	8	2, 11
Business and commercial		
5.1 Business case	3	
5.4 Procurement	3, 5	11
Organisation and governance		
6.1 Project life cycles	3	13, 17
6.5 Handover and closeout	12	
6.6 Project reviews	12	9, 13
6.7 Organizational structure	14	
6.8 Organizational roles	15	
6.9 Methods and procedures	17	13
6.10 Governance of project management	17, 18	
People and the profession		
7.1 Communication	12	
7.2 Teamwork	16	13
7.3 Leadership	16	15, 19
7.4 Conflict management	16	
7.5 Negotiating		3, 16

Table I.4 **PRINCE2 Methodology: Principles, Themes, Processes**

PRINCE 2	CHAPTERS ADDRESSING PRINCIPLES, THEMES, PROCESSES	
	MOST RELEVANT	RELATED
1. Seven principles		
Continued business justification	18	
Learn from experience	17	4, 13
Defined roles and responsibilities	15	
Manage by stages	3	2, 4
Manage by exception	9	
Focus on products	4,5,9	

Tailor to suit the project environment	1	I, 17
2. Seven themes		
Business case	3	
Organization	5, 14	1, 4, 8, 13, 15
Quality	9	11, 13
Plans	5	6–10
Risk	10	7, 11, 18, 19
Change	11	9, 13
Progress	11	11, 19
3. Seven processes		
Starting up a project	3, 4	16
Directing a project	11	12, 13, 19
Initiating a project	3, 4	
Managing a stage boundary	4	
Controlling a stage	11	
Managing product delivery	11	
Closing a project	12	

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?

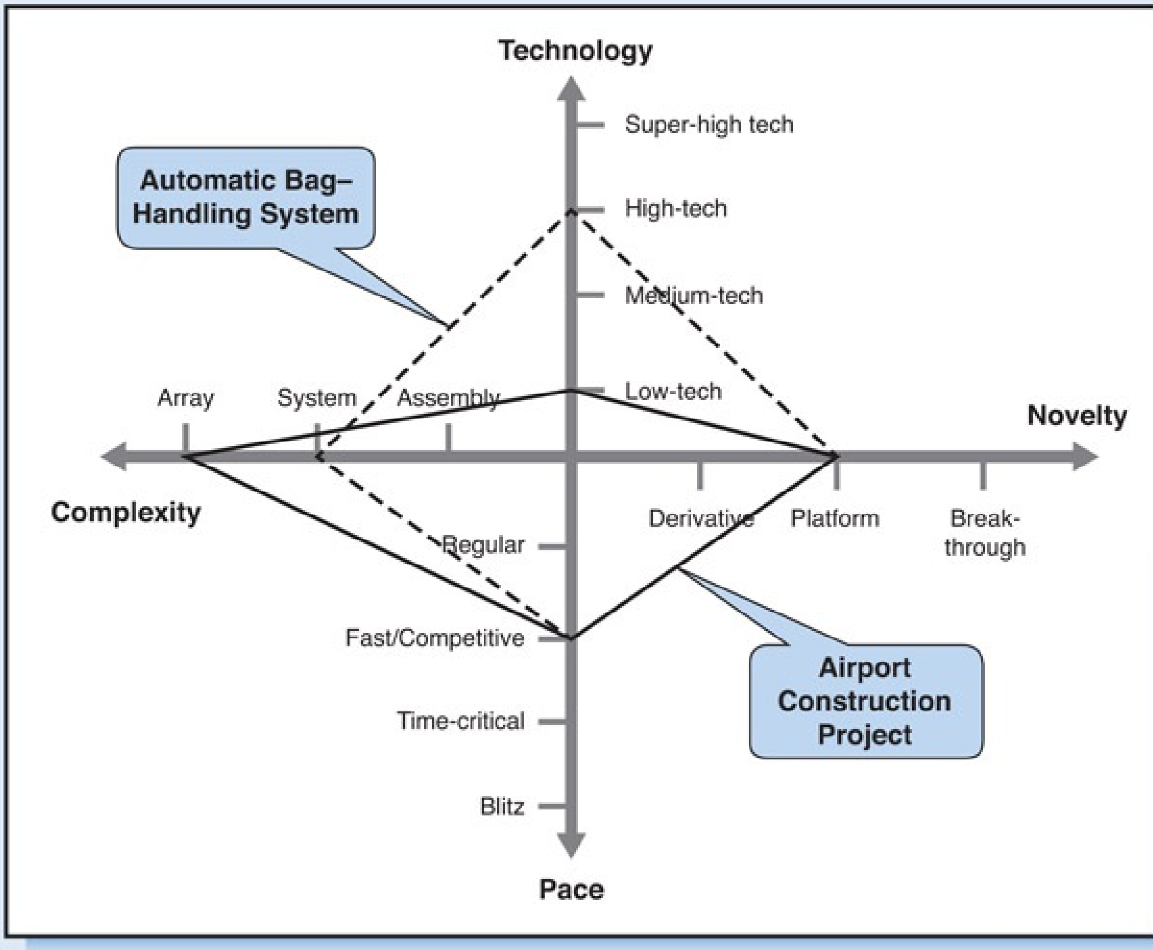
Case I.1 The Denver Airport[15]

When the Denver Airport project was initiated in 1989, the planned 4-year timeframe seemed adequate. However, despite abundant political backing and adequate funding, the project suffered a 16-month delay and a $1.5 billion cost overrun. 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 relatively simple NTCP analysis of the project and its sub-projects at an early stage (and adjusting the management style accordingly) might have significantly improved performance.

To enable aircraft turnaround around in less than 30 minutes as requested by United Airlines, one of the airport's largest tenants, an automated baggage sorting and handling system was necessary to improve efficiency over the traditional manual handling system. In December 1991 BAE Automatic Systems was contracted to design and implement the automated system in an estimated 2.5-year timeframe.

By August 1994 the system was 11 months late and was severely hampering airport operations. Management decided to build an alternative, more traditional baggage system as a backup at an additional $50 million cost, and only United would use the BAE system for its own terminal concourse. In January 1995 a full-scale practice run of the BAE system was successfully executed, 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 that one element—the automatic baggage-handling system: it was new technology and, thus, riskier than the rest of the project, a risk that was not considered. The system was the first of its kind (it had been used before only on a much smaller scale) and required several design cycles and intensive testing. It therefore should have been considered "High-tech" and managed accordingly. 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" profiles for the Denver Airport and for the Baggage-Handling System.

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

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 them to accept a schedule of 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.

Endnotes

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.

2. *Ibid.*, 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*, 6th edn. Association for Project Management, 2013; *IPMA Competence Baseline: ICB*, International Project Management Association. Available for download at http://ipma.ch/certification/competence/ipma-competence-baseline/ (accessed December 30, 2014); *A Guide To The Project Management Body of Knowledge (PMBOK Guide)*, 5th edn. Project Management Institute, 2013.

13. *Managing Successful Projects with PRINCE2*, 2009 edn. Office of Government Commerce. Available for download at https://www.prince2.com/downloads (accessed December 30, 2014).

14. Snow C.P. *The Two Cultures and a Second Look*. Cambridge, UK: Cambridge University Press; 1969.

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

Part I
Philosophy and Concepts

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

Chapter 1
What Is Project Management?

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

1.1 Functions of Management[1]

The role of management is to plan, organize, and integrate resources and tasks to achieve the organization's 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 the five functions identified in Figure 1.1. First is deciding what has to be done and how it will be done. This is the *planning* function, which involves setting a purpose or goal and establishing the means for achieving it consistent with higher-level organizational goals, resources, and constraints in the environment.

Second and related to planning is arranging for the work to be done; this is the *organizing* function. This involves (1) hiring, training, and gathering people into a team with specified authority, responsibility, and accountability relationships; (2) acquiring and allocating materials, capital, and other resources; and (3) creating an organization structure with policies, procedures, reporting patterns, and communication channels.

Third is directing and motivating people to attain the goal. This is the *leadership* function.

Fourth is monitoring work performance with respect to the goal and taking necessary action whenever work deviates from the goal; this is the *control* function.

The four functions are aimed at the goal, which implies a fifth function: assessing the four functions to determine how well each of the functions is doing and whether the functions or the goals need to be *changed.*

On a day-by-day basis, rarely do managers perform the functions in Figure 1.1 in sequence. Although planning logically precedes the others, there is always a need to organize activities, direct people, and evaluate work, regardless of sequence. Managers constantly face change, which means that plans, activities, performance standards, and leadership styles must also change.

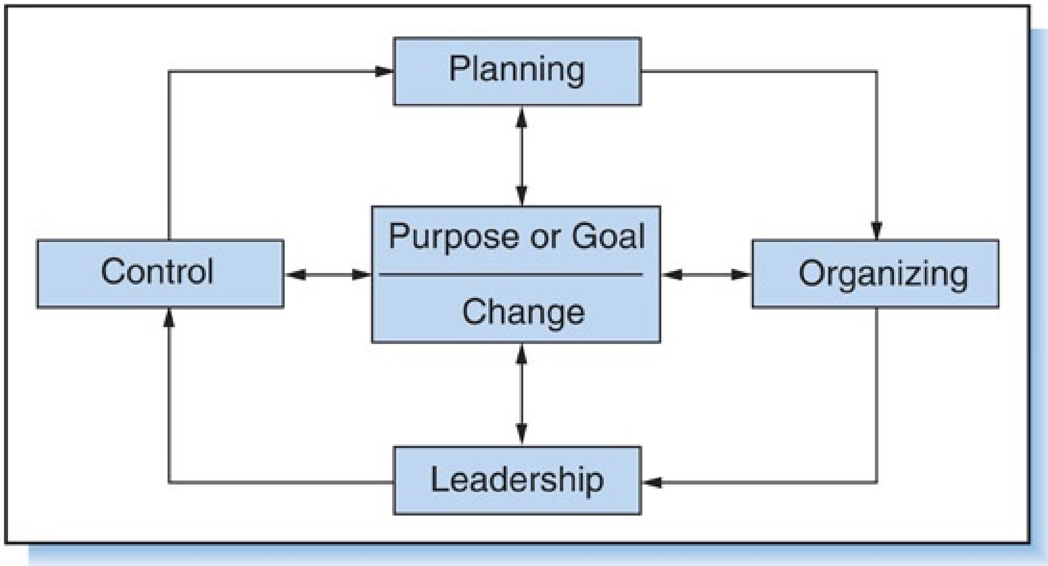


Figure 1.1 The functions of management.

Different managers' jobs carry different responsibilities depending on the functional area and managerial level of the job. Some managers 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.

1.2 Features of Project Management

Project management is a *systems approach* to management. A system is a collection of interrelated components or elements that in combination do something. 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 some goal. The focus of the systems approach is to optimize the overall *system* (*not* its individual elements) so as to achieve the goal. The approach starts by defining the goal, identifying components or elements of the system that contribute to or detract from meeting the goal, and then managing the elements to best achieve the goal. It involves all the functions of management—planning, organizing, leadership, and so on.

As described in the Introduction, projects differ from non-projects. Non-project activities such as mass production in factories or delivery of routine services are routine and seldom change; there is little uncertainty or risk involved. They tend to involve the same people doing the same procedures, day-in, day-out. In contrast, every project is unique and unfamiliar in some sense, and requires people or teams from different functions or organizations. This creates uncertainty and risk and makes it harder to achieve the desired goal. So the question is: How do you manage such a thing as a project? The answer: Use project management.

The key features of project management are:[2]

1. A single person, the project manager, heads the project organization and works independently of the normal chain-of-command. The project organization reflects the cross-functional, goal-oriented, temporary nature of the project.
2. Because each project requires a unique variety of skills and resources, project work might be performed by people from different functional areas or outside contractors.
3. The project manager is responsible for integrating people from the different functional areas or outside contractors.

4. The project manager works directly with functional managers or contractors who might be responsible for the individual work tasks and personnel within the project.
5. Whereas project managers focus on delivering particular products or services on time and on budget, functional managers might be responsible for providing project workers and resources from their departments. As a result, conflict may arise between project and functional managers over the people and resources allotted to projects.
6. A project might have two chains-of-command, one functional and one project, so people working in a project report to both a project manager and a functional manager.
7. Decision making, accountability, outcomes, and rewards are shared between the project team and supporting functional units and outside contractors.
8. Although the project organization is temporary, usually the functional or subcontracting units from which it is formed are 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 different individuals and units from within and outside the organization, managers and workers in different units and at different levels work directly with each other. Formal lines of communication and authority are frequently bypassed and a *horizontal hierarchy* is created. This horizontal hierarchy enables people in the project organization from different functional areas and outside organizations to work directly with each other as needed.

In non-project organizations, managers tend to be specialized and responsible for a single functional unit or department. A project, however, since it might involve many departments, needs someone from beyond these departments to take responsibility for meeting the project's goals. That person is the project manager. The emphasis on project goals versus the goals of each functional unit 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 areas of the organization as needed. This potentially makes being a project manager more complicated (and difficult) than being a departmental manager. Project managers must know how to use diplomacy, resolve conflicts, and be able to function without the convenience of always having the same team reporting to them.

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 probably first appeared in the major construction works of antiquity, such as the Pyramids and the Roman aqueducts, and were later modified for use on other projects such as shipbuilding. Starting in the early twentieth century, managers developed techniques for use in other kinds of projects, such as for designing and testing new products, and building and installing specialized machinery. During World War I a new tool called the *Gantt chart* for scheduling and tracking project-type work was developed (examples in Chapter 5), followed about 40 years later by the *project network diagram* (discussed in Chapter 6).

By the 1950s, the size and complexity of many projects had increased so much 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 "network-based" methods for planning and control were developed, one called *PERT*, the other called *CPM* (described in Chapters 6 and 7). A decade later, network-based methods were refined to integrate project cost accounting with project scheduling. These methods came into widespread usage in the 1960s when the US government mandated their usage in projects for the Department of Defense, NASA, and large-scale efforts such as nuclear power plants. In the 1970s, the *earned value* method of project tracking was developed (see Chapter 12); this led to performance measurement systems that simultaneously track work expenditures and work progress.

See Chapters 5 and 6

See Chapters 6 and 7

See Chapter 12

The last 50 years have witnessed the increased computerization of project management. Whereas initial project planning and tracking systems cost $10,000 to $100,000, today relatively low-cost software and freeware make it possible to use a variety of tools for planning, scheduling, costing, and controlling virtually any size project.

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 (described in Chapter 14), and it was not until the 1960s that companies began to use the term "project manager" as a formal title and role (see Chapter 15).

In recent years, project management has proliferated throughout all industries around the world. The most widespread applications and examples of each are discussed in the following sections.

See Chapters 14 and 15

1.4 Where is Project Management Appropriate?[3]

Fact is, project management is applied everywhere, and there are few industries or situations where it is not applied at least some of the time. This section identifies conditions and situations where a project-type organization applies or is essential.

Project management can be applied to any "ad hoc" undertaking. As shown in Figure I.3 in the Introduction, "ad hoc" includes activities that range from writing a term paper or remodeling a kitchen, to fundraising and constructing theme parks. Generally, the more unfamiliar or unique the undertaking, the greater the need for project management; the more numerous, interdisciplinary, and interdependent the activities in the undertaking, the greater the need for project management to ensure everything is coordinated, integrated, and completed, and nothing is overlooked.

Customers such as major corporations and governments frequently request or mandate formal project management because they believe it offers better cost, schedule, and quality control, and they prefer having a single point of contact—the project manager—with whom to deal.

See Introduction

Criteria

Cleland and King list five criteria for determining when to use project management methods and organization:[4]

1. Unfamiliarity

By definition, a project involves doing different things, doing the same things but differently, or both. For example, whereas continuous minor changes in products such as small improvements in automobile parts can usually be accomplished without project management, modernizing an automotive plant, which calls for non-routine efforts such as upgrading facilities, replacing equipment, retraining employees, and altering procedures, would certainly require project management.

2. Magnitude of the Effort

When a job requires substantially more resources (people, capital, equipment, etc.) than are normally employed by a department or organization, project management may 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. For example, although a corporate software installation project might *seem* to fall entirely within the functional area of information technology, in fact it might require a meshing of the procedures and resources of all departments affected by the installation and involve hundreds of people.

3. Dynamic Environment

Industries such as aerospace, biotechnology, computers, electronics, pharmaceuticals, and communications face continual change driven by an environment characterized by high innovation, intense competition, and shifting markets and consumer demands. Project management provides the necessary flexibility to deal with emerging threats and opportunities in such environments.

4. Interrelatedness

Functional areas tend to be self-serving and work independently. When the effort requires that they work together, project management is necessary to build relationships between the areas, expedite work, and reconcile conflicts. The project manager coordinates the efforts of internal functional areas and with outside subcontractors and vendors.

5. Reputation of the Organization

If failure to satisfactorily complete a project would result in financial ruin, loss of market share, damaged reputation, or loss of future contracts, the case for using project management is strong. Project management cannot guarantee success, but it does improve the odds by reducing the inherent risks in large, complex undertakings.

Example 1.1 Renovating the Statue of Liberty[5]

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 restoration of the statue and other buildings on nearby Ellis Island, the US Department of Interior established a foundation.

Very little of the restoration work qualified as "standard." It involved highly specialized skills such as erecting the scaffolding, constructing a new torch, building windows for the crown, and replacing the interior framework—expertise that tends to be found in smaller firms. As a result, the work was accomplished by 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 renovation effort 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; such a situation calls for project management. (Chapter 16 discusses the company responsible for managing the renovation.)

See Chapter 16

Where Project Management is 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 end-item, and the lower the stake in the result, the less the need for project management. 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 standardized, repetitive operations, there is much certainty in the process and outcome; for such operations, standardized, routine procedures for production planning, scheduling, and budgeting are well-suited, and project management is not.

1.5 Management by Project: A Common Approach

Though not appropriate for every situation, project management applies to a great many, and not only large-scale, infrequent undertakings, but also all kinds of smaller, more frequent activities as well. Whenever an undertaking involves activities that are somewhat 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 more the applicability of project management.

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. If the work requires coordinated efforts from different parties, project management applies.

Think about these situations for a moment and you start to realize the many cases where projects happen and project management applies.

Managing any kind of work as a discrete project is referred to as "managing by project," or MBP.[6] With MBP, an undertaking or set of activities is planned and managed as if it were a project. In particular, MBP implies that the undertaking will have well-defined objectives and scope, firm requirements for the end-results, a plan of work, a completion date, and a budget for the required resources. A team is formed for the sole purpose of performing 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 stable repetitive industries, small projects involving a few individuals are always in progress: new machines are installed, old ones are repaired; the office is remodeled; the cafeteria is renovated. When these or larger project efforts arise, a formalized project group is formed and a project manager appointed.

Example 1.2 Relocation of Goman Publishing Company

Many companies, regardless of size (headquarters for a multi-billion dollar corporation or a storefront family restaurant), 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 or ruin. 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 expense, distance from current location, prestige and size of the new location, and access to major highways. The next task was the relocation planning, which had two major phases: design and construction of the new facility, and the physical move, each involving numerous considerations. For example, Goman wanted to retain its current employees, and so as to maximize the new facility's appeal it chose to build 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 move. 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, including systems management, task force management, team management, ad hoc management, matrix management, and program management. All these forms share two features: (1) a *project team* or project organization created uniquely for the purpose of achieving a specific goal, and (2) a single person—a *project manager*—assigned responsibility for seeing that the goal is accomplished. Beyond these, features of the forms are somewhat different.

The first section below covers "basic" project management, the most commonly understood concept of project management. The other sections cover management forms similar to project management or related to project selection.

Basic Project Management

Commonly the project manager and functional managers are on the same organizational level, and both 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. The project manager may work directly with any level and functional area of the organization to accomplish project goals. She reports to the general manager or owner and keeps him apprised of project status. The project manager sometimes has authority to hire personnel and procure facilities, although more often she has to negotiate with functional managers to "borrow" them.

Basic project management is implemented in two widely used forms—pure project and matrix. 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. In matrix management, the project organization 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. These two project management forms will be described further in

Chapter 14.

See Chapter 14

Often found in construction and technology industries, basic project management is also readily applied to small, nontechnical activities, including in the arts and social sciences. Adams, Barndt, and Martin cite examples:[7]

- Health, Education, and Welfare (HEW) performs social work largely on the basis of grants allocated through state and local agencies. Associated with each grant are time, cost, and performance requirements for the funding agencies. In essence, each grant results in a project to which the concepts of project management can be applied.
- An advertising firm conducting promotional campaigns utilizes the services of the marketing research, accounting, graphics, sales, and other departments. The campaigns are similar to the projects in other industries: they require planning and coordination of the 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 or objectives about what must be accomplished and both require plans, budgets, and schedules for accomplishing the goals.

Nonetheless, 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 work efforts or projects* working to meet a program goal. The projects within a program share common goals and resources and, often, are interdependent. As examples, an urban development program may include several projects such as housing rehab, job and skill training, and small business consulting assistance; a Mars exploration program may include several projects for unmanned probes to the red planet and its moons, Phobos and Diemos, followed by a manned mission to Mars.

Another distinction is that projects are oriented to producing and delivering a product or service, but after that the project organization is dissolved and responsibility is handed off to someone else for operating the end-item. In a program, however, it is the responsibility of program management to ensure the end-item is not only delivered but is integrated with other systems and operational for as long as needed. For example, program management would oversee not only development of a satellite and its booster rocket and launch, but also ongoing operation and monitoring of the satellite—whatever is needed to achieve the overall satellite program goal.

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 has published a Standard for Program Management that aligns with its PMBOK; the UK Office of Government has produced one that aligns with Prince2.[8] Program management is discussed more in Chapter 17.

See Chapter 17

New Venture Management

Project management resembles *new venture management*, a type of management used in consumer-oriented firms for generating new products or markets. In new venture management a team is created to find new products or markets that fit an organization’s specialized skills, capabilities, and resources. Once it has defined the product, the team may go on to design and develop it, then determine the means to produce, market, and distribute it.

Product Management

The term *product management* refers to a special type of program management where 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 managing conflicts and resolving problems that would degrade manufacturing capability, forestall distribution, alter price, harm sales, or in any way affect financing, production, and marketing of the product. For products with long life cycles, the product manager role is filled on a rotating basis.

Project Portfolio Management

Many organizations select and group projects and programs into "portfolios," each similar to an investment portfolio, with the goal of maximizing the value of the portfolio in terms of, for example, profit, rate of return, or meeting company strategic goals. The portfolio manager helps the organization make decisions about adding, cancelling, or changing projects or programs based on 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 really about managing projects per se, but selecting and retaining the *right* projects. Thus, to portfolio managers, project management experience is less important; of greater importance is the financial and analytical skills needed to select and group projects and programs based on their contributions to strategic and financial goals. The portfolio manager must know how to diversify a project portfolio to take advantage of available resources and the risk–reward tradeoff. The PMI created the Standard for Portfolio Management and offers a Certification in Portfolio Management; the UK OCG has also created a standard.[9] Portfolio management is covered in Chapter 18.

See Chapter 18

1.7 Project Environments[10]

Project management practice also varies depending on the project environment, which author Daniel Roman classifies as commercial/for profit, government/nonprofit, and military. All the project forms described above are found in the commercial environment. The two forms most commonly found in government and the military are basic project management and program management.

Commercial/For-Profit Project Management

In a commercial project the end-item is a clearly defined product or service, typically customized to satisfy a customer, and motivated by profit criteria. The project manager usually guides the project start to finish, coordinating efforts of the project team with functional areas, subcontractors, and vendors, and keeping the customer and top management informed of progress toward project and profit objectives.

Government and Nonprofit Project Management

Government and nonprofit projects differ from commercial activities in several ways. First, there is no profit incentive in government and nonprofit work, and economic factors may be of lesser importance. Project managers are frequently reassigned midstream in their projects, which is problematic in terms of administrative continuity. For government work, project continuance often depends upon political considerations and funding that is legislatively appropriated.

Second, most of these projects focus on evaluation or testing of products or services procured from commercial contractors or vendors. Because design and development work in government projects is usually done by contractors, the project manager's role is largely administrative. Though she is responsible for checking on the contractors' progress, the project manager may have little control over technical matters. Project managers frequently oversee and coordinate multiple, related projects; in other words, they are program managers.

Military Project Management

Most military projects involve testing and evaluating hardware developed by contractors. Evaluation is often based on the "weapons systems" approach whereby each project is part of a larger systems program and hardware is evaluated for its contribution to the mission of the overall system. The major criteria for evaluating projects are technical and political; costs are of lesser importance; profit is not a consideration. Each civilian contractor has a project manager who works with a military project manager. The latter are military officers and, because of their limited tours of duty, typically do not oversee a project for its full.

The following sections describe situations where project management is most common: development of new products and systems, construction, services, and public sector works.

1.8 New Product and Systems Development Projects

The development of every new product and system is a project. Examples include projects for developing products (such as household appliances, industrial machinery, and computers) and systems for defense, aerospace, energy, and telecommunication. Projects in medicine and pharmaceuticals include the development of new medicines, medical procedures, and medical equipment. Projects in IT/ICT include the development of information systems and software products.

When the development of new products and systems comprises new technologies, the early phases of the projects typically involve testing and experimentation. Although the main purpose of each project is to create a newly designed product or system, the actual project deliverable could be either the physical product or system, or merely documents showing the product design or how to produce it. These projects all have a significant engineering component; the examples described in Chapter 2 would fall into this category.

Following are two examples of product and systems development projects.

See Chapter 2

SpaceShipOne and the X-Prize Competition[11]

In April of 2003 SpaceShipOne (SS1) and its mother ship White Knight were rolled out to the public. Simultaneously it was announced that SS1 was entering the $10 million X-Prize competition against 23 other teams from seven countries to be the first manned vehicle to successfully make two trips into space in less than 2 weeks (Figure 1.2). Space is internationally recognized as beginning at 100 km, or about 62 miles (commercial jets fly at a height of about 8 km). The brainchild of celebrated aerospace engineer and visionary Burt Rutan and the culmination of almost 8 years of design and development work, it was but the first step in Rutan's broader dream to build vehicles to carry paying passengers into space. 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 having many hundreds of engineers and many millions of dollars in government support to do it. Rutan would try to do it with his own company of 130 people, a handful of subcontractors, and the $25 million backing of billionaire Paul Allen, cofounder of Microsoft.

Figure 1.2 SpaceShipOne beneath its mother ship, White Knight.

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

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 is the sponsor of the X-Prize competition. Its long-term goal is to spur innovations that will make space travel safe, affordable, and accessible to everyone, and its X-Prize requirements were 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, is the program's customer; his plan is to buy spaceships and the associated technology for his fledgling space airline, Virgin Galactic. Branson has estimated Virgin will 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 preflight training, custom-molded seats, and 5 minutes of floating weightless while in space. (By comparison, a civilian trip aboard the International Space Station costs about $50 million.) Paying passengers are another stakeholder group. Although none would be aboard SS1, the vehicle was designed with them in mind. For instance, SS1's cabin is designed to provide a "shirtsleeve" environment so passengers would not have to wear spacesuits. The FAA is also a stake-holder; it imposes a long list of requirements necessary for the spaceship to be "certified" and commercially viable.

As in most technical projects, a project manager shared oversight of the project with a project engineer. The project engineer was responsible for identifying technical requirements and overseeing design work, system integration, and testing. All this, and what else is left for the project manager to do, will become clear in later chapters.

The Development of "Product J" at Dalian Company[12]

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, home and commercial appliances, computer and entertainment electronics, and communications that must continuously generate new products to survive in a competitive environment.

Dalian Company was concerned about maintaining market share for its "Product H," but it knew that competitors were developing substitutes for Product H that might be less expensive. To beat the competition, Dalian had to develop its own improved substitute, "Product J."

The product development process at Dalian is facilitated by the New Product Development Department. The department 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. The directors facilitate, coordinate, and monitor the project efforts of the various departments—research and development, engineering, marketing, manufacturing, and legal.

For each new product concept a team is created with representatives from functional departments. A director works with the team to assess the project's progress and requirements. Functional managers decide about what is to be done and how, but the directors have final say over project direction. The directors always know the status of their projects and report any problems or delays to upper-level managers who manage the projects as a "portfolio" (i.e. they are portfolio managers). Projects facing big problems or signs of failure are cancelled so resources can be allocated to more promising projects.

Similar to all new product developments, development of "Product J" involved the participation of several departments: R&D developed a product prototype and prepared specifications; engineering defined where and in what ways the product would be used; marketing defined the commercial market and determined how to position the product; manufacturing developed a new process for making the product that would be difficult for competitors to copy; finance determined the initial product costing and performed profit/loss forecasts; and legal obtained regulatory approval and performed patent research.

The director for "Product J" was involved from project conception. She worked with R&D scientists and marketing experts to determine project feasibility and was active in gaining upper management's approval. She worked with scientists and managers to prepare project plans and schedules. When additional labor, equipment, instruments, or raw materials were needed, she wrote requests for funds. When additional people were needed, she wrote personnel requests to upper management. During the project she scheduled and chaired project review meetings and issued monthly and quarterly progress reports.

1.9 Construction Projects

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 all 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 examples of project management in construction, for example: Case 8.2, the Chunnel Project; Case 9.1, Big Dig; Case 10.1, Sydney Opera House; Case 10.3, Nelson Mandela Bridge; and Case 19.1, the Mozal Aluminum Smelter. These cases describe large projects. The following case study illustrates project management in a smaller project.

Small Projects at Delamir Roofing Company

Delamir Roofing Company installs and repairs roofs for factories and businesses. Like other companies associated with the construction industry, Delamir considers each job a project and assigns a project manager to oversee it.

Involvement of the project manager begins when a request for work is received from a potential customer. The project manager examines the blueprints to determine how much material and labor time will be needed (called “prepping the job”) and then prepares a budget and a short proposal. After the contract is signed, the project manager visits the site ahead of the crew to make arrangements and accommodations for work to begin. The project manager has discretion in work crew selection to ensure that the size and skills of the crew fit the requirements of the job. After work begins, he is responsible not only for supervision of work and delivery of supplies, but for maintaining budget records and reporting progress to the home office. The project manager performs the final inspection with the customer and signs off when the job is completed. Overall, his responsibility is to see that the job is done well.

1.10 Service-Sector Projects

Project management is employed in a broad range of services, including banking, consulting, insurance, national revenue services, stock exchanges, and accounting, and includes projects for developing new service systems and processes. The next two examples show how it is used in a corporate audit and in a nonprofit fundraising campaign.

Auditing at CPAone[13]

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 skills, expertise, and personalities involved, a project manager is needed to oversee the audit. Every audit begins by assigning the client to a 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.

The project director begins by studying the client's income statement, balance sheet, and other financial statements. If the client has a bad financial reputation, the project director can make the decision for CPAone to refuse the audit. If the client is accepted, the director prepares a proposal that explains the general approach for conducting the audit and designates the completion date and the cost estimate.

In determining the general approach for conducting the audit, the project director considers the company's size and number of departments. Auditors are then assigned on a department-by-department basis. The audit team is a pure project team, created anew for every audit, composed of people who have the skills best suited to the needs of the audit. Generally, each audit team has one or two staff accountants and one or two senior accountants. During the audit the director monitors all work to ensure it adheres to the Book of Auditing Standards and is completed on schedule. Each week the client and project director meet to review progress. When problems cannot be solved immediately, the director may call in people from CPAone's tax or consulting divisions. If the IRS (Internal Revenue Service) requests an examination after the audit is completed, the project director sees to it that the client is represented.

Nonprofit Fundraising Campaign Project: Archdiocese of Boston[14]

The Archdiocese of Boston contracted the American Services Company, a fundraising consulting firm for nonprofit organizations, 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 had to work 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 and church leadership 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 Archdiocese community to achieve the $30 million goal. The study indicated that the goal was achievable, and pastors were invited to a kickoff luncheon at which time the Cardinal of the Archdiocese introduced the campaign. During the meeting, influential church personnel were signed up and the process of identifying potential donors and volunteers was started.

The project manager provided guidance for establishing a campaign leadership team and project office, enlisting volunteers, forming campaign committees, and recruiting and training volunteers. In addition to organizational matters, he 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.

1.11 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 involve the labors of numerous organizations. A large disaster such as the December 2004 tsunami in the Indian Ocean impacts many countries and requires the support and coordinated efforts of host governments, non-governmental agencies (NGOs), local business, religious, and community organizations, and international aid, charitable, and funding organizations.

Almost by definition, post-disaster recovery is a program—or several programs—a host of efforts devoted to the goals of rescuing and providing immediate relief to victims and, ultimately, of returning the lives of people in the areas affected back to normal. Each program involves many projects to address the multiple aspects of a recovery effort, including projects to provide:[15]

- 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—a managed program with numerous projects—that enables quick assessment of the scope of the situation, identification and organization of needed and available resources, and effective 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 has been blamed on a lack of leadership and coordinated management at all levels of government—federal, state, and local.

In the months and years following a disaster, the focus turns to obtaining and allocating aid funding; reconstruction, redevelopment, and rebuilding (infrastructure, organizations, facilities); permanently situating (returning home or relocating) victims; dealing with waste and debris; and providing opportunities, jobs and ongoing support. To accomplish this requires numerous projects, for instance, to obtain and allocate financial assistance to individuals, businesses, and local government, and provide subsidized housing and building materials. Often the goal is to employ the victims in many small-scale, labor-intensive projects to provide jobs and income.

For example, the December 2004 tsunami caused severe damage to coastal areas in Sri Lanka, Thailand, Indonesia, the Maldives, and other countries around the Indian Ocean, and in India alone affected an estimated 2.7 million already-poor people, 80 percent of whose livelihoods depended on fishing and 15 percent on agriculture. The government of India launched the Emergency Tsunami Reconstruction Project, estimated to cost US $682.8 million, to help repair or reconstruct about 140,000 damaged houses in two coastal regions and assist with the reconstruction of public buildings and the revival of livelihoods in fisheries and agriculture.[16] It is a project–a program really–that consists of many hundreds of projects, takes many years, and continues for as long as the funding holds out.

NASA Project and Program Management[17]

NASA has had a successful history of working in partnership with researchers in universities, industry, and the military. NASA and industry work closely together on technical problems, but technical initiatives and technical 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, which conduct the programs and their projects either on-site or at universities for contractors. NASA is divided into four mission directorates or offices: Exploration Systems, Space Operations, Science, and Aeronautics Research, shown in Figure 1.3.

Each directorate is responsible for development, justification, and management of *programs* that support broad NASA goals. Directorates are assigned field installations to carry out permanent activities for the directorate, but, still, also carry out projects or tasks under the direction of other directorates. For example, though Ames reports to Science, it also contributes to projects in Space Operations.

In a typical non-NASA government agency project, the agency prepares specifications for a program, lets a contract, and then relies on the contractor for results. NASA uses a different approach since no one company likely has all of the capability to execute a large project. Although NASA relies upon industry to build, integrate, and test-fly hardware, it relies upon its own in-house management and technical competence to monitor and work with contractors. Because NASA projects call for a diversity of technical and managerial competency, 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. The concept utilizes teamwork, central control, and decentralized execution, but respects the semi-autonomous status of NASA’s field installations.

NASA defines a *program* as a series of undertakings that over several years are designed to accomplish broad scientific or technical goals. It defines a *project* 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.

NASA uses a dual system of responsibility. The single greatest contributor to a project’s success is the person upon whom final responsibility rests, the *project* *manager*. She is the official responsible for executing the project within the guidelines and controls of NASA, and for day-to-day supervision, execution, and completion of projects. Although most of the workers on a project are outside of the administrative authority of the project manager, nonetheless they take directions from her *on any project matter*.

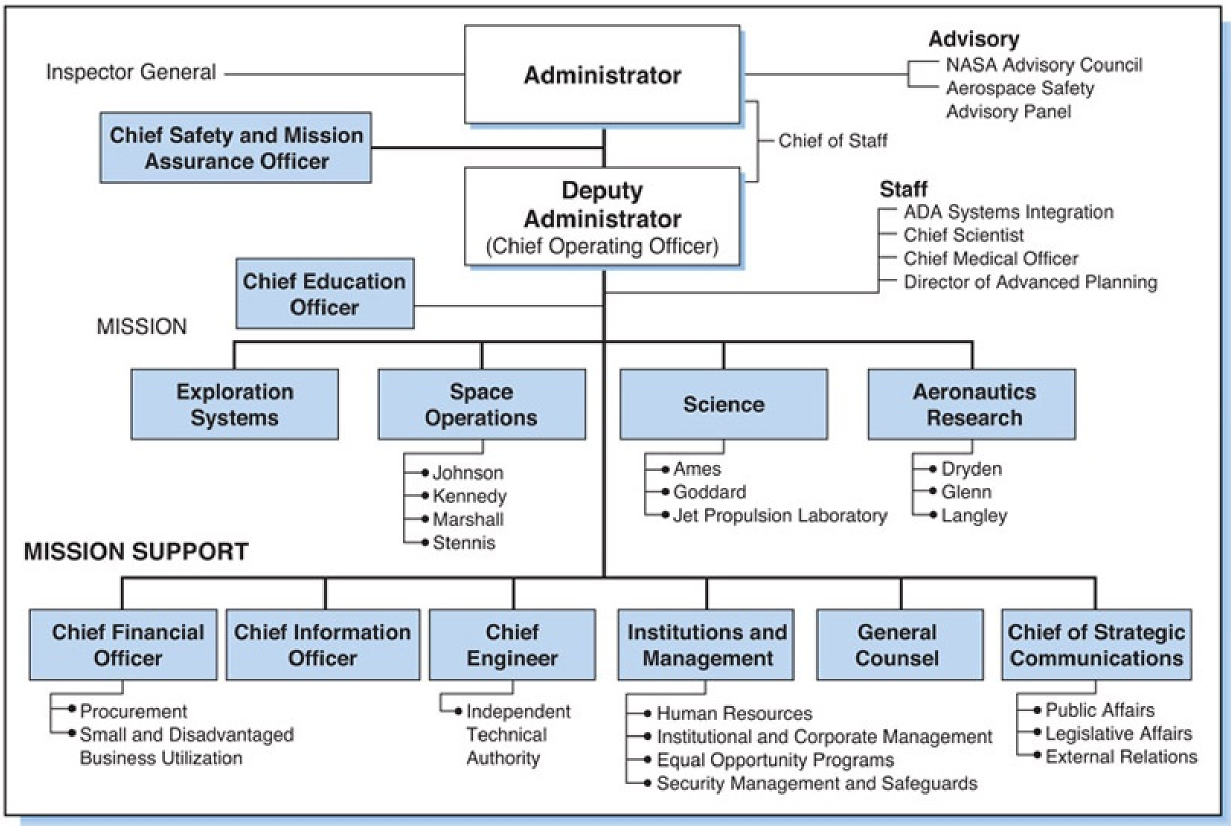


Figure 1.3 NASA program and organization chart.

Each project manager has a counterpart in Washington, the *program manager*, who is the senior NASA official responsible for developing and administering headquarter's guidelines and controls with respect to a given project. She must fight the battles for resource allocation within headquarters, work with all organizations participating in the project, relate the project to NASA's broader goals, and testify to or justify authorizations from Congress or the President. The success of a project depends on the project and program managers working together and the quality of their relationship. An example is Case 17.4, the Mercury Exploration Program.

See Chapter 17

1.12 Miscellaneous Projects

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

Maintenance

All major facilities and machines require maintenance work that takes on project proportions—everything from small repairs and preventative maintenance jobs to scheduled shutdowns of big facilities like chemical plants and power generating plants. Airplanes are removed from service after a certain number of flight hours, stripped down to the component level, inspected, and portions replaced, repaired, or rebuilt. Projects like this put facilities and equipment out of operation and require project management to do quality work, fast.

Events

Mentioned in the Introduction are fundraising and political campaigns, and even weddings. Such projects if small can be handled without project management, but as they grow in size and complexity so do the merits of using project management. Many major sports events like the Olympic Games certainly need project managers, but so do smaller ones like league championships. Case 9.2, the FIFA 2010 World Cup, is an example.

Implementation of Change

Any form of large-scale change effort is a project. Examples include implementing a new corporate strategy, upgrading the safety and health provisions in the workplace, and establishing a new business unit. Applications of project management to such projects are illustrated throughout this book; see for example: Case 1.2: Flexible Benefits System, and Case 3.1: West Coast University Medical Center.

1.13 Summary

The most important 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 in particular the organization, systems, and procedures to 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 a variety of forms such as pure project, matrix, and program management forms. Consumer-oriented firms use new-venture and product-management forms that are similar to basic project management. Project management is applied in much the same way in commercial, nonprofit, government, and military 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 the systems philosophy and methodologies that underlie much of project management theory and practice.

Review Questions

1. Making a film and carrying out a space mission are both expensive projects conducted by teams and subject to budgetary and schedule constraints. The technical expertise for landing a spacecraft on a planet is similar to that required to create the illusion of a spacecraft landing in a motion picture. Use the NTCP model described in the Introduction to indicate ways in which the two project types differ.
2. Describe five functions of management. Are any of these not performed by managers? How do you think each of these functions comes into play in the course of 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: pure project, matrix, program, new venture, product, and portfolio. Give at least one illustration of an organization where each one is used.
10. What are some of the problems of being a project leader in commercial,

government, and military projects? Where do organizations in these environments get project leaders?

11. 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?
12. 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.

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, new venture, 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 have an effect on how projects are managed?

Case 1.1 Disaster Recovery at Marshall Field's[18]

Early one morning basements in Chicago's downtown central business district began to flood. A hole the size of an automobile had developed between the river and an adjacent abandoned tunnel. The tunnel, built in the early 1900s for transporting coal, runs throughout the downtown area. When the tunnel flooded, so did the basements of buildings connected to it, some 272 in all, 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 the manager of maintenance, 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 take over coordinating the team's efforts. The team's goal was to ensure the safety of employees and customers, minimize flood damage, and resume normal operations as soon as possible. They hoped to reopen the store to customers in a week.

An attempt was made to pump the water out; however, as long as the tunnel hole remained unrepaired, the Chicago River 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 of Chicago and OSHA (Occupational Safety and Health Administration) gave approval to reopen the store.

After water was drained from Marshall Field's basements, damaged merchandise was removed and sold to a salvager. The second basement had to be gutted to assure removal of contaminants. 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. Throughout the ordeal, 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 disbanded as the store recovered.

This case illustrates crisis management, an important element of which is having a team that can move fast to minimize losses and quickly recover damages. At the beginning of a disaster there is little time to plan, though 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 Center[19]

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 multifunctional team of 13 members was formed with representatives from four departments—Human Resources (HR), Financial Systems (FS), and Information Services (IS)—and six technical experts from the consulting firm of Hun and Bar Software (HBS).

Early in the project a workshop was held with participants from Shah Alam and HBS to clarify and finalize project objectives and develop a project plan, milestones, and schedule. Project completion was set at 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 on-line, tested, and approved; HR workers had to be trained how to operate the system and load existing employee data; all Shah Alam employees had to be educated about and enrolled in the new benefits process; and the enrollment data had to be entered in the system.

The director of FS was chosen to oversee the project. She had the technical background and had previously worked in the IS group in implementing Shah Alam's patient care information system; everyone on the team approved of her appointment as project leader. 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.

Members of the Shah Alam team worked on the project on a part-time basis, spending roughly half their time on the project and the other half on their normal daily duties. The project manager and team leaders also worked part-time on the project, although each gave the project priority. Shah Alam's senior management had made it clear that meeting project requirements and time deadlines was imperative. The project manager was given 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 closely resemble?
2. The project manager is also the director of FS, one of several departments that will 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 the success of this project depends on the performance of team members who are not employed by Shah Alam, namely the HBS consultants. They must develop the entire hardware/software benefits system. Why was an outside firm likely chosen for such an important part of the project? What difficulties might this pose to the project manager in meeting project goals?

Endnotes

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. 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.

4. Cleland and King, *Systems Analysis and Project Management*, p. 259.

5. Based upon Hofer W. Lady Liberty's business army. *Nation's Business* July; 1983: 18–28.

6. Sharad D. Management by projects, an ideological breakthrough. *Project Management Journal* March; 1986: 61–63.

7. Adams J., Barndt S. and Martin M. *Managing by Project Management.* Dayton, OH: Universal Technology; 1979, pp. 12–13.

8. Project Management Institute. *The Standard for Program Management*, 3rd edn. Newton Square, PA: PMI; 2013; Office of Government Commerce. *Managing Successful Programmes (MSP).* UK: The Stationery Office; 2007.

9. Project Management Institute. *The Standard for Portfolio Management*, 3rd edn. Newton Square, PA: PMI; 2013; Office of Government Commerce. *Management of Portfolios.* UK: The Stationery Office; 2011.

10. This section is adapted from Roman D. *Managing Projects: A Systems Approach.* New York, NY: Elsevier, 1986; pp. 426–429, with the permission of the publisher.

11. This and examples in later chapters of SpaceShipOne illustrate concepts. Much factual information about the project and the systems is available from published sources, but design and development information of the systems 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 contributed by Daniel Molson, Mike Billish, May Cumba, Jesper Larson, Anne Lanagan, Madeleine Pember, and Diane Petrozzo.

15. Disaster Response. Lesson 7: Emergency Operations Support. University of Wisconsin, Disaster Management Center, http://dmc.engr.wisc.edu/courses/response/BB08-07.html.

16. 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

17. 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.

18. Information about this case contributed by Jennifer Koziol, Sussan Arias, Linda Clausen, Gilbert Rogers, and Nidia Sakac.

19. Information about this case 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 *system* of people, equipment, materials, and facilities organized and managed to achieve a goal. Much of the established theory and practice about what it takes to organize and coordinate a project comes from a perspective called the "systems approach." At the same time, work done in projects is often done for the purpose of *creating* systems, and such projects commonly employ methodologies such as "systems analysis" and "systems engineering." This chapter introduces concepts that form the basis for project management and the systems methodologies used in technical projects.

2.1 Systems and Systems Thinking

By definition, the term *system* refers to "an organized or complex whole; *an assemblage of things or parts interacting in a coordinated way*." The parts could be players on a football team, keys on a keyboard, or components in a machine. The parts can be physical entities or they can be abstract or conceptual, such as words in a language or steps in a procedure. Beyond being an "assemblage of parts," however, a system has three other features:[1]

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

The first feature means that, in a system, the whole is more than the sum of the parts. The human body, for example, is comprised of separate components—the liver, brain, heart, nerve fibers, and so on; yet if any of these are removed from the body, both they and the body will change. Parts of the body cannot live outside the body, and without the parts, the body cannot live either. The idea of the parts affecting the whole and vice versa is central to systems thinking.

The second feature of systems is that the parts work in combination to *do* something. What they do can usually be observed in the outputs of the system or the way the system converts inputs to outputs (although sometimes that conversion process may be quite obscure). In a human-made system the parts are designed to interact to achieve some purpose or goal.

Third, systems are conceived by the people looking at them, which means they exist in the eye (or mind) of the beholder.[2] What this says is that the conception of a system can be altered to suit one's purpose. For example, in diagnosing the illness in a patient, a doctor may see the entire human body as "the system." The doctor may send the patient to a specialist, who sees only the digestive tract as "the system." If the diagnosis is food poisoning and the patient files suit, her attorney might expand the view of "the system" to include the restaurant where the patient last ate.

Systems thinking is a particular way of viewing the world, its key feature being a focus on “the big picture—the whole system or organism,” rather than just the parts of the system. Systems thinkers also look at the parts and try to understand the relationships among them, but they *always* step back to see how the parts fit into the whole.[3] Systems thinking means being able to perceive the “system” in a situation; to take a seemingly confused, chaotic situation and perceive some 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 large projects.

Although project managers must be familiar with and able to coordinate the individual parts of the project, most responsibility for each of those parts is delegated to the managers and technicians who specialize in them. Project managers are concerned with the “big picture”—the whole project with its goals, work tasks, and the people involved; as such, they must be 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* 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.

A project can be conceptualized as a system that exists for the purpose of creating a human-made system. 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," "select prime contractors," "train crew," "launch components into orbit," "assemble components," "do project for cost $15 billion," 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 overall goals and objectives.

Elements and Subsystems

Any system can be broken down into smaller parts. These parts in combination form "the assemblage of parts" that constitutes the system. The smallest part of a system is an *element*. A system can also be broken down into parts that are themselves systems, called *subsystems*. A subsystem is a system that functions as a component 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 becomes a subsystem 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 end-item being produced by the project.

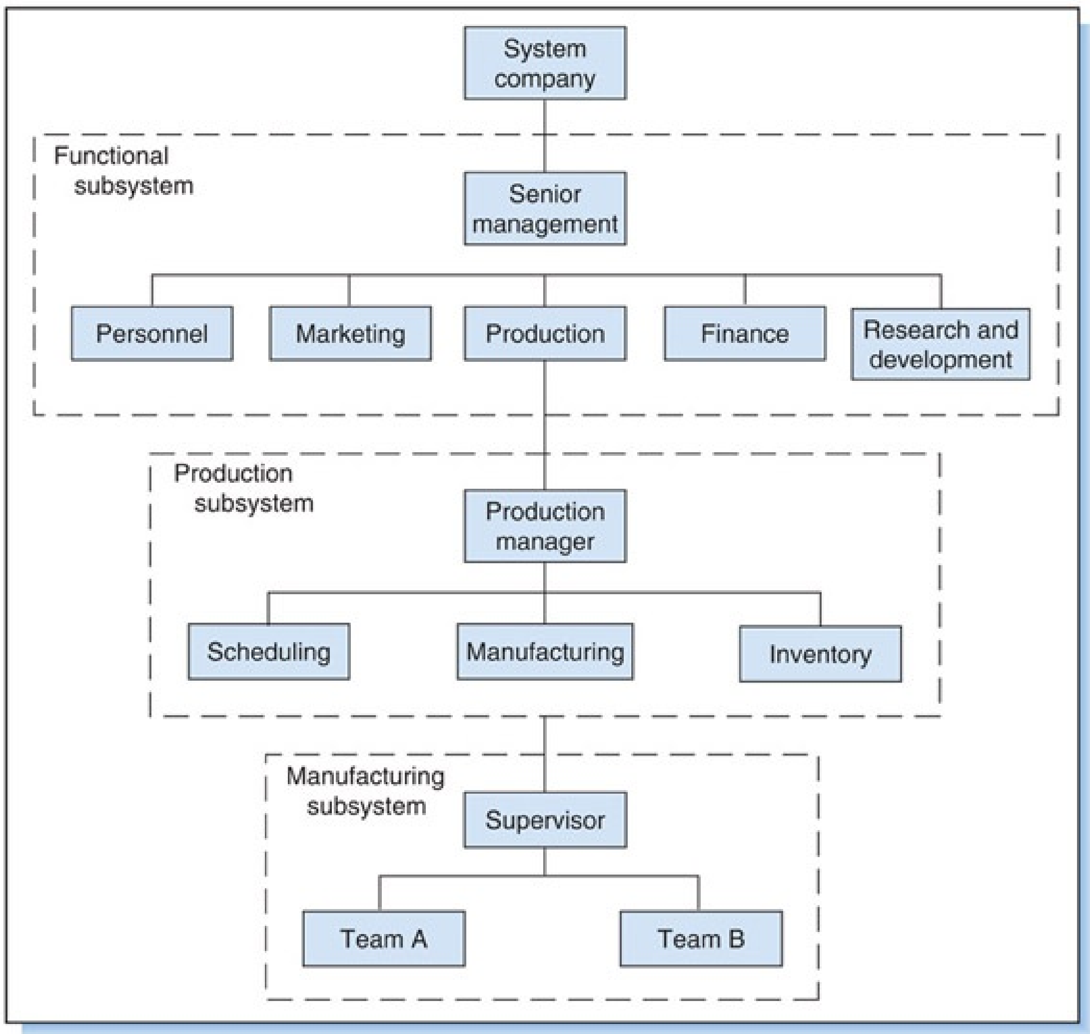


Figure 2.1 A company portrayed in terms of systems, subsystems, and elements.

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, the attributes are designed into the system so the system will perform as required. Often, the attributes of a system and its components are monitored to keep track of the system's behavior and performance. In a project, time and cost are universal attributes of most of its elements, 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 behavior or outcome of the system. In human-made systems it usually refers to things over which system designers and managers have no control. The environment can include, for example, the community or society we live in, the air we breathe, or the people with whom we associate—although it is not necessarily any of these. A system is separated from its environment by a *boundary*. In many systems the boundary is somewhat obscure and it is difficult to separate the system from its environment. To determine what the environment is, ask the 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 but "yes" to the second, then "it" is part of the environment. The following table shows how to distinguish a system from its environment:

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

"Irrelevant environment" includes all things that do not influence the system and that do not matter. To a project manager the planet Jupiter is in the irrelevant environment—unless her project is to send a space probe there, in which case Jupiter is relevant and, hence, 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 have to be lived with.

System Structure

Elements and subsystems are linked together by relationships. The form taken by the relationships is referred to as the structure of the system. The functioning and effectiveness of a system is largely determined by the "appropriateness" of the structure to the system's objective or purpose. Most complex systems have hierarchical structures consisting of organized levels of sub-elements within elements, elements within subsystems, and so on.

Figure 2.2 is an example of a hierarchical structure. It shows a project as a hierarchy of tasks and responsibilities. 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 structure is called a *work breakdown structure* and is explained more in Chapter 5.

See Chapter 5

Inputs, Process, Outputs, Interfaces

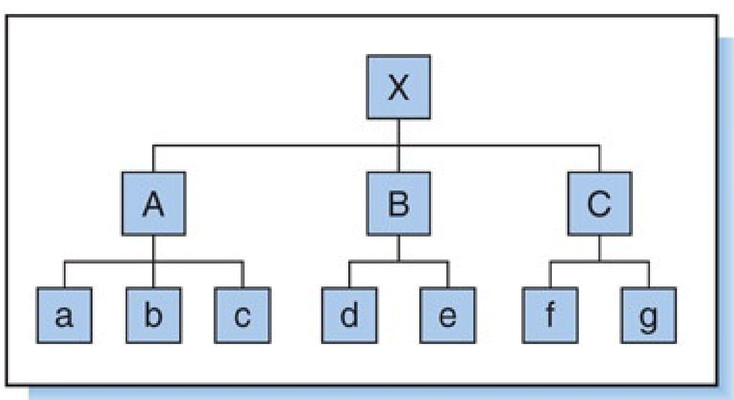


Figure 2.2 One way to conceptualize project structure.

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

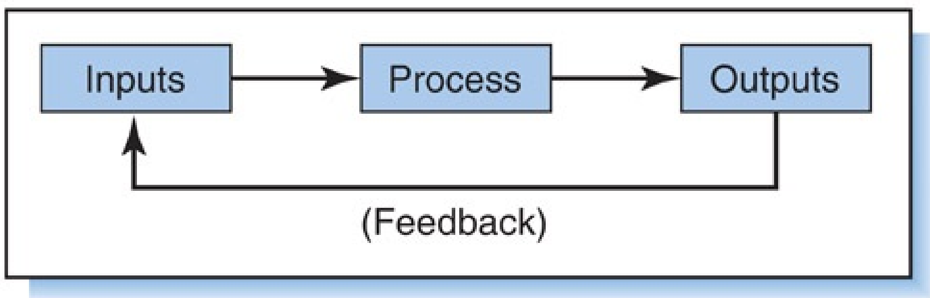


Figure 2.3 Input–process–output relationship.

Inputs are the raw materials, resources, or steps 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 the means by which the system physically converts or transforms inputs into outputs. An important aspect of system design is to create a process that effectively produces the desired outputs and meets system objectives, yet minimizes consumption of inputs and production of wasteful outputs.

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 where the output of one becomes the inputs 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.

In human-made systems, and especially in projects, the objectives of the subsystems are sometimes in *conflict*, which reduces the chances that they or the goal of the overall system will ever be realized. Removing conflict from objectives to enable meeting the goal of the overall system is called integration.

Integration

For a system to achieve its goal, all of its elements, the “assemblage of parts,” must work in unison. Designing, implementing, and operating a system that achieves its prespecified 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 project goals. In technological projects, project management also addresses the integration of the phys cal components and modules that compose the project end-item. The subject of systems integration is covered in Chapter 14.

See Chapter 14

open Systems and Closed Systems

Systems can be classified as *closed* or *open*. A closed system is one that is viewed as self-contained, and “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 systems that must be adaptable to the environment, including projects, must be treated as open systems.

Organizations and Environment[4]

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

As an open system, any organization must choose goals and conduct its operations so as 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. integrate these 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, be able to guide the project to its goal. A project that is predominantly influenced by divergent forces in the environment will be difficult to control and likely to fail.

Natural Systems and Human-Made Systems

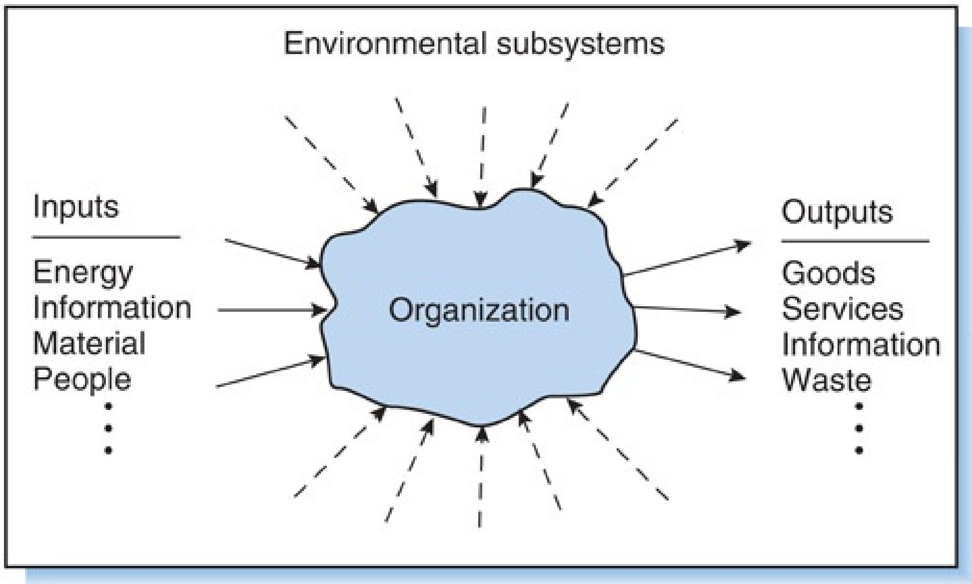


Figure 2.4 Organization as an input–output system.

Systems can also be classified as either *natural* or *human-made*. Natural systems originated by natural processes (e.g. organisms and planetary systems). Human-made systems are designed and operated bypeople (e.g. communication systems and human organizations). Projects are human-made systems (organizations) created for the purpose of creating other human-made systems.

Natural systems can be altered by or become intertwined with human-made systems. An example is the alteration of a river system and formation of a lake by building a dam; another is the alteration of the composition of atmosphere and ecosystem through CO_2 introduced by human-made machines.

Human-made systems are embedded in and utilize inputs from natural systems, and both systems interact in important and significant ways. In recent years the appearance of large-scale human-made systems has had a significant, mostly undesirable, impact on the natural world. Examples include global warming, acid rain, and toxic contamination of water systems. Such consequences, referred to as “side effects,” arise largely because system designers and users fail to consider (or chose to ignore) the impacts of human-made systems on the natural environment.

2.3 Systems Approach

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

Systems Approach Framework

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

Managers who adopt the systems approach recognize the multitude of "elements" in the systems they manage and the problems they need to solve, 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 full magnitude of a problem and anticipate 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 ultimate goal; it allows focus on the parts of the system, but only in regard to the parts' 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 take action regarding 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 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. Similarly, enacting laws is one way to reduce air pollution, but laws that restrict industry can damage local economies. Every problem is inextricably united 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 for 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 in 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 Appraisal[8]

The systems approach is a methodology for solving problems and managing systems. By its holistic nature, it 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 mandates hardheaded thinking about the *goals* and *objectives* of the system and real ways to measure them. Project management uses this kind of thinking: it begins with the mission or objectives of the system and, thereafter, organizes and directs all subsequent work to achieve those objectives. The stated objective 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, groups, or persons 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 or innovations 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 also 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. When resources are depleted they 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. In a project there are actually *two* systems, the one *being produced* by the project (this is the project end result or end-item) and the one *producing* the end-item (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 4.

See Chapter 4

The output of a system results not just from the individual elements, but from the way the elements interact. Thus, designing a new system or resolving problems in a human-made or natural system requires understanding *the way the elements interface and 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.

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

The preceding concepts are not necessarily addressed in the sequence they are listed. In actuality 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, objectives, and criteria? What are the elements? What are the relationships among them? What functions should each perform? What are the resources? What are the tradeoffs among resources?

Systems Models

Systems thinkers use "models" to help understand systems and assess alternative plans and solutions against goals. A model is a simplified representation of a system; it 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 exterior components) and excludes others (interior components and crew). Another kind of model is a *conceptual model*; it depicts the elements, structure, and flows 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 make 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. Engineers use model airplanes in wind tunnel tests, for instance, to try out design alternatives and measure the effect of different design parameters on airplane performance. A good model allows designers and analysts to ask "what if" questions and explore the effects of altering the various inputs. It takes into account the requirements, relevant 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 9.

See Chapter 9

Systems Life Cycles

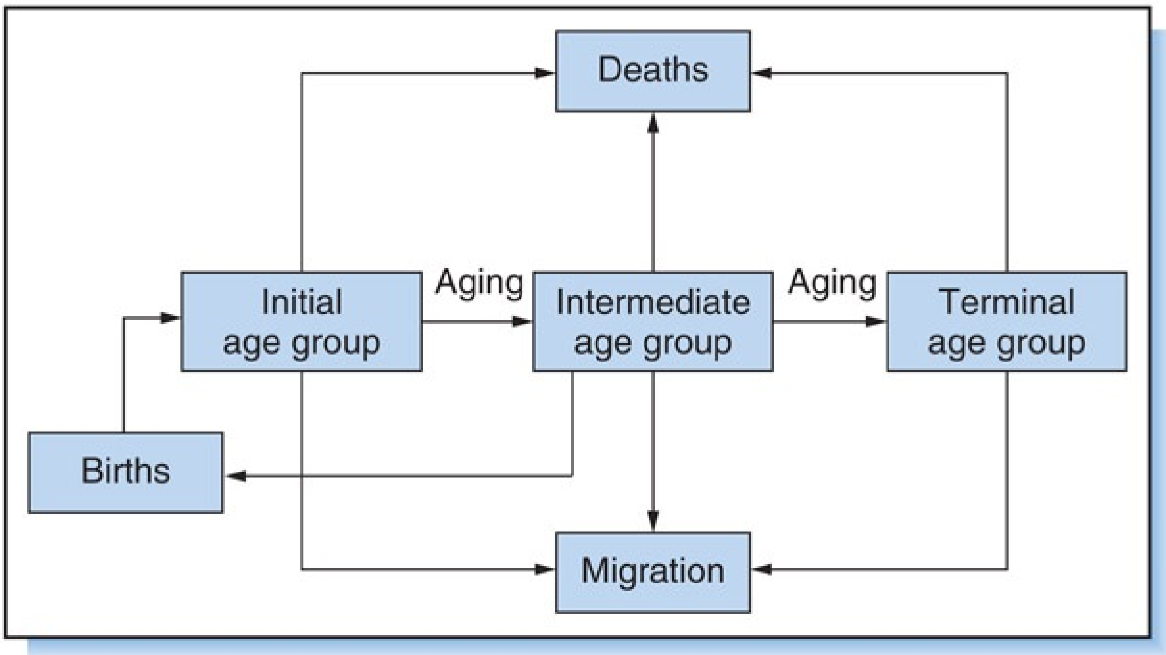


Figure 2.5 A generalized population sector model.

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 cycles of evolution. One basic 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" or event. 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 conception, design, development, production, launch into the market, capture of market share, then decline and discontinuation. Products such as cell phones may have life cycles only months-long; others (Kool-Aid and Levi’s jeans) have decades-long cycles.[11] As mentioned in Chapter 1, virtually all human-made systemsstart out as projects, and most projects also follow a cycle called the *project life cycle*.[12] This is discussed in Chapter 3.

See Chapter 3

2.4 Systems Engineering

Systems engineering, an application of the systems approach, is defined as "the science of designing complex systems in their totality to insure that the component subsystems making up the system are designed, fitted together, checked and operated in the most efficient way."[13] It refers to the conception, design, and development of a complex system wherein the *components themselves* must be designed, developed, and integrated together to fulfill the system objectives. Systems engineering 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.

All Systems Go

An example of systems engineering is the design and operation of a space vehicle. The expression "all systems go," popularized during the early US space flights, means that the overall system of millions of components that make up the space vehicle and its support systems, and the hundreds of people in its technical and management teams, is ready to "go" to achieve the objectives of the mission.

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 detailed, focused requirements, and have designed the components and subsystems that meet the requirements. They must then have built and combined the components into subsystems, and the subsystem into the total system 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.

Systems engineering applies to any system (hardware or software) that must be developed (perhaps from scratch), implemented, and operated to fulfill some immediate or ongoing future purpose. Examples can readily be found in the design and implementation of local, national, and global systems for communication, transportation, water purification and supply, power generation and transmission, research, and defense.

Overview[14]

Systems engineering can be described in terms of the three dimensions illustrated in Figure 2.6.

First, it is a multidisciplinary effort. Systems engineers (parties responsible for oversight of designing and building the system) work with the system's stakeholders to determine their needs and what the system must do to fulfill them. A stakeholder is any individual or group that affects or is affected by the system; the 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 requirements that specify *what* the system will do. The practice of involving key stakeholders in the early phases of the system conception and development is called "concurrent engineering," discussed in Chapters 4 and 14.

See Chapter 4 and Chapter 14

Second, systems engineering addresses *all aspects* 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. This aspect of systems engineering focuses on *how* the system must function to meet the requirements. Of course, none of the elements and subsystems function independently. All rely on the outputs of other functions and, in turn, provide inputs to still others; in a word, they *interface*. Systems engineering addresses how they should interface and the necessary interactions between them.

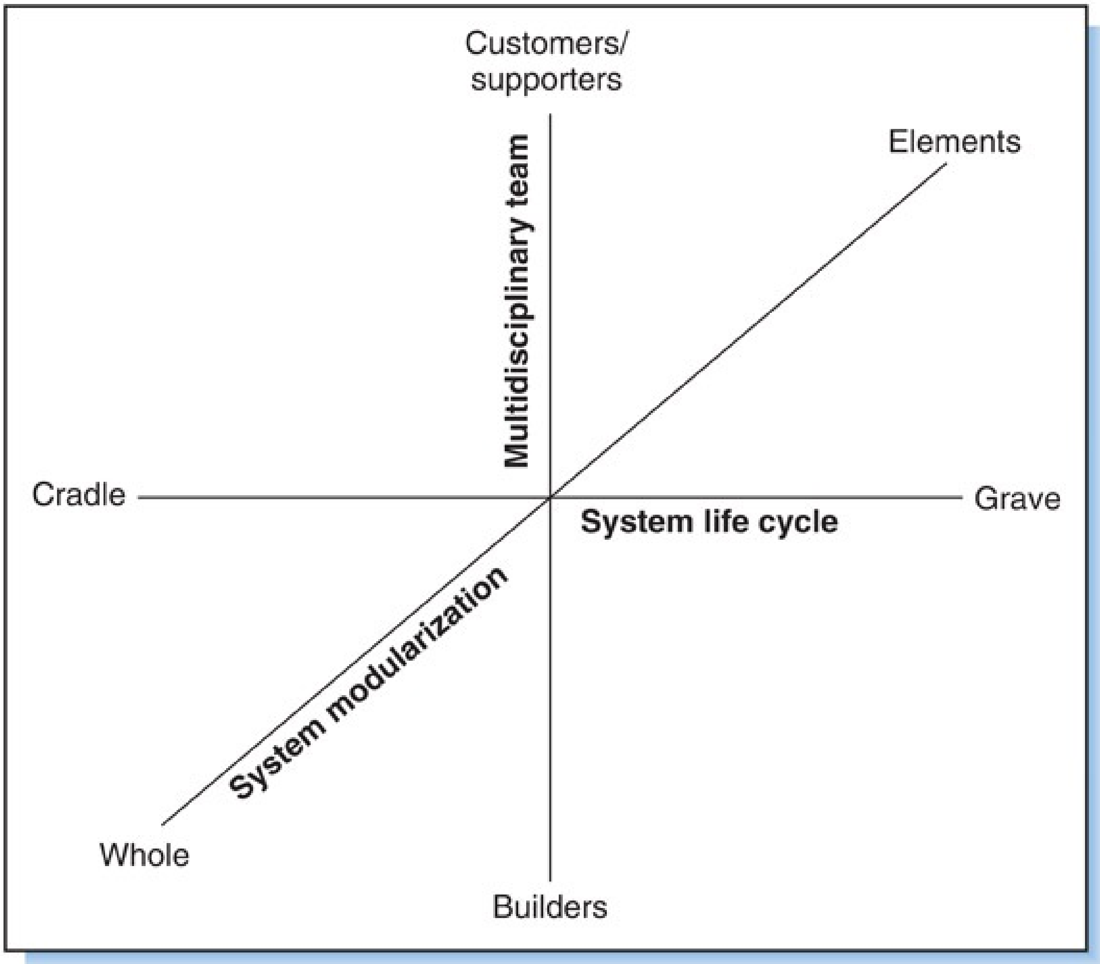


Figure 2.6 Dimensions of systems engineering.

Finally, in the creation and development of the system, systems engineering also takes into account how the system will be produced, operated, maintained, and finally disposed of—the system's full life cycle, cradle to grave. This helps assure that the system will be economical to develop, build, operate, and maintain, and friendly to users and the environment. A multidisciplinary team approach that involves all the system's stakeholders promotes this life cycle kind of thinking.

Once systems engineers have learned what stakeholders want and defined the objectives and requirements of the system, they then look for ways to meet the requirements. This involves research, analysis, and studies of alternative approaches to the system design, and the estimated costs, schedules, risks, and benefits associated with each. 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 so cripples the resulting system if done wrong. No other part is more difficult to rectify later."[15]

Example: Advanced Automation System[16]

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 the multidisciplinary teams sometimes include behaviorists and psychologists. Developing the flight deck for a commercial aircraft, for example, would 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 Cycle[17]

The process of creating a system concept is a series of steps to define the subsystems and elements that will comprise the system. The process is illustrated by Forsberg and Mooz's "V-model" in Figure 2.7.[18] It involves iterative cycles of (1) *top-down analysis* of details (i.e. decomposing the system into smaller parts),

(2) *bottom-up synthesis* (building up and integrating the parts into successively larger parts), and (3) *evaluation* (checking to see that results meet requirements).

Systems are designed and assembled from subsystems that themselves are designed and assembled from subsystems, and so on. The practice, called *modularization*, is what makes the design, assembly, and operation of complex systems feasible and practical. Herbert Simon gives the example of a watch-maker 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.[19]

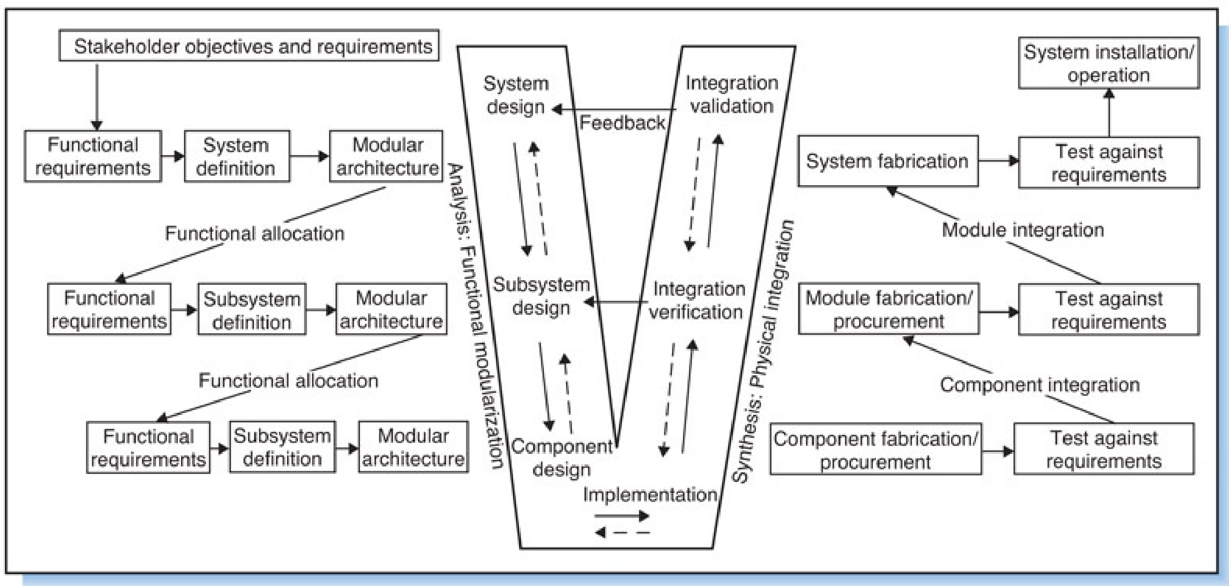


Figure 2.7 Forsberg and Mooz's V-model.

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

The top-down stroke of the V represents subdividing the functions of the system into subfunctions and requirements. At each lower level the process of working with customers to define requirements repeats, except the "customer" becomes the function at the next higher level and the question becomes: "What must the lower-level functions do to meet the requirements of the higher-level function?" In this way, requirements are defined for functions at all levels.

Systems are designed by designing subsystems or modules that each performs a necessary function of the system. Functions are the means by which 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 is almost completely modularized: it has a processor and controllers, drives, and peripheral devices that each performs a specialized function such as data processing, data storage, and input/output processing.

The way in which system functions are grouped 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; the visibly familiar modules of engines, wings, and fuselage, respectively, serve these functions. But each function is itself a composite of several subfunctions, hence each module is comprised of several submodules. A wing, for example, is subdivided into ailerons, flaps, spoilers, etc., each of which performs a specific aerodynamic function.

The bottom-up stroke of the V represents assessing "design alternatives" to satisfy requirements, implementing design decisions, converting designs into physical parts, integrating the parts, and verifying that the integrated parts meet the requirements. Design alternatives are the potential solutions to problems; they are the courses of action for meeting requirements; ultimately they show up in the final system as pieces of hardware and software. The chosen alternatives result in procuring or designing and building component parts. Components are checked individually and then assembled into modules; modules are tested, then combined with others and tested again.

If tests reveal that parts or modules are not meeting requirements, then the process returns to the downstroke of the V 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 top-down/bottom-up stroke; 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 considering alternative solutions; they combine knowledge from experts in disparate areas and can generate alternatives that transcend any one person's or field's area of expertise.

The design and development of complex technical systems can be vexing, but systems engineering offers a way to do it. In practice, systems engineering follows a process very 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, systems engineering applies more specifically to complex, usually technical, projects. Steps and tools that characterize systems engineering are covered in Appendix A to Chapter 4.

See Chapter 4

2.5 Project Management: A Systems Approach[20]

Project management is a systems approach to management: it is total-system oriented and emphasizes achievement of the *overall* mission and objectives of the project; it emphasizes decisions that optimize the *overall project* rather than the elements or subsystems of the project; and it recognizes interaction and synergy among elements of the project—that outputs from one element provide inputs to other elements. The project manager recognizes interactions and interdependencies between project elements and with the environment, and he works to ensure that organizations, responsibilities, knowledge, and data are integrated toward achieving overall project objectives. This contrasts with the more typical management view, which is to focus narrowly on individual functions and tasks and on the performance of individual departments, even if at the expense of the total organization.

In *Winning at Project Management*, author Robert Gilbreath[21] describes the "right" way to visualize a project. From an outsider's perspective, he says, a project may look like something with no separate discernable parts, like a barrel containing thousands of earthworms. Obviously, if you have to manage the project such a perspective is not very useful and you need another perspective, one that involves subdividing the continuum into a collection of elements and defining the characteristics of each.[22] Good project managers, says Gilbreath, conceptually subdivide the project into pieces and make sure each piece is well managed. The project manager knows all the pieces of the project and how each impacts the others and the overall project.

Gilbreath discusses 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 is essential for it enables the project manager to direct the project toward its goals and not get hung-up with the pieces.[23] In other words, the project manager needs to be a big-picture person who knows how to balance focus between technical elements of the project and the administrative aspects of schedules, budgets, and human relations. 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. Whether or not you call it the “systems approach” the point is, in managing a project it helps to think of a project 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 discern a degree of order and structure in a seemingly confused or chaotic situation. Systems thinking includes the "systems approach," which is a way of conceptualizing physical entities and addressing problems. The principle components of the systems approach are (1) the *objectives* and the *performance criteria* of the system, (2) the system *environment* and *constraints*, (3) the *resources* of the system, (4) the *elements* of the system, their functions, attributes, and performance measures, (5) the *interaction* among the elements, and (6) the *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 has given you an overview of project management. Projects are of finite duration—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 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 relevancy 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 process used? Is there a section, department, or task in the project called systems engineering? If so, elaborate. Are there functions or phases of the project that seem to resemble the systems engineering process?
5. As described in this chapter, besides the main end-item or operating system (i.e., the output objective of the project), systems engineering 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 is located 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 primarily 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 is able to 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 also is 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 or measures of performance to be used 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 Life and Death of an Aircraft

Development Project

Law and Callon[24] describe the history of a large British aerospace project in terms of two entities: the global system and the project itself. The *global system* comprised parties and organizations *outside* the project that had a stake in the project; the *project* comprised everything *within* the project, including all work and the organizations contracted to do it.

The Global System

The principle stakeholders in the global system were:

1. The Royal Air Force (RAF), which initiated the project with a request for a new supersonic aircraft with short take-off capability. The aircraft would be a "tactical strike and reconnaissance fighter" called TSR.
2. Ministry of Defence (MOD), which wanted an aircraft that would best fit the nation's current overall defense needs.
3. The Treasury, which wanted an inexpensive aircraft that would have market appeal for sale outside the UK, such as to the Royal Australian Air Force (RAAF).
4. The Royal Navy, which wanted to buy *a different* aircraft but was under pressure by MOD to buy the TSR.
5. The Ministry of Supply (MOS), which wanted an aircraft that would be produced by a consortium of several UK airframe and engine manufacturers.

As typical of most projects, each stakeholder in the global system conceptualized the project differently: to the RAF and MOD it would yield an aircraft for a specific mission; to the Treasury it would fit the defense budget and generate revenue; to the Navy it was a competitive threat to the aircraft they really wanted; and to the MOS it was an instrument of industrial policy. The parties had different reasons for contributing resources and support: some were economic (in return for funds, an aircraft would be built); some political (in return for a demonstrated need, objections of the Navy would be overruled); some technical (in return for engineering and technical effort, the aircraft would meet RAF performance requirements); and some industrial (in exchange for contracts, the aircraft industry would be consolidated).

The Project

The Treasury would not approve project funding until the aircraft's basic design, manufacturer, cost, and delivery date were defined. The RAF and MOD sent requests to the aircraft industry for design ideas and selected two manufacturers; Vickers Corp. and English Electric (EE). They favored Vickers for its integration capability (combining aircraft, engine, armaments, and support equipment into a single weapons package), but they also liked EE for its design experience with supersonic aircraft. So they decided to contract with both companies and adopt a design that would utilize features from both. The idea was approved by all other parties in the global system and funding for the project was released.

The project grew as Vickers and EE hired subcontractors and expanded their teams for design, production, and management. The two companies and several other contractors merged to form a single new organization called the British Aircraft Corporation (BAC).

Relationships Between the Global System and the Project

As the project grew so did the problems between it and the global system. MOS wanted centralized control over all aspects of the project and all transactions between the project and stakeholders in the global system. Although BAC was the prime contractor and ostensibly responsible for managing the project, MOS would not confer upon it the necessary management authority. Rather, MOS formed a series of committees with members from the global system and gave them primary responsibility to manage the project. This led to serious problems:

1. The committees were allowed to make or veto important project-related decisions. They, not BAC, awarded important contracts; when the RAF wanted to change its requirements, it consulted with the committees, not with BAC.
2. The committees often lacked sufficient information or knowledge. Technical committees made decisions without regard to costs; cost committees made decisions without regard to technical realities. Decisions focused on particular aspects of the project; seldom did they account for impacts on other parts of the project or the project as a whole.

Distrust grew between BAC and MOS; neither was able to effectively integrate the resources, information, and decisions flowing between parties in the project and the global system. Subcontractors became difficult to control. Many ignored BAC and worked only with MOS and RAF to get favorable treatment.

Global System Reshaped

Everyone knew the project was in trouble. Project costs doubled. One of the test engines exploded and the RAF recognized it would take years to understand the cause. In addition, the RAAF announced that it would not order the TSR but instead was buying the US-built F-111. Opposition to the project grew and in the upcoming general election the Labour Party promised that if elected it would review the project. When the Labour Party won, it immediately began an assessment of the project, which included comparing the TSR to the F-111—considered by now an alternative to the TSR. As cost overruns and schedule delays continued, MOS slowly withdrew support. Then the RAF withdrew its support when it discovered that the F-111, which was already in production, would meet all of its requirements. The project was canceled.

Questions

1. In this case history, what is the "system"? What are its elements? What is the "environment"? What are the elements of the environment?
2. Describe the interaction between the system and its environment.
3. Do you feel that important decisions made in this project represent "system thinking"? Explain.
4. Comment on the concept of "integration" in the project. How were aspects of the project integrated or not integrated?
5. What are the main factors that contributed to cancellation of the project? Which of these factors would you characterize as project management?

Case 2.3 Jubilee Line Extension Project[25]

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 9.1 and 15.3).

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:[26]

- 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 faced political pressure to complete JLE in time to serve as a main transport link to the Millennium Dome, which was then in-construction. Consequently it set an overly ambitious project deadline of 53 months.

The project was managed through the project director, project manager, and a large 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. Substantial 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.* Despite the fact that 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 2.4 Santa Clara County Traffic Operations System and Signal Coordination Project[27]

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.

Endnotes

1. Naughton J. and Peters G. *Systems Performance: Human Factors and Systems Failures*. Milton Keynes, UK: The Open University; 1976, pp. 8–12.

2. Ibid., 11. 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. Ibid., 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 M.I.T. 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. Ibid., pp. 192–197.

18. 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.

19. Herbert Simon, quoted in Auyang, *Engineering—An Endless Frontier,* p. 194.

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. Ibid., pp. 95–96.

23. Ibid., pp. 98–102.

24. From Law J. and Callon M. The Life and Death of an Aircraft: A Network Analysis of Technical Change. In Bijker W. and Law J. (eds), *Shaping Society/Building Technology.* Cambridge, MA: MIT Press; 1992.

25. Information for this case is derived from three sources, all downloaded Oct. 30, 2014: (1) *Project Profile, Jubilee Line Extension.* Omega Centre, University College London (no date), http://www.omegacentre.bartlett.ucl.ac.uk/studies/cases/pdf/UK_JLE_PROFILE_120909.pdf; (2) *Jubilee Line Extension, London, UK* (no date), http://www.omegacentre.bartlett.ucl.ac.uk/studies/cases/pdf/UK_JLE_2P_080911.pdf; and (3) Systems Engineering Case Study # 8 Jubilee Line Extension, *Systems Engineering in Transportation Projects: A Library of Case Studies,* INCOSE Transportation Working Group, 2013, http://www.incose.org/practice/techactivities/wg/transport/docs/CaseStudyLibrary_6_0.pdf.

26. Mitchell, B. *Jubilee Line Extension: From Conception to Completion.* London, UK: Thomas Telford Publishing; 2003.

27. 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, 2013, http://www.omegacentre.bartlett.ucl.ac.uk/studies/cases/pdf/UK_JLE_2P_080911.pdf; downloaded Oct. 30, 2014.

Part II
Project Life Cycle

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 two chapters in this section introduce the systems development cycle and describe its first two phases, conception and definition.

Chapter 3
Project Life Cycle and Project Conception

There is … a time to be born, and a time to die; a time to plant, and a time to reap; a time to kill, and a time to heal; a time to break down, and a time to build up …

—Ecclesiastes 3:1

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. The next chapter covers the second phase; subsequent chapters cover the others.

3.1 Project Life Cycle

Systems are dynamic—they change over time. The change tends to follow a distinct pattern that is repeated again and again. 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.

See Chapter 2

Projects follow a cycle called the *project life cycle*. Each project has a starting point and progresses toward a predetermined conclusion; during this time, activity in the project continues to grow, reaches a peak, and then declines—the pattern shown in the lower curve in Figure 3.1 (the upper curves" shows cumulative activity). Activity or effort can be measured in various ways such as the amount of money being spent on the project, number of people working on it, amount of materials being used, and so on.

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

Managing the project life cycle requires special treatment. Unlike non-project, repetitive operations where everything tends to be somewhat familiar and stable, many things in projects—resources, schedules, work tasks, etc.—are 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 tailored to fit each phase and stage of the project life cycle. Unforeseen obstacles can cause missed deadlines, cost overruns, and poor project performance. Managers must try to anticipate the problems, plan for them, and adjust activities and shift resources to mitigate or overcome them.

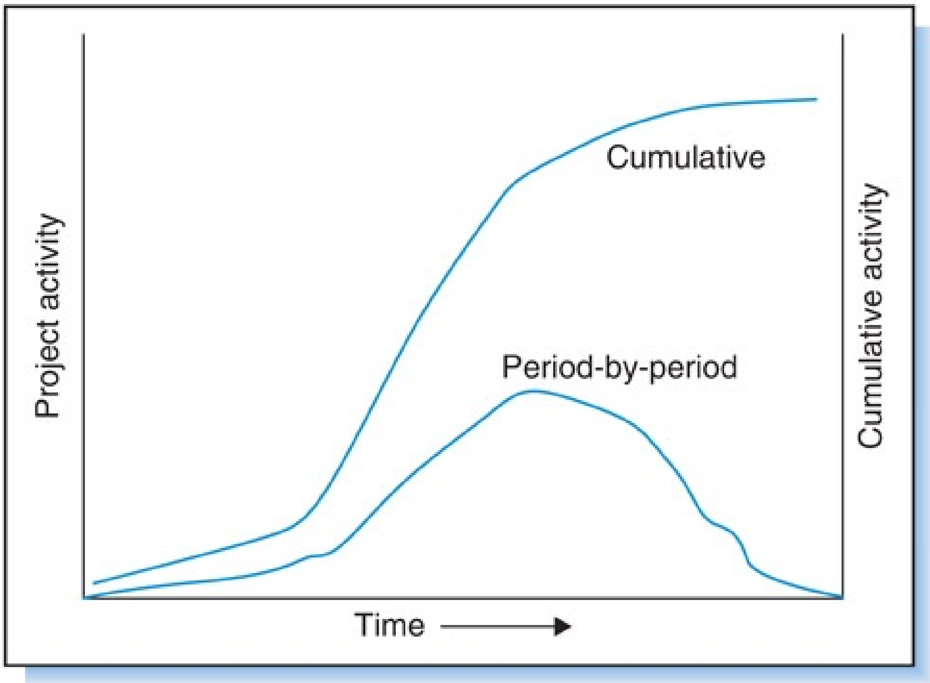


Figure 3.1 Level of effort during the project life cycle.

3.2 Systems Development Cycle

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

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

The project life cycle typically spans Phases A, B, and C—conception, definition, and execution. When Phase C ends, so does the project. At that point the system enters Phase D, operation; the system transits from being the end-item of a project to an operational entity.

Phase A: Conception

Every project is an attempt to solve a problem, and a first step in solving the problem is recognizing that the problem exists. After that, the individual facing the problem—the customer and users—seeks out someone who can help. The steps they take—soliciting contractors who can do the work, evaluating their proposals, and reaching an agreement—are all part of the *procurement management* process.

If the customer organization has an internal group capable of doing the work, it turns to them. If not, it looks to outside contractors, possibly by sending them a formal request for help called a *request for proposal*, or *RFP*. Each contractor examines the customer's problem, objectives 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 presents to the customer a proposed solution (system concept) in a *proposal* or *letter of interest*. The customer then examines the proposals and makes a choice. The result is a formal agreement between the customer and the chosen contractor. Most ideas or proposals never get beyond Phase A; the problems addressed are judged as insignificant, or the proposal as impractical, infeasible, or lacking benefits to justify funding and resources. The few that are approved and reach a contract agreement move on to Phase B.

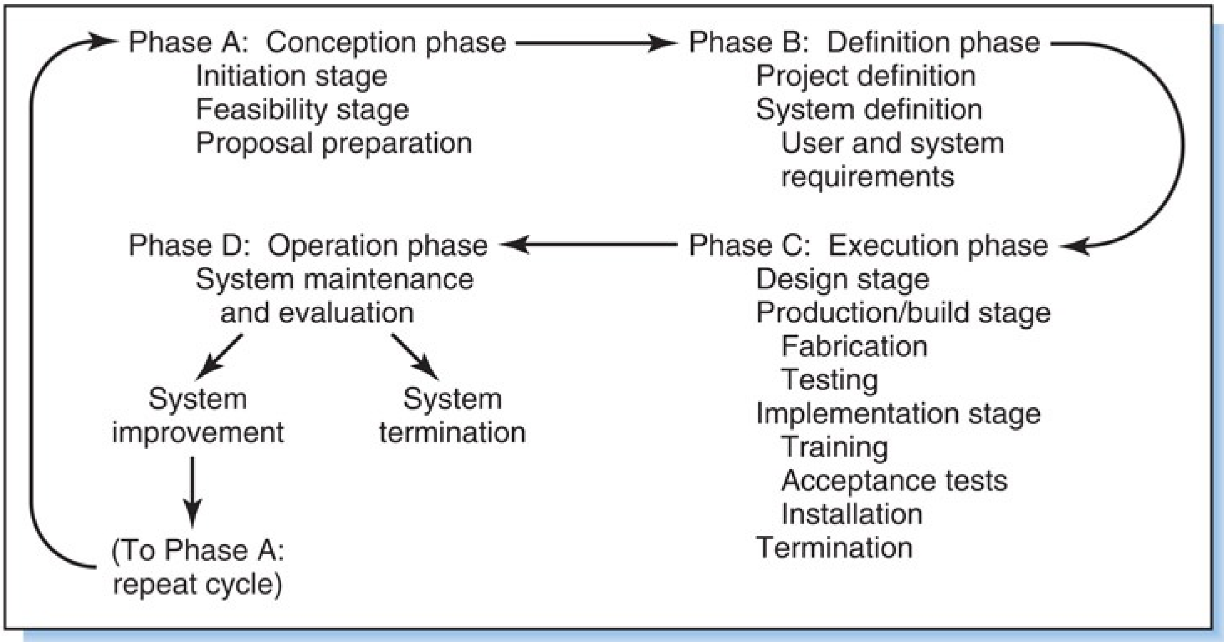


Figure 3.2 Four-phase model and stages of the systems development cycle. The project life cycle is Phases A, B, and C.

Phase B: Definition

Having reached agreement with the customer, the contractor begins a detailed analysis of the system concept, during which it defines requirements the system must fulfill to meet the customer's needs, and the functions and elements the system must possess to meet those requirements. This definition results in 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. Meantime, project management assembles a comprehensive project plan that defines the activities, schedules, budgets, and resources to design, build, and implement the system. Contractor top management reviews the plan for acceptability and then forwards it to the customer who also reviews it for acceptability.

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 the project prior to the work execution in Phase C. FEL is discussed in Chapter 4.

See Chapter 4

Phase C: Execution

The execution phase is when the work specified in the project plan is put to action; it is sometimes referred to as the "acquisition" phase because the user acquires the system at the end of the phase and most system resources are acquired then. The execution phase often includes the stages of "design," "production," and "implementation," referring to the progression through which a system moves from being an idea to a finished, physical end-item. All systems are comprised of elements arranged in some configuration, pattern, or structure, and it is in the *design* stage that the elements and configuration necessary for the system to fulfill requirements are defined. Following design the system enters *production* where it is built as either a single-item or mass-produced item. During the execution phase the system is *implemented*; it is installed in and becomes a part of the user's environment.

Phase D: Operation

In the operation phase the system is deployed; the customer takes over to operate the system and maintain it. For systems such as products and equipment that people use and rely upon daily, Phase D may last for years or decades, in which case the phase includes not only operation and maintenance of the system, but improvement and enhancement to keep the system viable and useful. Every system eventually outlives its purposes or simply wears out. When that happens there are two choices: scrap the system or modify it so it remains useful; in the latter case the "modification" becomes a new system concept, the beginning of a new SDC and a new project.

For some systems Phase D is short or nonexistent: examples are a political campaign, rock concert, and gala ceremony (the project ends on Election Day or upon completion of the concert performance or ceremony).[1]

Virtually all projects progress through Phases A, B, and C, though not necessarily through the stages shown in Figure 3.2. In some projects, certain stages receive little emphasis or are entirely skipped; many projects, however, move through all the stages, even if informally. For instance, although not every project requires proposal preparation, every project does start with a proposal from *someone*. Similarly, while many projects do not involve "production," every project involves the production of something—even if only information. A great many projects follow a pattern similar to the cycle in Figure 3.2. The next two examples illustrate.

Example 3.1: New Product Development Cycle at Jamal[2]

Jamal Industries is a medium-sized manufacturing firm that produces products for major retailers under the retailer's own labels such as Sears and True Value. It develops and produces its products in the phases of *initiation, feasibility, analysis, design*, and *manufacturing*. Jamal initiates and implements most projects internally, though sometimes it contracts out development and manufacturing work. This example is such a case.

A competitor had introduced a computerized timer that would have a major impact on 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 feasibility study indicated that the first alternative was not feasible, although the second one —selling the product under its own label—was.

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 make a superior timer that Jamal could market at a price $12 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 as long as 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 System Development Cycle at Microsoft[3]

New software development at Microsoft commonly follows the phases of *planning, development,* and *stabilization*, although some of the phases pass through a series of iterations.

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 new or an upgrade. 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. The document starts out small (sometimes it can be described in a single sentence) but expands as the project progresses. The document and plans are combined with time estimates to create a project schedule. The phase concludes when management approves the plans and schedule.

The *development phase* is nominally 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 to enable the sub-phase to be completed by the milestone date. The first three sub-phases are devoted to development and coding, testing for bugs and functionality, and documentation of the product features. The goal of each sub-phase is to meet the requirements for a set of product features that would be fully ready to "ship," even though shipping isn't yet possible because the features have yet to be integrated into the product. In the event that 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 accurately correspond to the product when released. The sub-phases of the developmental phase are a variation of the "agile" approach to system development, described in Chapter 13.

See Chapter 13

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 remaining bugs are removed from the product (to be fixed later and included in subsequent product releases). This phase concludes with the release of a "golden master" disk from which manufacturing will make copies. The project concludes with a project team meeting to review the project and what was learned from it.

Most project-oriented companies undertake projects in ways best suited for them, and they prescribe or mandate ways to manage and perform tasks in those projects; i.e., they create their own project *methodology*. The two above examples illustrate such "homegrown" project methodologies. Throughout this book we will repeatedly refer to the methodology that encompasses phases A thorough C in Figure 3.2. We use this methodology not because it is always the best but because it conveys a common pattern that is very similar to methodologies we have seen in many companies. Another methodology similar to Figure 3.2 is the systems engineering process; this process, which relates to tools and topics discussed in Chapter 4, is described in the Chapter 4 Appendix. Other methodologies are discussed in Chapter 17.

See Chapters 4 and 17

Phased Project Planning and Fast-Tracking

The phases and stages in a project life cycle are sometimes undertaken in a stepwise fashion called *phased project planning* or project *gating*. At the end of each phase the project objectives, costs, and outcomes are evaluated, and a decision is made to continue, suspend, or cancel the project. Resources are committed only after a management review. This is elaborated in Chapter 4.

See Chapter 4

The project phases as described are not always performed in discrete sequence but can be overlapped in a practice called *fast-tracking*. Before Phase B is completed, elements of Phase C are started; before Phase C is completed, portions of Phase D are started. Fast-tracking compresses the time for systems development and implementation, though it poses the risk of overlooking or misdefining tasks and having to repeat or undo them.

In projects using the so-called *agile* methodology some phases are repeated. The execution phase and, sometimes, also the definition phase are repeated in cycles, each cycle intended to refine, enhance, or build upon the results of the previous cycle. Agile is covered in Chapter 13.

See Chapter 13

Stakeholders

The SDC has many stakeholders (actors and interested parties). The main stakeholders groups are:

1. System *customers* (buyers, clients, or owners), including:
    a. Customer management
    b. Users and operators
2. System *contractors* (the systems development organization (SDO), developer, promoter, or consultant), including:
    a. Contractor top management (corporate and functional managers)
    b. Project management (project manager and staff)
    c. The "doers"—professional, trade, assembly, and other workers

Customers are the persons or groups for whom the project is being done and who will acquire and/or operate the system when it is completed. Customer management pays for and makes decisions about the project; users and operators will utilize, maintain, or in other ways be the recipients of the project end-item. It is important to identify the actual users since, ultimately, it is for them the end-item is being created. The terms *customer* and *user* are used somewhat interchangeably, but it's 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 or developer or consultant is the party that studies, designs, develops, builds, and installs the system. The contractor is usually external to the customer organization, although, of course, 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 sometimes is referred to as the system 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 user-buyer). The project manager usually works for the contractor, although the customer might also have a project manager.

Besides these, the life cycle involves other key 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 he is affected by it, or potentially can alter its outcome is a *stakeholder*. Project stakeholders are discussed throughout the book and somewhat in depth in Chapters 15 and 17.

The remainder of this chapter will focus on the first phase of the project life cycle and how projects are conceived and started. For externally contracted projects, most of what happens in this phase is part of what is called the *procurement management* process.

See Chapter 15 and Chapter 17

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

The process begins when the customer or user perceives a *need*;[4] i.e., it recognizes a problem or opportunity and, possibly, ways to deal with it. Sometimes the need is expressed as a vision.

Example 3.3: Vision Statement at Microsoft[5]

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 a recent version of Excel it was just five pages long.

The purpose of the vision statement is to communicate the concept and requirements of the product to the development team, other product groups, and management. At Microsoft the vision 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 informs everyone about what they will do and not do, and gives them a common overview.

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 cull for the few good ideas, the customer organization undertakes a brief, initial investigation.

Initial Investigation

Many users know a problem exists but do not know what it is or 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, potential 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 or not to proceed with the idea. Most ideas never get farther 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 comparative 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 particular 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 18.

See Chapter 18

The initial investigation is usually conducted by the customer and is brief, requiring a few days or weeks at most. Sometimes 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 is approved for the next stage, feasibility.

3.4 Project Feasibility

Feasibility assessment is the process of studying a need, problem, and solution in sufficient detail to determine if an idea is economically viable and worth developing. The initial investigation is a form of feasibility study, a *pre-feasibility* study, which 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 (system concepts) and the benefits and costs of each. The customer typically performs 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, highway, or tunnel are examples where the feasibility studies are undertaken by contractors and are themselves big, expensive projects.

When a number of alternatives exist for the project, feasibility may depend on the life cycle cost (LCC) of the end-item system, which is the system's total cost over its entire useful life, including installation, operation, and disposal. For systems that must be newly developed, the cost includes *all* costs associated with all phases of the SDC—conception, definition, execution, and operation. The topic of LCC is covered in Chapter 8.

See Chapter 8

Some projects involve multiple feasibility studies. Product development and research projects often have a *technical feasibility* (to assess the risk that the technology might not work) and another, *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 for the project as compared to other projects under consideration. For large projects the execution plan (see Chapter 5) should include a section on project financing that addresses funding arrangements and means for controlling cash flow and managing money.

See Chapter 5

If the feasibility study indicates that the concept is viable, one of two things happen (Figure 3.3). Theme A: if the concept is something the customer can handle itself, it is passed along to an internal group for development and execution; Theme B: if the concept cannot be executed internally, it 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 Swisshotel, after deciding to build a hotel at a specific location, hire outside contractors to execute the project. In the latter case, they solicit proposals and bids from multiple contractors with an RFP document (described in the next section) and select the best.

Each contractor competing for the project must perform its own feasibility study to assess the merits of the project and whether or not it wants to participate. If a contractor decides to go forward, it will investigate alternative possible solutions (system concepts) to the customer's problem, choose one, and describe it in a proposal. This is called the "proposal preparation process." Upon receiving the proposals, the customer reviews them and selects the one that best fits the selection criteria, i.e., is the "most feasible."

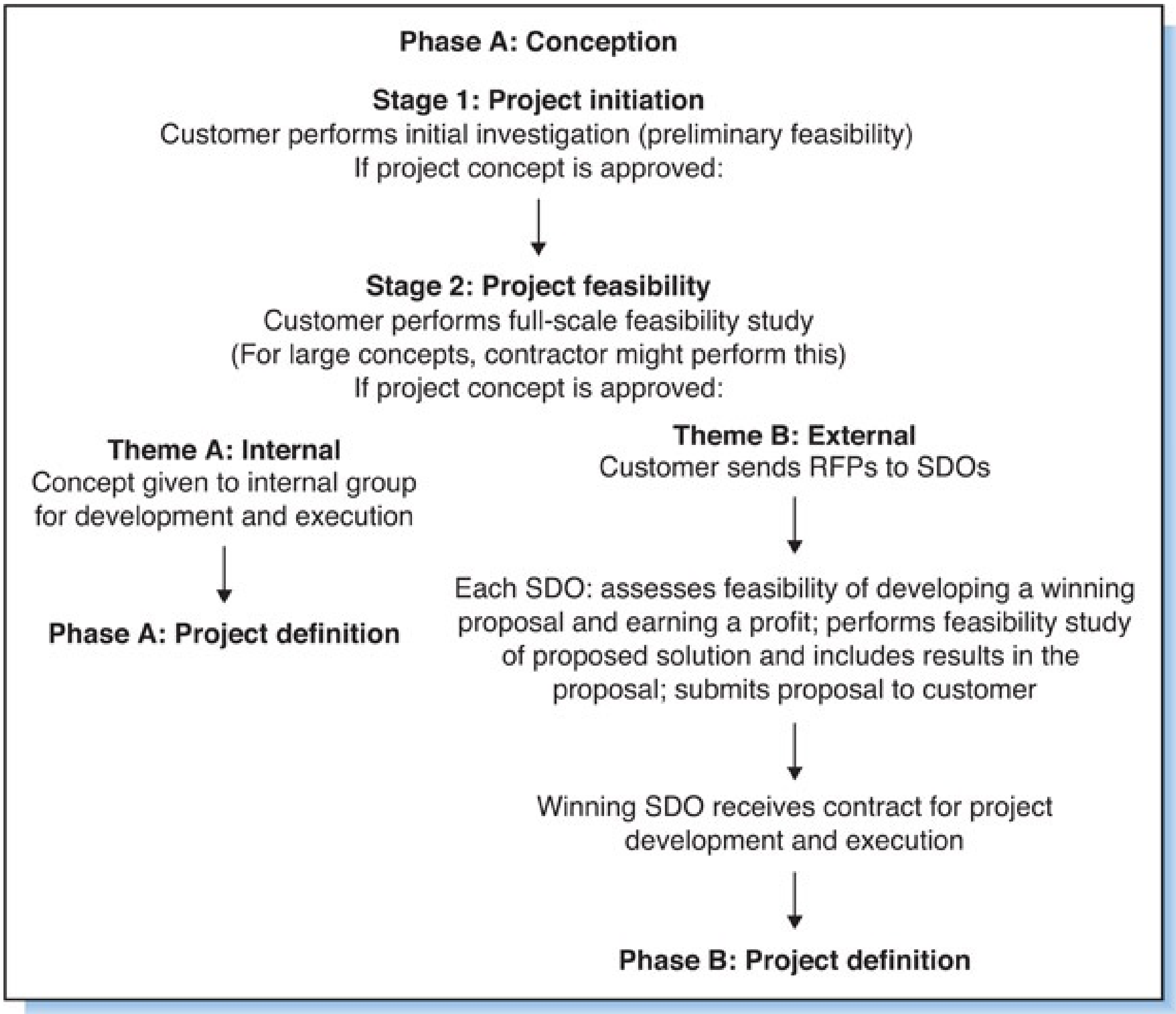


Figure 3.3 Feasibility study as part of the conception phase.

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 alternative solutions and picking the "most feasible" one to propose; 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 Proposal

The RFP—request for proposal (or *request for bid, request for quotation, invitation for bid (IFB)*, or similar term)[6] 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 how the winning proposal will be selected (proposal evaluation criteria) (see Figure 3.4). The process to select the best proposal is described later and in Chapter 18.

See Chapter 18

One 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 prequalified 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 scanning online newsletters and bulletins, and requesting RFPs. For example, *Commerce Business Daily* is a web publication that gives a synopsis of all US federal jobs over $10,000. Businesses scan the jobs and request RFPs for those they might be interested in bidding on.

Often the customer will precede the RFP with a *Request for Qualifications* or *RFQ*, which is a request for contractors to describe their qualifications. The customer sends RFPs only to contractors deemed qualified for the work.

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.

Usually the customer gets just what he asks for, and project foul-ups later can be traced 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 of preparing a winning proposal and, should it win the contract, of performing 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 possibly win the project, doing so to maintain its relationship with the customer, remain on the customer's bidders list, or keep the field competitive. Sometimes a customer sends out an RFP with *no intention* of ever signing with a contractor, doing so simply to gather ideas—obviously a situation of which respondent 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.

The 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 worth supporting; if the project work is to be done externally, the contractor also performs one to determine if the job is worth pursuing. In this section we consider the latter, although the same steps would apply equally to the customer or anybody doing a feasibility study.

The statement of the problem as defined in the RFP is frequently incomplete, vague, or even incorrect. If an RFP has been received it will likely contain such a statement. Thus, one of the contractor's first steps in 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 user. It is thus important that the contractor identifies who the *user* really is. Surprisingly, this is not always obvious. The real user, the party that will operate, maintain, or be the main beneficiary of the system, is often confused with persons who only represent the user. If the customer is an organization, the contractor must determine the individuals whose needs are to be met. The contractor should be working closely with the user throughout the feasibility study, so it is important to find users who are familiar with both the problem and the workings of the organization. Sometimes, however, the RFP specifies that in order 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. The feasibility study sometimes results in a document called a "business case."

The Business Case

The business case document 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 the process of choosing between projects, some companies use the business case to compare the value of and risks associated with a project to those of other projects. This is discussed in Chapter 18.

See Chapter 18

The content of the business case is based on the findings of a feasibility study; if the feasibility study indicates the project is viable, then the business case is written to justify the project. Whereas the final report of a feasibility study compares alternative solutions, 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 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 in some cases these estimates are updated as the project moves through its early phases. For example, the PRINCE2 methodology specifies developing an *outline business case* at the project start and thereafter reviewing and updating these estimates after each project phase. In some industrial mega-projects the first two phases of a project are referred to as FEL-1 (Front-End Loading-One) and FEL-2; FEL-1 concludes with a *preliminary business case*, FEL-2 with a verified, *detailed 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

Problems originate from needs (Definition: a problem is an unsatisfied need), and so do solutions (Definition: a solution is a way to satisfy a need), so it is important that the solution adopted for the project addresses the right needs. Hence, conducting a feasibility study and preparing a proposal should begin with defining user needs. J. Davidson Frame[7] suggests the following steps:

1. *Ask the user to state the needs as clearly as possible.*

Ask the user a complete set of questions to further elicit the needs. For example:

> Are these real needs, or are there other, more fundamental ones?
> Are they important enough to pursue?
> Are we are capable of fulfilling these needs, or is someone else better

suited?

If the needs are fulfilled, will they give rise to other needs?

Will satisfying these needs also satisfy other needs too?

What effect do the unmet needs have on the organization and the user?

What other parties are affected by these needs and how will they react to our efforts?

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

The steps are repeated as often as necessary, concluding with a statement of needs that the user accepts as best representing 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.[8]

- *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 actually *feels* the need and is most affected by it, or is it the party

who *pays* to resolve it? Usually they are different. The needs statement should reflect the opinion of the party to which the solution will be directed—the user. Do not be content for one party to tell you the need of another. Talk to the other party, too.

- *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:

  1. Extend the list of needs to be much broader than the user thought. This increases the size of both the problem, and, no surprise, the contractor's billable work.
  2. 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.
  3. Not ask for but rather *state* the user's needs (because, after all, the contractor *is* the expert).

Sometimes users are resistant to clarifying needs and 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 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 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 you said the room would be bright. It isn’t.”

Both cases illustrate user-contractor disagreements about end-results. Misunderstandings like these delay project completion, drive up costs, and sometimes become legal disputes that put the outcome in the hands of the courts. The problem is lack of clear *user requirements*. User requirements should describe in unambiguous terms what the user wants in the finished solution. Derived from user needs, the requirements are the measure by which the user determines whether or not the end result or solution is acceptable. Formally documented, they are the quality measures for the project. In the above examples, they would include the *functions* that the installed computer system and overhead lighting must serve.

Ideally, user requirements address the needs not only of users but also builders, suppliers, and other stakeholders that will benefit from, manage, maintain, or otherwise be impacted by the system. Perhaps obvious, 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.”

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.

Requirements for Objectives and Life Cycle

Every project and the end-item system to which it is directed start with a statement of objectives that elaborate on the needs and 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:

See Chapter 1

Develop a spaceship that can:

1 attain a minimal altitude of 100 km (where "space" begins)
2 be reused (launched) every 2 weeks
3 carry three people.

Each objective is then 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, cradle to grave, which means they should incorporate issues regarding how the system will be developed, built, used, marketed, financed, maintained, and disposed.

Requirements for Operational Modes

Included in this life-cycle thinking are the different ways and kinds of environments in which the system will be used or operated, 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.

Requirements for Constraints and Interfaces

Every system is subject to limitations imposed by the environment and other systems with which it must interface. These include 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

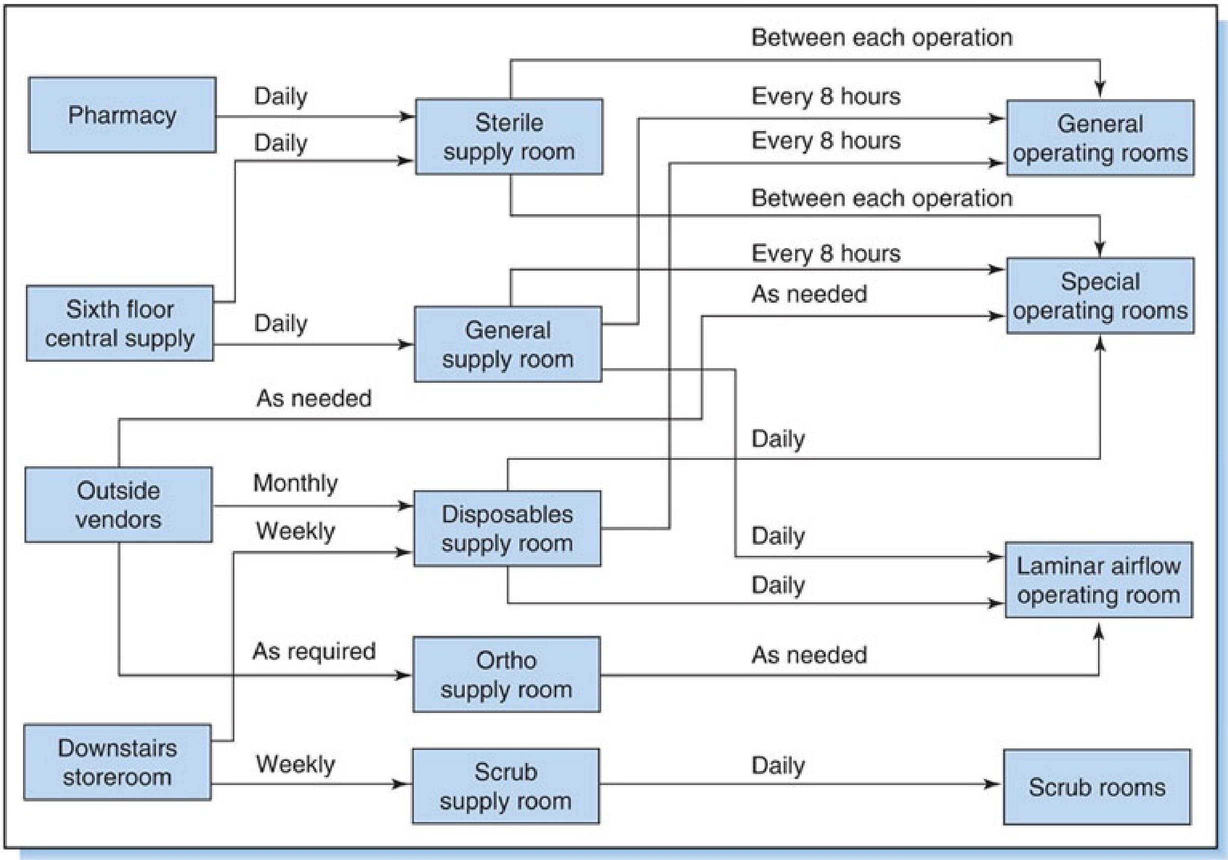


Figure 3.5 System schematic: Flow of supplies to the operating room.

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. A purpose of the feasibility study is to *understand* and *document* the current system, including its inputs, outputs, functions, flows, subsystems, components, relationships, attributes, resources, and constraints. The system schematic in Figure 3.5, for example, shows the elements and flows of a hospital system; it was developed in a project to reduce the cost of supplies inthe operating room. Of course, sometimes needs arise simply because there is no current system. That would be the case for SpaceShipOne.

Analysis of Alternative Solutions

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

Alternative solutions are analyzed for ability to satisfy objectives and user requirements within available resources and imposed constraints. The best solution is chosen and proposed to the customer. The following example illustrates.

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:

See Chapter 1

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

Associated with each of these requirements are many issues, problems, and alternative solutions. One issue that impacts all of 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
    b. Do not enter Earth orbit
3. Getting back to Earth
    a. Follow a wide parabolic arc
    b. Follow a narrow arc going almost straight up and almost straight down
4. Landing
    a. Land in a "zone" using a parachute
    b. Land at an airport like an airplane

After considering the requirements and constraints, 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. Choosing these alternatives involved analysis of cost, risk, technology, time, and ability to meet the requirements.

Environmental Impact

Part of a project's feasibility is determining the project's or end-item system's impact 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).

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, such as on:
    - Quality of air, soil, watersheds, wetlands, flood plains
    - Fisheries; sensitive plants; sensitive, endangered, or threatened species
    - Scenic resources; societal and aesthetic experiences
    - Heritage resources (sites, structures, buildings, districts, objects)
    - 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

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 greenhouse gas emission. Projects themselves and the end-items they produce contribute significantly to such effects. Consequently, one way to mitigate the damage is to design and build products—and manage the projects that do so—with an eye on sustainability.

Many industries have taken strides to incorporate environmental and social responsibility into the role of project management. 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 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 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 CO_2 from site activities.
- Reduce trips to/from the site and use local suppliers to reduce transport CO_2.

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:[9]

- 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 near transportation alternatives and within walking distance to ten basic services.

Matters of sustainability arise throughout the project life cycle—in project initiation, feasibility, and definition; in the RFP, proposal, contract, requirements, and project plan; in risk analysis; and in project execution. Although in some cases project managers have little ability to influence such matters, in others they do; they can influence the designers of the end-item and select the right contractors and suppliers with the goal of sustainability and minimizing the environmental impact of the project.[10] The project plan should minimally ensure that the project and its outcomes comply with local, state, and federal environmental laws; where the laws are inadequate, project managers can take the lead.

The result of the feasibility study is a statement of the problem, a list of needs and user requirements, a description of the current situation, and a preferred solution and reasons for its selection. The feasibility study, when combined with the project plan, bid price, and contractor qualifications, form the project proposal.

3.5 The Project Proposal

Proposal Preparation[11]

The proposal tells the customer what a contractor intends to do; it is the basis for selecting the project contractor. The effort to prepare the proposal is itself a project and, thus, it 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 preparation of the proposal; often, this person becomes the project manager if the contract is won. She might be entirely responsible for managing the 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; usually the bulk of the project team is not specified until after the contract is won.

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

The project 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 culled 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, the contractor should state that in the proposal.

During proposal preparation the project 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 one 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.

As the proposal is being prepared, the contractor should establish a dialogue with the customer to determine which solutions it prefers and which requirements are dominant among time, cost, and performance. Even when the RFP is clear, this will help ensure that the proposal will conform to the RFP specifications and 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 3.5: Writing Proposals for Real Estate Projects at Cwutzrite Company

The real estate department at Wutzrite Company helps clients choose among real estate investment alternatives. A meeting is set up with the client to define the client's investment "problem" and goals; 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 good rule of thumb is: Do not try to do the entire project while preparing the proposal! In some technical projects this may be unavoidable since the proposal includes a full-scale demonstration of the proposed solution. Developing the system for demonstration may itself be tantamount to a good-sized project.

To assure nothing is overlooked in the proposal preparation, project managers typically employ checklists that, over the years, grow to accumulate all the important items on a proposal, including, for example, key considerations for design, assembly, test, shipment, documentation, facilities, subcontractors, supplies, travel, labor rates, training, and payment. Before the proposal can be submitted 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.[12] If the proposal is prepared 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 the proposal for the LOGON project prepared by the Iron Butterfly Company.

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 Appendix A

The amount a contractor spends on preparing proposals and the proportion of contracts it wins significantly affect its company overhead since expenses for proposal preparation must be charged to overhead. Only in rare cases such as major defense contracts are the winning contractors reimbursed for proposal expenses.

Selecting the Winning Proposal

Upon receiving proposals from multiple contractors, the customer evaluates and compares 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 is more involved and 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; interested readers should see Chapter 18 where they are covered in depth. Here we give a brief overview of the project selection process.

See Chapter 18

In general, selection of projects is based upon consideration of the following criteria (sometimes provided to contractors in the RFP):

- Project price
- Ability of contractor to satisfy stated needs (solution or technical approach)
- Return on investment
- Project plan and 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 the proposed solution or technical approach. Thus, each proposal should include a rudimentary project plan showing key activities, and start and end dates and deliverables for each. Contents of the plan and methods for preparing the plan are discussed in Chapters 5 through 10.

See Chapters 5 to 10

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*, employs rating proposals according to several evaluation criteria on a checklist. Each proposal is given a score s_j for each criterion *j*. The overall score for the proposal is the sum of the scores for all criteria,

$$S = \sum s_j, \text{ where } j = 1, 2, \ldots, n$$

The proposal receiving the highest overall score wins.

One 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,

$$S = \sum s_j w_j, \text{ where } j = 1, 2, \ldots, n$$
$$\sum w_j = 1, \text{ and } 0 \leq w_j \leq 1.0$$

The procedures for the two methods are illustrated in Example 3.6.

Example 3.6: 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 the following four-point scale:

Criteria	1	2	3	4

Technical solution approach	Poor	Adequate	Good	Excellent
Price of contract	>1.8	1.6–1.8	1.4–1.6	<1.4
Project organization and management	Poor	Adequate	Good	Excellent
Likelihood of meeting cost/schedule targets	Poor	Adequate	Good	Excellent
Reputation of contractor	Poor	Adequate	Good	Excellent

Simple Rating

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

	Scores		
Criteria	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:

Criteria	Weight
Technical solution approach	0.25
Price of contract	0.25
Project organization and management	0.20
Likelihood of meeting cost/schedule targets	0.15
Reputation of contractor	0.15
	1.00

Taking the weights into account, the proposals scored as follows:

		Iron Butterfly		Modicum	
Criterion	Weight (w)	S	(s) (w)	s	(s) (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.

Assessment of proposals might also include 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 10.

See Chapter 10

Sometimes the contract award depends more on the contractor's qualifications than on the proposed solution. Among factors the customer might consider are:[13]

- 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 presentations or live demonstrations of their proposed solution or system. If several contractors receive close marks or some aspects of their proposals are unspecified or questionable, then the parties must negotiate to settle upon the final terms. If none of the proposals are acceptable or the feasibility studies show the project would be too costly, risky, or time-consuming or not provide adequate benefits, the process ends and nobody gets a contract.

When a contractor does not get the job, a good practice is to conduct a proposal "post mortem" to determine why not, lessons learned, and what it would do differently next time.

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 perform the work on its own, the initiation might happen with the business case study 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 market, competition, proposed product, 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 dual purposes: it is the feasibility study and the proposal combined. IT projects are similarly initiated by business case studies.

The department that would do the project if it were approved (PD or IT) 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 18. If the project is approved, a project charter is created, as described in Chapter 4.

See Chapters 4 and 18

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 stake-holders the process is more protracted. Examples include projects for infrastructure and transportation systems (Boston Big Dig, Delhi Metro, telecommunication systems, Chunnel), 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—where it is more difficult to define the stakeholders and meet their multiple and sometimes conflicting needs—the RFP/proposal process includes a "front-end" component to identify the important stakeholders and incorporate their needs into a list of stakeholder requirements. The contractor gathering the stakeholder requirements might be hired solely for that purpose and not be the same contractor as the one to perform the project work. After the stakeholders and their needs have been identified, other contractors are requested to review the requirements and suggest solutions, possibly through the RFP/proposal process. The process of identifying stakeholders and their requirements is a systems engineering effort that takes into account the system's far-reaching effects and the many stake-holders that will touch (or be touched by) the resulting system throughout its life cycle.

3.6 Project Contracting

Contracting environment and Process[14]

Contracting is ubiquitous in project management since all work is done in accordance with formal or informal contracts. Even in internal projects where there is no contract, an agreement exists between user and developer groups about the work to be performed to meet pre-specified objectives and requirements.

Most projects, even internal ones, involve some degree of external, legal contracting because the customer often must hire someone externally to perform at least some of the work. In many projects, *everything* in the project is done or provided by external organizations. As Figure 3.7 illustrates, these "external organizations" might be linked by numerous contractual agreements. The customer might contract with a principle party (prime contractor or SDO) to oversee the entire project; in turn, this party might contract with other parties—subcontractors, consultants, material suppliers, and contract labor (union trade professionals)—to be responsible for portions of the project; these parties might then contract with still others.

The *RFP/proposal* process addresses the question of *who will do the work*, not only for the customer but also for any party seeking to hire another to do work. Whether the customer, prime contractor, or other company down the line, each follows a process identical or analogous to the RFP/proposal process to document its needs, solicit ideas, and choose between potential contractors. Thus, any contractor farming out portions of work to subcontractors or acquiring material or services from suppliers will follow the RFP/proposal process to identify and choose the most qualified subcontractors and suppliers. Just as the customer follows the process to hire a contractor, so the contractor follows it to hire subcon-tractors and so on down the line.

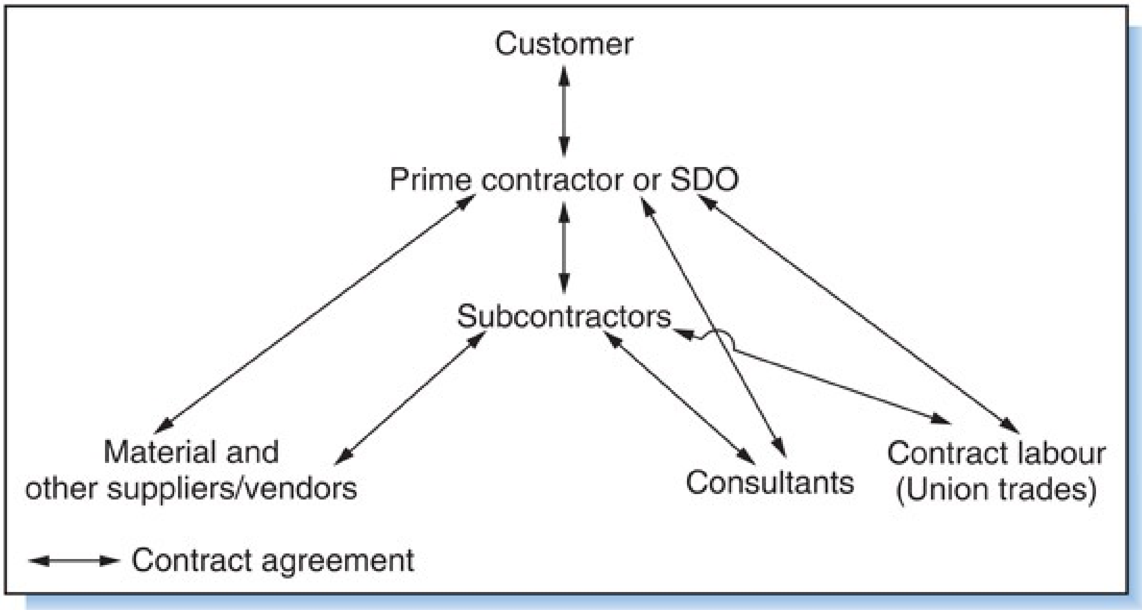


Figure 3.7 Contracting parties in a project.

The effort required to review and hire qualified subcontractors can be quite time consuming, especially in international projects, so extra time should be allotted for it in the project schedule. If not, the process to select subcontractors can delay the start of portions of the project and then cause the entire project to slip behind schedule.

The customer should seek to retain a measure of control over contracted work. To that end, the contract should clearly specify the areas of the project over which the customer has ultimate authority for supervision and decisions, and the customer's role in tracking project progress. From the time when the contract is signed until when the project is closed out or terminated, the contractual agreement must be *managed*, which means keeping the agreement up-to-date with respect to ongoing changes to the project, the customer's needs, and the contractor's capability, and checking that all work conforms with the agreement. This process, called *contract administration*, is discussed in Chapter 11.[15]

See Chapter 11

Subcontracting[16]

Each party in Figure 3.7 must decide what portions of the project it will do itself and what portions it will contract to others. Some contractors do the work; some hire others to do it, i.e., they subcontract. For example, a customer hires a general or prime contractor to manage a construction project and, perhaps, to assemble the building structure, but then the contractor hires other companies for specialized work such as wiring, plumbing, ventilation, and interior details.

Example 3.7: Project Contracts in Construction

Large construction projects are often in the news—sometimes because of problems owing to cost overruns or schedule slippages. Although many factors are cited (labor union problems, materials shortages, weather, and inflation), the cause is frequently poor management and lack of control. Often, the project manager is either the architect, engineer, or builder's prime contractor. This works on small construction jobs, but on big jobs it does not because designers and contractors each represent separate interests. When things go wrong and arguments arise, both tend to be self-serving and there is no one who is impartial and can reconcile differences in the best interests of the customer (owner or developer).

A better arrangement is when the customer appoints an independent construction project manager to represent her interests during the entire design and construction process. Figure 3.8 shows two possible ways to arrange this, EPCM and EPC ("Engineering, Procurement and Construction Management" vs. "Engineering, Procurement and Construction"). In the figure, EPCM and EPC represent the independent project manager. In the EPCM case the customer contracts directly with the designer and building contractor; in the EPC case the independent project manager does this. In either case the project manager's central position in the project organization enables her to monitor and coordinate all design and building tasks in accordance with the customer's goals.

The main role of the EPCM project manager is to ensure that the designs are within the customer's cost allowances and building requirements, and the builder's work is executed in accordance with contract specifications and at a fair price. The project manager is involved in everything—overseeing preliminary design, subcontracting, and controlling site work according to design specifications, schedule, budget, and safety rules. Although the customer signs contracts with the designer and the builder, the project manager acts as the customer's agent to facilitate relationships among the parties.

An EPC project manager acts in much in the same manner as in EPCM, however, having contracted directly with the designer and the builder, the EPC project manager has *full authority and accountability* for all elements of the design and building of the structure, typically as a "turnkey" facility. As a consequence, the customer is much less involved in the project than in the EPCM case.

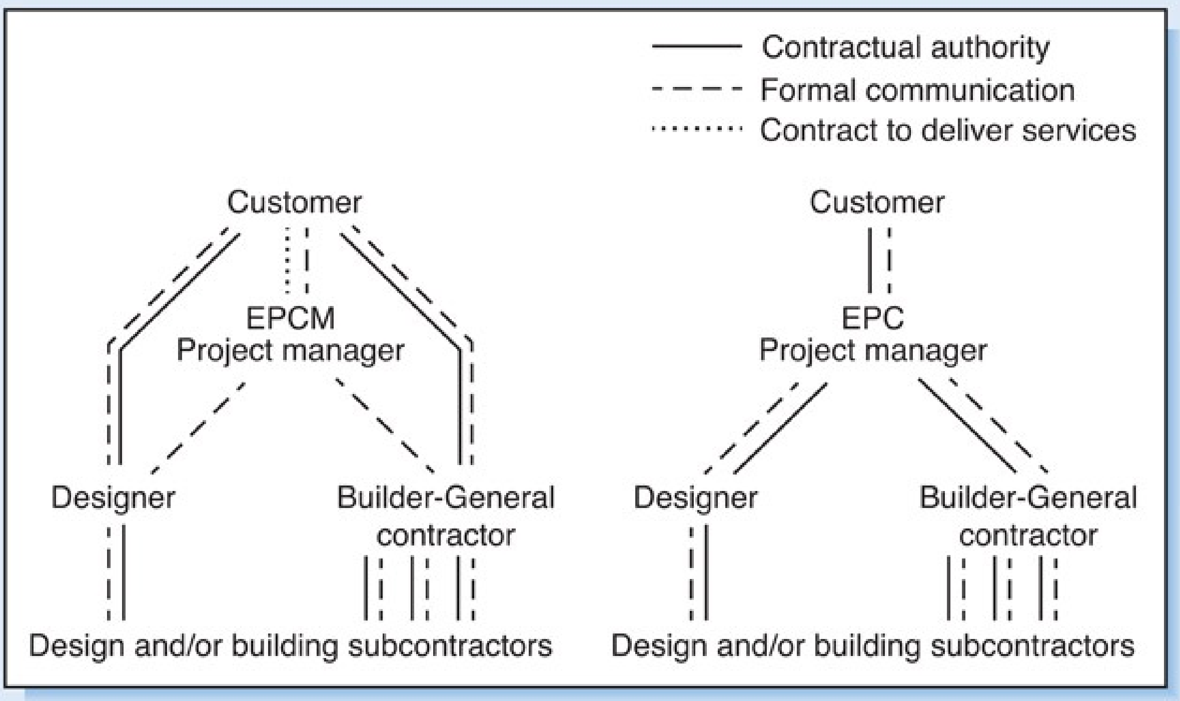


Figure 3.8 Alternative types of project manager in a construction project.

Even a contractor that might be capable of doing all the work itself may choose to subcontract because it has limited capacity or believes a subcontractor could do the work for lower cost or less risk. For development projects of large-scale systems, the prime contractor will usually design the overall system and major subsystems, and will produce some elements of the system itself but subcontract the production of all others. In projects where significant portions of project work are to be subcontracted, the customer often mandates the scope of the subcontracted work and the criteria for selection of suitable contractors or 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 inspection at any subcontractor offsite facilities. Similarly, any communication about customer changes to requirements is channeled through the contractor to the subcontractor. (If however you are a subcontractor and are having trouble getting paid, you might appeal directly to the customer to pressure the contractor into paying you.)

Work that is contracted, subcontracted, or procured from suppliers is referred to as "procured" goods, work, and services. Like everything else in projects, procured work must be planned, scheduled, budgeted, and controlled—it must be managed. Called *procurement management*, this is described in Chapter 5.

See Chapter 5

Contract Negotiation[17]

Contract negotiation is the process of clarifying technical or other terms in the contract and reaching agreement on time, schedule, and performance obligations. Negotiation is not necessary for standardized projects for which the terms are simple and costs are fairly well known, but it is for complex systems that require development work or are somewhat risky. In fact, when an IFB (invitation for bid) for a standard or well-defined item is sent out and price is the only criterion, negotiation would be unethical and is not allowed. Different contractual agreements offer advantages to the customer and contractor, depending on the nature of the project. These agreements are discussed in the Appendix to this chapter; they are, briefly:

- *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-Plus 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*: The amount paid depends on the contractor's performance in comparison to the target price, schedule, or technical specification. The contractor either receives a *bonus* for exceeding the target or must pay a *penalty* for not meeting it.

The specific type of contractual agreement between the customer and the prime contractor determines the type of agreement negotiated with subcontractors. If the prime contract agreement is fixed price (FP), then subcontract agreements should also be FP else the prime contractor risks being charged more by subcontractors than will be received from the customer. If the prime contract is cost-plus or incentive, the contractor has latitude to use any type of agreement for subcontracts.

Negotiation is the last activity before a contract agreement is reached, but the negotiation *process* often begins earlier during proposal preparation because the terms in the proposal and the contract must be mutually consistent and acceptable to both customer and contractor. During negotiation, terms in the proposal related to specifications, schedules, and price are converted into legal, contractual agreements. Performance, schedule, and cost are interrelated, and a "package" agreement must be reached wherein all three parameters are acceptable to both parties. Final negotiation is the last formal opportunity to correct misperceptions that might have slipped through the RFP/proposal process. (Customers are *always* negotiating—informally—to get a better deal, even after the project is underway. The contractor must always be wary of saying and writing anything that might be construed as a promise to deliver more than is specified in the contract.) 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.

Throughout negotiation, the project manager pushes the merits of his proposal. His goal is to obtain the best possible agreement for his company. In countering 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. Details of the plan are used to define which parts of the schedule, work, or price are relatively "fixed" and which are flexible and can be negotiated.

To be able to negotiate from a knowledgeable and competitive position, the project manager must learn as much as possible about the customer and the competition. She should determine if the customer is under pressure to make a particular decision, faces an impending fiscal deadline, or historically has shown preference for one particular approach or contractor over others. The project manager should also know the competition—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 competitors. (Relying on the last source is ethically questionable and, of course, works against the contractor whenever competitors hire its employees.)

To be able to negotiate tradeoffs, the project manager must be intimately familiar with the technical details of system 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 schedule and time–cost tradeoffs.

The signed contract becomes the binding agreement for the project. Any changes thereafter should follow formal change procedures, including change notices, reviews, customer approvals, and, sometimes, contract renegotiation—topics discussed in Chapter 11.

See Chapter 11

Contract Statement of Work and Work Requisitions

The contract contains an SOW that is similar to the SOW in the proposal or the original RFP, or is a restatement of either 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, and so on. The CSOW specifies the conditions under which the deliverables or end results will be accepted by the customer. Failure to clearly state these conditions can lead to later disputes and delays in completing the project.

Contracts with suppliers and subcontractors also include a CSOW, plus for each party the responsibilities and liabilities. Contracts for procured items include specifications, quantities, delivery schedules, costs, payment schedules, and ways to handle changes or variations.

When the customer and the contractor both agree on the CSOW, the project is considered "approved" and ready to go. Before work can actually begin, however, it must be divided among the involved departments and subcontractors of the SDO, 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 to each party the work expected of it and to authorize the work to begin. This topic is discussed further in Chapter 11.

See Chapter 11

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

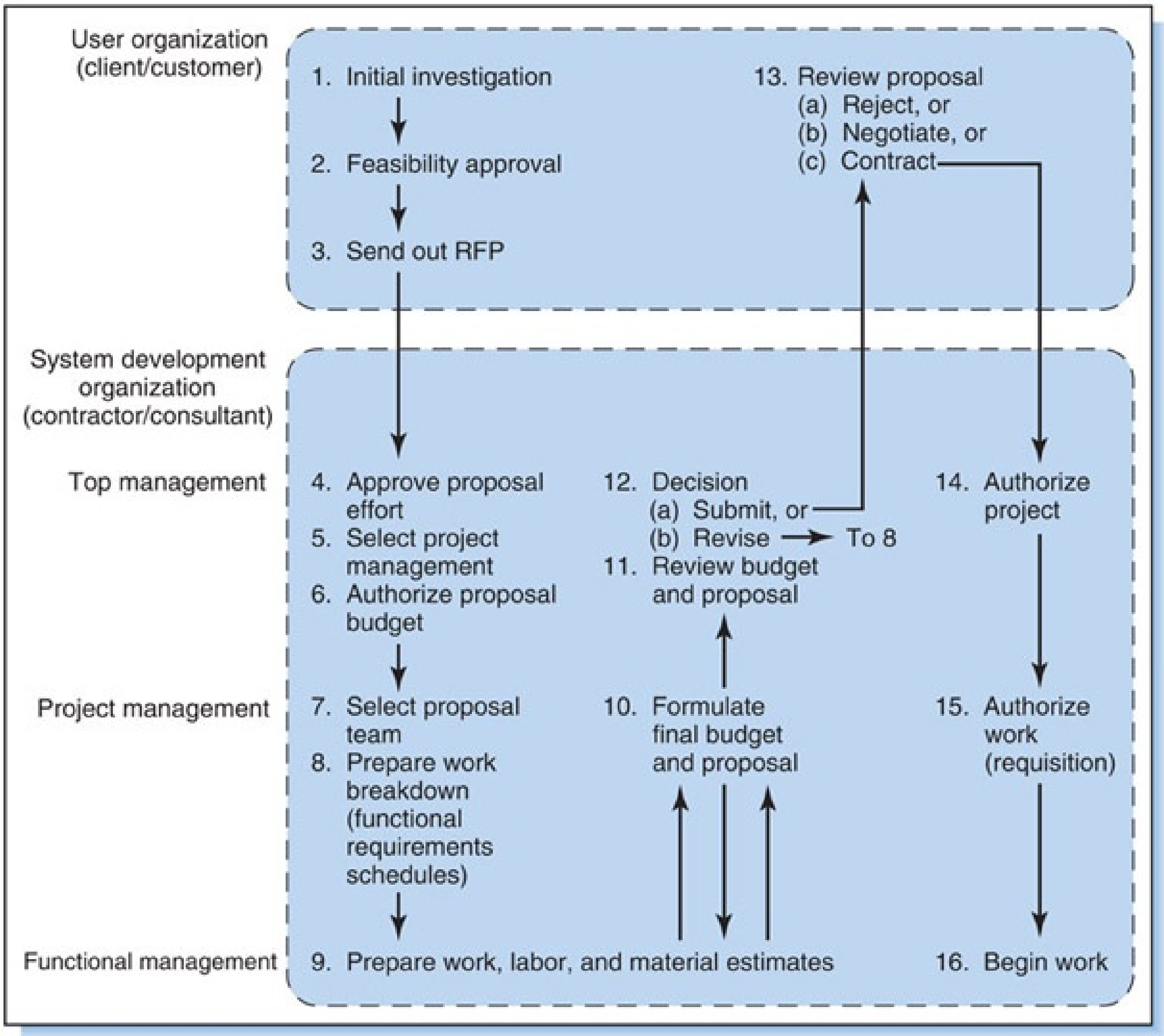


Figure 3.9 Project initiation, proposal preparation, and project authorization process.

The process of initiating projects, preparing proposals, and negotiating and finalizing contracts often involves convolutions that cannot possibly be covered in a chapter. The following story illustrates.

Example 3.8: Proposal for the Apollo Spacecraft[18]

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 the Apollo spacecraft. His division had already been working feverishly to solve difficult technical problems for a proposal to build the rocket's second stage. The technical requirements were so demanding that only a handful of contractors had stayed in the competition. Most managers in the middle of 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 men to the moon and back. 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, old "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 took it anyhow. Now NA would bid on *both* the second stage *and* Apollo (Figure 3.10).

The RFP for Apollo allowed competing contractors 10 weeks to submit their proposals. Three competitors had already done Apollo studies and had a 12-month head start; one already had assigned 300 people to work on the concept, spent $3 million, and prepared a 900-page report. In large bids like this companies partner up to add muscle to the proposal, but all of the big aerospace contractors had already teamed up and NA was left to go it alone. Nobody believed Storms' proposal team had a chance, including the company president.

Storm's team labored feverishly, 7 am to 11 pm. Feeding them information were scores of engineers in shops, offices, laboratories, and wind tunnels throughout the company; to oversee them Storms picked the best leaders at NA he could find—smart, practical people with solid experience that others looked up to.

Good news arrived: NASA announced that NA had been chosen prime contractor for the second stage. But jubilation settled into gloom over prospects of also winning Apollo; practically nobody felt that NASA would award two major contracts for the lunar program to the same company.

The allotted $1 million had long since been exceeded—maybe by three times, but no one knew. Back then cost statements ran 30 to 60 days behind billings, and Storms gambled that NASA would receive the proposal before his boss saw the final bill. With less than 6 weeks to go he picked John Paup to be Apollo program manager, someone he thought perfect for the role, a "witty, engaging person" who understood the technology. For the next month Paup listened to presentations 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, no seats, he wanted to hear 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 pm 2 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, some of them legends. Undaunted, Paup hit all the presentation high points and finished 10 minutes early.

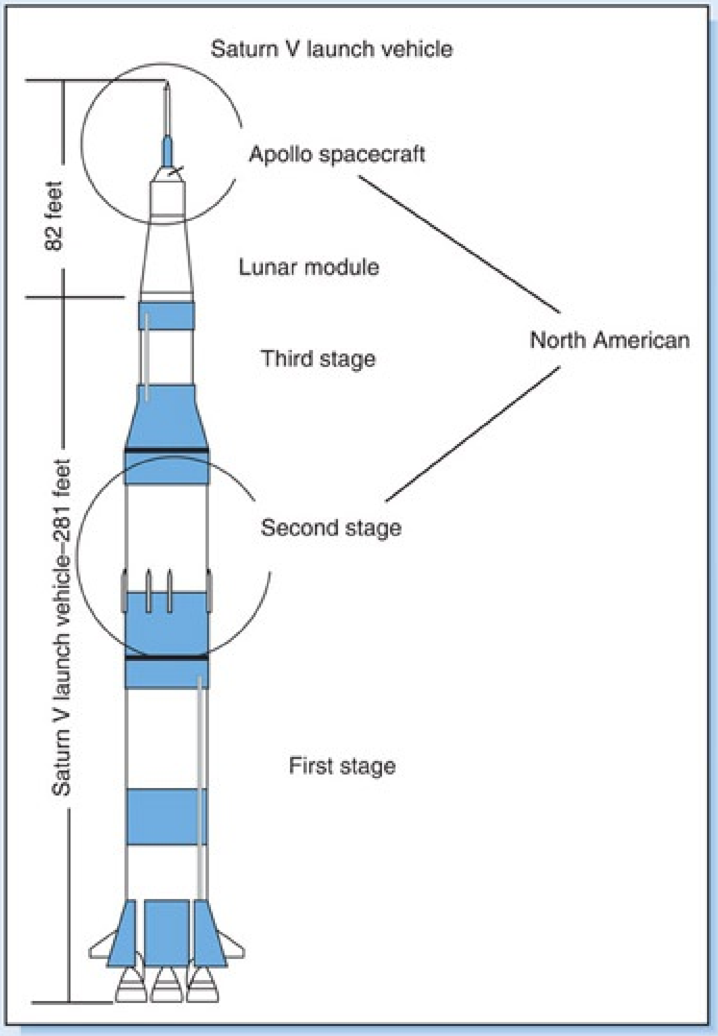


Figure 3.10 Apollo/Saturn moon rocket and North American components.

Picture courtesy of NASA.

Days later Storms received a telegram: NASA wanted to know how, given NA's second-stage contract, it could possibly handle Apollo too? The written response was too long to telegraph back so Storms and Paup jumped on a plane to hand-deliver it. This violated an unwritten rule that a contractor does not meet with the customer evaluating the proposal. But Storms had little regard for such rules, especially with so much at stake.

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 understatement. North American won the contract, although it would take another year to formalize the details: 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 3.11). The risks of sending humans to the moon were overwhelming, so the contract was cost-plus. By the time the lunar program ended 10 years later with the return of the seventh crew from the moon, NA as prime contractor had earned $4.4 billion—over $27 billion in 2015 dollars.

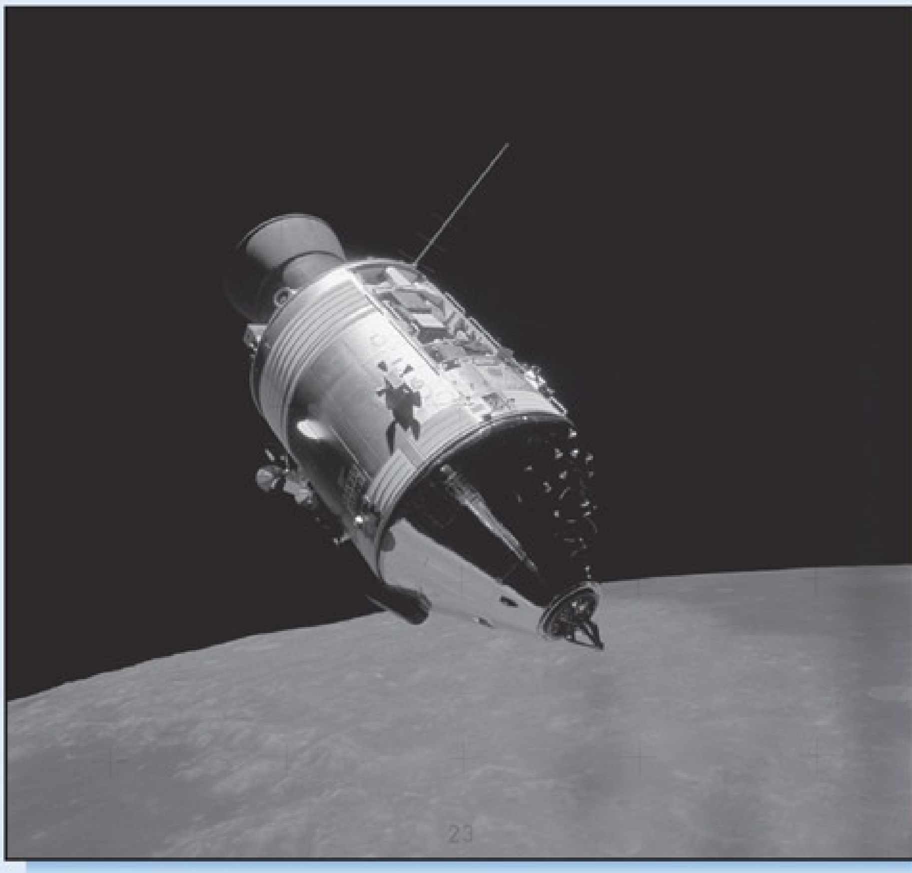


Figure 3.11 Apollo spacecraft.

Photo courtesy of NASA.

3.7 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, the 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 and contract negotiation.

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 a crucial phase and the place where, often the seeds of project success or failure are planted.

Appendix: Kinds of Contracts

A contract is an agreement between two parties wherein one party (the seller—the project contractor) promises to perform a service for another (the buyer—the client or customer), typically in return for payment.

Requirements about the service and the payment are clearly spelled out in the contract, which typically includes the following:

- Scope of work to be done or items to be sold, including support and ancillary (side) items such as manuals, documentation, and training. Any specifications and standards referenced are considered as part of the contract.
- Duties of the contractor in providing the work or items.
- Time schedule allowed.
- Duties of the client regarding payments (including a schedule for milestones).
- How changes to the contract and disputes will be handled.
- How risks will be handled, including warranties, penalties, or bonuses/incentives.

Different kinds of contracts provide different advantages to the client and the contractor, depending on the project risk and difficulty in estimating costs. Each party tries to negotiate the kind of contract and the terms that best serve its own interests.

The two fundamental kinds of contacts are *fixed price* and *cost-plus*. In the fixed price contract, the price is agreed upon and remains fixed as long as there are no changes to the project scope or provisions of the agreement. In the cost-plus (or *cost reimbursement*) contract, the contractor is reimbursed for all or some of the expenses incurred during the project, and as a result, the final price is unknown until the project is completed. Within these two types are several variations including some with incentives for the contractor to meet cost, time, or performance targets.[19]

Most projects involve multiple contractors and, hence, multiple contracts and a combination of contract types, for example, cost plus for engineering and design, and fixed price for construction. This is often a good way to contract project work, especially for large projects.

Variables

The variables specified in a contract may include the following:

C_{ex} and C_{ac}	Target (expected) cost and actual cost of the project under normal circumstances. "Cost" represents monies expended by the contractor in performing the work.
Fee	Amount paid to the contractor in addition to reimbursable costs.
Price	The price the client pays for the project. Price includes reimbursable costs (or a percentage thereof) incurred by the contractor, plus the contractor's fee.

Fixed Price Contracts

Fixed Price Contract

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, he might win the job but make no profit; if he overestimates, he may lose the job to a lower priced bidder.

Example A1: Fixed Price Contract

Contract agreement:

Cost estimate, C_{ex} = \$100,000

Fee = $10,000
Price = $110,000

No matter what the project actually ends up costing (C_{ac}), the price to the client remains $110,000.

When project work is straightforward and can be specified in detail, everyone prefers this kind of contract. Clients like it because they are less concerned about project costs. Contractors like it because clients tend to request fewer changes to the contract.

The disadvantage of an FP contract is that it can be more difficult and costly to prepare. The contractor risks underestimating the cost and losing money on the project, which might motivate him to increase the bid price or cut corners (use cheaper quality materials, perform marginal workmanship, or extend the completion date) during the project to reduce costs. To counteract this, the client can specify in the contract rigid end-item specifications and completion dates, and closely supervise the work. If, however, the project gets into serious trouble, bankrupts the contractor, and leaves the project incomplete, the client may be subject to litigation from other stakeholders.

Fixed price contracts do not work well in high-risk projects. Project sponsors often impose an FP contract, thinking that it transfers the risk of overruns to the contractor. Sometimes it does, but when a large project gets into trouble and the contractor cannot absorb the losses, the sponsor will have to keep paying to sustain the project.

FP contracts can be short-sighted. A project's success often depends on the performance of the end-item long after the project is completed, yet the "fixed price" may force contractors to jettison things (cut corners) that diminish that performance.

Fixed Price with Redetermination[20]

Projects with long lead times such as construction or production have contract *escalation provisions* that protect the contractor against increases in materials, labor rates, 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 *redetermined* as actual costs become known. In the latter case, the initial price is negotiated with the stipulation that it will be redetermined later to accurately reflect actual cost data. There are a variety of redetermination contracts: some establish a ceiling price for the contract and permit only downward adjustments, others permit upward and downward adjustments; some establish one readjustment at the end of the project, others allow multiple, periodic readjustments. Redetermination contracts are appropriate wherever design efforts are difficult to specify or the final price cannot be estimated for lack of accurate cost data. The redetermined price may apply to future items and items already produced.

Because the only requirement to renegotiate the price is substantiating cost data, redetermined contracts tend to induce inefficiencies. 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-plus" kind of contract and is subject to abuse.

Any contract wherein all costs are reimbursed is called a "cost reimbursable" contract; these include the cost-plus and incentive contracts discussed next.

Cost-Plus Contracts

In complex, uncertain, or risky projects where it is difficult to accurately estimate project costs, cost-plus type contracts allows work to begin before the costs are fully determined.

Cost Plus Fixed Fee (CPFF)

Under a CPFF contract, the contractor is reimbursed for all direct allowable costs plus an additional, fixed amount to cover overhead and 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, usually computed as a percentage of the target or estimated cost, C_{ex}.

Example A2: Cost Plus Fixed Fee Contract

Contract agreement:

Cost estimate, C_{ex} = \$100,000
Fee = \$10,000
Target price = \$110,000

In addition to the fee, the client will pay for all allowable costs (perhaps "all" costs, C_{ac}). Thus, if the project ends up costing C_{ac} = \$200,000, the price to the client is \$210,000.

In contrast to FP contracts, CPFF agreements put the burden of risk on the client. The client 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 he gets paid the same fee regardless. A major factor motivating the contractor to control costs and schedules is the negative effect of overruns on his reputation. Another is that as long as the contractor's workforce and facilities are tied up, he cannot work on other projects.

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 \$10 million but then profit another \$100 million from fees added to materials and services. The client will learn about these added costs only through auditing the project. Contractors sometimes argue that the costs in a CPFF agreement are proprietary, however that is nonsense and the customer needs a good auditor to check on every cost during the project and before the contractor is paid. The client may also specify who is to be project manager or assign her own project manager to work alongside the contractor's project manager.

Despite the risks, the client 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 the costs are difficult to estimate.

Guaranteed Maximum Price

With a CPFF the final price is unknown until the project is completed and the costs tallied, so a more appealing agreement to clients is the guaranteed maximum price (GMP) contract, which is a CPFF contract with a price cap. The GMP amount is negotiated and the client agrees to pay for actual project costs *up to the GMP*; for costs beyond that, the contractor is responsible. The GMP includes the contractor's fee, which can be fixed or a percentage of costs.

Suppose the fee is set at $10,000 and the GMP at $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 Contract

A time and materials (TM) contract is a simple agreement that reimburses the contractor for labor and materials costs incurred in the project. It provides for payment of direct labor hours at an hourly rate that includes direct labor costs, indirect costs, and profit. Sometimes a ceiling price is established that may be exceeded, depending on the agreement. Charges for private consultants and the services of electricians, carpenters, mechanics, etc., are usually based on TM.

Incentive Contracts

When a contractor is unwilling to enter into an FP agreement and the client does not want a CPFF contract, an alternative is an *incentive* arrangement. This has features of both kinds of contracts: it is similar to CPFF in that costs are reimbursed, but the amount reimbursed is based on *shared savings*, i.e., any savings from C_{ac} being less than C_{ex} are split between the parties.

The sharing split is determined by the *cost sharing ratio* (CSR). 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 he *pays* 20 cents on every dollar spent above C_{ex} but *earns* 20 cents more on every dollar saved below

C_{ex}. As further incentive to reduce costs, the ratio might be changed for costs above C_{ex} such that the contractor must pay a higher percentage. The contract appeals to both parties since the contractor can earn greater profit and the client pay a lower price.

Cost Plus Incentive Fee Contract (CPIF)

In a CPIF contract the *project price* is based partially on a percentage of the actual cost, C_{ac}, and the CSR. The contract specifies the target costs, C_{ex}, and the CSR, which specifies how any savings or overruns will be divided between the client and the contractor.

Example A3: Cost Plus Incentive Fee Contract

Contract agreement:

Cost estimate = Target cost = C_{ex} = \$100,000
Fee = \$10,000
Target price = \$110,000
Cost sharing: CSR = 50/50

CSR of *m/n (m + n = 100)* means that for any difference between target and actual cost, client gets *m* percent of any saving and pays *m* percent of any overrun. Price is computed as

Under target cost: Price = $C_{ac} + (C_{ex} - C_{ac}) \times n +$ Fee
Over target cost: Price = $C_{ex} + (C_{ex} - C_{ac}) \times m +$ Fee

The incentive is for the contractor to keep costs below \$100,000. Suppose C_{ac} is \$80,000 (\$20,000 under C_{ex}).

Price = \$80,000 + (\$20,000)(.50) + 10,000 = \$100,000

The client saves \$10,000 on price and the contractor earns a \$10,000 bonus. The client must be vigilant to ensure that the incentive doesn't lead 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)(.50) + \$10,000 = \$160,000
The contractor is \$200,000 − \$160,000 = \$40,000 in the red.

A variation of CPIF is guaranteed maximum price—GMP. To continue the previous illustration, suppose the GMP is \$130,000. If C_{ac} is \$200,000, the customer pays only \$130,000, the GMP. With the GMP, the contractor is \$200,000 − \$130,000 = \$70,000 in the red.

Commonly two CSRs are used, a different one each for under target and over target. See end-of-chapter review question 27.

Fixed Price Incentive Fee Contract

A Fixed Price Incentive Fee Contract (FPIF) is similar to a CPIF contract but has a ceiling on both price (GMP) and profit. The contractor negotiates to perform the work for a target price based upon a target cost (C_{ex}) plus a fee, and for a GMP and a maximum profit. If the project cost ends up being 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 A4 Fixed Price Incentive Fee Contract

Contract agreement:

Cost estimate, C_{ex} = \$100,000
Fee = \$10,000
Target price = \$110,000
GMP = \$125,000 (fee + reimbursement), client will pay no more than this Maximum profit = \$15,000, contractor profit cannot exceed this

Cost sharing: CSR = 50/50, therefore:

- If C_{ac} < \$100,000, client 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, client 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 $5000 for the cost savings (50 percent of the $20,000 savings is $10,000, of which only $5000 is allowed because the maximum allowable profit is $15,000). Total price to client: $95,000, a $15,000 savings 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 GMP is $125,000, which is all the client pays. 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 profits can be made through the incentive features. But unlike cost-plus contracts, they provide some assurance about 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. They do not apply to R&D or other projects where the target cost is difficult or impossible to estimate.

Other Incentives Contracts[21]

Incentives can be applied to schedules and performance as well as cost or price. Similar to the CPIF and FPIF contracts, these agreements specify target project completion dates or target performance parameters for the end-item. The final price of the project "rewards" the contractor for exceeding the target or “penalizes” the contractor for missing the target.

Incentive agreements sometimes have negative effects, such as cost incentives leading to schedule slippage. To offset this are so-called *multiple incentives contracts* that reward contractors for achieving targets associated with multiple criteria such as schedule, cost, *and* performance. Fee weights assigned to the criteria are used to determine the amount of “fee swing” allocated to each criterion. Consider the example shown below where the fee structure is similar to the CPIF example. Here the “fee swing” (F_{min} to F_{max}) is between 2 percent and 14 percent, or 12 percent.[22]

C_{ex} = $100,000
F_{ex} = $8 (8%)
F_{max} = $14 (14%)
F_{min} = $2 (2%)

The 12 percent fee swing is then divided among the criteria; for example:

Criterion	Weight	Fee Swing
Performance (x)	0.5	6%
Cost (y)	0.25	3%
Schedule (z)	0.25	3%
Total	1.00	12%

In engineering contracts, typically the largest weight is given to performance, followed by schedule and cost. To assess performance, several measures might be used at once, such as accuracy, range, reliability, and speed; an index is devised so all measures can be represented by a single performance factor.

In this example, performance is given a weight of 0.5, which yields a profit swing of 6 percent; schedule and cost are each given weights of 0.25, so each have a profit swing of 3 percent. The profit percentage is computed as a function of the three criteria according to the formula

$$P = (8 + x + y + z)\% \, (C_{ex}).$$

Values for *x*, *y*, and *z*, determined at the end of the project, would be based on the curves in Figure 3.12.

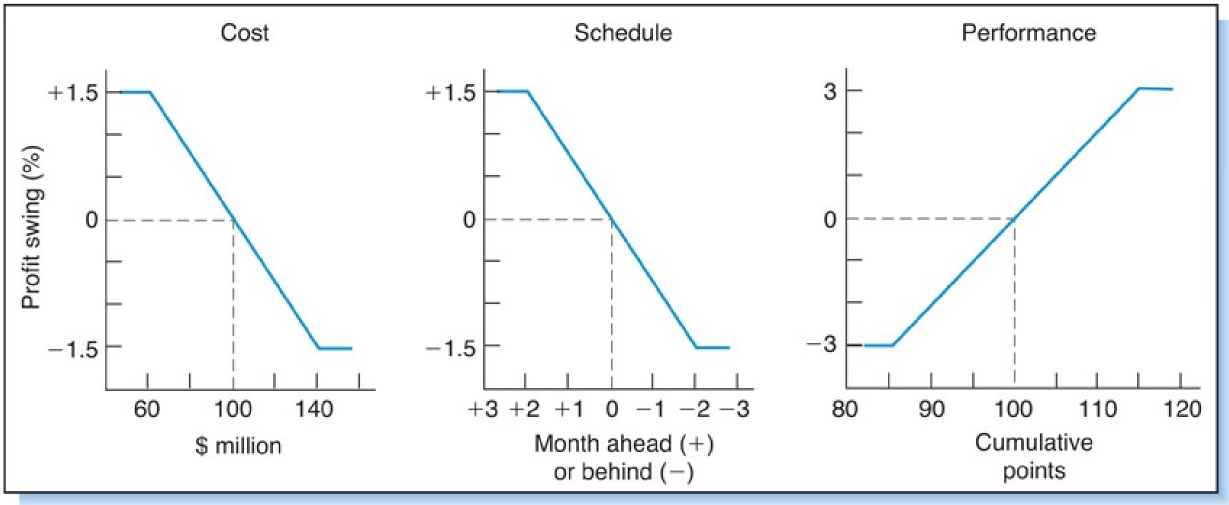


Figure 3.12 Multiple incentive contract.

Since the multiple criteria in this kind of contract tend to be interrelated (e.g. performance targets can be met, but for greater time and cost), the terminology and computations involved in structuring the contract tend to be tricky; hence, this type of contract is rarely used.

Review Questions

1. How are projects initiated? Describe the process.
2. What factors determine whether or not an idea should be 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 role 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.
25. Describe the different kinds of contracts (refer to chapter Appendix). What are the relative advantages and disadvantages of each to the customer and the contractor?
26. A customer accuses a project manager for cost overruns and a delay in delivery. Why is it relevant whether the relationship between the customer and the project manager is governed by an EPC or EPCM contract?
27. Refer to the CPIF and FPIF example problems in Examples A.3 and A.4 in the chapter Appendix.

    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 profit?

c. Commonly 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

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. The system would be linked to WCMC's servers and be 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 performances. 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 select 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 on 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 have to 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 Spacecraft[23]

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	Business Strength (30%)	Weighted Total (40%)
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	5.53	5.67	7.62	6.41

Corporation

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?

Case 3.4 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 Polanski 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 ample 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 substantial experience in nuclear power plants. Moreland would be responsible for virtually 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 take on the job, but on a cost-plus basis only. 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 ever done.

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?

Endnotes

1. 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 political term—but that would be stretching the analogy.

2. Based upon information collected and documented by Cary Morgen from interviews with managers of Jamal Industries (factual case, fictitious name).

3. Cusumano M. and Selby R. *Microsoft Secrets.* New York, NY: Free Press; 1995, p. 210.

4. A need is a value judgment that a problem exists. Different parties in an identical situation might perceive the situation differently; as a consequence, 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.

5. Cusumano and Selby, *Microsoft Secrets.* p. 210.

6. 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.

7. Adapted from Frame J.D. *Managing Projects in Organizations.* San Francisco, CA: Jossey-Bass; 1987, pp. 109–110.

8. Ibid., pp. 111–126.

9. Sundblad D. "Sustainable Construction Techniques." http://greenliving.lovetoknow.com/Sustainable_Construction_Techniques, accessed November 12, 2014.

10. 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.

11. 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.

12. 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.

13. Murphy O. *International Project Management.* Mason, OH: Thompson; 2005, pp. 159–161.

14. 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.

15. 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 called "contract monitoring." See Hirsch W. *The Contracts Management Deskbook*, revised edn. New York, NY: American Management Association; 1986, Chapter 6.

16. Ibid., pp. 290–315.

17. See Hajek, *Management of Engineering Projects*. Chapters 8 and 9; and Rosenau, *Successful Project Management*, pp. 34–41.

18. 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, D.C.: NASA Scientific and Technical Information Office, SP-4205; 1979, sections 2.5 and 4.2.

19. A complete description of contracts is given in Hirsch W. *The Contracts Management Deskbook*, revised edn. New York, NY: Amocom; 1986, pp. 43–75. For construction contracts: Furst S. and Ramsey V. (eds) *Keating on Construction Contracts*, 8th edn. London: Sweet and Maxwell; 2006.

20. Hajek, *Management of Engineering Projects*, pp. 82–83.

21. Miller R. *Schedule, Cost, and Profit Control with PERT.* New York, NY: McGraw-Hill; 1963, pp. 173–184.

22. Example from ibid., 174–175.

23. Brooks, Grimwood, and Swenson, *Chariots for Apollo*, Chapters 2–5.

Chapter 4
Project Definition and System Definition

When one door is shut, another opens.

—Cervantes, Don Quixote

If you build it they will come.

—Field of Dreams

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 systems solution, (3) elemental plan in the project 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, and the previous chapter, concern conceptualizing and defining the “system”—the end-item of the project. This is also a major thrust of the systems engineering methodology, which is often applied to projects that involve more stakeholders, greater technical complexity, greater risk, and have farther-reaching consequences than other projects. Interested readers are encouraged to see Appendix A at the end of the 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 definition, design, production, and implementation of the solution. 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, Definition, 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 says "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 decide on 10 floors, after 8 floors have been built you cannot change the decision to 5 floors (without tearing down 3 floors) or to 15 floors (if the foundation was built for only 10).

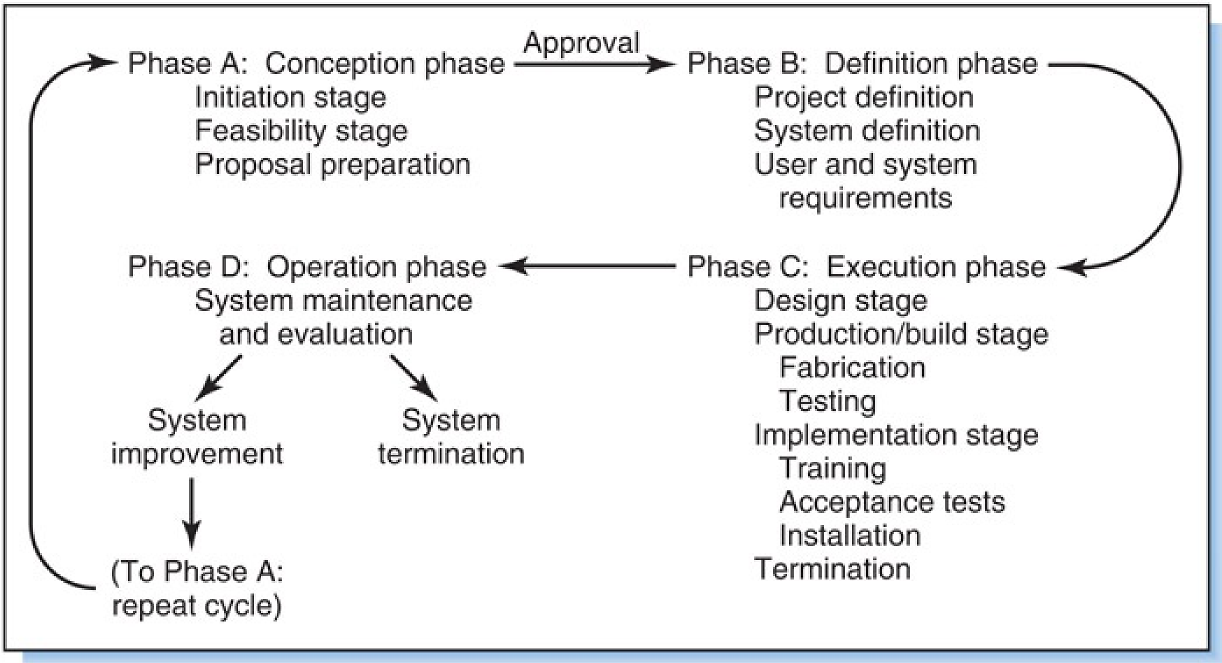


Figure 4.1 Four-phase model of system development cycle.

This is illustrated by three curves in Figure 4.2. Early in the project it is easy to make decisions that will affect the project outcome, and the cost of changing those decisions is little. Early on very 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 thoroughly 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 or *result* of the project, the other is to see the *effort* 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 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 *end-item* must do to satisfy user requirements; project definition aims at specifying what the *project team* must do in the project 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 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.

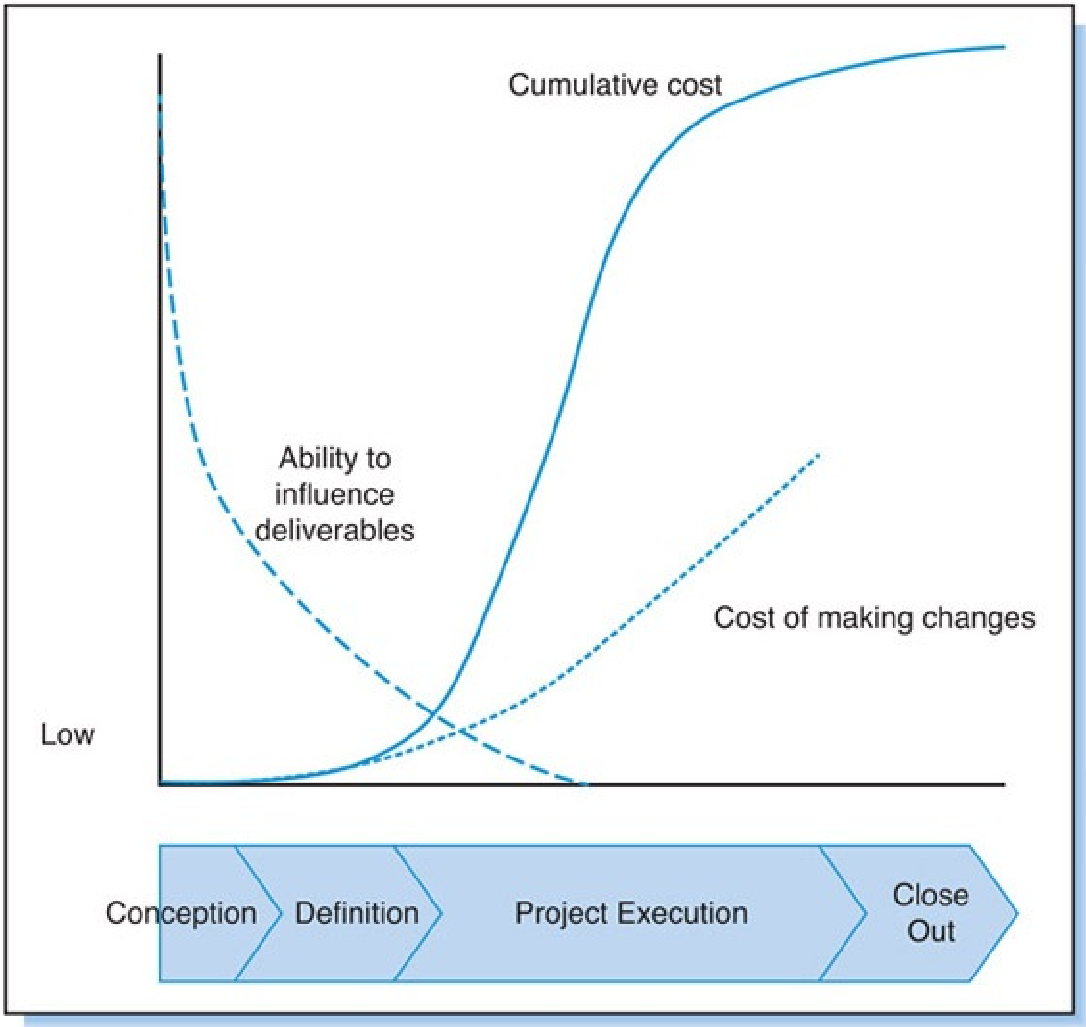


Figure 4.2 Project costs and ability to influence outcomes vs. project phase.

Project Kickoff

The project formally begins with a kickoff meeting—the first formal meeting of the project team and key stakeholders. The purpose of the meeting is to announce that the project is about to commence, communicate what the project 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 with questions and answers at the end.

Invited attendees should be formally notified in advance and provided 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 prior to the meeting.

Every project should start with a kickoff meeting. For a large project, the proposal preparation effort should be preceded by a kickoff meeting; similarly, each project phase should be initiated with a kickoff.

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 shortly after the kickoff. Team building is discussed in Chapter 16.

See Chapter 16

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 again and again 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 always a good idea to first check the project name's acronym before keeping the name; a serious project should not make people chuckle whenever they see its acronym (*A*utomated *N*etwork for *U*niform *S*ecurity).

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 project 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, creating schedules, budgets, and cost control systems, and the project team, its leaders, subcontractors, and support staff.

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 or people with the requisite expertise assigned to the project. Sometimes she seeks the customer's approval in adding members to the project team, which is advisable whenever the customer must work closely with the team or when the customer 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:

- A scope statement or SOW that includes high-level user requirements and system requirements.
- Work breakdown structure and work packages or tasks.
- Project organization and responsibility assignments.
- Assignment 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.
- Work review plan.
- Change control plan.
- Implementation plan to guide conversion to, or adoption of, deliverables.
- Health, safety, and environmental (HSE) policy/plan.

The execution plan is described more fully in Chapter 5, and elements of the plan are described throughout the book. Concerning the last element above, HSE, it is perhaps obvious that project management is responsible for protecting the project team, stakeholders, and society from injury during the project and long-term health hazards associated with the project and its outcomes. Minimally, the project plan must address measures to guard against accidents and health hazards as required to comply with industry standards and municipal, state, and federal laws and regulations, and, additionally, to meet the unique circumstances of the project.[2] Company policies regarding HSE should be included or referenced in the project plan, and the project manager is responsible for ensuring the policies are implemented, that specific HSE roles and responsibilities are defined, and staff receive appropriate H&S training.[3] Significant hazards that cannot be eliminated should be included in the risk management plan. Preparation of the HSE includes consideration of environmental and sustainability matters as discussed in

Chapter 3.

See Chapter 5

See Chapter 3

All of 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, starting with the next chapter. Appendix C at the end of the book is a sample project execution plan.

In large projects the planning is divided into sub-plans created by subordinate members of the project team. The project manager coordinates and oversees their efforts to ensure that the sub-plans are thorough and tie together. The final plan is reviewed for approval by contractor top management and the customer. 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 might contain elements to which the customer objects. 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 materials and work methods must be acceptable to the customer.

Once the project plan and system specifications have been approved by management and the customer, the project team turns its attention to the detailed design and building of the system; this is what happens in Phase C, as covered in Chapters 11 and 12. As will be explained later, however, project planning never stops; it continues throughout the project life cycle.

See Chapter 11 and Chapter 12

4.3 Phased (Rolling Wave) Project Planning

A major thrust of Phase B is to develop the project plan, but seldom does it produce a comprehensive, detailed 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 details in 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).

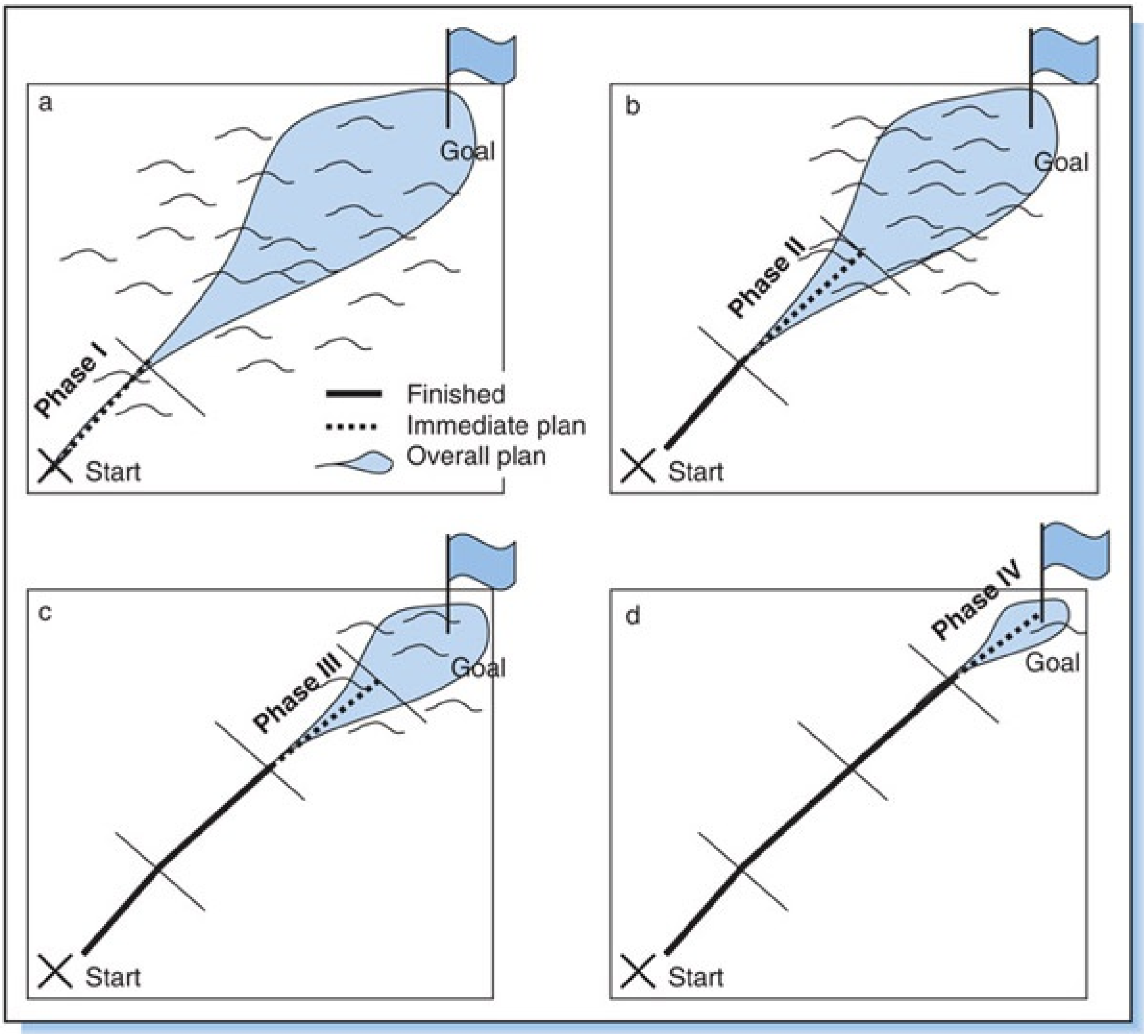


Figure 4.3 Phased project planning.

At the onset 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 estimateproject 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 line, 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 a detailed plan and make commitments to work, dates, and costs. Application of this rolling-wave planning is a major feature of Agile projects, described in Chapter 13.

See Chapter 13

In some projects each phase concludes with a *milestone* or *phase-gate* where the customer or executive managers review the deliverables and project performance; if satisfied, they approve the deliverables and pay for work done thus far. At the same time, they review the detailed plan for the next phase and assess the costs, risks, etc. of the updated high-level plan for the rest of the project. Note that 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; termination before then occurs only as the result of unforeseen events external to the project.

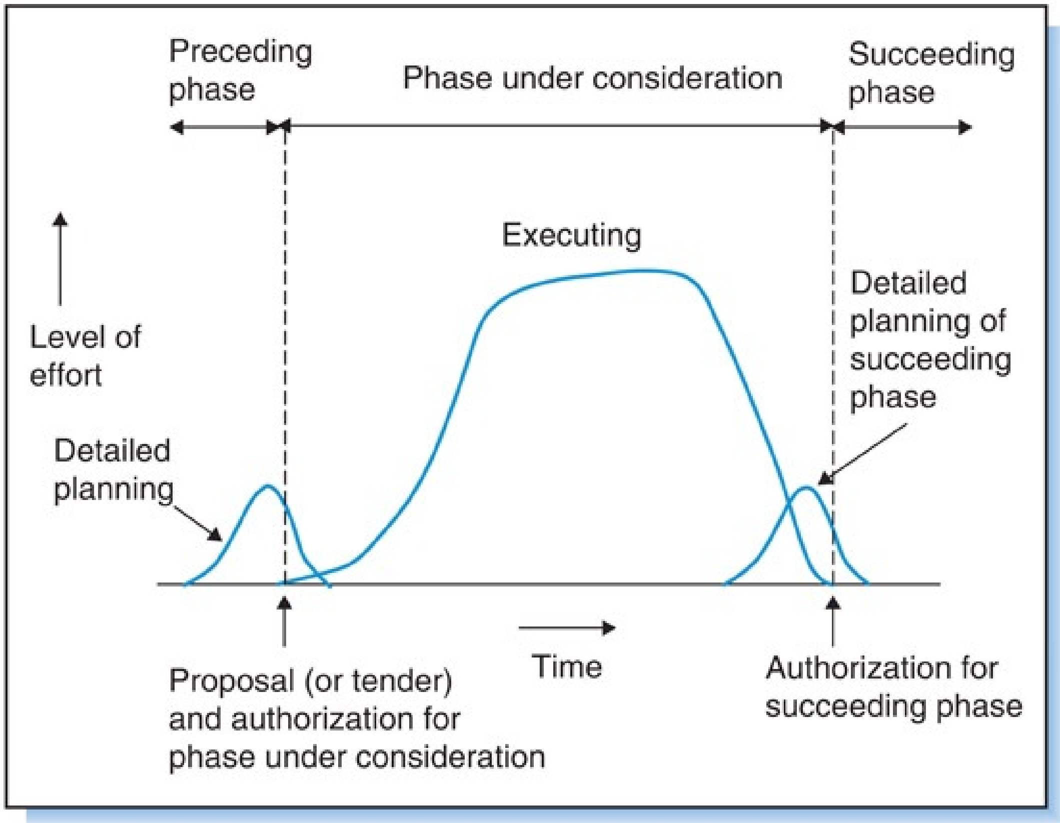


Figure 4.4 Detailed planning for each project phase.

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

In some organizations this "phase-gate" review process is a mere formality and 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:

(a) Green Light: all relevant stakeholders are satisfied with the work per form ed so far. Also, they accept the plans for the rest of the project and that the risks identified and the business impact of the project justify continuing with the project. The project is authorised to proceed to next phase.
(b) Yellow Light: the stakeholders feel the business impact of the project justifies continuing the project but do not feel that objectives of the previous 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 improve the plan.

(c) Red Light: due to changes in the business environment, risks, or disappointing results from the preceding phase, stakeholders consider the business impact of the project as insufficient. The project is terminated. If the possibility exists that conditions might improve later, the project can be "put on ice" for later reconsideration.

In companies that do multiple simultaneous projects, phase-gate reviews are sometimes used to compare the projects—their benefits, resource requirements, and relative performance, and to determine which get the green, yellow, or red light; this is discussed in Chapter 18. The phased approach to project planning and approval is illustrated with Mary and Peter.

See Chapter 18

Example 4.1: Mary and Peter's New House

Mary and Peter buy property to build a new house upon. They approach NewHome Construction and describe to Paul, the owner, what they have in mind. Among other things, they want to know what it would cost. Having been in the business for a number of years, Paul has an idea of the cost but is wary to quote a fixed price since he doesn't 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 an estimate of when the house would be completed. Nobody has as yet made any commitments. On the question, "Where do we go from here?" Paul answers that the first phase is to do a concept design, after which he will deliver sketches of the house. He also describes the other phases of the project he foresees, and the deliverables, approximate schedule, and approximate cost for each.

Mary and Peter sign a contract for Paul to provide a preliminary 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 the detailed design to include drawings that the construction team will use to build the house. Just like the first phase, the contract specifies the deliverables (drawings), the delivery date, and the 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 opportunity to inspect the work and verify that it has been completed and meets workmanship standards as 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 phases form the basis of project methodologies and phase-gating as discussed in Chapters 17 and 18. In organizations that have project methodologies, project managers follow the prescribed sequence of standard phases for planning and executing projects.

See Chapter 17 and Chapter 18

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 others, 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 start of the project. For *external* projects, that purpose is served by a contract, so, generally, no charter is required.

The charter describes the project to stakeholders in the organization and establishes the project manager's authority to organize and make use of 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 similar to the project plan. 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 documents ("project initiation documentation") authorizes subsequent stages.

For a small project or initial phase of a project, the charter can be a short document simply stating approval for the project or phase. 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.
- Plans for: procurement: safety, health, environmental protection; communication.
- Control procedures.

Despite similarities, the project charter and the execution plan differ in two important ways. First, the purpose of the charter is to *describe, justify,* and *authorize* the project; the purpose of the execution plan is to *give direction* to stakeholders working in the project. This leads to major differences in the content of each and, usually, an execution plan that is substantially longer, more detailed, and comprehensive than the charter.

Front-end Loading[4]

A variation of phased project planning and approval used by some industries (chemical, mineral, oil, and gas) in major industrial infrastructure projects (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 18). The output of FEL-1 is a preliminary business case that confirms the feasibility of the proposed capital investment.

See Chapter 18

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 as well as a scope statement that enables reliable cost and schedule forecasts. Typically only 1 percent of the total project costs are incurred during FEL-1 and FEL-2.

FEL-3 is "project definition" and 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," "pre-feasibility/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 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. Requirements are therefore 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 impact project work effort, time, cost, and risk. Unless the requirements are clearly defined and agreed upon, it will be impossible 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 Conception should be reviewed 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-22, taste better than Joe's beef jerky, obtain at least a 20 percent rate of return, or upgrade to the latest software release. General requirements such as these must be expanded before serious development work and project planning can be started. As shown in the next example, 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 set by studying competitors' products. The food processor as developed met all the requirements set by marketing and engineering, yet it was obsolete before it even 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" or "low cost").
- The user or contractor are unable to adequately describe the requirements because the end-result is complex, abstract, or artistic.
- The customer or contractor intentionally define requirements in ambiguous terms to allow latitude in results later in the project.

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

- Convince both the user and contractor groups of the importance of 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 both the user and the

contractor 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 or not to approve them.

Detailed user requirements come from one source: the user. The project manager, however, should not be accepting of just any requirements provided by the user, but should offer assistance in 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 cost, schedule, or other ramifications of requirements, 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. For example, besides being profitable the contractor might specify requirements to keep skilled workers and costly production facilities occupied.

System requirements define the system or solution approach—including the principle functions, system architecture, and resulting end-item (system, solution, or product)—and provide a common 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 systems must *do* to satisfy the user requirements. Following are examples contrasting user requirements and system requirements:

User Requirements	System Requirements Will Address
1. Vehicle must accelerate from 0 to 60 mph in ten 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 what the project's designers and builders must address in designing and building the end-item. The following illustrates this for the X-Prize/SpaceShipOne project introduced in Chapter 1.

See 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 specify 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 acceleration force must not exceed certain level
    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 sufficiently so that 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 that the new system must be able to 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 to 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, decibels of sound, 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 on specific subsystems and 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 whole 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*. Total cost of the system if a particular design is adopted. This includes the cost of the design as well as the cost for implementing and operating the design to achieve a given level of benefit.
4. *Reliability*. Ability of the system or component to function at a given level or for a given period of time before failing.
5. *Maintainability*. Ability of the system to be repaired within a certain period of time (i.e., the *ease* with which it can be repaired).
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 or strain, technical skill, training, or ability required for operating and maintaining the system.
9. *Robustness*. Ability of the system to survive in a harsh environment.
10. *Expandability*. Ability of the system to be easily expanded to include new functions or be adapted to new conditions.

These requirements are sometimes called "non-functional" requirements (!) because they are not tied to particular functions and are desired of the entire system and its components.

Requirements Priority and Margin

Two properties of each requirement are its priority and margin. 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 perform in a certain way and be a particular maximum height, but performance has priority. Knowing 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.

Requirements Breakdown Structure

During requirements analysis, system functions are sorted and assigned to 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 purpose of the RBS is to provide 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 detailed enough 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.*

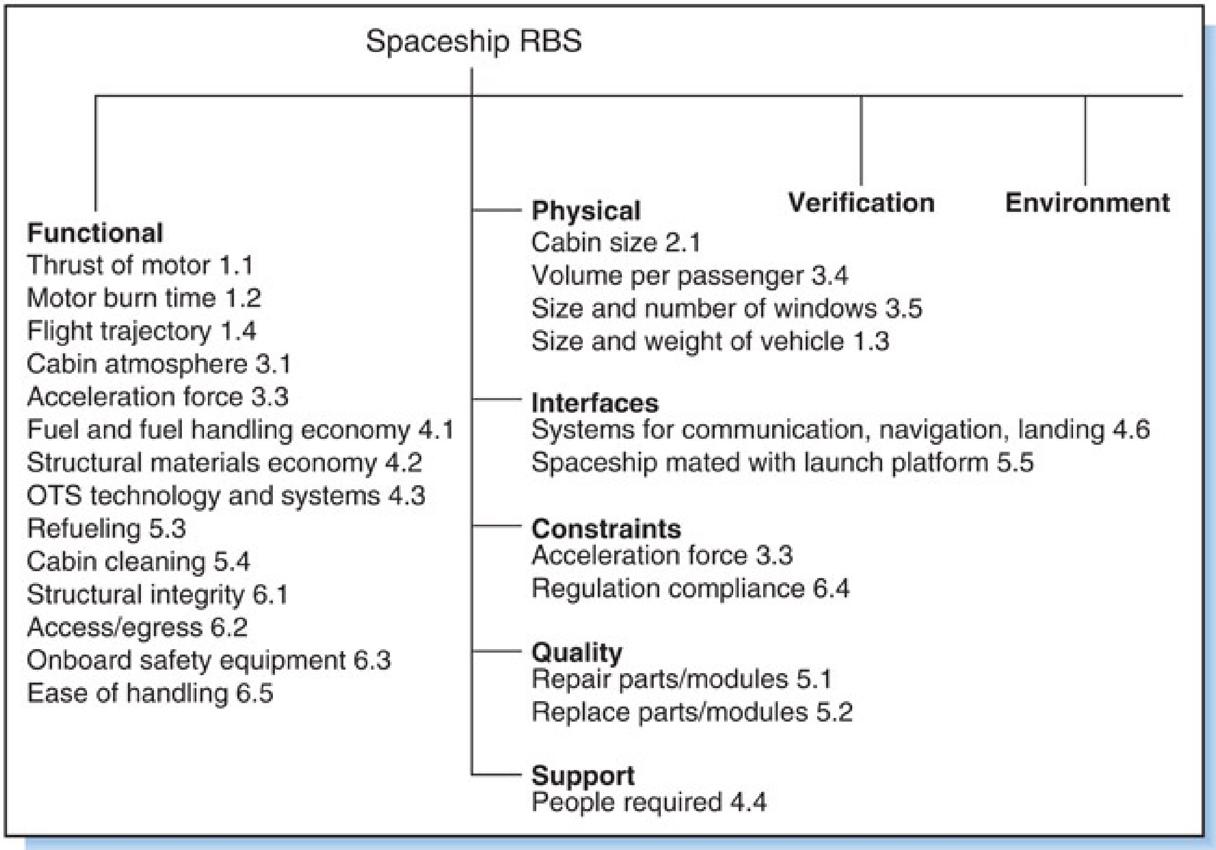


Figure 4.5 Requirements breakdown structure.

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 even lower-level subsystems. From the system specification for a new 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.

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 kilo newton, is a measure of force.) 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, this says the team will have to design and build one from scratch. The multiple arrows to each specification in the last column indicate that each specification must satisfy multiple requirements.

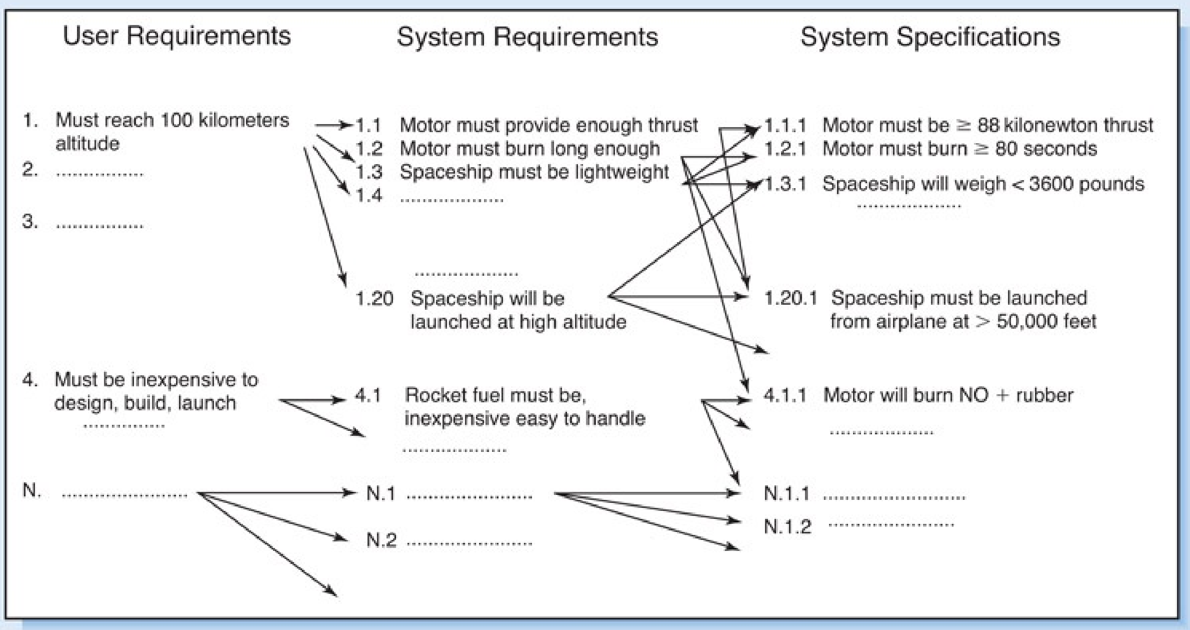


Figure 4.6 Relationships between user requirements, system requirements, and system specifications for spaceship.

Traceability

Developing clear specifications is important, but so is keeping track of their relationships to each other and to system requirements. Throughout the systems 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 traceability matrix, described in Appendix A to this chapter. The process of managing all of this—identifying specifications, tying them to physical components, tracing the impacts of changes, and *controlling* changes so requirements are met and do not conflict is called *configuration management* and *change control*, discussed in Chapters 9 and 11.

See Chapter 9 and Chapter 11

System specifications are the criteria that will guide actual project work; they are written by and for project subject-matter specialists—systems analysts, programmers, engineers, product and process designers, consultants, etc.—and they address all areas of the project—design, fabrication, installation, operation, and maintenance. System specifications should be set so as to meet but not exceed *customer baseline specifications,* which are high-level specifications the customer can understand. This is one way to prevent "scope creep," i.e., the growth of project requirements that causes project budgets and schedules to also grow.

Iterative Design-Testing and Rapid Prototyping

The definition of requirements and specifications, and the design and testing of the system usually happens iteratively, particularly when the project end-item is complex. The requirements cannot be completely defined without some amount of prior design work, and the design work cannot be completed without some amount of prior fabrication and testing. The overall process generally cascades down as illustrated in Figure 4.7, with occasional loops-back and repeated steps. Work that flows from stage to stage like this is called the *waterfall process*. To assess and modify specifications, often a *prototype* is used. A prototype is an early running model of a system or component built for purposes of demonstrating performance, functionality, or proving feasibility. It is built according to initial specifications and then tested; if, based upon tests, the specifications are changed, then the prototype is modified and tested again. This process ensures that the basic system design supports the system specifications.

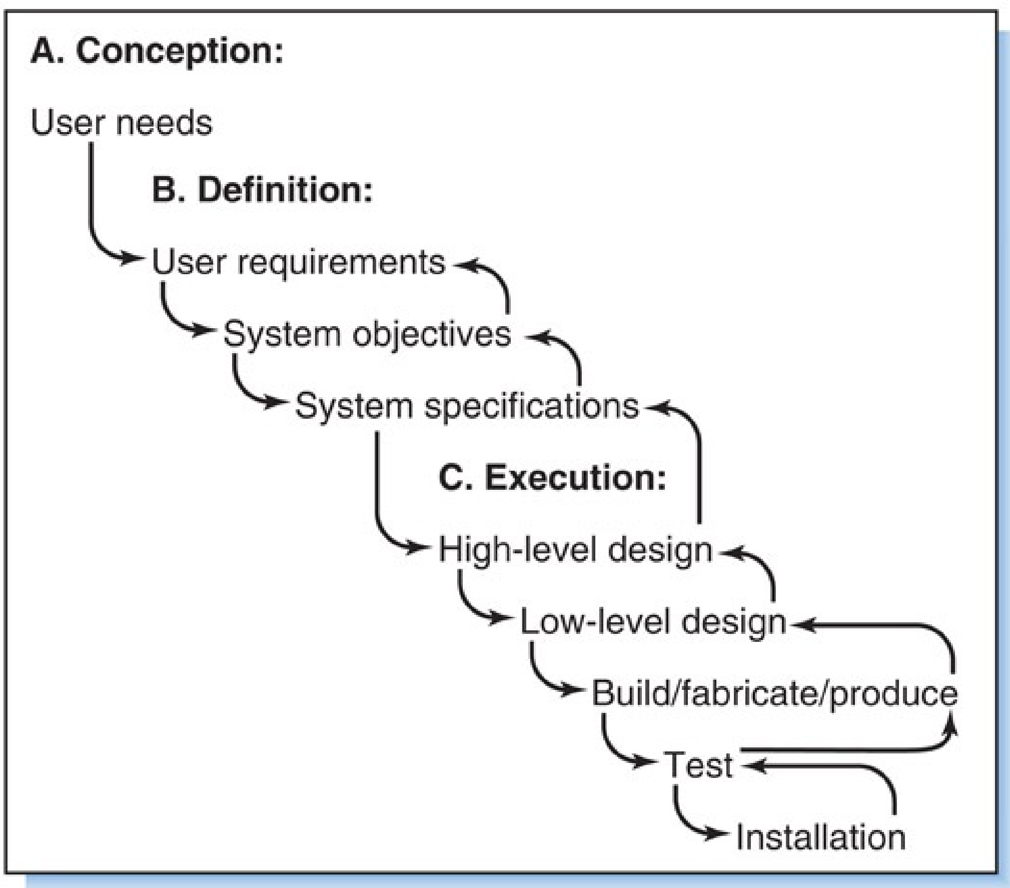


Figure 4.7 Iterative development cycle (waterfall process) for complex systems.

It can be difficult to conceptualize the system when no system exists like the one to be developed. The system that the customer "sees" might be very different from the one the developer envisions, yet without a physical or working model the difference might not be apparent. Requiring the customer to specify and sign-off on requirements early in the project only intensifies the problem. It forces the customer and developer to commit to decisions before they reach a mutual understanding about the requirements.

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 but *not* the complete system, and is somewhat easy to create and modify. The customer experiments with the RP to assess the system's functionality and determine any necessary modifications or additions. After a few iterations of experimenting and modifying requirements, the final requirements and design concept are firmed up. In software development, the RP might be a series of screens or windows with queries to allow a user to "feel" what the system would be like. Architects use physical scale models of buildings for the same purpose. They know that a physical model is always better than a drawing or lists of requirement 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 drawings and requirements are finalized only after the customer has accepted them as represented by the RP model.

Ordinarily, the RP process will not speed up the definition phase; instead, it might lengthen it. The first RP model will likely be incorrect, though it will enable the customer and developer to experiment, learn, and eventually select optimum requirements. RP models and mockups are used, for example, to demonstrate the form and functionality of the shapes and sizes of control panels used in plants and equipment, and to design the interior layouts of automobiles and aircraft cabins.

Agile Project Management

The waterfall approach in Figure 4.7 (and, generally, the phases of A, B, and C in

Figure 4.1) applies to projects where the requirements can be defined early in the project and will not change afterward. Such situations are like a waterfall: the projects move “down” from one stage to the next. But also like a waterfall, it is hard to go the other way, which is analogous to what happens when the stages of a project must be repeated. When the requirements in a project cannot be completely defined early on or will change significantly, then steps in the project have to be repeated. The waterfall process is able to accommodate this (the back-arrows in Figure 4.7), although not very effectively, because altering the requirements midstream is costly and time-consuming. Waterfall applies to projects where requirements can and must be defined early (e.g. designing and constructing a new building or airplane); for projects where the requirements cannot be defined or will certainly change (e.g. some software projects), so-called *agile* methods are better.

In *agile project management,* the project is divided into a sequence of small, iterative efforts, each conducted by a team devoted to meeting a limited set of requirements and releasing a partial result or solution. The end-item system is developed in a series of quick iterations, where, in effect, the stages of definition, design, development, and testing are repeated. Each iteration (called a “sprint” because it is short—a month or less) delivers a partial yet stand-alone and fully functional result. At whatever time the project is terminated, the customer is left with usable results as produced up to that point. Agile project management is the topic of Chapter 13.

See Chapter 13

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, suppliers, functional areas such as engineering, marketing, manufacturing, customer service, and purchasing, and users and operators. The more these individuals and groups have a hand in defining requirements and the project plan, the better the project plan will account for the requirements and the 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, a good way to generate product requirements is at an off-site workshop for all the key project stakeholders, including functional groups, users, and suppliers. Beginning with a list of customer needs or requirements, the team develops the system requirements (or, lacking adequate user requirements, develops them too). For complex systems, a better approach is to create a team comprised of all key stakeholders for the purpose of defining 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 because concurrent engineering involves not just engineering but marketing, purchasing, finance, quality, and more.

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

Example 4.5: Design-build Teams at Boeing[8]

At one time in the Boeing factory the production plant was located on the main floor and the engineering group was upstairs. Whenever a problem occurred in the plant, engineers just walked down to take a look. Today, Boeing employs many thousands of people at several locations, and such easy interaction isn't possible. Similar to 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, out-board 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. The concept of having so many people at design meetings—people from airlines, finance, production, and quality—was totally new, but with so many people representing so many interests there were actually few conflicts.

Since most components in an airplane interact (interface) with numerous others, most participants in the program had to be assigned to multiple DBTs (to insure their components would work with other DBT’s components). The manufacturing representative, for instance, belonged to 27. His duty was to tell engineers what would happen when their elegant designs met with the realities of metal, 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 a lot of electrical and hydraulic components that maintenance personnel would need to access, but engineers hadn’t noticed that repairing the components would require removal of the entire faring. The manufacturing rep noticed, however, and suggested adding two big doors, one on each side of the faring. This improved access to the components inside and greatly simplified their repair.

Concurrent engineering teams are discussed more fully in Chapter 14. Another practice for defining requirements and keeping the project focused on them is *quality function deployment (QFD)*. This is covered in Appendix B to this chapter.

See Chapter 14

4.5 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. 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. Third, 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 front-end phases of a project—conceptualization and definition—are important to the viability and success of the project. What is surprising is the short-shrift given to user and systems requirements definition in so many projects, and the impetus to begin preparing a plan without even knowing what the end result of the project is supposed to be. 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 he wants it, and how much he 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 end result must be and do—the user requirements and the system requirements.

Appendix A: Stages of Systems Engineering[9]

The systems engineering methodology described in Chapter 2 follows a series of stages that closely 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 ultimate deployment and operation by a user. The process, outlined in Figure 4.8, 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).

See Chapter 2

Stage 1. Needs Identification and Conceptual Design[10]

The main tasks of this stage, which are analogous to project life cycle Phase A, are to define stakeholder needs and requirements, perform 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 of the intended end-item system.

Stakeholder and Needs Identification

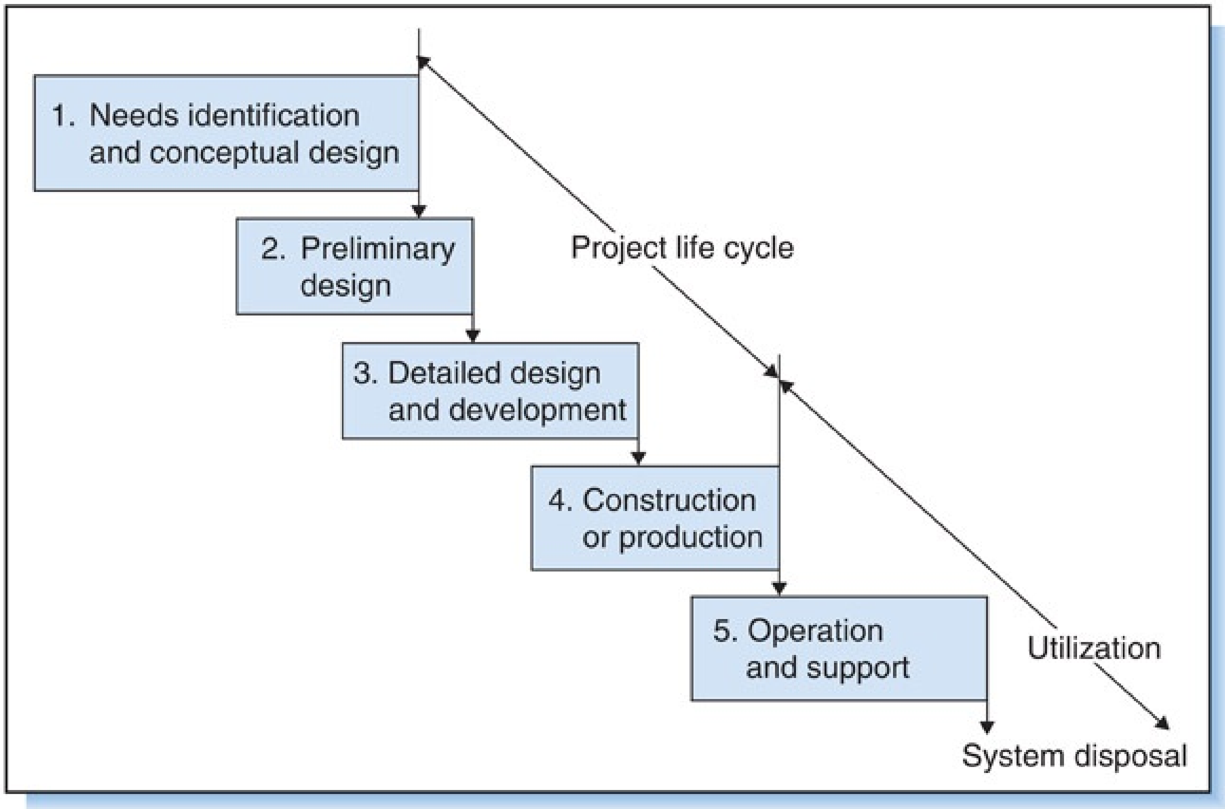


Figure 4.8 Stages of the systems engineering process.

Systems engineering deals with poorly defined problems. The customer may feel that something is wrong, or something new is required but be unclear about the source of problem or need, or what thesystem should look like or do. Sometimes it is not even clear *who* has the problem or need. The first step in systems engineering is identifying the parties who will be affected by or able to impact the system. Even identifying the "customer" is not trivial; it might be an organization, but within the organization only certain parties will have the authority to make decisions relating to the system, or will use, operate, or be impacted by it. These parties must be singled out and their needs identified.

Developing a clear conception 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? What is the value of the system that will solve the problem?
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 back and forth 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 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: 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, parts availability, etc.; 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 principle stakeholders have reviewed and endorsed the SRD.

Example A4.1 SRD for the Spaceship[12]

As an example, let’s revisit the X-Prize/SpaceShipOne project described in Chapter 1. The criteria of the competition were to send a reusable vehicle capable of carrying three people into space twice within 2 weeks. Besides winning the X-Prize, a 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:

See Chapter 1

1. Can minimally attain 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* to be able to meet the requirements in the SRD. For example, the stakeholder requirement that the spaceship “provide comfortable flight” implies system requirements that the spaceship’s cabin temperature, humidity, and pressure all remain 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 designer 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

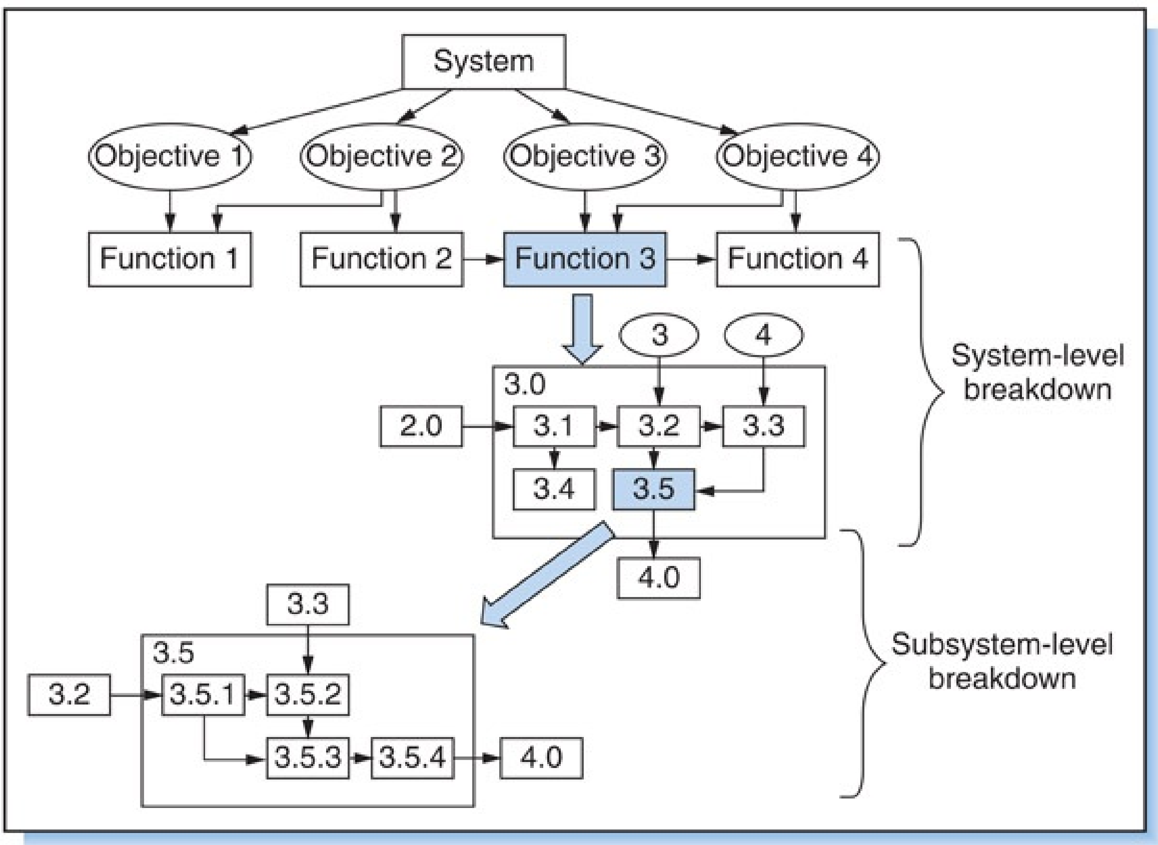


Figure 4.9 FFBD for decomposing system-level functions into lower-level functions.

Functional requirements specify the functions that the new system 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 4.9. 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 conceptualdesign 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.

Notice the numbering scheme used in Figure 4.9: each and every function has a unique identifier that enables it to be traced to the original system-level function; e.g. function 3.5.4 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 on, 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 astronauts.

Example A4.2: Functional Requirements Breakdown for the Spaceship

Figure 4.10 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.

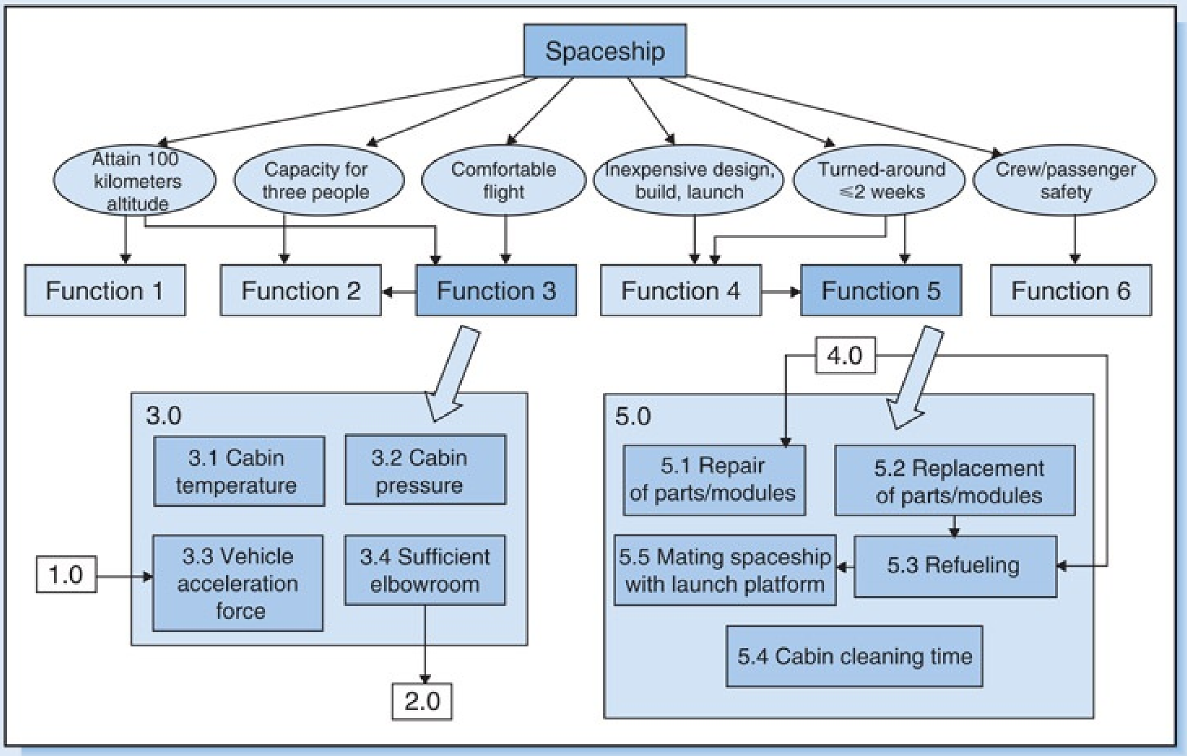


Figure 4.10 System-level breakdown of functions for spaceship.

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 for 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, measures, 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 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 different 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 less costly. 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 Fuel, fuel system, and fuel 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 simple and inexpensive to fuel and operate, safe, 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 Design[13]

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 Phase B. Studies are performed of the high-level elements comprising the system, and the system-level requirements are *allocated* among the subsystems.

Functions of Subsystems

The FFBD process as illustrated in Figure 4.10 is now repeated to decompose the system-level functions into subsystem-level functions and, as before, to define functional, performance, and testing requirements for each functional block. The functions are decomposed to whatever level necessary to completely define each subsystem and permit decisions about whether each function can be met with an OTS design or product, or whether it must be designed and built from scratch. In preliminary design, there is a subtle shift in focus away from *what* the system will do to *how* it will do it. The shift is from *functional* design to *physical* design.

Example A4.4 Decomposing Functions into Subfunctions

Figure 4.11 shows the FFBD for function 5.5, mating (attaching) the spaceship with the launch vehicle. This requirement is derived from the system-level requirement of "turnaround in 2 weeks or less."

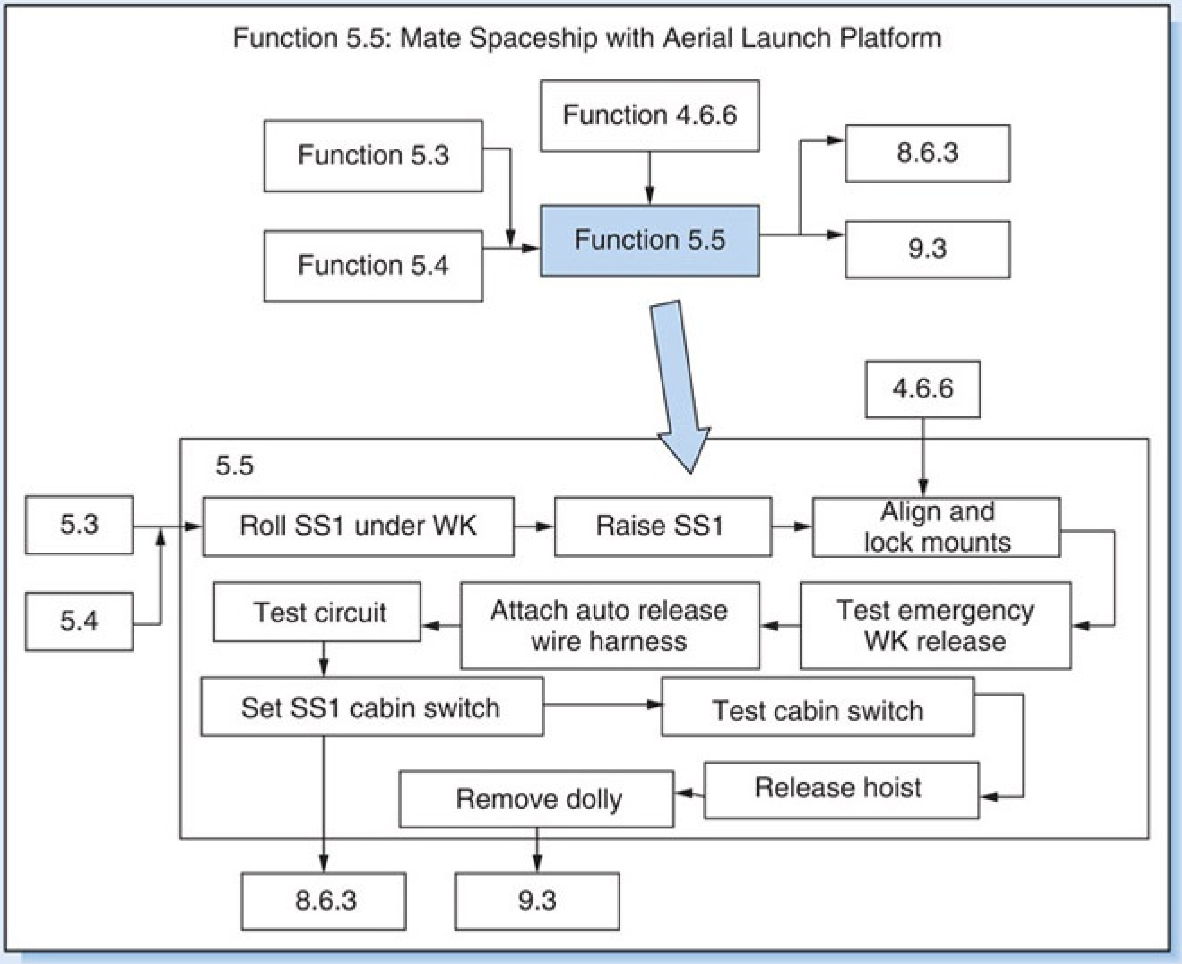


Figure 4.11 FFBD for mating SpaceShipOne and White Knight.

Suppose the performance requirement for mating the spaceship to the underbelly of the mothership is set at 10 hours. Having decomposed the function into all of the subfunctions in the procedure, planners are then able to set time requirements for the subfunctions such that the overall mating procedure will not exceed 10 hours.

Grouping of Functions: Architecture and Configuration Items

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, handle bar.
- *Configuration:* wheels attached at ends of frame; front wheel pivots on frame; seat mounted on frame between wheels; pedals attached to frame, linked by chain to rear wheel; etc.

Sometimes the architecture "looks right," sometimes not. Often, in order to satisfy unique requirements, designers are forced to stray from the commonplace architecture, the result being a "funny-looking" architecture.

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.

Each major subsystem will perform a major function or set of system-level functions as listed in the functional baseline. From this point onward, each of these subsystems will be called a *configuration item* or CI. In general, a CI is a subsystem or component whose history is documented and monitored throughout the system's complete life cycle—its design, production, and usage. The purpose of this documenting and tracking (i.e., *configuration management*) is to ensure that any changes in the design, production, or usage of the CI do not alter or degrade its 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 any items identified as critical to performance, risky, or costly.

Requirements Allocation

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 CIs (the system architecture). 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 4.12, the matrix columns are the subsystems responsible for meeting the requirements; the matrix 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 4.12 that responsibility for some requirements is shared by more than one CI. For example, the weight of the system (requirement 1.5) is shared by all the CIs. That is to say, the spacecraft weight is the sum of the weights of all the CIs, and if the weight of any one is changed, so is the weight of the spacecraft. If the maximum loaded weight of the spacecraft is set at 3,600 kg, the CIs must be designed so that all of them combined will not exceed that weight.

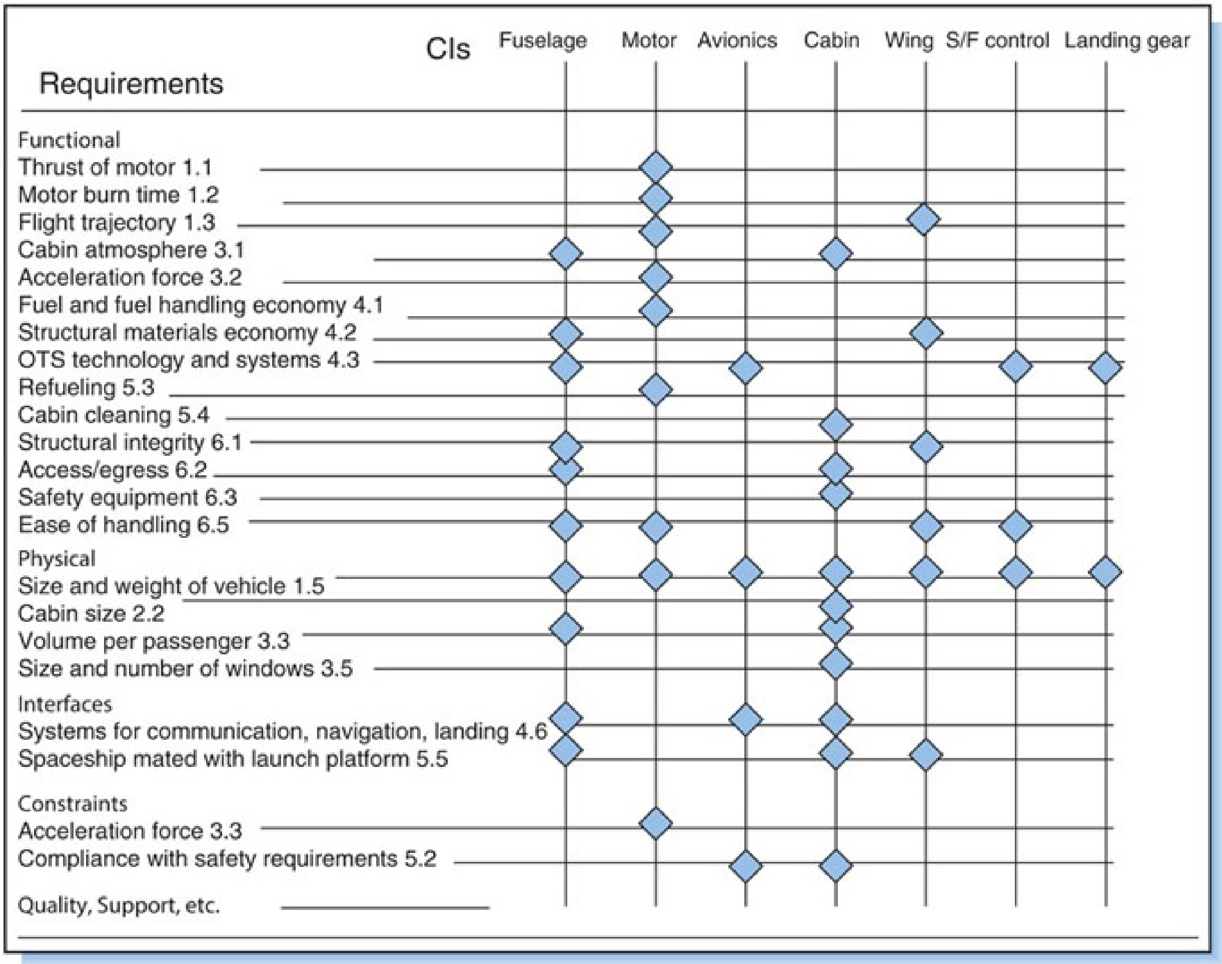


Figure 4.12 Allocation or traceability matrix.

Source: Adapted from Falconbridge R. and Ryan M. *Managing Complex Technical Projects*. Boston: Artech; 2003, p. 78.

Example A4.6: Allocation of Weight among CIs

Question: How do you design and develop all of the CIs such that in the end the total weight (shared requirement) does not exceed 3,600 kg? Answer:

estimate the percentage of the total spaceship weight that each 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 x 3,600 kg = 1,080 kg, the motor target weight is 0.20 x 3,600 kg = 720 kg, and so on. Since achieving targets is critical, each is designated as a *Technical Performance Measure*, or TPM, 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 by or the target spaceship weight increased by 30 kg. Throughout the design process it will be necessary to adjust the CI targets and allocations as guided by the TPM process. This process is described in Chapter 11.

See Chapter 11

Interfaces

None of the subsystems function independently; 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 4.11 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 and the "flow" of something between functions. The "thing" flowing 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 fuselage of the spacecraft contains the motor and also supports the wings, neither wings nor motor can be designed without also considering the design of the fuselage, 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 the subsystems 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 buy or modify an OTS design or product, or to develop a new design from scratch. An OTS design or product that meets all or most of the requirements for a CI and is not too costly will be purchased; otherwise the CI must be designed 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 a big deal 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 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. Suppose also that to propel a vehicle of weight w into space will require a rocket motor with thrust of y (Figure 4.13, 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.

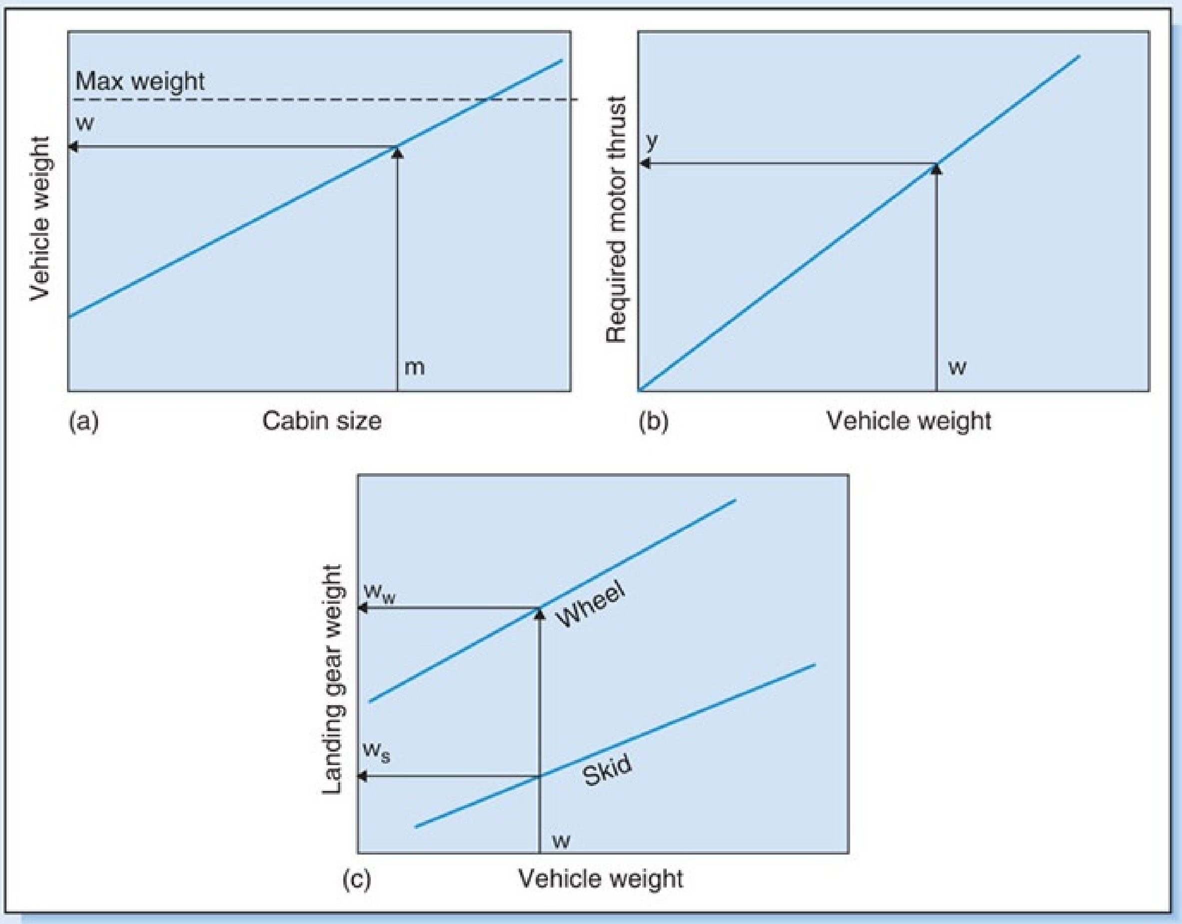


Figure 4.13 Impact of cabin size on vehicle weight, rocket thrust, and landing gear.

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 4.13, 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 physical appearance of the system begins to take 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 4.14).

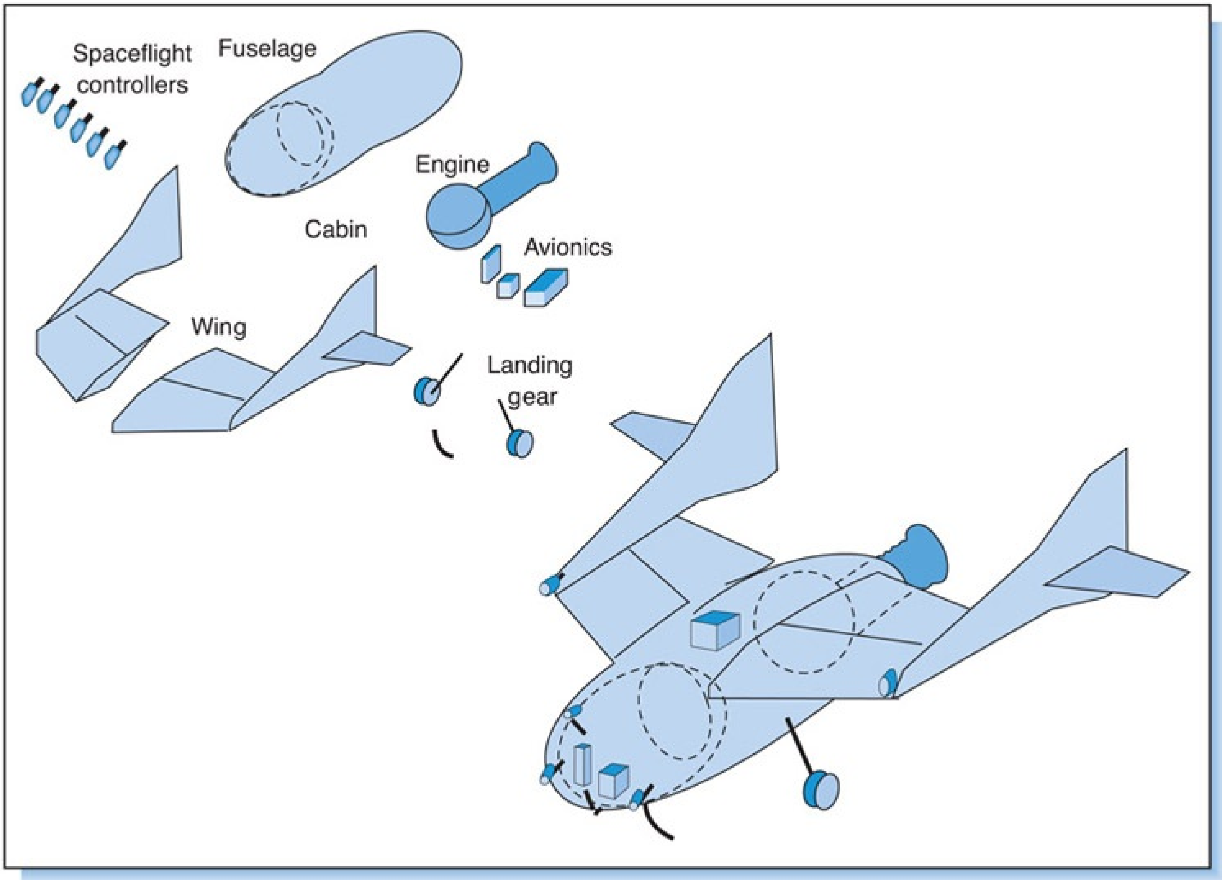


Figure 4.14 Pictorial representation of major subsystems (CIs) and allocated baseline design. (The "funny-looking" architecture derives from the spaceship having to meet many requirements. On re-entry, the wings rotate upward, making the spacecraft one big airbrake that floats to Earth like a shuttlecock, thus avoiding high speed and high temperature. Nearer to the ground the wings tilt back and the ship glides to a landing.)

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.

Everything up to this point has been analytical in nature. With detailed design, the development process moves from "concepts on paper or computer"—the SRD, system specifications, FFBDs—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 input-output will be manual, mechanical, electronic, and so on. Available OTS components are selected on the basis of 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 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 9.

See Chapter 9

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.
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 come to light during these tests. Designs are modified to eliminate deficiencies, or to correct them, 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 any of 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, phasing out, and disposing of the system at the end of its life cycle. The last way, system closeout and disposal, is a major consideration in the design and operation of systems that have potential to degrade 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 wastes, and remove toxins from soil and water, which can be expensive, time-consuming, and extend the system life cycle by years or decades.

Example A4.9: Life Cycle of SpaceShipOne

Preliminary development of SS1 and its support systems began in 1999, and full development began in April 2001, albeit in total secrecy. Exactly 2 years later Dick Rutan announced intentions to capture the X-Prize and flight-testing began (Figure 4.15).

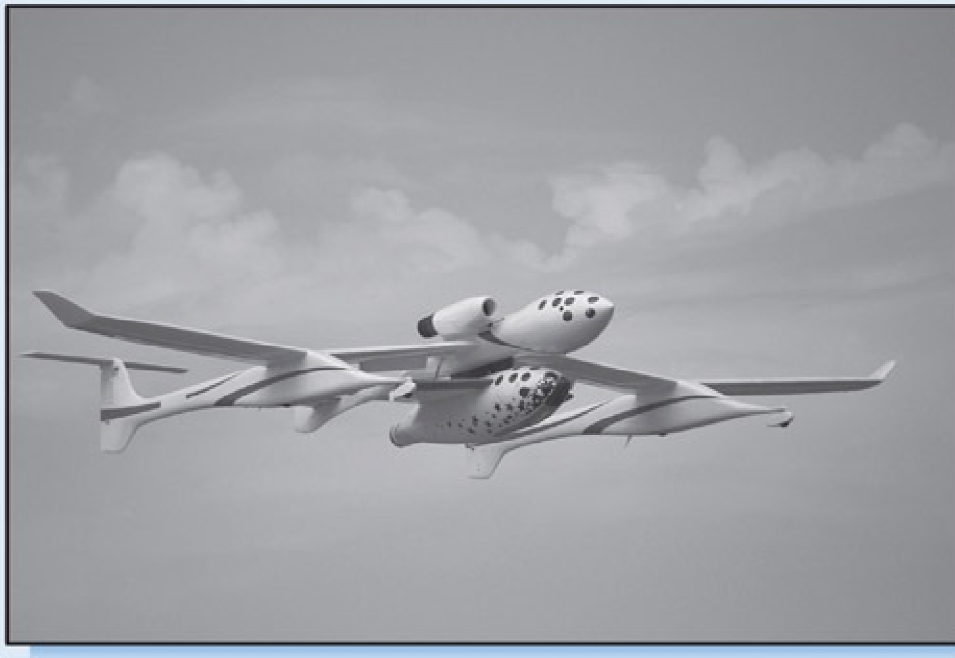


Figure 4.15 SS1 beneath mothership White Knight.

Photo courtesy of John Nicholas

In May 2004, Mike Melville piloted the craft on a test above 100 km, making him the world's first civilian astronaut. On October 29 he again flew SS1 into space, and less than 2 weeks later so did pilot Brian Binney, winning the $10 million X-Prize for the SS1 team (Figure 4.16). Today SS1 hangs in display at the Smithsonian Air & Space Museum in Washington, D.C. 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 eventually to operate a fleet of them.

Figure 4.16 Designer Burt Rutan (center) and pilots Mike Melville (left) and Brian Binney. Photo courtesy of John Nicholas.

Appendix B: Quality Function Deployment[15]

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 throughout the world.

House of Quality[16]

QFD mandates that the project team articulate the means by which the product or system being designed will achieve customer requirements. The process starts with market needs or customer requirements and then uses a planning matrix called the *House of Quality* to translate the needs or requirements into technical requirements. The structure of the house is shown in Figure 4.17.

- 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.

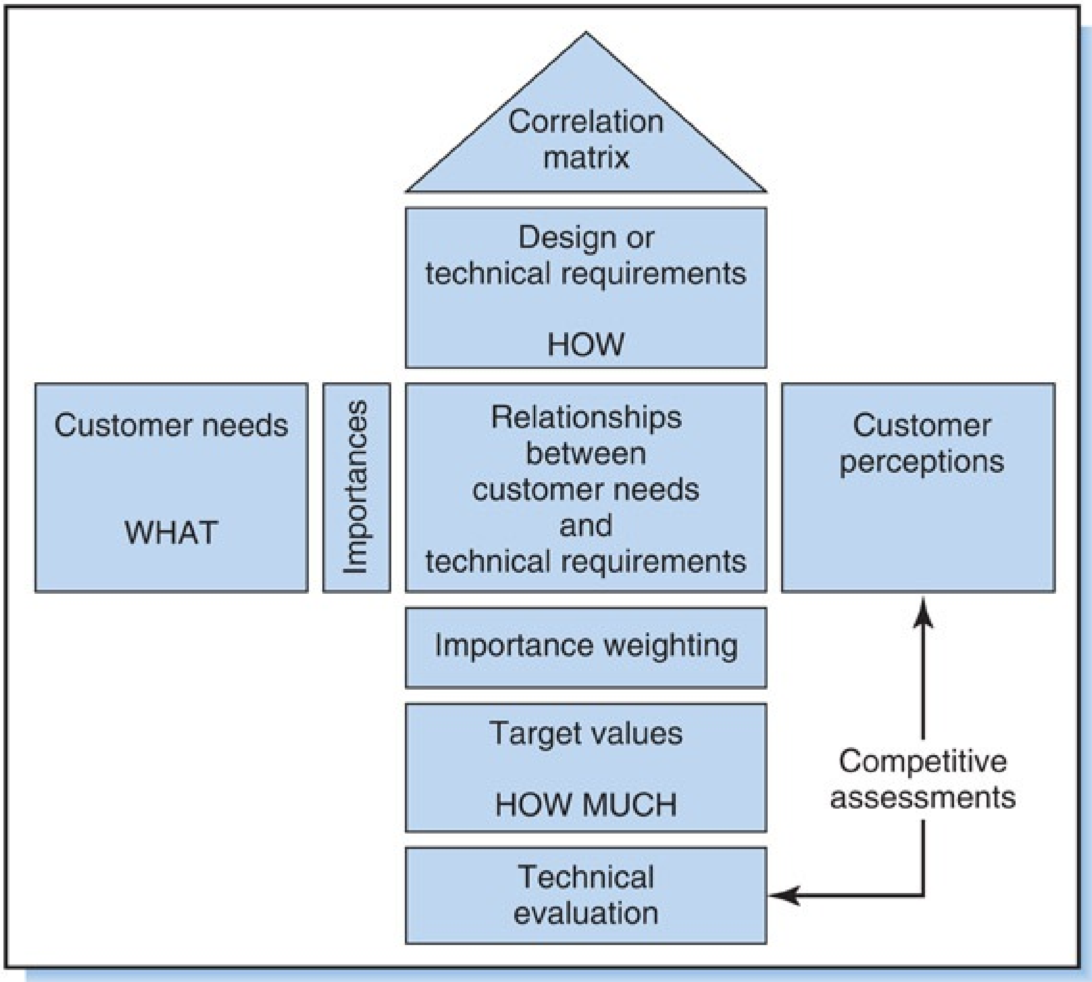


Figure 4.17 Structure of the house of quality.

Features of the house of quality are illustrated in Example A4.10.

Example A4.10: House of Quality for a TV Remote-Control Switch

Figure 4.18 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 are what customers think is important about the product. They are the product "whats."
- Importance to customer: The requirements have been rank ordered 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."
- Central matrix: Contains symbols that show the strength of the relationship between the whats and the hows (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).

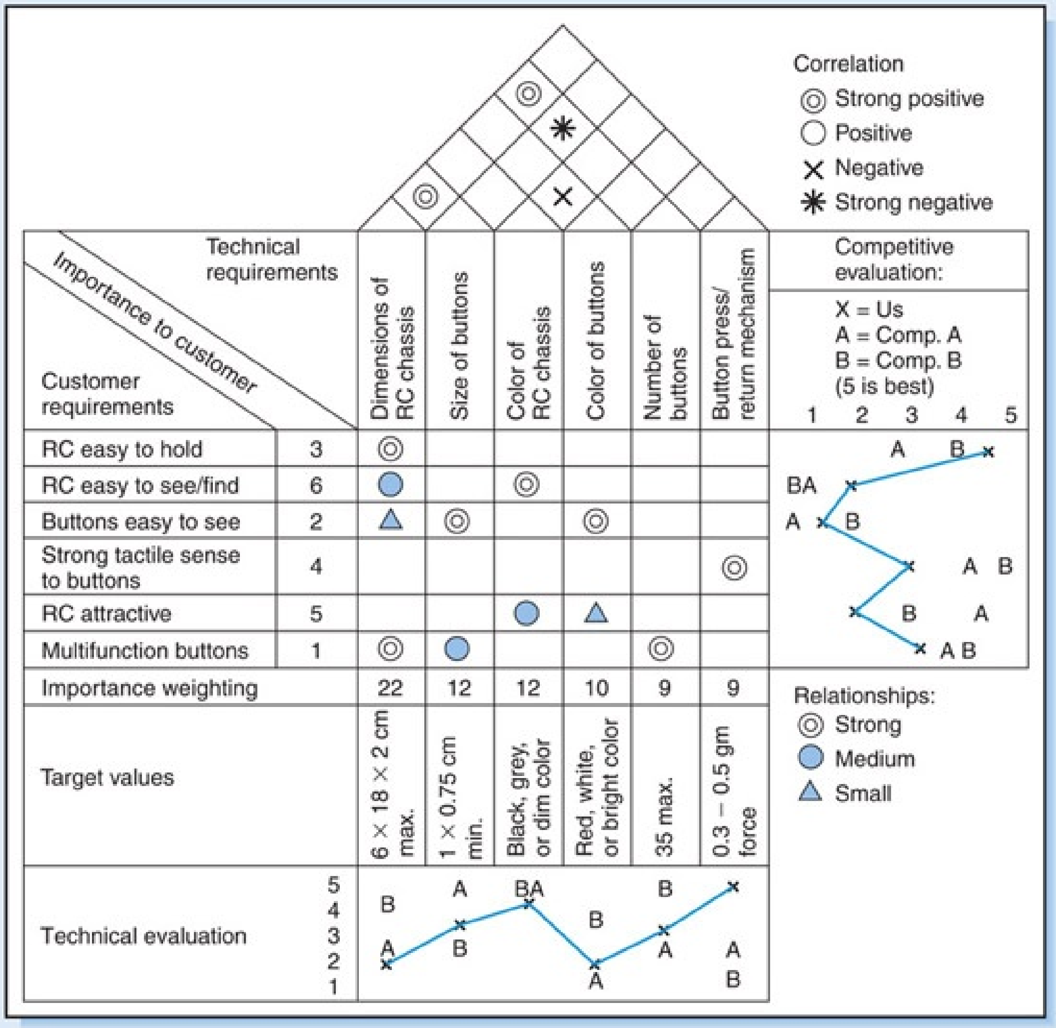


Figure 4.18 House of quality for television remote-control switch.

- 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 4.18 indicates that no company does particularly well in terms of "buttons easy to see" despite the fact that 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 visibility of the buttons by increasing their 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 go into the house. At minimum, the 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 objectively meets technical requirements. The data come from many sources, including 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 technical parameters of greatest importance to performance. These parameters become the ones for which designers seek the "optimum" values. Once the optimum values have been set, the analysis moves on to identify important factors in the manufacturing process necessary to achieve the design requirements. The house of quality is just the first of several steps in the QFD process that leads to a project plan.

QFD Process[17]

The QFD process employs a series of matrices in a multi-phased approach to project planning. The process, shown in Figure 4.19, utilizes four matrices that correspond to four project phases: project planning, product design, process planning, and process control planning. The phases (circled numbers) are as follows.

1. Create the "house of quality" matrix (A). This matrix converts customer needs or requirements into technical requirements.
2. Develop an initial version of the project plan based upon the house of quality requirements. The house matrix does not have to be completed; start with a rudimentary plan using information available from the house matrix, then expand the plan as new requirements emerge in the updated matrix.
3. Create the design matrix (B). This matrix converts technical requirements from the house matrix into product design features and requirements.
4. Create the process matrix (C). This matrix converts design features and requirements from the design matrix into process steps or production requirements.
5. Create the control matrix (D). This matrix converts process steps or production requirements from the process matrix into process tracking and control procedures.
6. Refine the project plan to incorporate aspects of the design, process, and

control matrices.

The matrices highlight the information needed to make decisions about product definition, design, production, and delivery, and they link work requirements in the four phases so that customer needs and technical requirements, as defined early in the process, are translated undistorted into design features and production requirements. Shown in Figure 4.19, the link happens by taking the requirements or steps from the top of one matrix and putting them on the left side of the matrix in the next phase. This linking of matrices ensures traceability (that word again)—that any project activity can be traced to the customer need or requirement that it fulfills, and, conversely, that every customer need and requirement can be traced to the necessary project activities. Put another way, QFD ensures that every activity serves a requirement, and every requirement is served by at least one activity. The result is a plan where every task throughout the project is integrated with the technical requirements listed in the original house matrix. The next chapter describes further aspects of an integrated project plan.

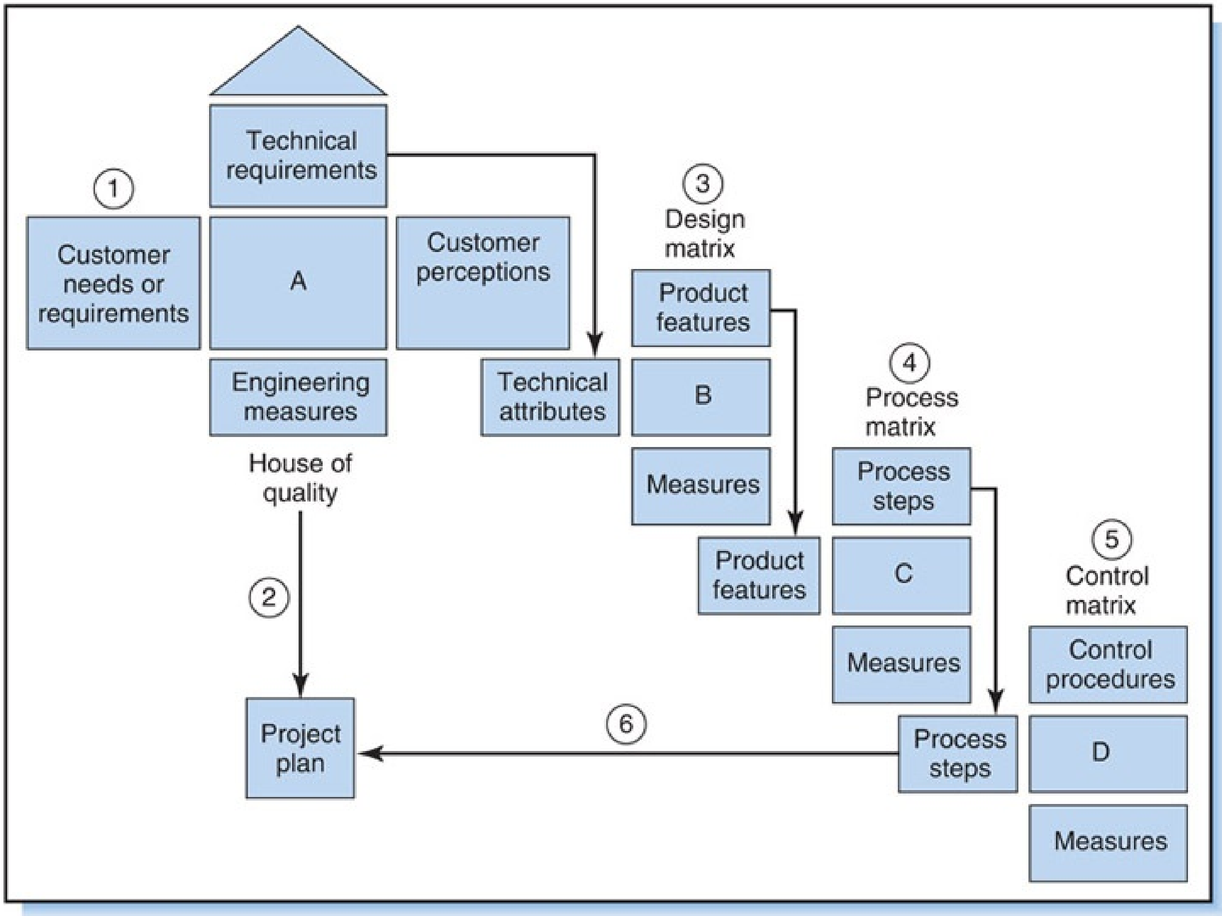


Figure 4.19 QFD multiphase, multi-matrix approach.

The additional time required by the QFD process to produce a project plan and an initial product design is offset by the reduced time to produce the *final* design because less redesign and fewer engineering changes are needed after the product goes into production.

Example A4.11: Chrysler Development of the LH Car Line[18]

Chrysler first applied QFD in the design and development of its LH-platform cars (Chrysler Concorde and Dodge Intrepid). Early in the product concept stage a program team was formed to establish overall design guidelines. The program team allocated responsibility for the different major automobile systems to different design groups (as did Boeing in its teams), and each group set up a QFD team to determine system-level requirements. Once requirements were set, smaller groups were formed to focus on designing the components within the system.

The QFD methodology was part of a broader concurrent engineering effort that yielded impressive results: The total LH design cycle took 36 months versus the historical 54 to 62 months; prototype cars were ready 95 weeks before production launch versus the traditional 60 weeks; and the program required 740 people compared to the usual 1,600 people. The cars received numerous awards and magazine citations for design excellence.

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. Describe briefly 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 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 4.8. 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”?

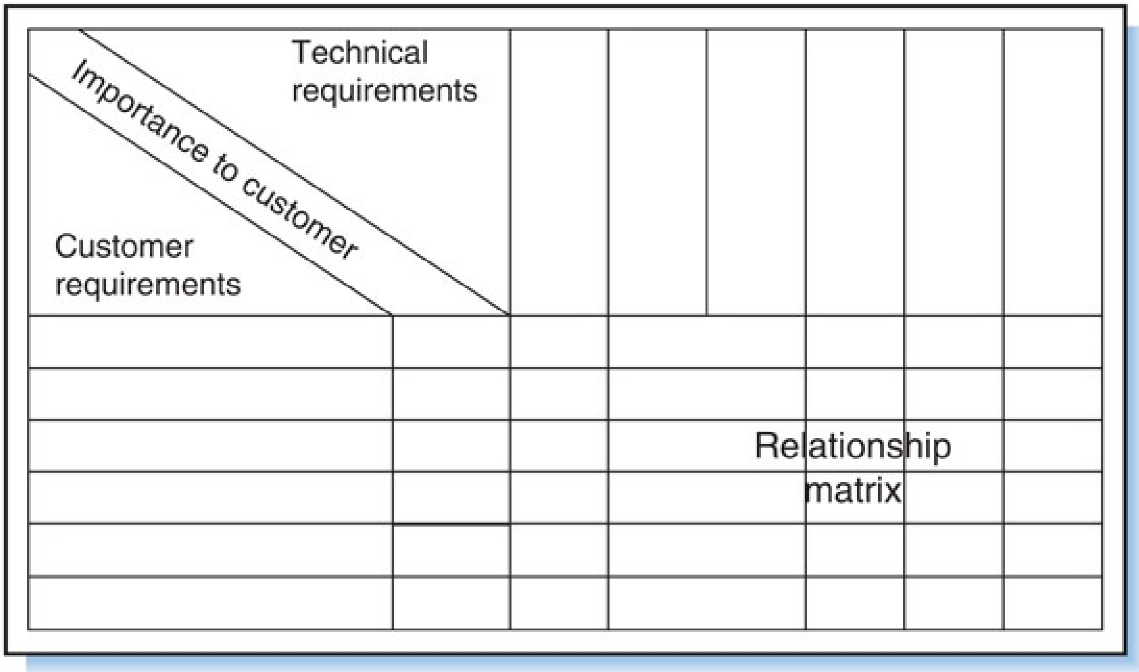


Figure 4.20 QFD matrix for Question 19.

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. Was QFD or a similar process used to define requirements and/or create the project plan?
11. 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) is the prime contractor for a large skyscraper project in downtown Manhattan. SBC is working directly from drawings received from the architect, Santaro Associates (SA). Robert Santaro, owner and chief architect of SA, viewed this building as similar to others he had designed. However, one difference between this building and the others he overlooked was the building's facing, which was to consist of large granite slabs—slabs much larger than anything with which either he or SBC had prior experience.

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 are 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 granite slabs too big to install with such precision. As a result, the spacing between slabs is often too small, making it difficult or impossible to install window units. Most of the 2,000 window units for the building have already been manufactured so it is too late to change their specifications, and most of the granite slabs have been hung on the building. The only recourse for fitting window units into tight spaces is to grind away the granite. It is going to be very expensive and will certainly 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 wants 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, as a way 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. Revcon decided to outsource the development and design of the new valve. It prepared an RFP with the following 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 work 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 for. 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 prototype to Revcon. Because the prototype description in the proposal was vague, Revcon expected the prototype to be an almost-finished product, whereas Welbar understood it to be a simple working model 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, it went ahead and prepared production-ready models. The finished prototype later demonstrated that the production models could not be produced.

Eventually Welbar did design a truly innovative valve; however, the design would require 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. Because of that the product could not be priced low enough 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 out when a few 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 Sihgh, feels that the kind of system best suited to the user's needs is more or less obvious, and to address the needs she creates some bullet points and flow-charts. 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 to 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 of the additional people added to staff the project, 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, 2006: 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, 2006: Peter meets with Bruce, a mining engineer with 20 years of experience in Australian gold mines, and Sam, a geologist who a few years back 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 a figure of 150,000 ounces at $400 per ounce, 3 years from now, would lead to large losses that could ruin the company.

Current information about the ore body is inadequate, however, and it will be necessary to drill exploration holes to learn more about the general geology of the area.

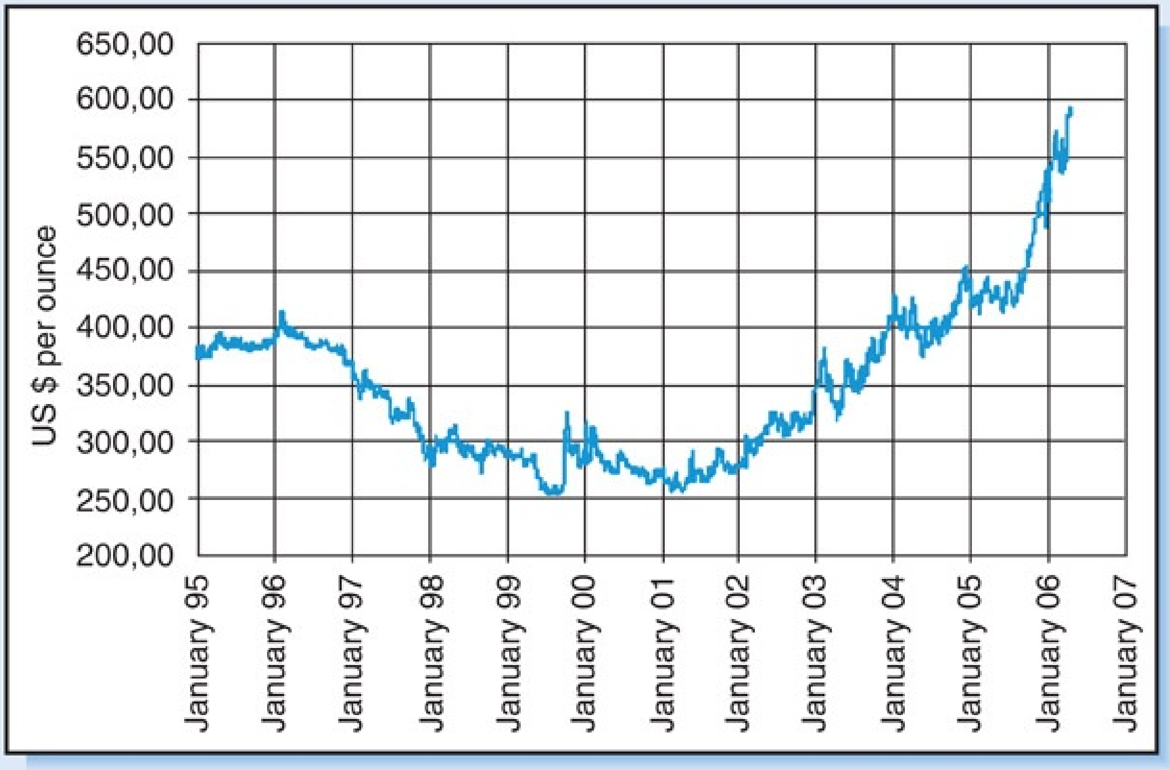


Figure 4.21 Gold price.

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 US $150 millionto $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 between $1.2 million and $1.6 million. Rock samples from these holes will be analyzed in a laboratory to determine the gold content.”

Peter instructs Sam to review the data from his previous exploration work and to prepare a report of his recommendations concerning the future exploration. He is authorized to spend no more than $25,000 on this “paper exercise.” 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 as of that stage—would full-fledged shaft development be authorized.

Questions

1. List the phases of the project and indicate the minimum and maximum cost of each phase as foreseen in August 2006.
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 risk of the project.
4. Comment on the problem that, once money has been allocated to the process, people might become "hooked" into the project and be 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 at this time?

Refer to Case 2.4, Santa Clara County Traffic Operations System and Signal Coordination Project

See Chapter 2

Question

The INCOSE group determined that a requirements traceability matrix, which was not used in the project, could 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.

Endnotes

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. *APMP Syllabus,* 3.1 edn. Buckinghamshire, UK: Association for Project Management; 2012.

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: Yourdan 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.I. 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) Chestnut H. *Systems Engineering Methods.* New York, NY: John Wiley & Sons; 1967, pp. 1–41; (4) 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, www.scaled.com/projects/tierone/index.htm.

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.

17. See Bicknell B. and Bicknell K. *The Road Map to Repeatable Success: Using QFD to Implement Change.* Boca Raton, FL: CRC Press; 1995, pp. 97–110.

18. Lockamy A. and Khurana A. Quality function deployment: a case study. *Production and Inventory Management Journal* 36(2); 1995: 56–59.

Part III
Systems and Procedures for Planning and Control

Project management extends beyond defining project objectives and requirements; it involves forming a project organization, identifying the necessary tasks and the resources to do them, and providing leadership to get the tasks done. Overall project objectives and system requirements need to be articulated into detailed plans, schedules, and budgets to accomplish the objectives and requirements. Methods are then needed to make sure 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 specifying, scheduling, and budgeting project activities, assessing risks, monitoring and controlling work, and organizing and keeping records to achieve project quality, time, and cost requirements.

Procedures should be conducted within a framework to ensure that everything to be done is accounted for, properly organized, and executed. These frameworks and the structures, activities, and systems that comprise them—work breakdown structures, cost accounting systems, information systems, and many others—are described in this section of the book.

Chapter 5
Basic Project Planning Techniques

Big fleas hath smaller fleas upon their backs to bite'em.
And these fleas have smaller fleas
And so ad infinitum.

—Jonathan Swift

Every project is somewhat unique because it is aimed toward an end-item or result that is in some way unique. Because of its uniqueness, basic questions about the project must be addressed and answered before work can begin. Satisfactorily answering these questions so that the project will achieve its goals is the function of *project planning*.

Project management can be broadly divided into two parts: (1) in the conception and definition phases, *preparing a plan* that specifies the project requirements, work tasks, responsibilities, schedules, and budgets; (2) during the execution phase, *executing the work in the plan* and *tracking progress* versus the plan. This chapter gives an overview of the first part and covers the topics of scope and work definition, elemental scheduling, and procurement management.

5.1 Planning Steps

After a business need, contract request, or RFP has been received, top management releases funds to prepare an initial plan, schedule, and cost estimate for the project proposal. Approval of the project and signing the contract authorize the project to begin, starting with the definition of system requirements and preparation of a project execution plan. For internal projects, a project charter is sent to stakeholders to announce and describe the project.

The project manager, if not already assigned or involved, is now identified to oversee the planning process and produce a plan that elaborates on any earlier plans as prepared for the proposal, business case study, or charter.

Because each project is unique, there is never an a priori, established way about how the project should be done. Each project poses new questions regarding *what, how*, by *whom*, in *what order*, for *how much*, and by *when*, and the purpose of planning is to answer them. The planning process answers the questions in the following steps:

1. **What is the desired end result?**
   Define the project *objectives, scope,* and *system requirements*. These specify the project deliverables, end-items, and other sought results, as well as 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 activities include everything necessary to create and deliver the end-item or deliverables, including planning, control, and administration activities.
3. **Who will do it?**
   Specify the *project organization*—the individuals or departments, subcontractors, and managers that 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 a method for tracking and controlling project work, which is necessary to keep the project conforming to the schedule, budget, and user and system requirements.

This chapter and the next seven chapters discuss these steps in detail.

5.2 The Project Execution Plan

Project planning begins early in the project life cycle—in most cases with preparation of the proposal. While preparing the proposal a rudimentary project team is organized, and the team prepares a brief summary plan for inclusion in the proposal. They prepare this plan using the same, albeit more abbreviated, procedures as used to develop more elaborate and detailed project execution plans. 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.[1] The planning effort in preparing the proposal is directed at estimating the project duration, cost, and needed resources. The proposal summary plan includes just enough information about the project and the price to enable the customer to make a decision.

In contrast, the project execution plan lays out the specifics of the project that will serve as a roadmap to *guide* the project team throughout the project execution. As mentioned in Chapter 4, usually the plan contains details only for the immediate upcoming phase of the project, about which the most is known.

See Chapter 4

Details for later project phases are filled in later as more information becomes available.

Contents of Execution Plans

Contents of execution plans vary depending on the size, complexity, and nature of the project. Figure 5.1 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 outlined here;[3] in small or low-cost projects it is possible to bypass some of the items, being careful not to overlook the crucial ones. It is good practice to carefully review every item in the template even if only to verify that some are “N/A” (not applicable). An example project execution plan for the LOGON project is at the end of the book in Appendix C.

See Appendix C

You might notice similarities between the sections of the plan and the contents of the proposal described in Figure 3.6. Although the format is different, there are similarities. Sometimes the proposal, after revision to reflect updates, agreements, and contract specifications, becomes the project execution plan. More often, however, the proposal serves as an outline for the execution plan, and the execution plan is more expansive than the proposal. Because the primary audience of the execution plan is the project team and not the customer, the sections on work definition, schedule, and budget in the plan are much more detailed than in the proposal.

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 uncertainity, 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 5.1 Template for Project Execution Plan.

As illustrated in the following example, sometimes development of the project execution plan is an evolutionary, cross-functional process.

Example 5.1: 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 from vendors and then combines them to meet customer requirements. The company was awarded a large contract for a system to place, store, retrieve, and route shipping containers by the MPD Company. 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 CreativeRobotics, will provide the computer and robotics systems plus assistance with their installation and checkout. Frank Wesley is the IBC project manager put in charge of preparing the proposal.

Most of the project execution plan for LOGON originated in IBC's project proposal. In preparing the proposal, engineers from IBC, CRC, and CreativeRobotics worked together to design a system that covered all of MPD's requirements. The design included schematics, operational specifications, and a bill of materials. The design managers at CRC and CreativeRobotics estimated the labor expertise needed and the costs for parts and labor. Frank Wesley and his engineers prepared a work breakdown structure and estimates for IBC's time and costs. He then combined these with CRC's and CreativeRobotics' estimates to arrive at an overall 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, project plan, costs, and schedules, and prepare a detailed execution plan. This plan contained similar information to the proposal, but was updated and expanded to include schedules for procured materials and parts, plans for labor distribution across work tasks, a task responsibility matrix, a detailed WBS and associated budget, and a master schedule.

In the LOGON project the execution plan evolved in stages: it was initially created during proposal preparation but was then 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 the definition phase). Often, project planning is itself a significant effort that requires substantial time and labor.

Learning from Past Projects

Oftentimes organizations approach each project as being *too* unique and ignore the lessons of history—the mistakes, solutions, and lessons learned of the past.[4] No project is ever 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 is provided with planning assistance in the form of lessons learned, best practices, suggested methodologies and templates, and even consultation based upon experience from past projects. In some cases this assistance comes from the project management office (PMO), described in Chapter 17. Lessons learned and best practices are compiled from the *post-project summary* or *project postmortems* of past projects; these are formal retrospective reports created at project termination that describe what went well, what went wrong, and any lessons derived from the project experience (described in Chapter 12). They provide useful guidance in planning future projects and help managers avoid reinventing the wheel and repeating mistakes.

See Chapter 17

See Chapter 12

5.3 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 or extent of work it encompasses, the concept of *project scope.* Determining the project scope happens during project conception, first when the project is initiated and during preparation of the RFP and the proposal, and again during project definition. In each case, user needs and requirements are compared to time, cost, resource, and technology constraints to determine what the project should and can encompass. The process of setting the project scope is called scope definition.

Scope Definition

Scope definition involves specifying the breadth of the project and the full span of its outputs, end-results, or deliverables. The defined end-items to be produced or delivered by the project are termed "inclusions," meaning they are *included* in the project. To ensure clarity, the items, conditions, or results *not* to be included in the project, i.e., "exclusions," are also defined; for example, a project to construct a building might exclude the building's landscaping and interior decorating. Distinguishing between inclusions (contractor responsibilities) and exclusions (possible customer responsibilities) prevents misunderstanding and false expectations.

Scope definition focuses primarily on determining outputs and deliverables, not on time and cost. Of course, time and cost delimit or dictate the potential deliverables; as such, in the scope definition they must be accounted for as "constraints."

The outcome of scope definition is a *scope statement* that describes the main deliverables of the project, criteria for acceptance of the deliverables, assumptions and constraints (to provide rationale as to 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 for scope definition includes a set of user needs and requirements, a business case or other expression of needs, and constraints and assumptions; ideally the principal subsystems and components of the end-item will have been identified and also serve as inputs. Everything to be included in the project or contract, including support and side-items, as well as work or deliverables *not to be included* in the project (exclusions) are stated. The scope statement sometimes also 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 refers to or incorporates other documents that contain them.

For a unique project, the preliminary scope statement defined during project initiation might be somewhat vague; it should however be expanded and clarified during project definition as detailed plans for the first phase of the project are developed. For programs, separate scope statements are developed for the overall program and the individual projects that comprise the program.

Once the scope statement has been approved, it becomes a controlled document that can be modified only through a formal change process (Chapters 9 and 11).

See Chapter 9 and Chapter 11

Example 5.2: Scope Statement for the LOGON Project

The RFP for the LOGON project (see Appendix A at the 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, check-out, 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 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: a 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. Storage-bucket system is Model IBS05 adapted to requirements C.1 through E.14.
    c. Robotic 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 (a) through (g) above 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, the detailed designs noted in points 3(b) and 3(c) include reference to requirements C.1 through E.14 and F.1 through G.13. The requirements must be sufficiently comprehensive to enable subcontractors to produce the specified systems and components. Elsewhere the scope statement lists any project exclusions as noted in the RFP or identified by IBC.

The scope statement is the reference document for all project stakeholders; it becomes the basis for making decisions about resources needed for the project, and, later, determining whether or not required or requested changes to work tasks and deliverables fall within the agreed-upon project scope. A common tendency in projects is *scope creep*, which means the project keeps growing 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

The statement of work is a description of the project; it includes a scope statement, but often goes far beyond that. It describes, for example: deliverable specifications and requirements; deliverable schedules; management procedures for communication, planning, and handling risks and changes; project budget; and key personnel responsible for administrative and work tasks. As such, 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 the previous document. The project charter described in Chapter 4 might also contain a SOW.

5.4 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 overlook or duplicate activities. To insure that every necessary activity is identified and clearly defined, and that no activities are missed, a procedure called the "work breakdown structure" is used.

Work Breakdown Structure

Complex projects consist of numerous smaller sub-projects, interrelated tasks, and work elements. As the rhyme at the beginning of the chapter alludes, the main end result or deliverable of a project can be thought of as a system that consists of subsystems, which themselves consist of components, and so on. The method for subdividing the overall project into smaller elements is called the *work breakdown structure* or *WBS*, and its purpose is to divide the total project into "pieces of work" called *work packages*. Dividing the project into work packages helps in preparing schedules and budgets and assigning management and task responsibilities.

Creating a WBS begins with dividing the total project into major categories. These categories then are 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 of the breakdown gets 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 the following four levels:

Level	Element Description
1	Project
2	Sub-project

3	Work Package
4	Activity

Level 1 is the total project. Level 2 is the project broken down into several (usually four to ten) major elements or sub-projects. 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 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,* a fifth level is added at the top and the levels are renumbered: Level 1 is the program, Level 2 the project, and so on.

When the process is completed, tasks at the bottom levels, whatever the levels might be, are called *work packages.* In the table above, the term "work package" appears at Level 3, but that is for illustration only. Later in the planning process, larger work packages might be subdivided into more-detailed activities or tasks,

The actual number of levels in the WBS varies by project, as do the actual names of the element descriptions at each level. (The level and names are often prescribed by the project methodology in use.) Figure 5.2 shows a typical WBS. Note the different levels and descriptions for each work element.

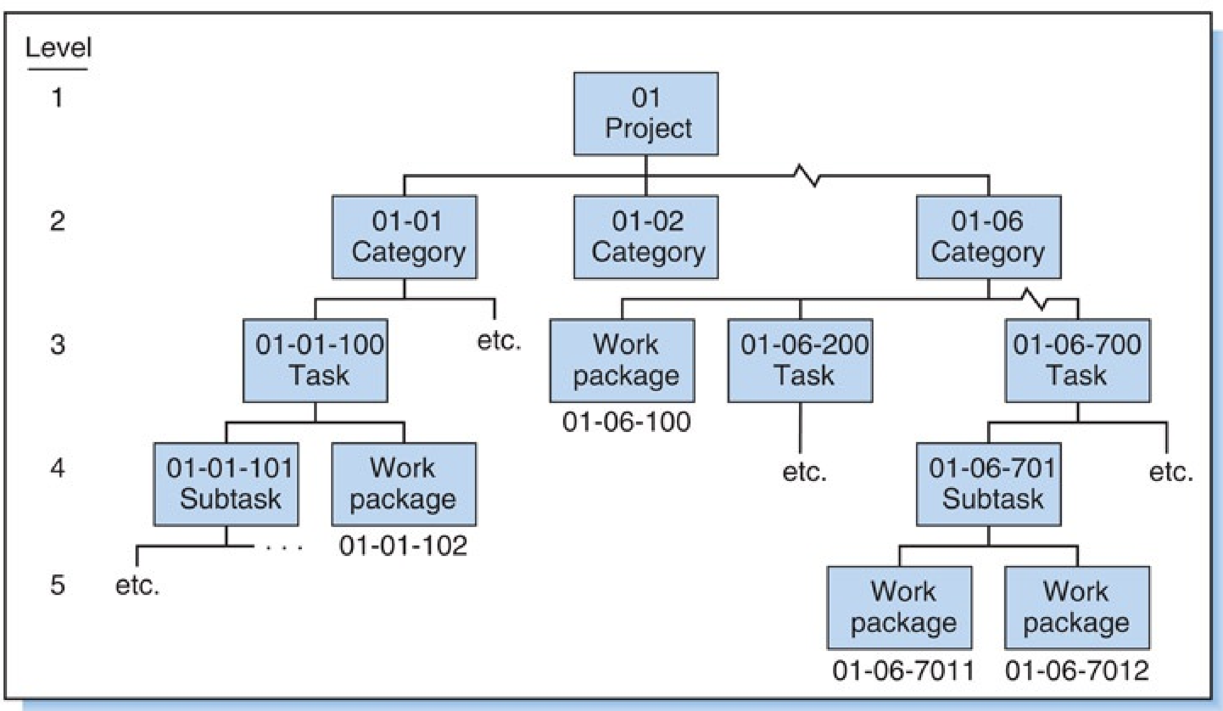


Figure 5.2 Elements of a WBS.

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 suggest 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 of the configuration items (CIs) and major components of the system, as well as the work tasks to design, develop, build, and test them.[5]

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 subcontractor through a contract agreement between the subcontractor and the project manager. For internal projects, responsibility for each sub-project or activity is assigned to an in-house department, through a formal agreement between the department manager and the project manager.

To avoid unnecessary complexity, the number of levels in the WBS should be limited. A five-level WBS might be appropriate for large projects, but for most small projects a three-level WBS is adequate. To help organize and track project activities, each work element is coded with a unique identifier or number. Usually the number at each level is based on the number at the next higher level. In Figure 5.2 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 numbering scheme.

Figure 5.3 illustrates ways to create the WBS for constructing a house. The top part of the figure shows the main project end-item (Level 1) and the major categories of work (Level 2) necessary to build it. For the most part, the items at Level 2 are physical pieces or components of the house. In other words, they identify the deliverables or products to be produced. This is called a *product-oriented WBS*. 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 these. That is, creating a WBS in this way assists in preparing other parts of the plan, including the project budget and schedule. The bottom part of Figure 5.3 shows how the product-oriented WBS would be subdivided into four levels.

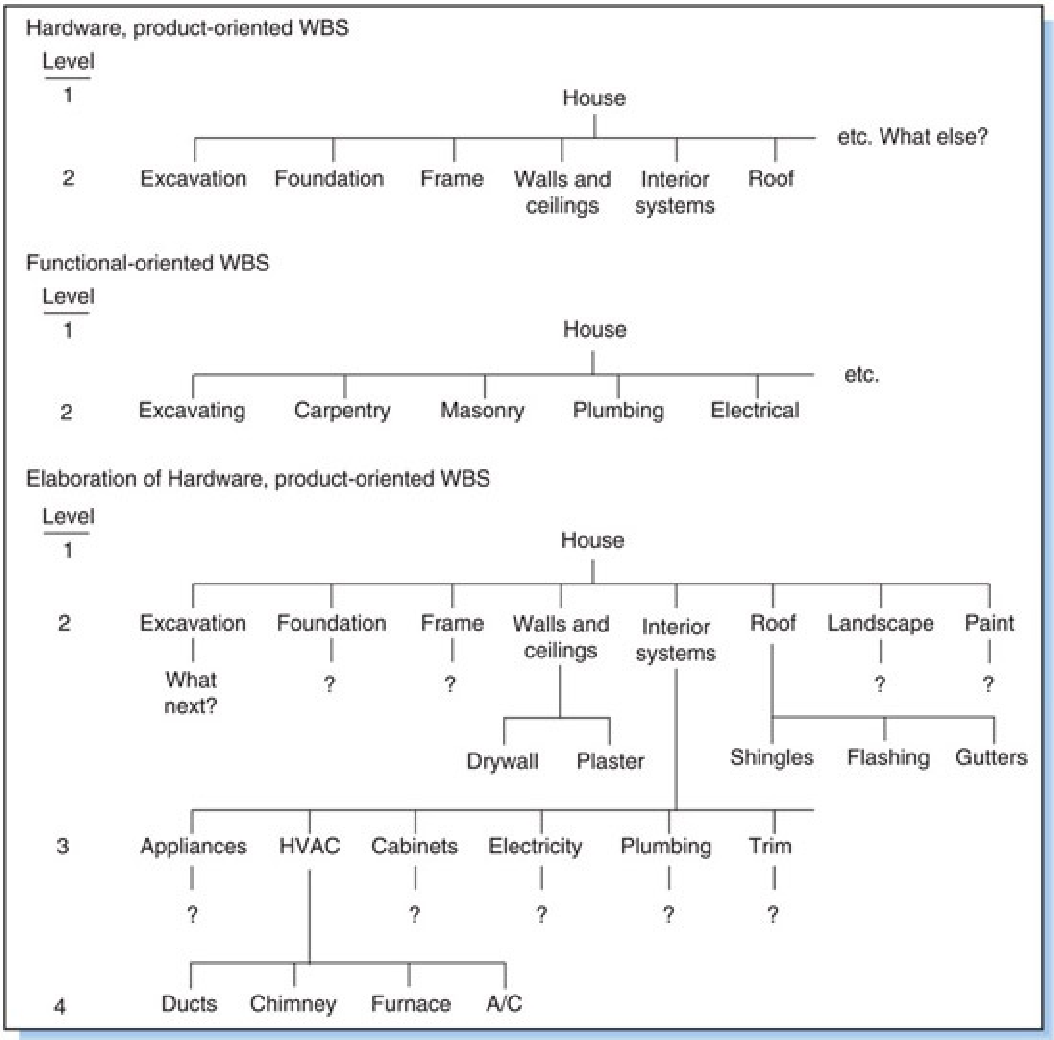


Figure 5.3 Example of a WBS for building a house.

Sometimes the WBS or portions of it are *function-oriented* or *task-oriented* (rather than product-oriented). For example, the middle part of Figure 5.3 shows the project subdivided according to work functions (e.g. carpentry and plumbing), not deliverables. Functions or tasks such as management and overhead, design, engineering, training, and inspection that apply to multiple deliverables or to *integrating* multiple deliverables should be identified as separate work packages. Whether the WBS uses product-, functional-, geographical-, task-based, or other breakdown is a matter of preference, or is stipulated by the organization's project methodology or WBS templates.

During the WBS process the questions "What else is needed?" and "What's next?" are constantly being asked. Supplementary or missed elements are identified and added to the WBS at appropriate levels. For example, the WBS in Figure 5.3 does not include blueprints, budgets, and work schedules, even though the house cannot be built without them. These are deliverables associated with planning the project and designing the house, which could be included in the WBS by expanding Level 2 and inserting categories for "Design" and "Project Management," and then at Level 3 by inserting "blueprints" under Design and "budget and work schedules" under Project Management, respectively. Somewhere in the WBS, considerations such as site location, permits and licenses, environmental impacts, etc., must also be included. As described later, the WBS must also reflect any procured (contracted, outsourced) goods, materials, or services.

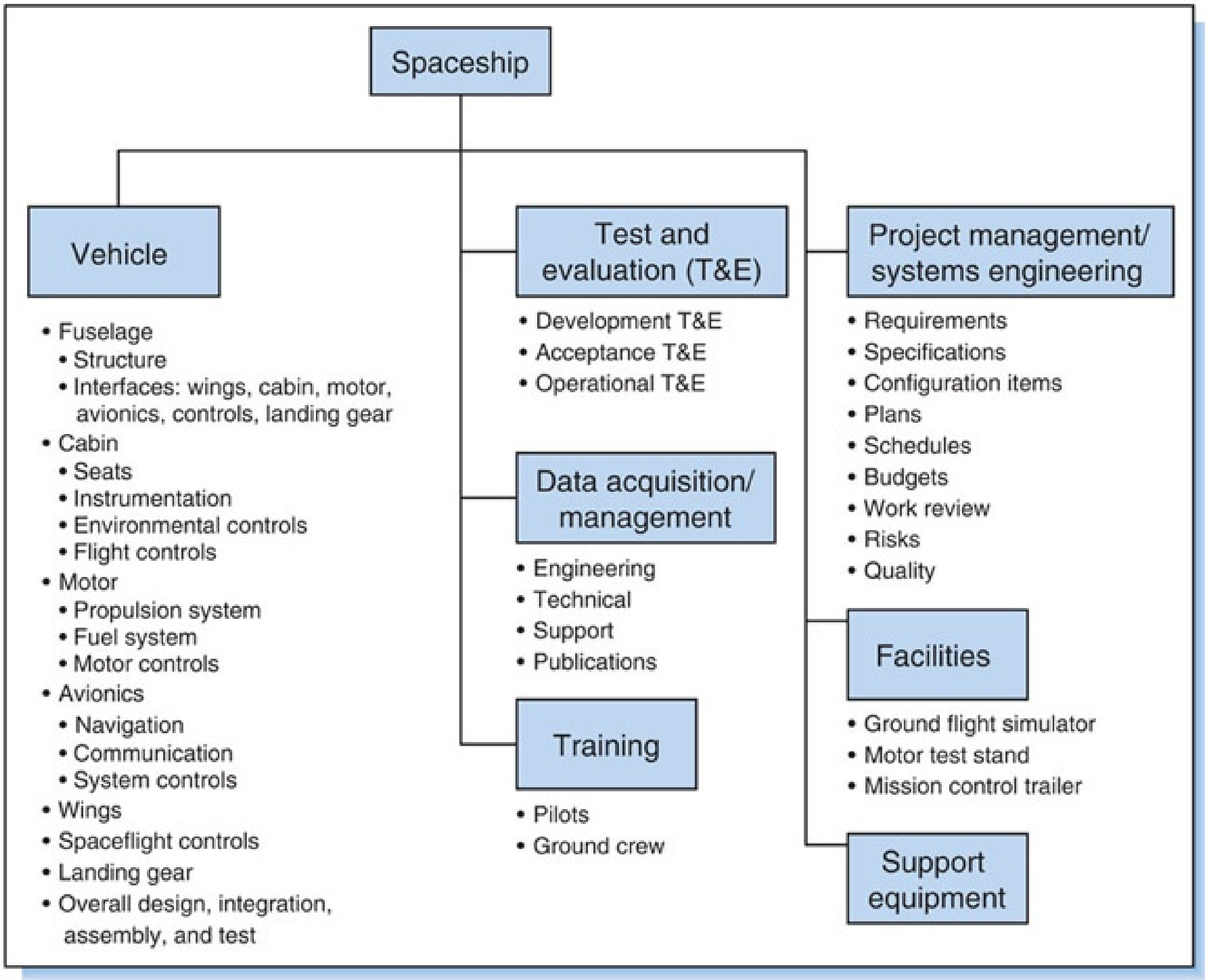


Figure 5.4 WBS for spaceship project.

The concept of traceability, mentioned in Chapter 4, also applies to the WBS. A test for 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 can't be, the question is, why is it there?

See Chapter 4

Figure 5.4 exemplifies the WBS for a large engineering project where the main deliverable and many of its subsystems and components must be developed, built, integrated, and tested from scratch. Notice, some portions of it are product-oriented (vehicle, facilities), while others are function-oriented (test/evaluation, project management/systems engineering).

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.

Example 5.3: Process of Developing the WBS for the LOGON Project

The project manager and 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 breakout the work packages into finer detail. In the first meeting they "rough out" the major categories of work and deliverables in the Level 2 breakdown from the SOW (described in Example 5.2) and requirements, and identify the responsible functional areas.

After contract signing the project manager meets with managers from the functional areas that will be contributing to deliverables in the Level 2 breakdown. These managers then meet with their supervisors and technical staff to prepare a Level 3 breakdown. Where necessary, supervisors prepare a Level 4 breakdown.

Figure 5.5 shows a WBS that 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 bottom of the branches are "work packages," denoted by letters in parentheses. Notice that the work packages are at different levels of the WBS; this is because each branch of the WBS is developed separately.

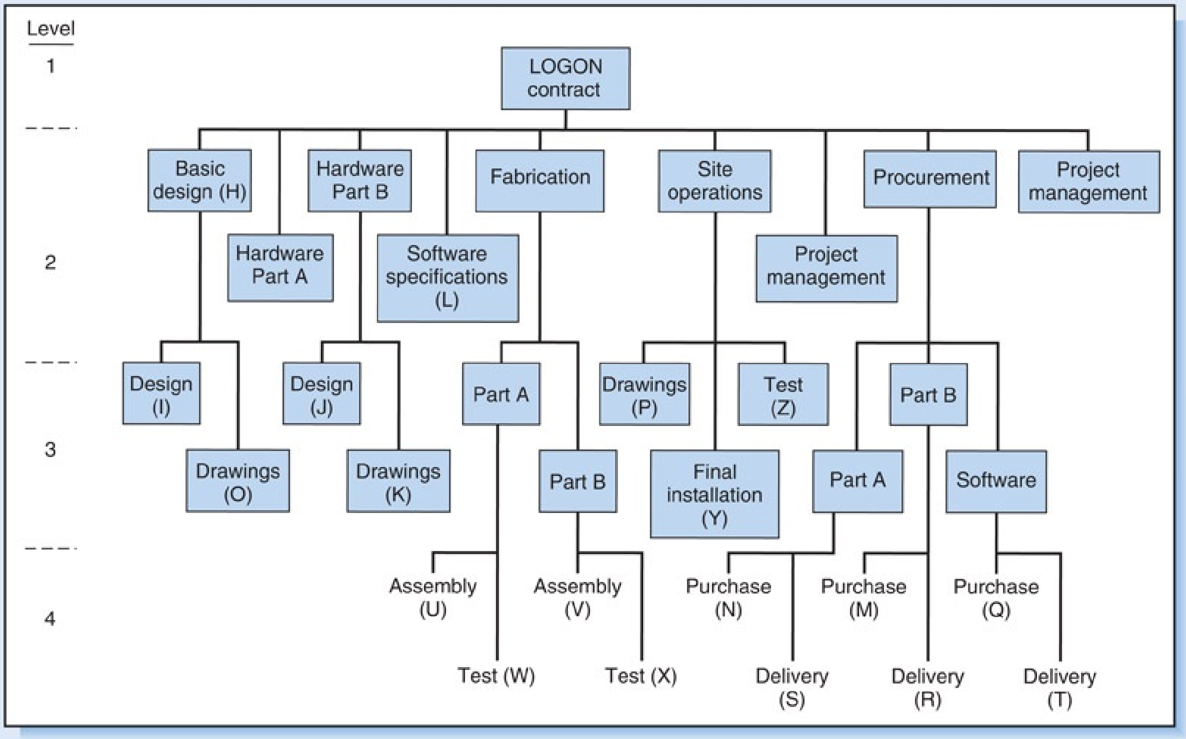


Figure 5.5 Work breakdown structure for the LOGON project. Work packages are lettered H through Z.

Work Packages

How far down does the breakdown go? Simply, for as far as needed to completely define all work necessary for the project. The work in each “box” or element of the WBS should be “well defined”; if it is not then the box must be subdivided into smaller boxes. For a box to be “well defined” it should include the following:

1. *Comprehensive SOW*: work task or activity to be done.
2. *Resource requirements*: labor, equipment, facilities, and materials needed for the task.
3. *Time*: estimated time to perform the task.
4. *Costs*: estimated resource, management, and related expenses for the task.
5. *Responsibility*: parties, individuals, or job titles responsible for doing

and/or approving the task.
6. *Outcomes*: requirements, specifications, and associated deliverables, end-items, or results for the task.
7. *Inputs*: preconditions or predecessors necessary to begin the task.
8. *Quality Assurance*: entry, process, and exit conditions to which the task must conform; as specified in the quality plan.
9. *Risk*. uncertainties about time, cost, and resources associated with the task.
10. *Other*: additional information as necessary.

These properties are summarized in Figure 5.6. If any of them cannot be defined for a given box, then the task or product in the box is too broad and must be broken down further. When all or most of the properties can be defined for a box or element, the element is considered “well-defined” and, by definition, a *work package*.

Inputs	Task	Outcomes
Predecessors	Statement of work	Deliverables
Preconditions	Time	Results
Resources	Cost	
Requirements/specifications	Responsibility	
	Quality assurance	
	Risk	

Figure 5.6 Properties of a work package.

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.

WBS Templates

A company that routinely performs similar kinds of projects might utilize a standardized WBS “template” at Level 2 or Level 3. The template is based upon experience from having done many of those kinds of projects. In some companies the template is created and maintained by the project management office (PMO). Even with a template, however, it is good to remember that every project is somewhat unique and that such uniqueness might not become apparent until Level 3 or 4. Hence, the 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 might 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 hiding. To reduce oversights, it is good practice to have two or more teams each create a WBS, and then to combine them into one.

Ideally work packages represent jobs of about the same magnitude 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. These are simply guidelines. Work package cost and duration can depend on many factors such as project size (smaller projects have smaller work packages).

Each work package represents a contract or agreement with a subcontractor, supplier, or internal functional unit. Although several functional or subcontracting units might share responsibility for a work package, ideally a work package has only one party responsible for it.

Example 5.4: Work Package Definition for LOGON Project

The LOGON project was divided into 19 work packages—the boxes lettered H through Z in Figure 5.5. Below is an example of the properties for a typical work package, Work Package X: Test of Hardware. Note how the defined properties correspond to those listed in Figure 5.6.

1. **Statement of work**: perform checkout, operational test, and corrections as necessary for sign-off approval of four Batman

transporter units, Model IBR04.

2. **Resource requirements**: labor (FT commitment, 3 weeks): test manager, two test engineers, three technicians.

   Procured materials: track for mockup; all other materials on hand.

   Facility: test room number 2 at Iron Butterfly for 3 weeks.
3. **Time**: 3 weeks scheduled; (time critical) start December 2; finish December 23.
4. **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

5. **Responsibility**: 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.
6. **Deliverables**: four tested and approved Batman robotic transporters, Model IBR04. Refer to specifications.
7. **Inputs**: predecessor: assembly of Batman robotic transporter (work package V). Preconditions: test room setup for robotic transporter.
8. **Quality Assurance**: refer to entry, process, and exit conditions for work package X in the LOGON quality plan.
9. **Risk**: RBU will fail test requirements because of assembly/integration problems/errors. Likelihood: low. Contingency reserve: additional week included in the schedule.
10. **Specifications**: refer to test document 2307 and LOGON contract

spec sheets 28 and 41.
11. **Work orders**: none, pending.
12. **Subcontracts/purchase orders**: no subcontracts; P.O. 8967–8987 for track tests.

A work package that produces a tangible deliverable or physical product as in the example should include specific start and finish dates for the work package.

WBS Process and the Integrated Project Plan

In an integrated project plan the elements of the plan—requirements, work tasks, schedules, budgets, risk, quality, communications, procurement, and so on—are interconnected. Once created, such a plan provides managers with a variety of ways to track the project, and to assess the impact of actions or problems with some elements of the plan on the other elements.

To better describe what an integrated plan is, we can compare it to what it is *not*, which would be: a list of work packages or tasks generated without much regard for user requirements; a budget that does not account for the resources required of the project tasks; and a schedule where the tasks do not match up with the tasks on the WBS or budget. To the outsider, it would appear that four people who never talked to each other had each come up with, respectively, a list of work tasks, a list of needed resources, a schedule, and a budget. Amazingly, that is sometimes the way plans are created, with the result that requirements, 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 list of work tasks developed in the WBS appears in schedules, budgets, and most other elements of the plan.

The process of creating the WBS and the resulting list of work packages thus integrates various elements of the project plan and project control in several ways:[6]

1. Managers, subcontractors, and others 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 that work.
2. Work packages in each phase are logically related to those in earlier and later phases; this ensures that predecessor requirements 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, suggesting which work packages are in need of corrective action.

The 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.

5.5 Project Organization and Responsibilities

Integrating WBS and Project Organization

During the WBS process each work package is associated with the area of 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 5.7: on the left is the company organization structure; across the top is the project WBS. For contracted project work (performed by external parties), the organization structure on the left would show contractors and suppliers instead of departments. The box at the intersection of each department and work package signifies a *control account* or *cost account.* Each account represents assignment of responsibility for a particular task or work package or portion of one to a department. Just like a work package, it includes a schedule and budget, resource needs, deliverables,

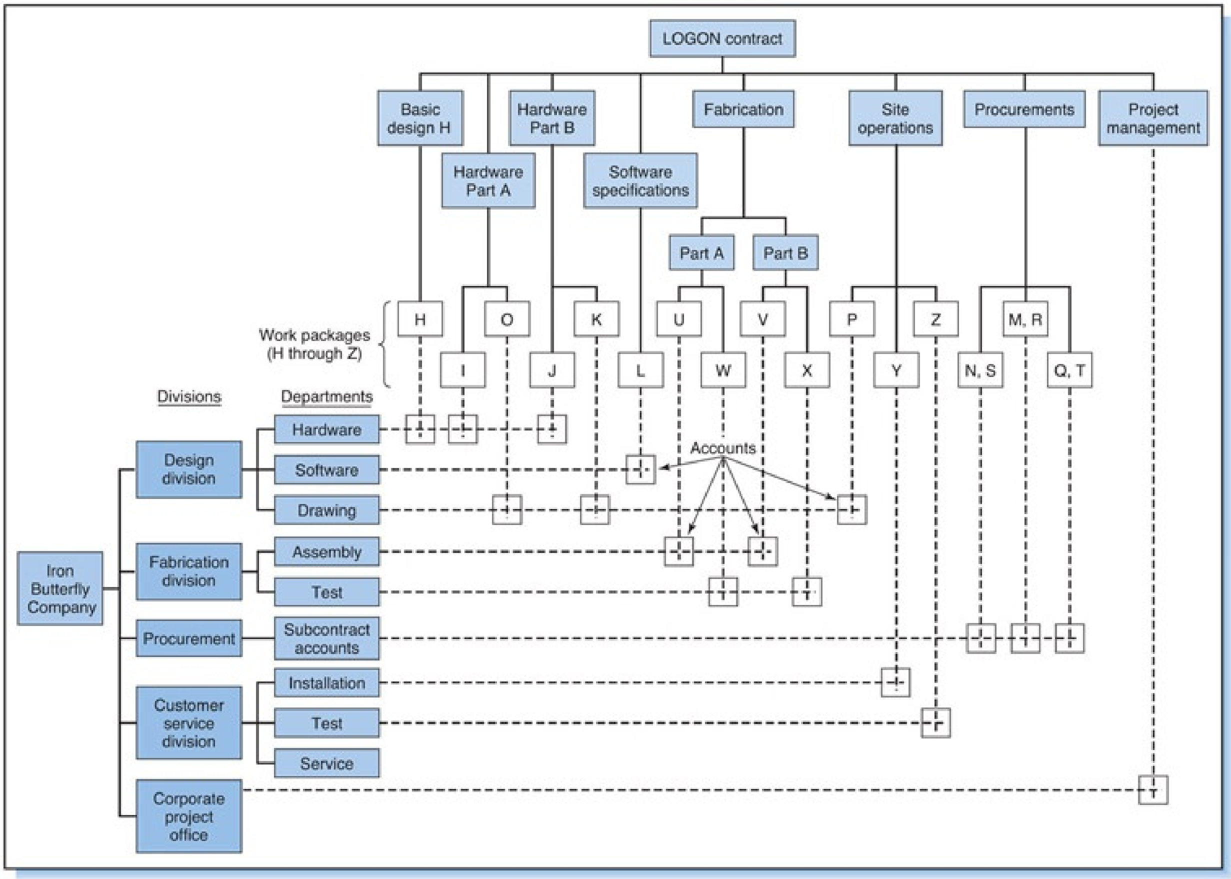


Figure 5.7 Integration of WBS and project organization.

requirements, and the manager or supervisor responsible for it. Each control account integrates the WBS with the project organization and represents an agreement or contract with departments or subcontractors to fulfill work package requirements. Control accounts are described further in Chapter 8.

See Chapter 8

Responsibility Matrix

The individuals responsible for work packages are shown in a chart called the *responsibility matrix* or assignment matrix. Figure 5.8 shows the responsibility matrix for the LOGON project. The rows represent the work packages or major project tasks and activities identified in the WBS. The columns represent the persons, groups, departments, or contractors responsible for them, and can also include other stake-holders who need to be notified about project matters. Letters within the matrix symbolize the *kind* of responsibility: primary (ultimate accountability for the work package); secondary (assistance or help); notification (must be notified about the work package's status); and approval (has authority to approve or reject work package deliverables). (For this reason the matrix is also called a RACI chart—Responsible, Accountable, Consulted, and Informed.) Note that for each task one and only one person is assigned primary responsibility. The matrix can also be used to signify who will *do* the work and any other conceivable kind of responsibility.

From the matrix, everyone associated with the project can easily see who is responsible for what. This helps avoid people shirking responsibility and "passing the buck."

To ensure everyone knows what is expected of them and what they can expect from others, the people, groups, or companies identified in the matrix should review and consent to the responsibilities. The assignments in the matrix can be roughed in during project conception and then detailed and firmed up during a team-building session held after project kickoff. Team building is described in Chapter 16.

See Chapter 16

5.6 Scheduling

The next logical step after requirements definition and work definition 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 show not only how to get to where you want to go, but also progress you have made along the way. Work packages are what you must do; combined with the schedule they are the road to project goals. Along that road are signposts called *events* and *milestones* that show how far you have progressed. Passing the last event signifies having reached the final destination: project completion.

Events and milestones should not be confused with work packages, activities, or other kinds of tasks. A task or work package is the actual work planned or being done and represents the *process* of doing something (such as *driving* a car to get somewhere); it consumes resources and time. In contrast, an *event* signifies a *moment in time*, the instant when something happens. In a project, events represent the *start* or *finish* of something (equivalent to beginning a trip or arriving at an intermediate destination). In most project schedules, each task is depicted as a line segment; the two ends of the line segment represent the events of starting and completing the task. For example, in Figure 5.9 the line segment labeled "Task A" represents the time to do Task A; events 1 and 2 represent the moments when Task A is started and finished, respectively. In project schedules, each event is attached to a specific calendar date (day, month, and year).

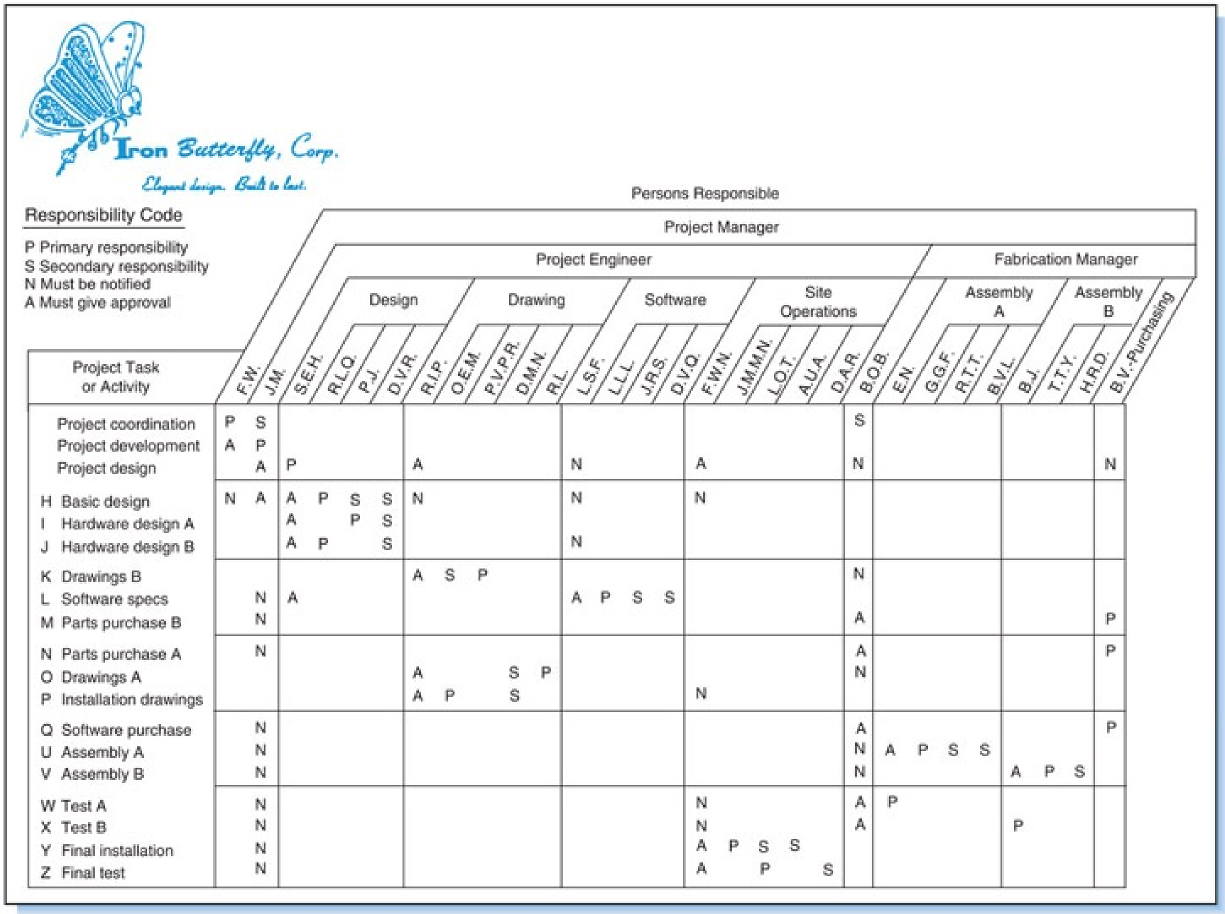


Figure 5.8 Sample responsibility matrix for LOGON project (with initials of persons responsible).

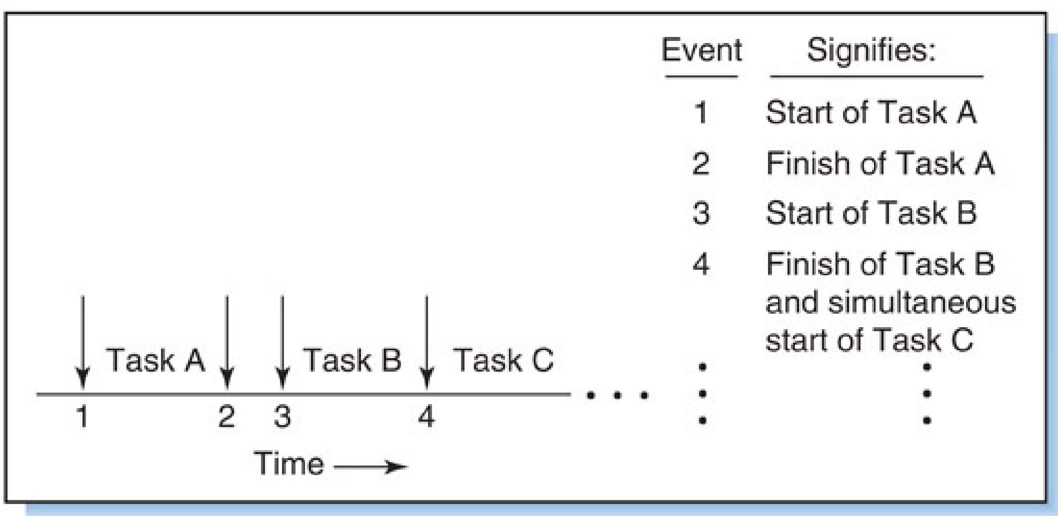


Figure 5.9 Relationship between tasks and events.

Two kinds of events in projects are interface and milestone.[7] An *interface event* denotes the completion of one task and simultaneous start of one or more subsequent tasks. Event 4 in Figure 5.9 is an interface event. An interface event often represents a change in responsibility: one individual or group completes a task and another individual or group starts the next task. It signifies approval of the task just completed and readiness to begin subsequent tasks.

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 signify progress and, as such, they are important measures. Often, approvals for 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 phase of the systems development cycle. Failure to pass a milestone is usually a bad omen followed by changes to the budget and schedule.

Kinds of Schedules

Two common kinds of schedules are the project schedule and the task schedule. Project managers and upper management use the *project schedule* (project *master* or *execution* schedule) to plan and review 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 scheduling the tasks identified from the WBS or in the scope statement. Later, they refine the schedule in a bottom-up fashion, taking into account the more detailed task schedules 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 that phase to begin.

An *activity schedule* shows the specific activities or tasks necessary to complete a work package. It is created for people working on the activities and enables lower-level managers and supervisors to focus on those activities, and not be distracted by others with which they have no interaction. Activity schedules are prepared by functional managers or subcontractors, but incorporate interface and milestone events as specified on the project master schedule. Project and activity schedules are prepared and displayed in many ways including with Gantt charts.

5.7 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 to most elements of a program plan and that it would be easy to assess progress by viewing each element's status with respect to time. His approach, which came to bear his name, became widely adopted in industry.

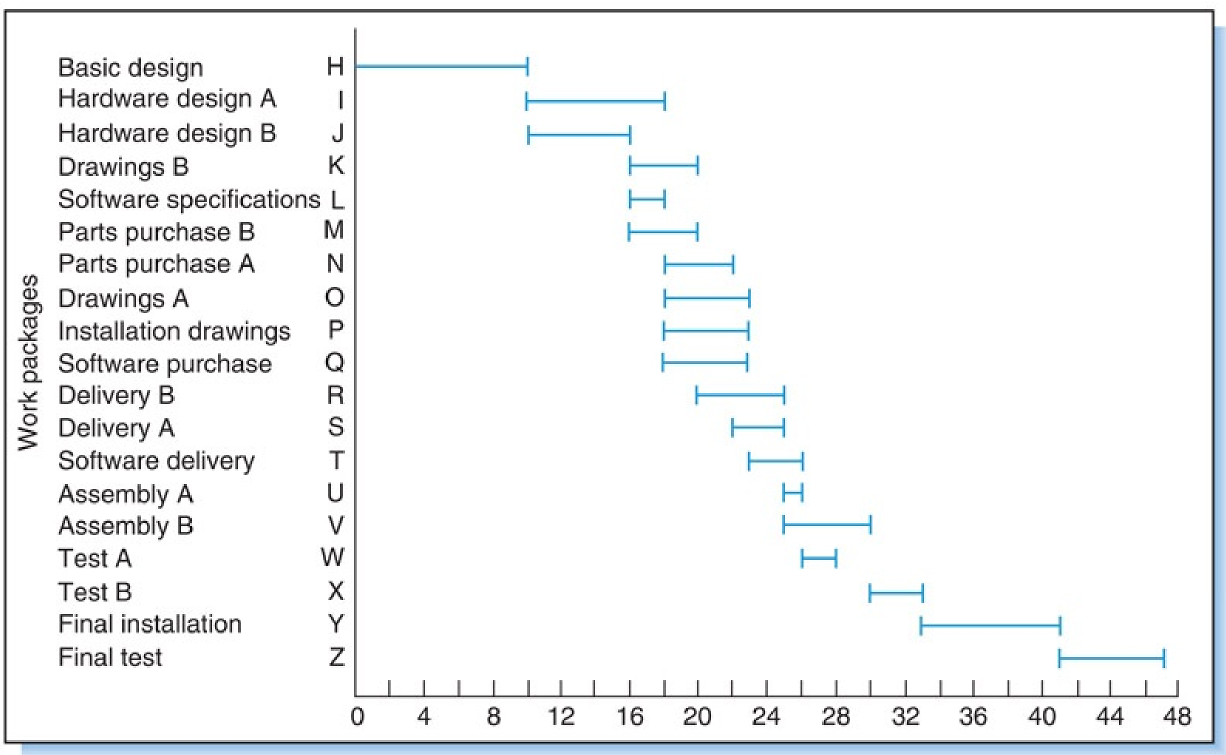


Figure 5.10 Gantt chart for LOGON project.

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 5.10 shows the Gantt chart for work packages in the LOGON project. Listed on the left-hand side are work packages and along the bottom are work weeks. The starting and completion times of the packages are indicated by the beginning and ending of each bar.

Preparation of the Gantt chart comes after a WBS analysis has identified the work packages or other tasks. During WBS analysis, the functional manager, contractor, or others responsible for a work package estimate its time and any prerequisites. The work elements are then listed in sequence of time, taking into account which elements must be completed before others can be started.

As an example consider how the first nine work elements in Figure 5.9 (work packages H through P) are scheduled. 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 "predecessor" inputs mentioned earlier in the discussion of work package definition. Suppose that during the WBS analysis for LOGON it was determined that before work tasks I and J could be started, task H had to be completed; that before tasks K, L, and M could be started, task J had to be completed; and that before task N, O, and P could begin, task I had to be completed.

Before these can be started…	This must be completed
I, J	H
N, O, P	1
K, L, M	J

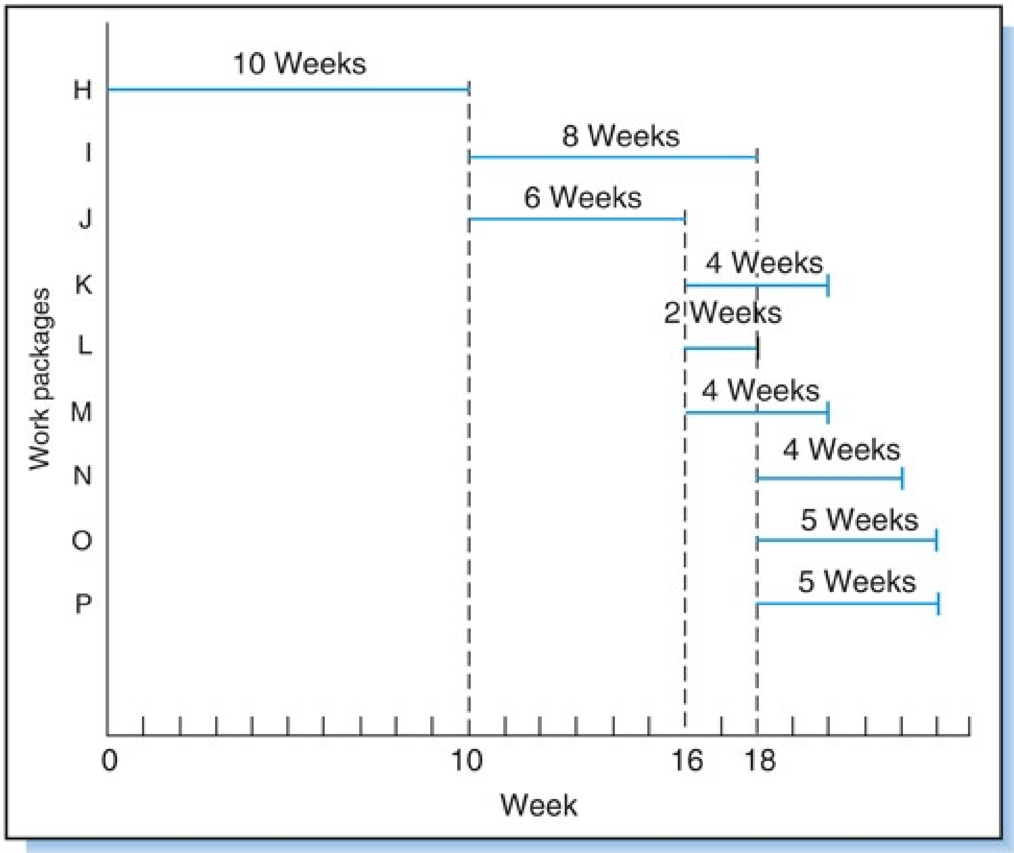


Figure 5.11 Setting up a Gantt chart.

This sequencing logic is used to create the Gantt chart. Thus, as shown in Figure 5.11 (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 task N, O, and P be started. As each new work task is added to the chart, care is taken to locate it following completion of all of its predecessor work tasks. This example uses work packages as the tasks being scheduled, but in fact 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 5.12 shows progress as of the "status date," Week 20. The heavy portion of the bars indicates the amount of work that has been completed. The thinner part of the bars represents work unfinished or yet to be started. This method is somewhat effective for showing which of the work 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 yet to be started.

When the Gantt chart is used like this to monitor progress, the information it reflects must be the most current possible and the chart should be updated daily or at least weekly. Tracking progress is important for identifying and rectifying problems. Posting progress like this is a good way to keep the team motivated.

Hierarchy of Schedules

For large projects with many work elements a *hierarchy* of schedules is used, as illustrated by the three levels in Figure 5.13. 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 bottom or task-level shows work packages or smaller tasks within an activity. Milestones and target dates can be displayed at any level.

Each level 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 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 used by work package leaders and correspond to the task schedules mentioned earlier. Figure 5.14 is a multilevel schedule, showing both the higher-level project

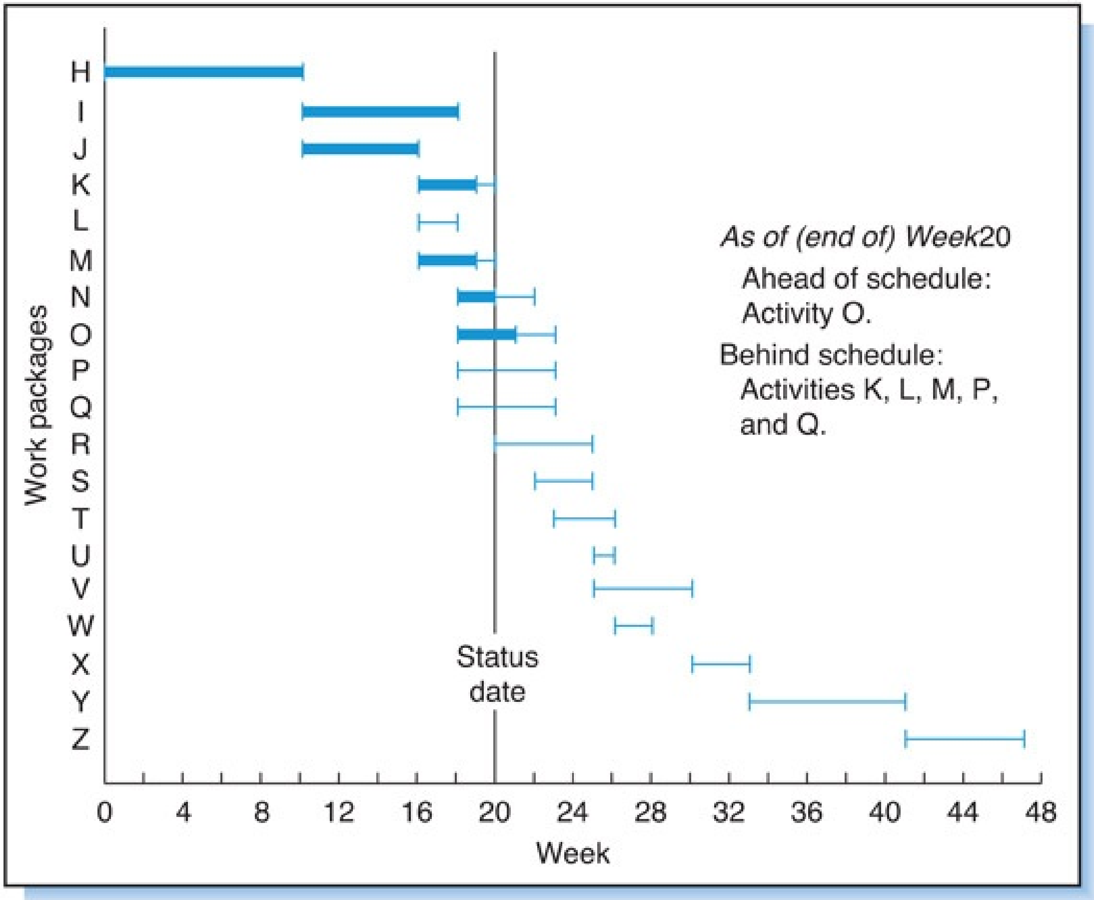


Figure 5.12 Gantt chart for LOGON project showing work progress as of Week 20.

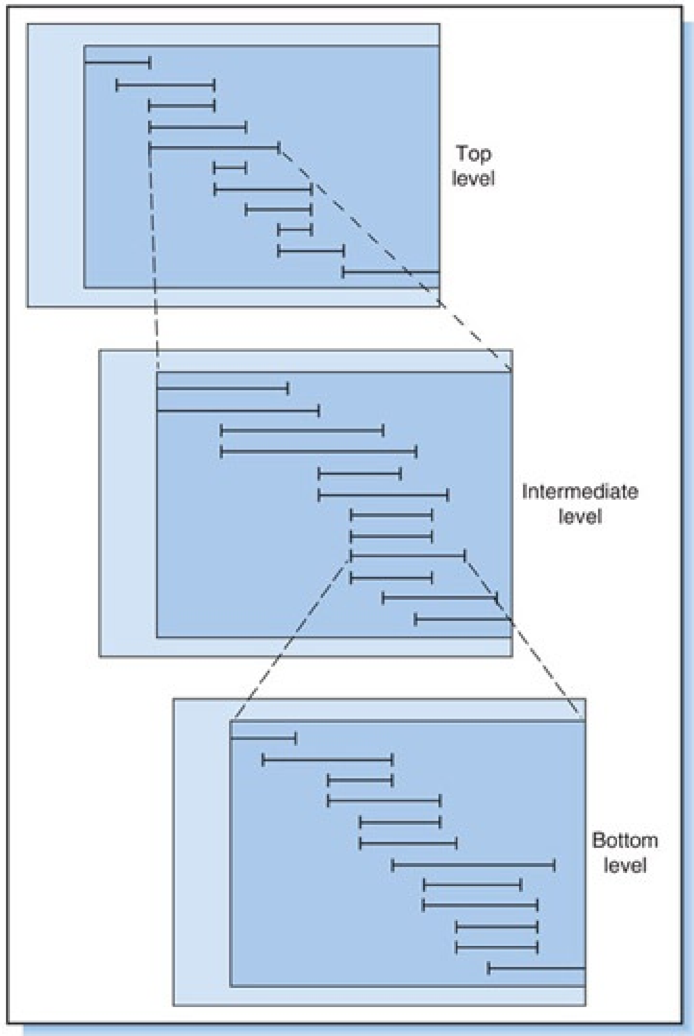


Figure 5.13 A hierarchy of schedules.

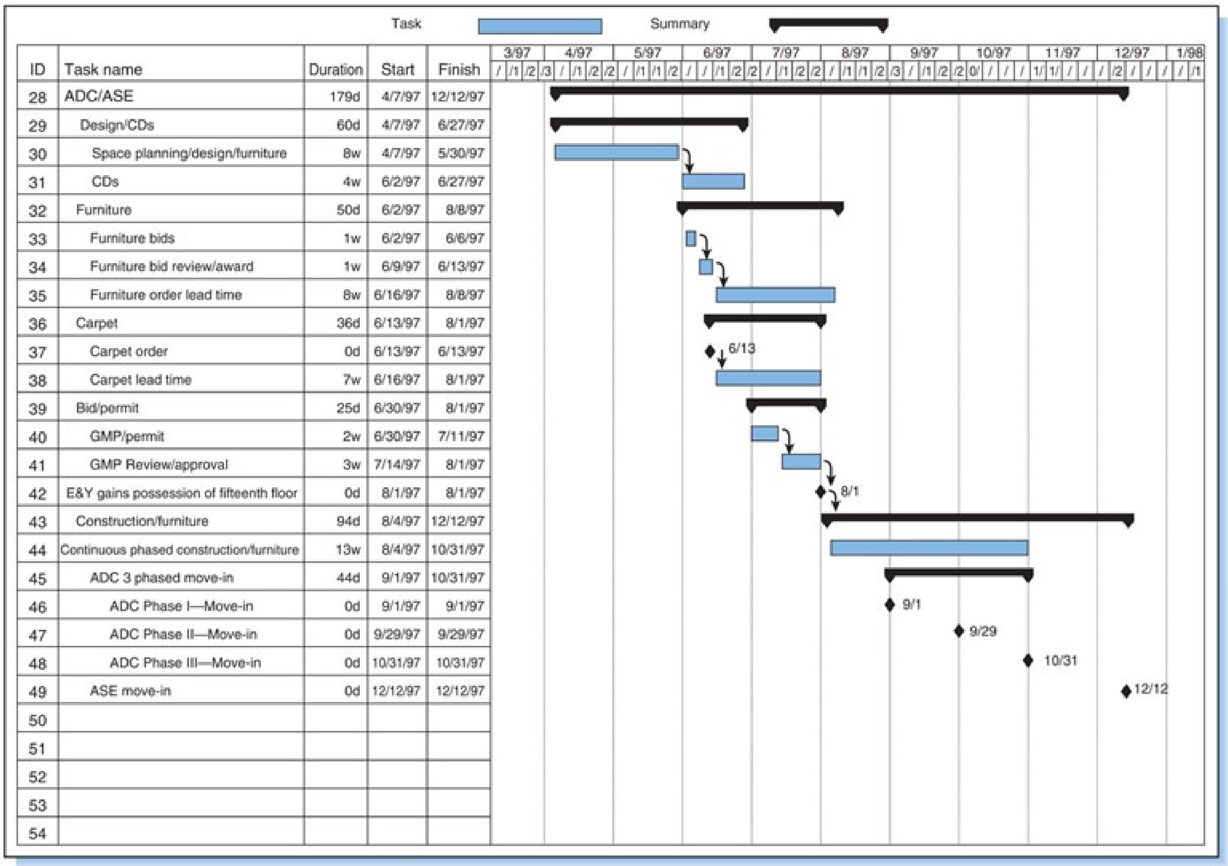


Figure 5.14 Multilevel schedule.

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 planning approach in Figure 4.2.

Disadvantages of Gantt Charts

A disadvantage of the Gantt chart is that it does not necessarily show 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 others be 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.

5.8 Line of Balance (Linear Scheduling Method)

While projects are by definition unique, one-time endeavors, they sometimes contain repetitive work activities. Examples include erecting numerous towers for a new transmission line, constructing multiple largely-identical housing units, and erecting a multi-story building. A method for planning and controlling these repetitive activities is the *line of balance*—LOB (also called the *Linear Scheduling Method* because it is often used on "linear projects" such as highways and pipelines where the physical location of the work can be represented in terms of miles or kilometers).

Example 5.5: Cranes for Construction

A supplier of construction cranes must deliver a total of 12 cranes according to the schedule in Table 5.1. Prior to delivery, a set of activities must be completed for each crane. These are shown as activities A–F in the Gantt chart in Figure 5.15.

Table 5.1 Delivery Schedule for Cranes

Week	Date	Delivery	Quantity Cumulative Delivery Quantity
1	February 7	1	1
2	February 14	2	3
3	February 21	4	7
4	February 28	5	12

Suppose we look only at deliveries on February 14; according to Table 5.1, a total of three cranes must be delivered by then. The question is, how far along should all the other activities be by then? For example, how many power units (activity B) should be bought by then (assuming one power unit per crane), how many components procured, and so on?

According to Figure 5.15, a power unit must be bought 2 weeks prior to the delivery of a crane. Now, look at the right-hand column of Table 5.1; moving down 2 weeks from February 14 shows the number 12: this means that 12 power units must be bought by February 14. Since activities A and C both must be completed 3 weeks prior to crane delivery (Figure 5.15), we see, referring to Table 5.1, that 12 sets of “other components” must be procured (activity C) and 12 sets of structural components must be fabricated (activity A) by February 14. In the same manner, since operators must be trained (activity E) 1 week before delivery, moving down 1 week from February 14 in Table 5.1 shows that seven operators must be trained by February 14. Likewise, we see that seven cranes (activity D) must be assembled by February 14. Also, since tests with operators (activity F) involves zero lead time, three tests should be completed by then.

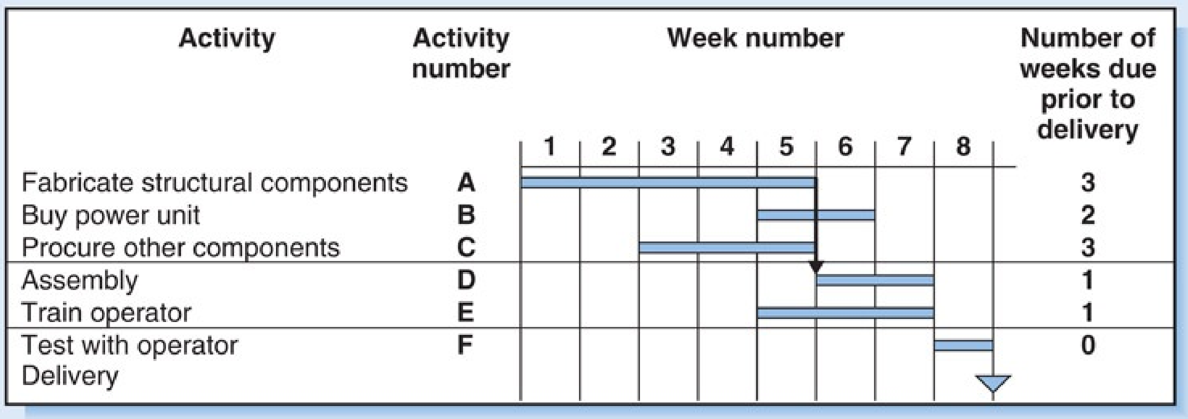


Figure 5.15 Gantt chart of tasks for delivery of one crane.

Figure 5.16 summarizes the LOB—the number of deliverables (completed units) per activity as of February 14. For the cost center, function, or supplier responsible for each activity, the LOB provides information necessary to estimate needed resources and plan work.

An alternative to Figure 5.16 is a diagram that shows the number of units to be completed by a specific activity per time period. For example, Figure 5.17 shows dates and quantities for fabricated structural components (activity A) and assembled cranes (activity D). The same kind of figure can be used to monitor actual units completed and track progress versus planned units.

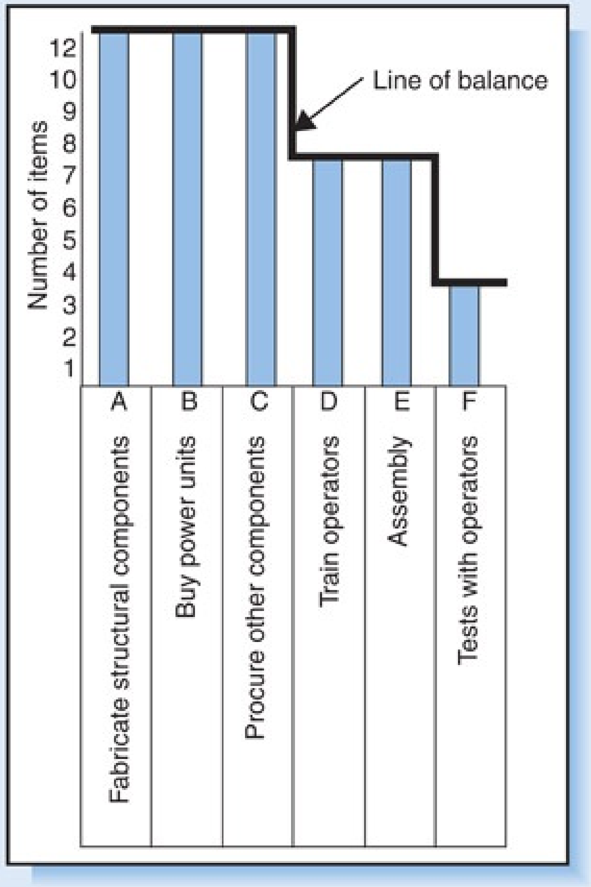


Figure 5.16 Number of deliverables required by February 14 per type of activity.

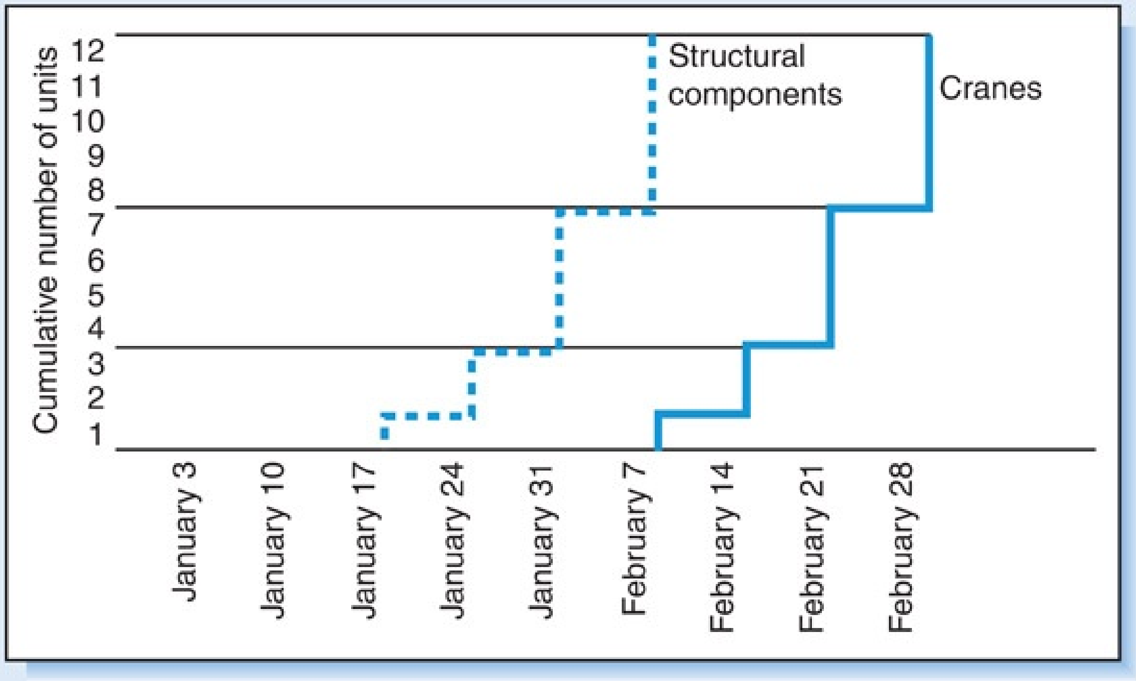


Figure 5.17 Alternative presentation of a Line of Balance schedule.

5.9 Procurement Management[8]

Most projects involve procurement of goods, materials, and subcontracted work. Indeed, in some projects everything is "procured" and virtually nothing is done or 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, subsystems, components, services, or other project deliverables, and of work packages and tasks identified in the WBS.

Certainly, the management of procured materials and outsourced work is every bit as important to the project as work done internally: procured items that exceed budget or schedule, or fail to meet requirements, cause cost and schedule overruns for the entire project.

The role of *procurement management* is to help the planning and control of the following:[9]

1. Equipment, materials, or components designed and built by vendors *specifically* for the project. These procured items might involve parts of, or entire, work packages (e.g. design work, environmental study, soil analysis). Major portions of a project might be wholly outsourced in a "turnkey" arrangement (i.e., subcontractors fully design, build, and install major equipment or components for the project end-item).
2. *Off-the-shelf* (OTS) equipment and components supplied by vendors. These represent products that are readily available and not produced specifically for the project.
3. Bulk materials (cement, metal tubing or framing, wire, stone, piping, etc.)
4. Consumables (nails, bolts, rivets, fuel) or tools for construction or fabrication.
5. Equipment for construction or fabrication not already owned by the contractor; includes cranes, supports, scaffolding, and equipment for shops, welding, and testing.
6. Administrative equipment not already owned by contractor; includes computers and project office facilities and equipment.

To simplify, we lump these items together here and refer to them as *procured goods, work, or services* (GWS). Goods refers to raw materials or produced items; work means contracted labor; and services includes consulting.

The term "procurement" represents activities related to bought, or subcontracted items, although other terms are also used, but with the following distinctions: whereas "acquisition" refers to the purchase of an *entire complex system* that is not well-defined (including its design, development, ramp-up, and production), and "buying" refers to the purchase of a standardized (off-the-shelf) *item or part*, "procurement" refers to the purchase of a component or subsystem *(less than entire system)*—including its design and/or production—according to specifications provided by the customer. Hence it would be appropriate to say the "acquisition of a nuclear power plant," the "procurement of an automatic shut-down safety device," and "buying a batch of standard one inch nails."

Procurement management includes most everything associated with contracting and contract administration: soliciting bids and selecting contractors, establishing legally binding contracts between parties, managing the execution of the contracts, and closing out of contracts. The first few of these topics will be covered here and the others in Chapters 11 and 12.

See Chapter 11 and Chapter 12

Soliciting and Evaluating Bids

Once the decision is made to procure GWS, potential vendors are solicited to offer bids or proposals. A customer who has a long-term relationship with a supplier or contractor, especially one with unique skills or capabilities, will usually approach the contractor and negotiate a contract. This is called *sole-sourcing* because only one contractor is considered for the contract. When the scope of the project is somewhat simple and the requirements are well defined, the customer can advertise for bids online and other media using an RFP or RFQ (Request for Quotation, a simple price quote). For a large and somewhat undefined system that requires design work or other intellectual input, an RFP or IFB (information for bid) is sent to a short list of qualified suppliers. The RFP or

IFB might be preceded by an RFI, a request for information to determine if the contractor is qualified and should be sent an RFP. Sometimes the RFP is accompanied with a *bidder's* or *contractor's conference* to explain the background and scope of the project, documentation required from contractors, and contractual requirements. Acceptance of a bid will result in a formal contract with content and conditions as described in Chapter 3. When the procured item is hardware, the contract should specify at what point the supplier is no longer responsible for damages and the buyer becomes responsible.

The basic types of contracts are described in Chapter 3, although certain industries require specific contract formats.[10] Procurement management is a specialized function that requires legal and contract administration skills; in some organizations a specialized procurement division handles it.[11]

See Chapter 3

Procurement Planning and Scheduling

The first step in procurement planning is to estimate the procured items, labor, or services needed for the project. Associated with every work package are procured GWS requirements, some that will be shared with other work packages. Items to be procured are identified during the WBS process, either from planning the work and resources needed for particular work packages, or from knowing that entire work packages must be outsourced. In the former case, managers responsible for each work package identify the GWS within the package that must be procured (e.g. the work package "build wing" will require procuring fiberglass, aluminum, and other materials from suppliers); in the latter case, managers will recognize that certain (sometimes significant) portions of the project (*entire* work packages) will have to be outsourced (e.g. the work package "develop rocket motor" will require development, fabrication, and testing—*all* to be provided by contractors).

Associated with each procured item is a schedule specifying when the item is needed and when procurement activities must begin. Everything to be procured in the project must be scheduled in advance to allow enough time to conduct the

RFP/proposal/supplier-selection process (described in Chapter 3), and for suppliers to deliver (or design, build, and then deliver) the items at the times needed. Figure 5.18 shows the considerations in scheduling a procured GWS item. The schedule is prepared by working backwards, starting with event 10, the date when the item *must* be available for the project. This schedule is then integrated with the project schedule to assure that the procurement process happens far enough in advance so that the item will be available when needed. This procedure is repeated to schedule all procured GWS items.

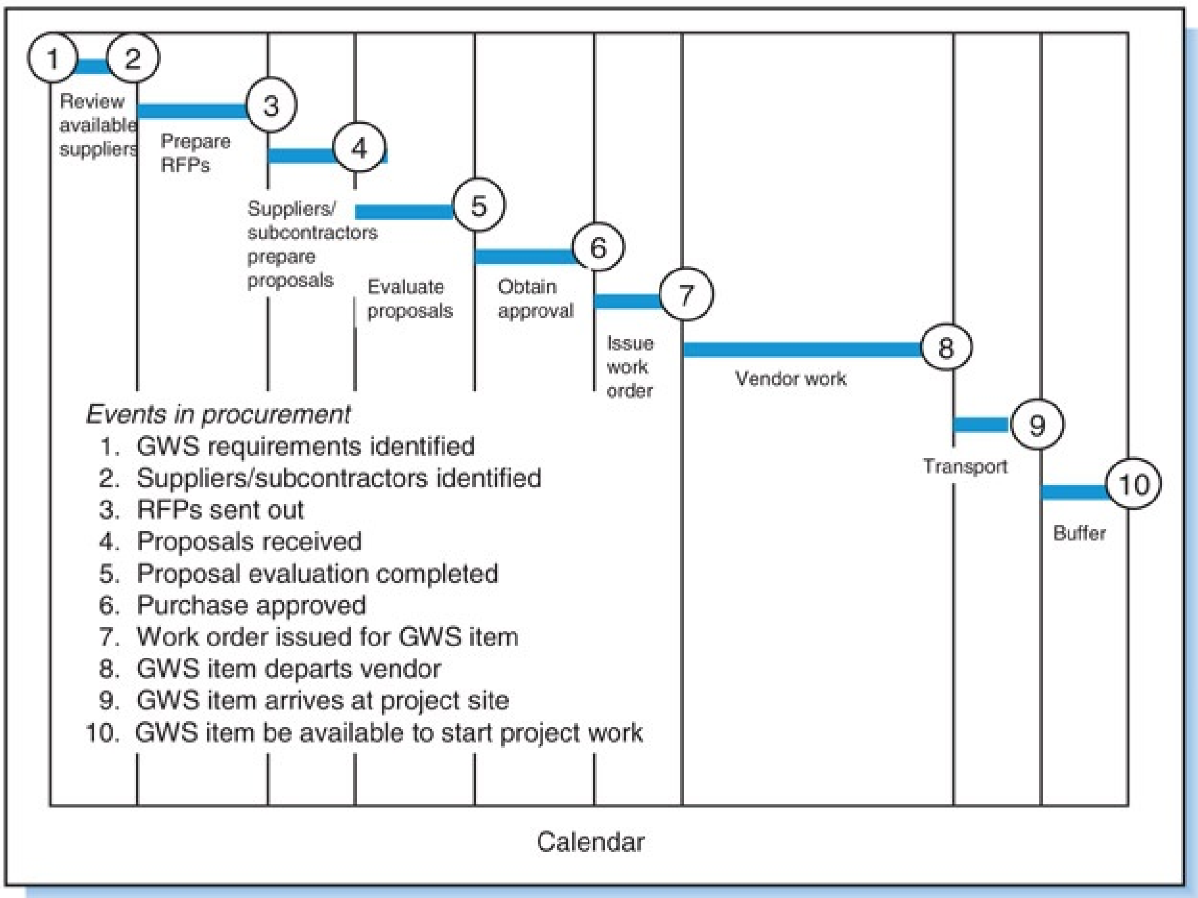


Figure 5.18 Procurement activities schedule.

Source: Adapted from Joy P. *Total Project Management.* Delhi: Macmillan India Limited; 1993, p. 383.

Of course, preparing such a schedule requires knowing the lead times for each of the procurement activities—the time needed for, e.g. suppliers and subcontractors to prepare proposals, the project manager to evaluate the proposals and issue contracts and work orders, and suppliers/subcontractors to fulfill the work orders (which could involve their designing, building, and testing of equipment or components). It's not uncommon, especially in international projects, for these times to be grossly underestimated and, subsequently, the project delayed.

The schedule in Figure 5.18 starts at the point where GWS requirements have been identified. To get to that point, however, the system requirements and specifications must first have been defined—another reason for careful attention to system definition in the project life cycle.

Procured GWS require the same treatment in project planning as internal aspects of the project; hence matters such as the responsibility, budget, quality, and risk for procured items must also be addressed in the plan—topics to be discussed in later chapters.

Logistics Plan

Logistics relates to the transport and storage of materials. In projects that are materials-intensive, the loading, unloading, transportation, inspection, clearances and approvals, and storage of materials can be major issues. For example, consider a large construction project and the importance of timing the arrival of materials (steel, pipes, concrete slabs) to coincide with when those materials 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 do you *put* them? Where do you put the truck that brings them? In congested urban areas there simply is no space; when there is, materials delivered early are subject to damage, deterioration, and theft. Whenever GWS items cannot be scheduled to arrive *just in time*, provision must made to store and protect them, which on large projects can be very costly.

5.10 Summary

The purpose of project planning is to determine the way in which project goals will be achieved—what must 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, planned, and controlled. Most all 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.

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 only high-level tasks and work packages; task-level schedules show the tasks needed to complete individual work packages. The most common form of schedule is the Gantt chart. As a visual planning device it is effective for showing when work should be done and whether work elements are behind or ahead of schedule.

Project plans must account for all resources and work necessary for the project, including those procured (provided by suppliers and contractors). Procured items and the procurement process must be included in all elements of the project plan—the WBS, schedule, responsibility matrix, budget, and so on.

The concepts and techniques in this chapter are foundation tools for planning and scheduling. The next few chapters look at additional techniques for planning and scheduling. Later chapters address the role of the WBS, work packages, and schedules in cost estimating, budgeting, and project control.

Review Questions

1. What questions need to be answered every time when 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 (a few weeks or months) with relatively 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 differences and similarities 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 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 5.5 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 5.8 shows some possible types of responsibilities that could be indicated on a responsibility matrix. What other kinds of responsibilities or duties could be indicated?
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. Distinguish an event from an activity. What problems can arise if people on a project confuse these terms?
22. Distinguish an interface event from a milestone event. Give some examples of each. When is an interface event also a milestone event?
23. How are project-level and task-level schedules prepared? What is the relationship between them? Who prepares them?
24. Construct a Gantt chart similar to the one in Figure 5.10 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?

25. How must the Gantt chart you drew in problem 24 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?
26. Is the Gantt chart adequate for planning and controlling small projects?
27. In a hierarchy of schedules, how does changing a schedule at one level affect schedules at other levels?
28. How do you decide when more than one level of schedule is necessary?
29. If a hierarchy of schedules is used in project planning, explain whether there should be a corresponding hierarchy of plans as well.
30. What aspects of the project fall under "procurement management"? Why is managing procured items just as important to project success as managing internal items? What are the issues in scheduling procured items?
31. Consider this statement: The management of procured items can pose greater difficulties than managing internal items. Do you agree or disagree, and why?

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 were activities in the WBS transferred to a schedule? How were times estimated? Who prepared the schedules?
15. Show examples of project-level and task-level schedules. Who prepared each? How were they checked and integrated?
16. 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?

Case 5.1 Barrage Construction Company: Sean's WBS

Sean Shawn was recently appointed project planner at Barrage Construction, a company that specializes in custom-made garages. He had 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 well enough from having talked to company employees. He sat down and drew the WBS in Figure 5.19. To estimate costs for each work category in the WBS he reviewed company cost records from three recent two-car garage projects he thought similar to the 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 $43,000 for a two-car garage and $64,500 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.

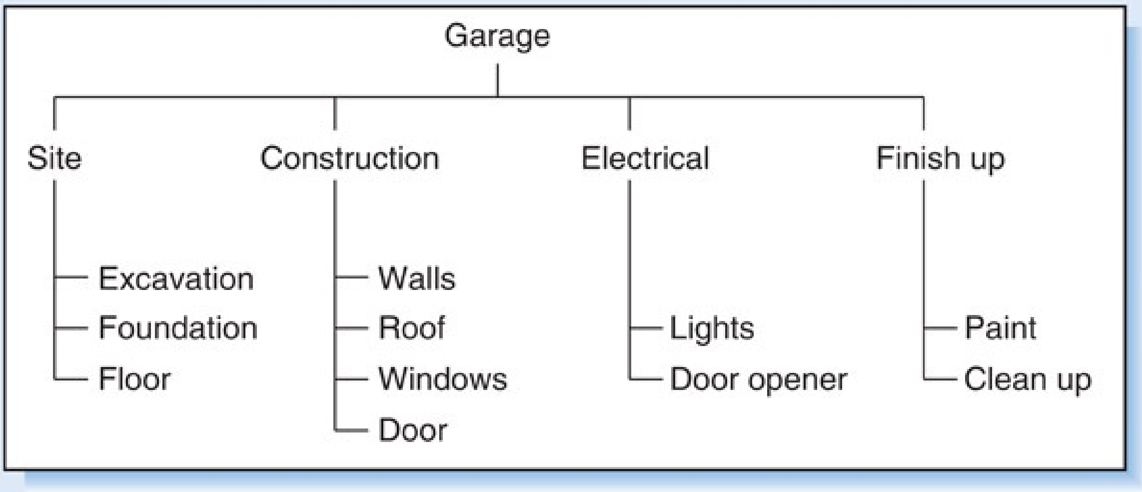


Figure 5.19 Sean’s garage project WBS.

Question

What is your opinion of Sean's approach to creating a WBS and estimating project costs? Please elaborate.

Case 5.2 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 1 week (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, gives it to a temp agency, and receives a price quote.

Deva shows the completed plan to her manager and asks him to review it. The plan consists of the Gantt chart and a budget that is largely 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 5.3 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 long list of requirements, but after reviewing the project SOW and requirements list, the first thing Walter wonders is who is going to be on his project team. He asks his manager, who gives him the names of three people in the department 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 they are the most qualified or experienced to do. Since he has worked with the 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, while persons B and C have 4 tasks and 5 tasks, respectively. To balance out the workload, he takes four of the tasks from person A and splits them between the other two. He is pleased because, he feels, with seven, six, and seven 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 they must be done.

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 soon forthcoming, to 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 5.4 Planning the Boca Implementation at Kulczyński Products

Tomasz Grabowski is newly hired as a project manager of IT at Kulczyński Products. His first project is to install the brand-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 need to be performed, shown below. He is proud of the list and thinks it's 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 console machines and prerequisite software.
11. Install Boca Business Systems Manager on console machines.
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 tasks:

    a. Extend the data model.
    b. Configure the instance placement.
    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 the Automated Business Systems file and XML definitions to verify resource inclusion and placement.
    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 31. 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 an earlier project. He takes that timeline and modifies 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, which is too late. He reviews the estimates and reduces several of them enough to shorten the timeline by 2 months. Figure 5.20 is the finished timeline.

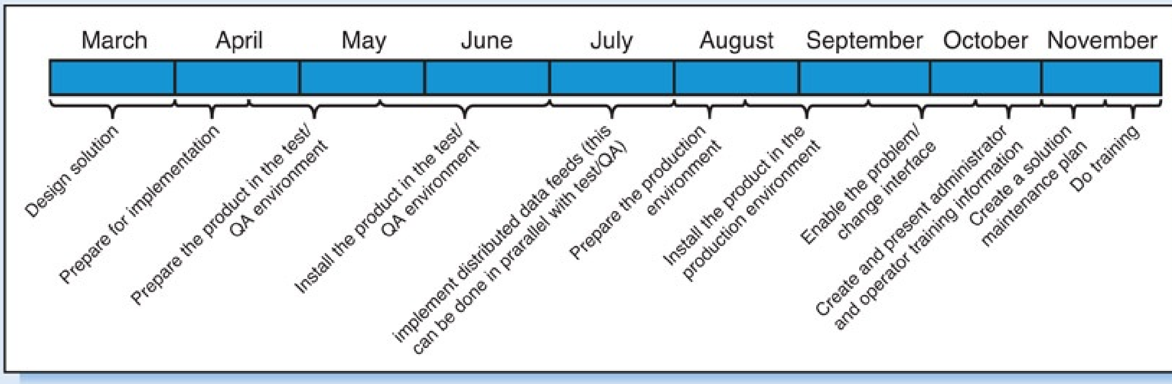


Figure 5.20 Boca Implementation Project timeline.

Tomasz intends to assign responsibility as follows:

- The Boca Team is responsible for everything on the task list with the word “Boca.”
- The Kulczyn’ski Team is responsible for everything else on the list.

Just before the project begins Tomasz gives the Boca Team and Kulczyn’ski Team each a copy of the task list and the timeline. He explains to them, “Here’s the list of what we need to do. If you are able to do everything within the timeline, we should be able to meet the project deadline.” Comment on Tomasz’s “plan.”

Endnotes

1. Some organizations use the term "project charter" to refer to an "execution plan." Our preference is for 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 that will guide the project team though project execution.

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; especially important is to anticipate and prepare for possible struggles, obstacles, and conflicts to be encountered.

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. Archibald R. *Managing High-Technology Programs and Projects.* New York, NY: John Wiley & Sons; 1976, pp. 65, 156.

8. Portions of this section adopted from Joy P.K. *Total Project Management.* Delhi: Macmillan India Limited; 1998, pp. 378–400.

9. Ibid., pp. 378–380.

10. Examples: NEC or New Engineering Contract, The Institution of Civil Engineers, *The Engineering and Construction Contract.* London: Thomas Telford; 1995; and FIDIC, International Federation of Consulting Engineers, Lausanne, Switzerland, http://www1.fidic.org.

11. See Whittaker R. *Project Management in the Process Industries.* New York, NY: John Wiley & Sons; 1995.

Chapter 6
Project Schedule Planning and Networks

You can't always get what you want.

—Rolling Stones

Project scheduling is more than just displaying tasks on a Gantt chart. It 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. A Gantt chart is good for communicating the project schedule, however 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.

6.1 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 6.1 is a network diagram for "getting up in the morning and getting dressed" (for a male). The boxes represent activities or tasks, and the arrows between them show 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 6.1 is of course for illustration purposes only; any real-life attempt to plan work in such detail would be micro management and 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, sub-projects belonging to a project, or work packages in a project, subproject, or specific facility.

See Chapter 5

Networks also show *events*. As described in Chapter 5, an event represents an *instant* in time, an "announcement" that something has happened or will happen. Typically it signifies the start of an activity 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*.

See Appendix A to this chapter

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.

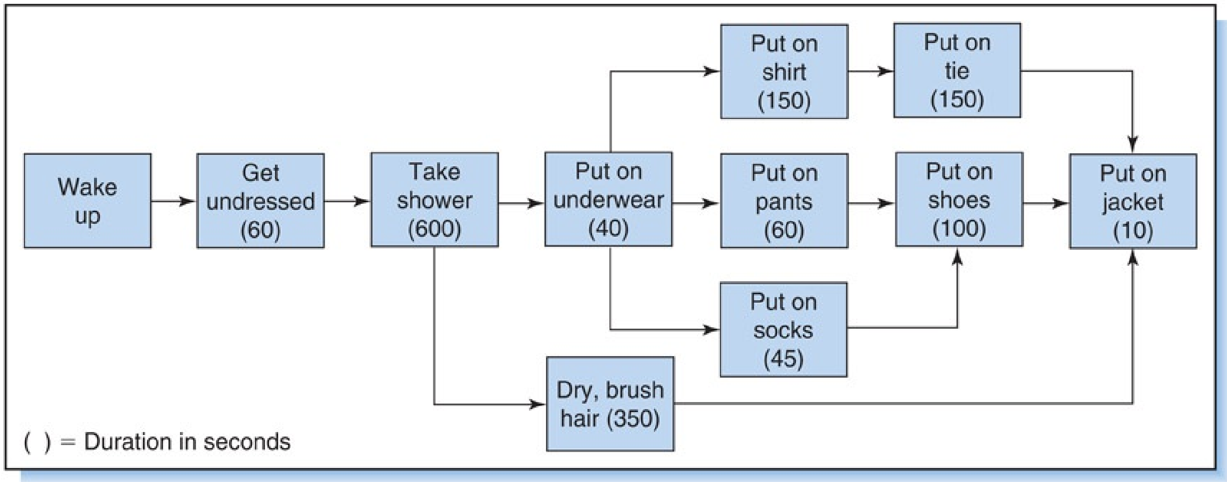


Figure 6.1 Network diagram for getting up and getting dressed.

Activity on Node (AON) (or PDM) Diagrams

Figure 6.2 shows an activity as represented in the AON method. Each *node* (box) in the network is an activity; shown inside the node is information about the activity such as its duration, start time, and finish time.

To construct an AON network, start by drawing the first activity in the project (e.g. "wake up"). From this activity draw arrows to the activities that happen next. As shown in Figure 6.1, activities are added one after another, in sequence or parallel.

But before you can actually create a network, you must first know the dependency *relationships* among the activities. For each activity you need to know, for example:

- 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 one has predecessors, or activities that must be completed ahead of it; in Figure 6.1, for example, "put on shirt" is a predecessor for "put on tie." Similarly, every activity except the last one 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."

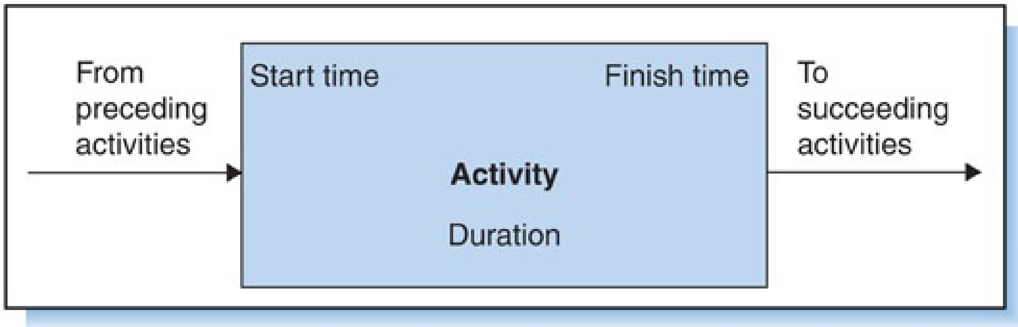


Figure 6.2 AON presentation for an activity and its start and finish events.

It is important to distinguish *mandatory* and *discretionary* dependency relationships:

- Mandatory: the sequence of two activities (which activity should precede the other) cannot be reversed. The relationship between "put on socks" and "put on shoes" in Figure 6.1 is an example.
- Discretionary: the sequence is a matter of choice. For example, "dry, brush hair" and "put on jacket" *can* 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*.

In another kind of dependency, called *external* dependency, an activity depends on some event or activity that is not in the network. For instance, in Figure 6.1 the activity "take umbrella" might be included at the end of the network; it would depend on an external factor, the "forecast for rain."

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 have to first "wake up.") Given information on immediate predecessors, it is easy to construct a network. For example, the network in Figure 6.1 can be constructed from the information in Table 6.1. Start with the first activity in the project (the one in Table 6.1 with no immediate predecessor) —"get undressed," then connect it to 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 6.1.

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* or *in any order*. Once the network has been completed, check the relationships among activities for completeness and logical consistency.

Another example is given in Table 6.2. The network diagram for this project, shown in Figure 6.3, 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 D has two immediate predecessors, B and C, it is connected to both of them; similarly, so is Activity E. Each node is labeled to identify the activity and its duration.

Table 6.1 Activities and Immediate Predecessors

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	100
	{ Put on socks	
Put on jacket	{ Put on tie	10

	{ Put on shoes
	{ Dry, brush hair

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 6.3, 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 6.2 Activities and Immediate Predecessors

Activity	Immediate Predecessor
A	–
B	A
C	A
D	B, C
E	B, C

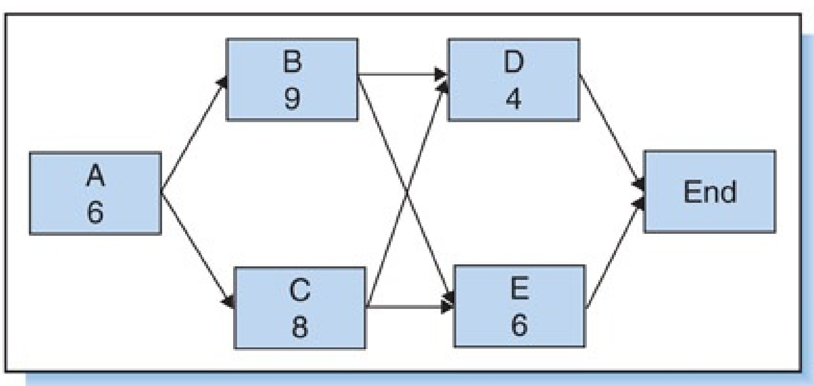


Figure 6.3 AON diagram corresponding to project in Table 6.2.

As a final example, Table 6.3 shows the immediate predecessors for the LOGON project using work packages from the WBS in Chapter 5. Figure 6.4

shows the corresponding network.

See Chapter 5

Tables 6.1, 6.2 and 6.3 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 6.2 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 need be known to construct a network.

Creating a Project Network

A project network is created using a list of activities from the WBS and their predecessors. If done by hand the process is trial and error and the network might have to be redrawn a few times before it is correct. Even if done by computer, good practice is to first sketch out the network by hand to create an initial ("coarse grain") network and then 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 sub- networks that represent, for example, sub-projects, work packages, 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 that the phases can be authorized and milestone payments approved.

Table 6.3 Activities and Immediate Predecessors for LOGON Project

Description	Immediate Predecessors	Duration (Weeks)
Basic design	–	10
Hardware design for A	H	8

Hardware design for B	H	6
Drawings for B	J	4
Software specifications	J	2
Parts purchase for B	J	4
Parts purchase for A	I	4
Drawings for A	I	5
Installation drawings	I, J	5
Software purchases	L	5
Delivery of parts for B	M	5
Delivery of parts for A	N	3
Software delivery	Q	3
Assembly of A	O, S	1
Assembly of B	K, R	5
Test A	U	2
Test B	V	3
Final installation	P, W, X	8
Final system test	Y, T	6

* Work packages from WBS, Figure 5.5.

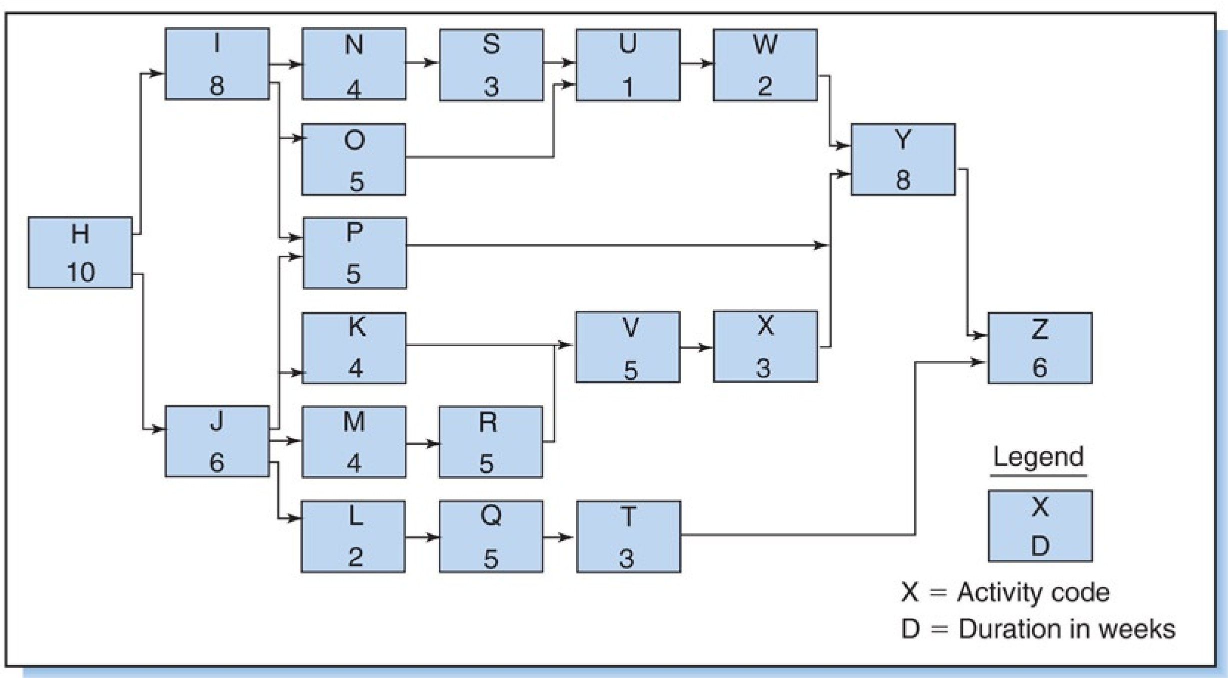


Figure 6.4 Network diagram for LOGON project.

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* or *progressive elaboration*) 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 a Gantt chart can be developed, as explained later.

6.2 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 take longer than planned.

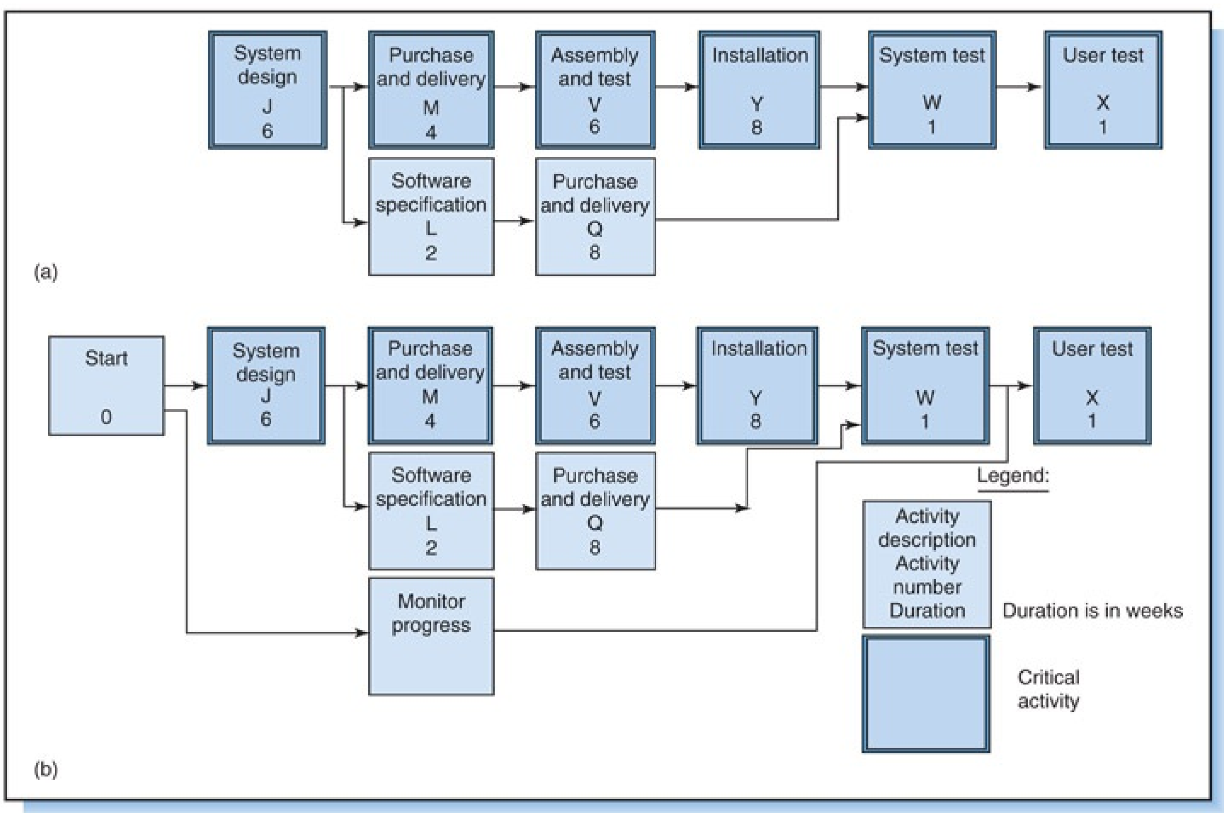


Figure 6.5 Network for ROSEBUD project.

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 6.5 shows the network. (Parts (a) and (b) of Figure 6.5 are very similar; for now look only at (a).) The first phase inthe project is systems design (Activity J), followed by the simultaneous phases of (1) purchase, assembly, and installation (Activities M–V–Y), and (2) software specification and purchase (L–Q). These two phases are followed by the last phase of the project, 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 6.5(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—called the critical path—gives the project duration.* The critical path is highlighted on the figure; activities with the darker "framed" boxes are critical.

If it becomes necessary to reduce the project duration, any reduction effort (e.g. 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 from their earliest possible dates 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 will be completed, at 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 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 provide warning signals when noncritical activities are at risk of becoming critical; this is done in the *critical chain* method, discussed in Chapter 7.

See Chapter 7

Figure 6.5(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 6.5(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" is also used to describe a *summary activity;* e.g., a set of activities aggregated into one work package.)

A final example is Figure 6.6. This network has four paths leading 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

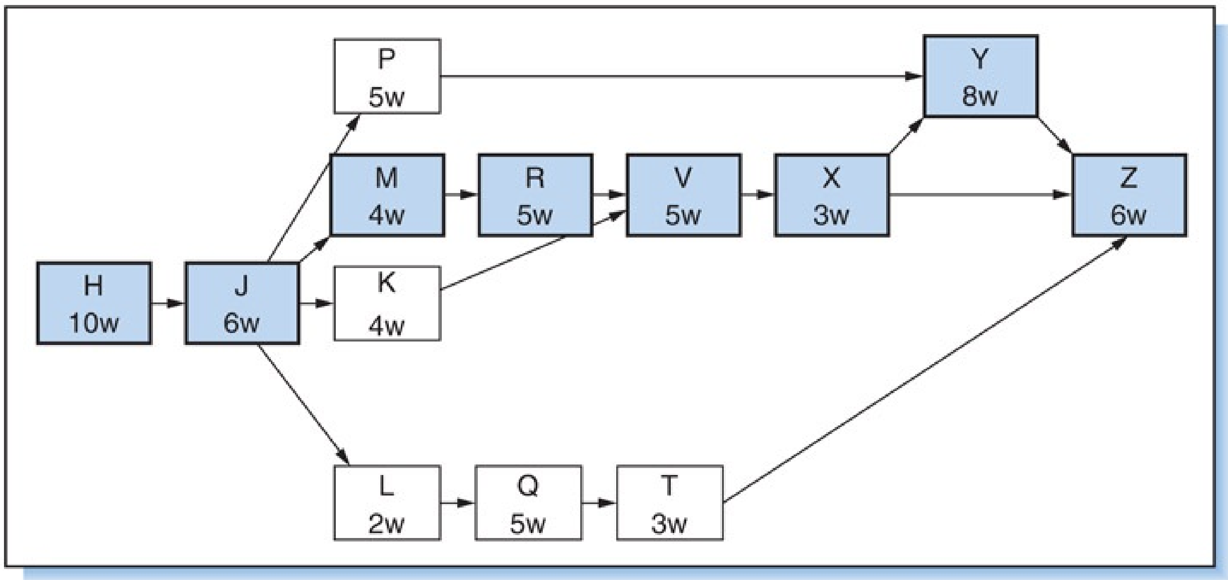


Figure 6.6 Example network showing the critical path (created with Project Scheduler 8.5).

The longest of the four paths is Path c (indicated by the "shadowed" boxes); hence c is the critical path and T_e = 47. (In Figure 6.6, notice the arrow between X and Z is unnecessary: if Z follows Y and Y follows X, it goes without saying that 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 6.6 were 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 more common "Time 0" assumption; Appendix B to this chapter describes scheduling under the "Day 1" assumption.

The formula for computing finish time given start time and duration is:

Finish time = Start time + 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). These times represent the earliest possible times that the activity can be started and completed.

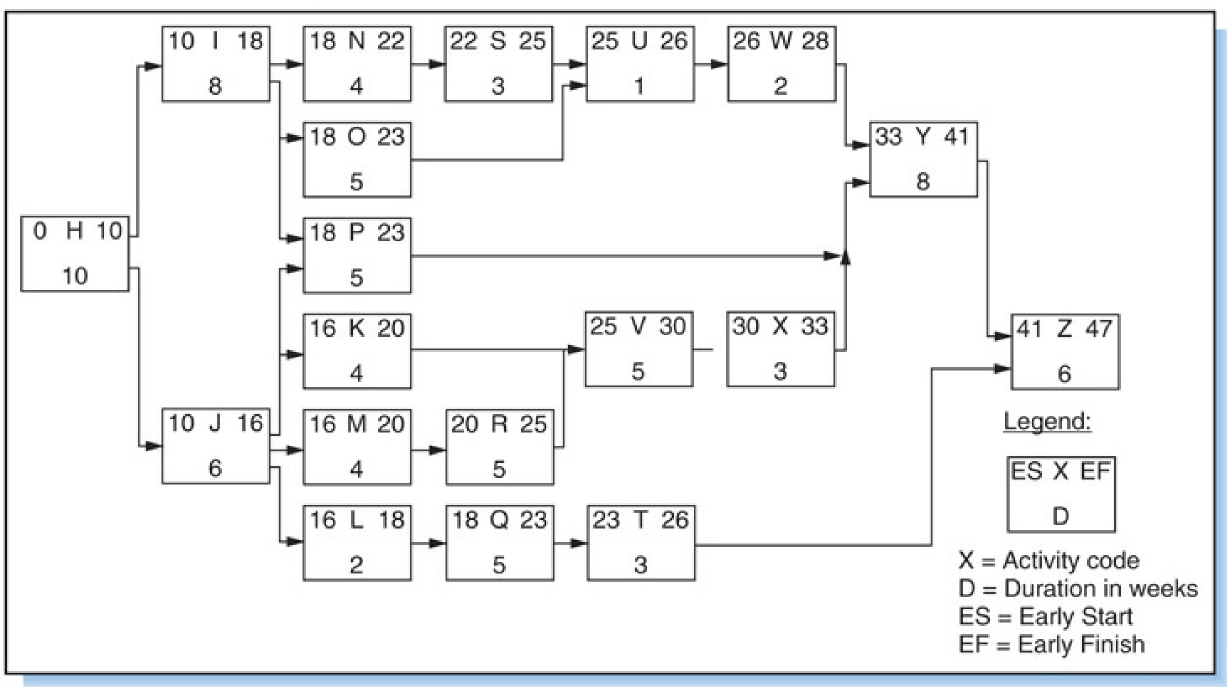


Figure 6.7 Example network showing ESs and EFs.

But the ES of an activity depends on the EFs of its immediate predecessor, which is found by summing the durations of all the predecessor activities along the path 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 immediate predecessor that has the *latest EF.*

All this is shown in Figure 6.7. 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

EF = ES + Duration

In Figure 6.7, 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, Activity J's ES will be Week 10 and its EF will be Week 16. Activity J is the only immediate predecessor for activities K, M, and L, so the ES for activities K, M, and L will be Week 16. Activity V has two immediate predecessor activities, 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 predecessor activities, which is R. Thus, ES for Activity V is Week 25.

The same happens at Activity P: it has two immediate predecessor activities, I and J. Since Activity I's EF is 18 and Activity J's ES is 16, Activity P's ES must be Week 18. Activity Y has three immediate predecessor activities, 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, the latest EF of its immediate predecessor activities and Y and T. Activity Z's EF is Week 47, which 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 the predecessor. When an activity has several immediate predecessors, its ES is based on the *latest* EF of all the immediate predecessors.

Late Times: Late Start (LS) and Late Finish (LF)

As earlier discussed, 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 6.8. 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 have to be completed *as soon as possible*, the date for T_s is the same as the 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 − Duration

In Figure 6.8, 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, for Activity Y and Activity T the LF is 41 weeks, and LS for Y is 41 − 8 = 33. Continue moving backward like this through each path, computing LF and LS for each activity.

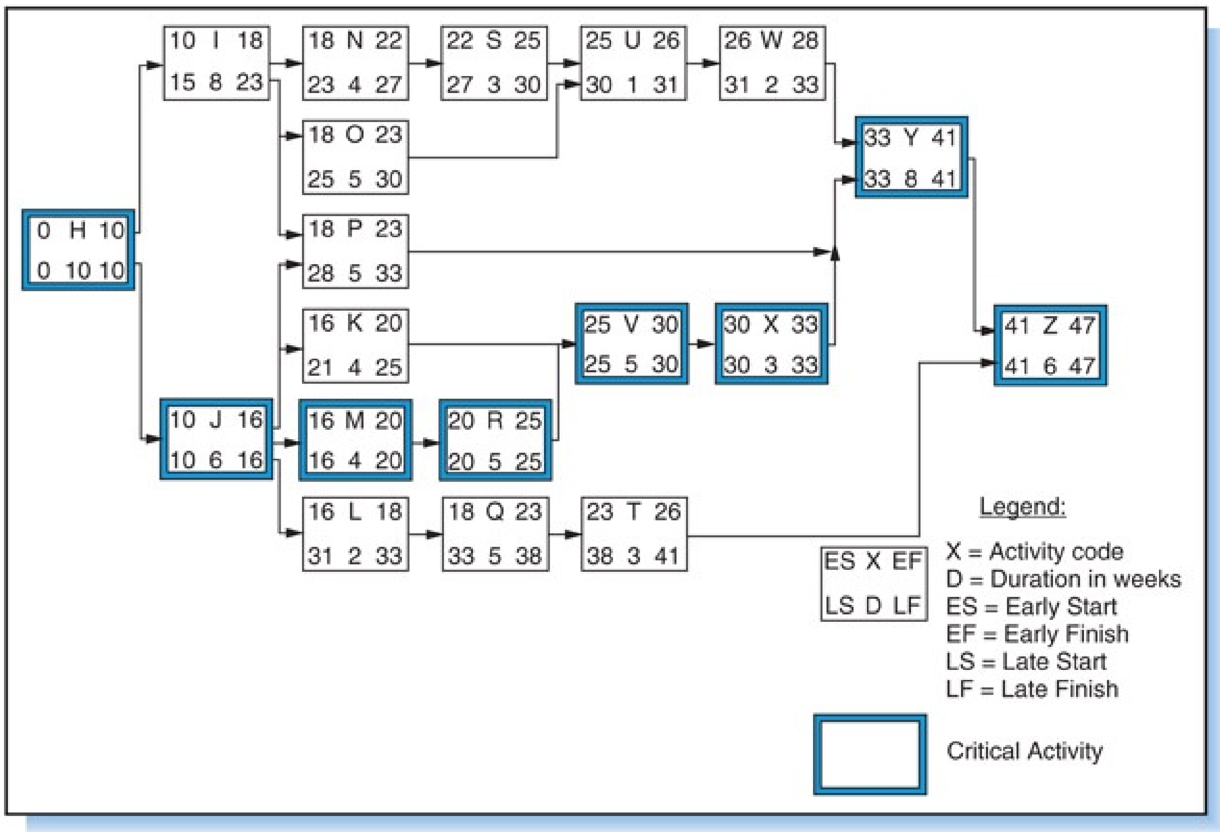


Figure 6.8 Example network showing LFs and LSs.

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):

- LF for Activity P = 33 weeks (because LS for Activity Y is Week 33); thus LS for P is 33 − 5 = 28
- LF for Activity K = 25 weeks (because LS for Activity V is Week 25); thus LS for K is 25 − 4 = 21 weeks
- Similarly, LF for Activity M = 20, so LS = 20 − 4 = 16 weeks
- LF for Activity L = 33, LS = 33 − 2 = 31 weeks.

Since the smallest (earliest) LS is 16 for Activity M, the LF for Activity J is16; this is the latest time Activity J can be finished to enable all of 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 6.8, the ESs and LSs 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 when an activity *must* take place at the latest and when it *can* take place at the earliest. It is the amount of time an activity can be delayed without delaying the project:

Total slack = LS − ES
= LF − EF

In Figure 6.8 the total slack for Activity H using 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 6.4.

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 6.8, 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 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 there are sufficient resources, 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 the *critical chain* and discussed in Chapter 7, schedules activities as late as possible.)

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, although they create Gantt charts using the early times. As discussed, however, activities should not necessarily be scheduled according to the early times.

Table 6.4 LOGON Project Time Analysis (from Figure 6.8)

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 and the activity delayed any further, delays successors and affects the end date of the project as a whole.

***Free slack* is the spare time on an activity that, if used up, does not affect the early start time of any succeeding activities (i.e., will not affect the total slack nor delay any successor).

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:

Free slack for activity = ES (earliest successor) − EF (activity)

For example, in Figure 6.8 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 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 (specified in the responsibility matrix) 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 can delay other activities and disrupt the schedules of everyone who 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 6.4 summarizes these concepts, showing ES, LS, EF, and LF, and total and free slack for the LOGON project in Figure 6.8. 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.

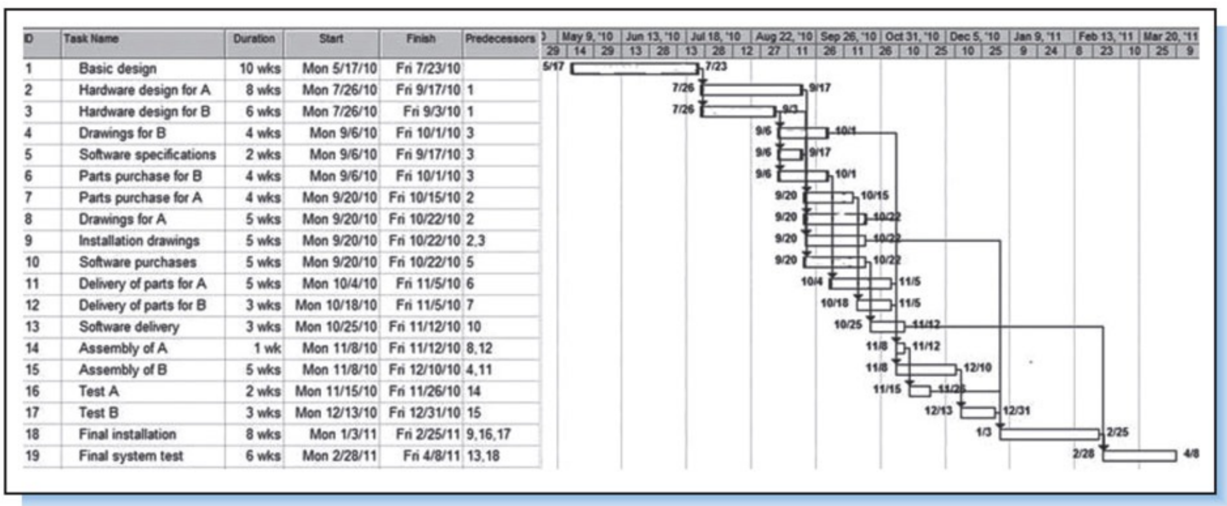
Setting the target date to *later* than T_e has the effect of *increasing* total slack for every activity in the project by the amount $T_s - T_e$. Although no longer zero, the slack on the critical path will still be the *smallest* slack anywhere in the network. For example, if the target completion date for the project in Figure 6.8 were increased to $T_s = 50$ weeks, then the total slack in Table 6.4 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_e, then the total slack times everywhere in the project will be reduced by the amount $T_s - T_e$ 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 customer target date. (Note that altering T_s has no influence on free slack times: these depend on early start and early finish times, both of which are affected by 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 resultant T_e. 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, noting the feasibility of speeding up activities in the project to eliminate negative slack times on the critical path.

6.3 Converting to Gantt Calendar Schedules

Using information from tables such as Tables 6.2 or 6.3 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.



ID	Task Name	Duration	Start	Finish	Predecessors
1	Basic design	10 wks	Mon 5/17/10	Fri 7/23/10	
2	Hardware design for A	8 wks	Mon 7/26/10	Fri 9/17/10	1
3	Hardware design for B	6 wks	Mon 7/26/10	Fri 9/3/10	1
4	Drawings for B	4 wks	Mon 9/6/10	Fri 10/1/10	3
5	Software specifications	2 wks	Mon 9/6/10	Fri 9/17/10	3
6	Parts purchase for B	4 wks	Mon 9/6/10	Fri 10/1/10	3
7	Parts purchase for A	4 wks	Mon 9/20/10	Fri 10/15/10	2
8	Drawings for A	5 wks	Mon 9/20/10	Fri 10/22/10	2
9	Installation drawings	5 wks	Mon 9/20/10	Fri 10/22/10	2,3
10	Software purchases	5 wks	Mon 9/20/10	Fri 10/22/10	5
11	Delivery of parts for A	5 wks	Mon 10/4/10	Fri 11/5/10	6
12	Delivery of parts for B	3 wks	Mon 10/18/10	Fri 11/5/10	7
13	Software delivery	3 wks	Mon 10/25/10	Fri 11/12/10	10
14	Assembly of A	1 wk	Mon 11/8/10	Fri 11/12/10	8,12
15	Assembly of B	5 wks	Mon 11/8/10	Fri 12/10/10	4,11
16	Test A	2 wks	Mon 11/15/10	Fri 11/26/10	14
17	Test B	3 wks	Mon 12/13/10	Fri 12/31/10	15
18	Final installation	8 wks	Mon 1/3/11	Fri 2/25/11	9,16,17
19	Final system test	6 wks	Mon 2/28/11	Fri 4/8/11	13,18

Figure 6.9 LOGON project schedule adjusted for holidays and weekends.

For starters, the Gantt or calendar schedule must account for *non-working time* such as weekends, holidays, and vacations. Figure 6.9 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:

- *Resource constraints*: a work package is delayed because the necessary resources are unavailable or must be shared with other, parallel 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, developing technologies, other design activities.
- *Logistics*: the acquisition of a bulky item for construction is delayed until space becomes available at the construction site.

Computer software can readily generate the project network, Gantt chart, and calendar schedule but unless issues like those listed above are accounted for, the project 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 determined the early and late dates, and issues and constraints surrounding the project have been addressed.

6.4 Management Schedule Reserve

In common practice, the contractual or committed target completion time T_s 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 *management schedule 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 no more than an estimate (a best guess). All 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 is added to the estimated duration. This reserve constitutes a "safety buffer" or "time buffer" to accommodate any project delays. Time buffers are discussed in Chapter 7.

See Chapter 7

6.5 Alternative Relationships[2]

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 6.10 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" should be able to start after *some* of the activity "move-in furniture" has been done; i.e., "move-in employees" can 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-tofinish (SF). It also allows for lags between the times when activities must be started or finished. These relationships are described next.

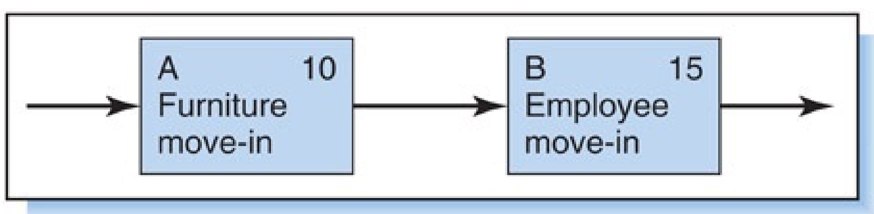


Figure 6.10 Example of FS relationship.

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 6.11. The *n* days delay is called *lag.* In the case of acceleration instead of a delay, *lead* is used instead of lag (lead is the mathematical negative of lag).

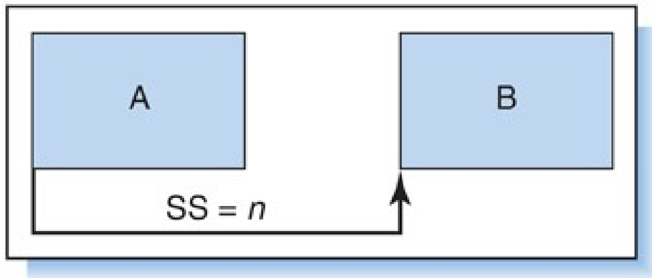


Figure 6.11 PDM representation of SS relationship with *n* day lag.

Using the example from Figure 6.10, 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 6.12.

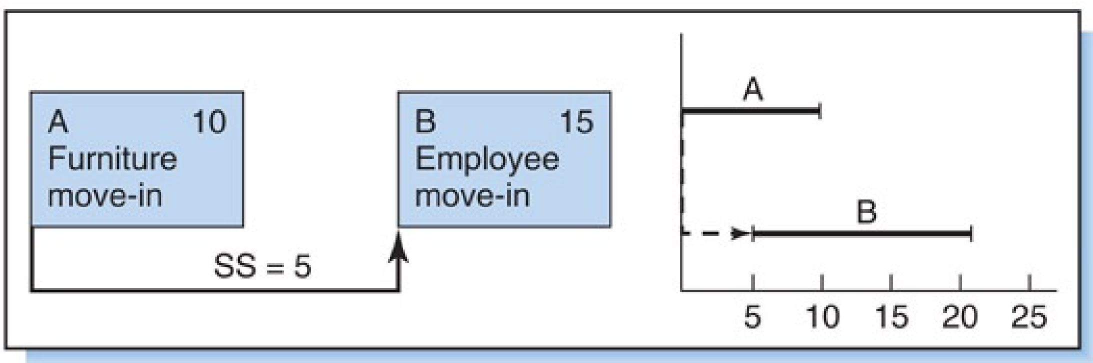


Figure 6.12 Example of SS relationship.

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 6.13 where the finish of "paint parking lines" (B) must occur within 5 days of the finish of "lay asphalt"(A). Two or more activities that must finish at the same time is an FF relationship with zero lag.

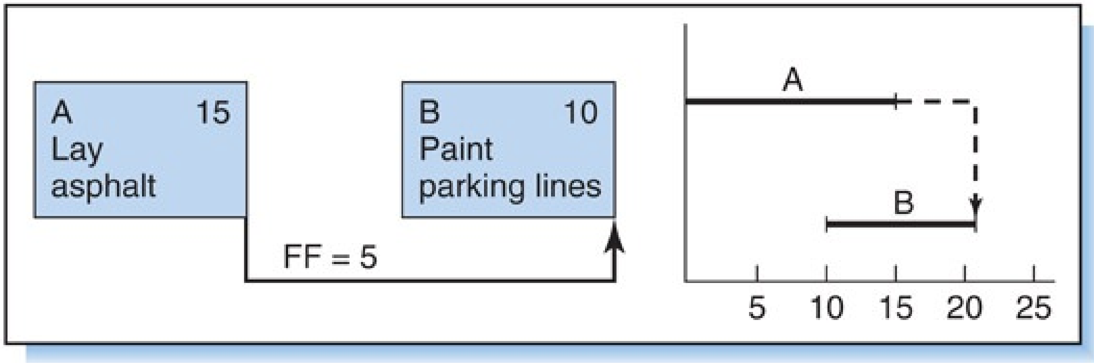


Figure 6.13 Example of FF relationship.

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 6.14.

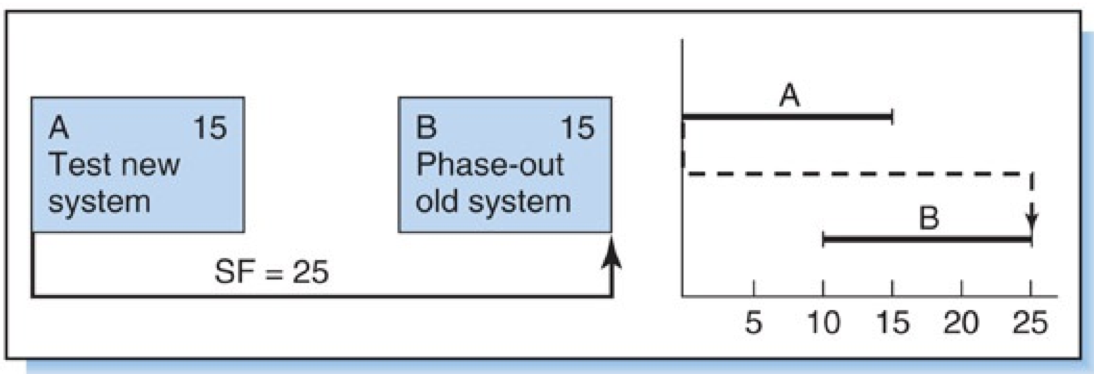


Figure 6.14 Example of SF relationship.

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 6.15. Note that when $n = 0$, the FS relationship becomes the same as traditional AON network method wherein the start of a successor coincides with the completion of its latest predecessor.

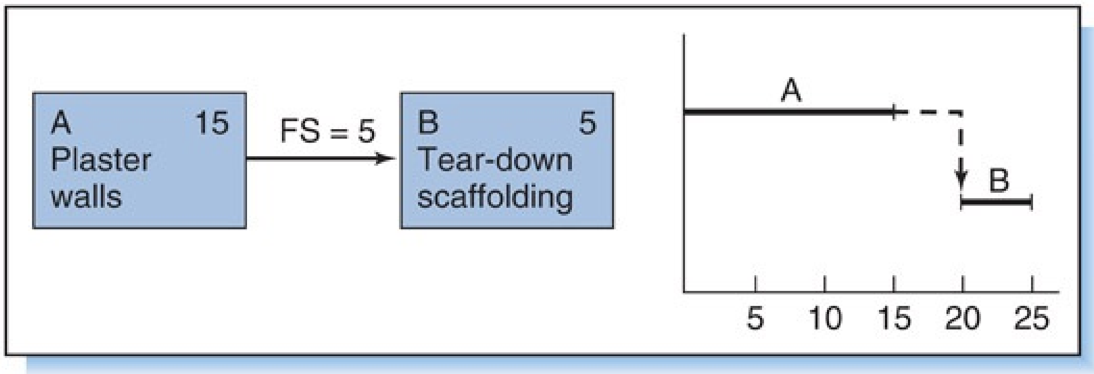


Figure 6.15 Example of FS relationship.

Multiple PDM Relationships

Two PDM relationships can be used in combination. Having both SS and FF is a rather common case. Notice in the example shown in Figure 6.16 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., the work in B can be stopped and then resumed). 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 6.17. 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.

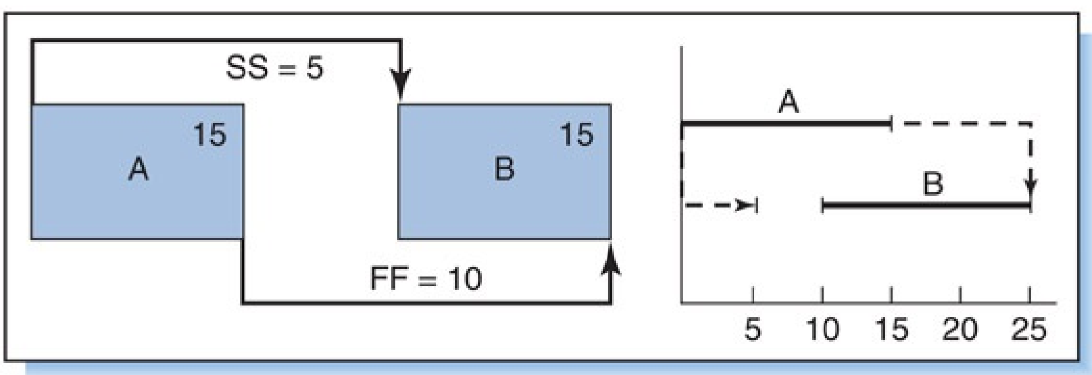


Figure 6.16 Schedule for non-interruptible activity B.

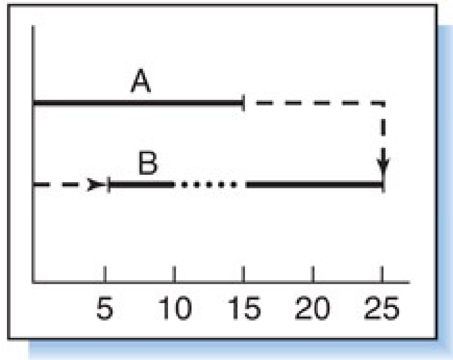


Figure 6.17 Schedule for interruptible activity B.

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.

Example 6.1: PDM in ROSEBUD Project

Figure 6.18 shows the AON diagram for the ROSEBUD project and Figure 6.19 shows the corresponding “time-scaled network,” which is a form of Gantt chart explicitly showing dependencies among the activities.

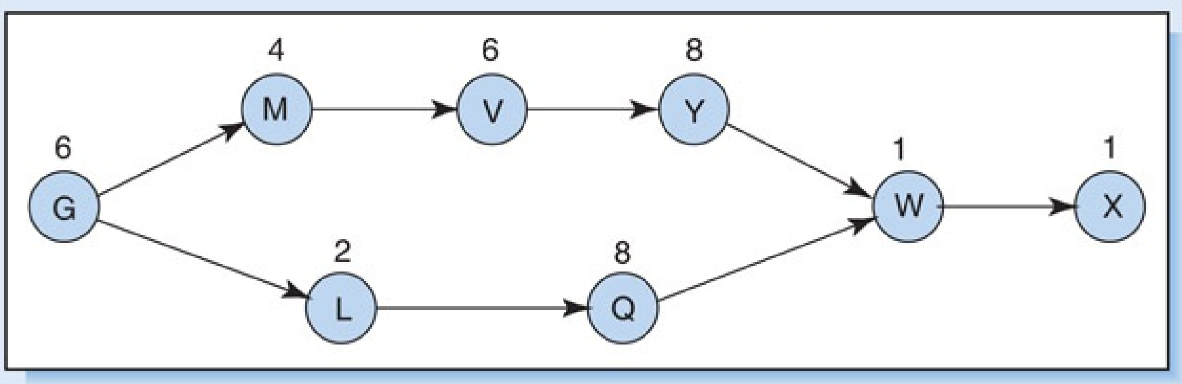


Figure 6.18 AON diagram for ROSEBUD project.

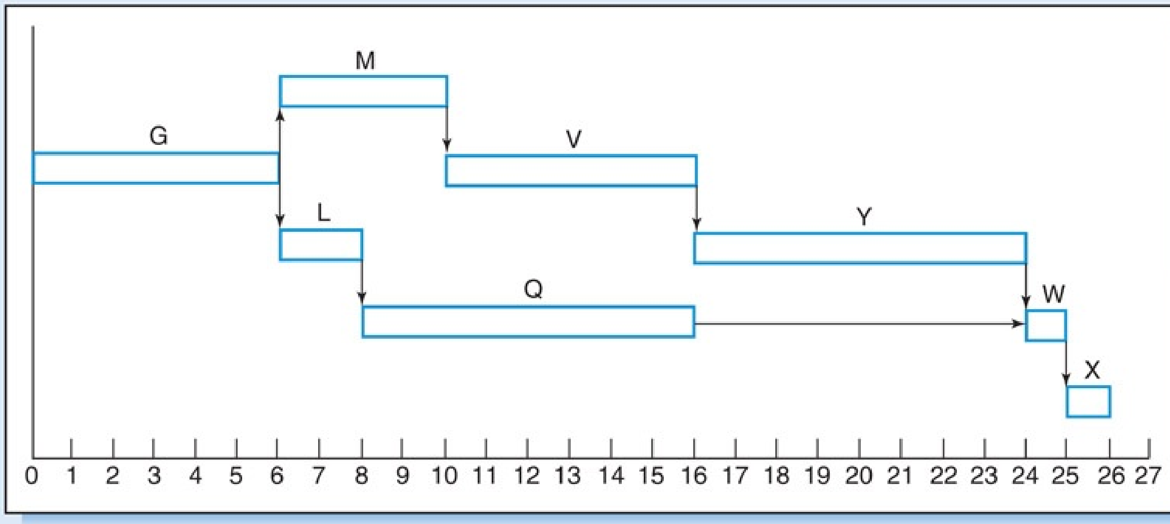


Figure 6.19 Time-scaled network for ROSEBUD project.

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 6.20 shows these relationships. Figure 6.21 shows the corresponding time-scaled network assuming earliest start dates and allowing for interruptible activities.

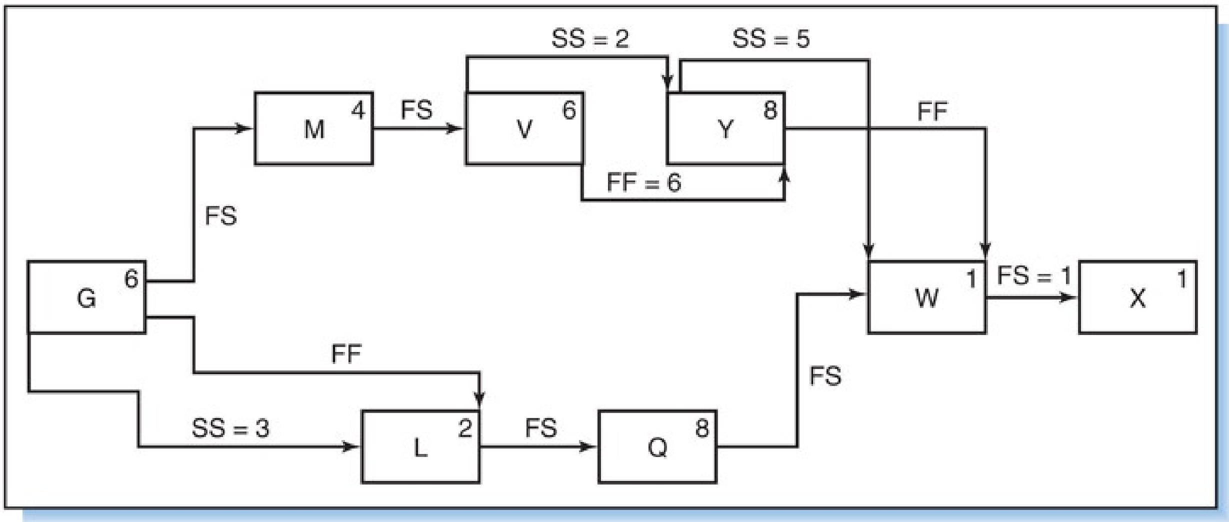


Figure 6.20 PDM network for ROSEBUD project.

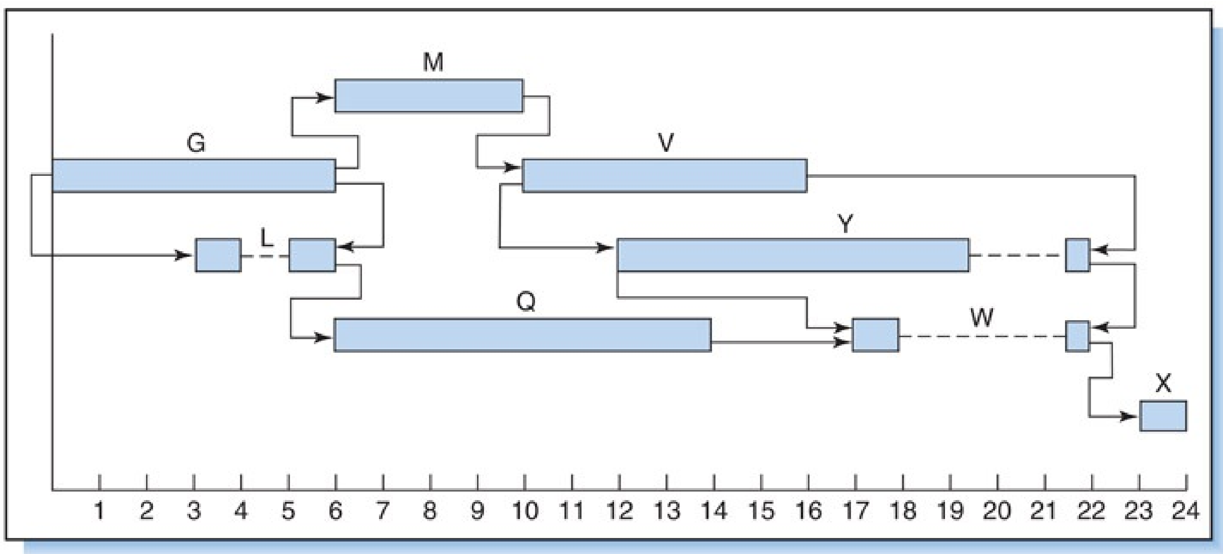


Figure 6.21 Time-scaled network for ROSEBUD project revised for PDM.

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 *move back* the start date of Y. Because of the precedence requirement, moving back the start date of Y results in moving back the start date of W and, as a result, the start date of X. In other words, shortening critical Activity Y actually causes an *increase in the project duration*.

In general, interpreting a PDM network requires more care than ordinary AON networks. However, such care is relatively inconsequential when the PDM network is created with project management software.

6.6 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*. That's because 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 would always be available when needed. But of course resources are not always available, and when they are not *the schedule must be changed to whenever they 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

Many times 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 resources are not sufficient to satisfy the needs of all of them, some activities must be delayed. Figure 6.22 illustrates this: (a) shows the network; (b) shows the project schedule, not accounting for the resources. Suppose Activities B and C both require the same resource, but the resource can be used by only one of them at a time. In that case, the schedule must be revised; (c) shows two alternatives.

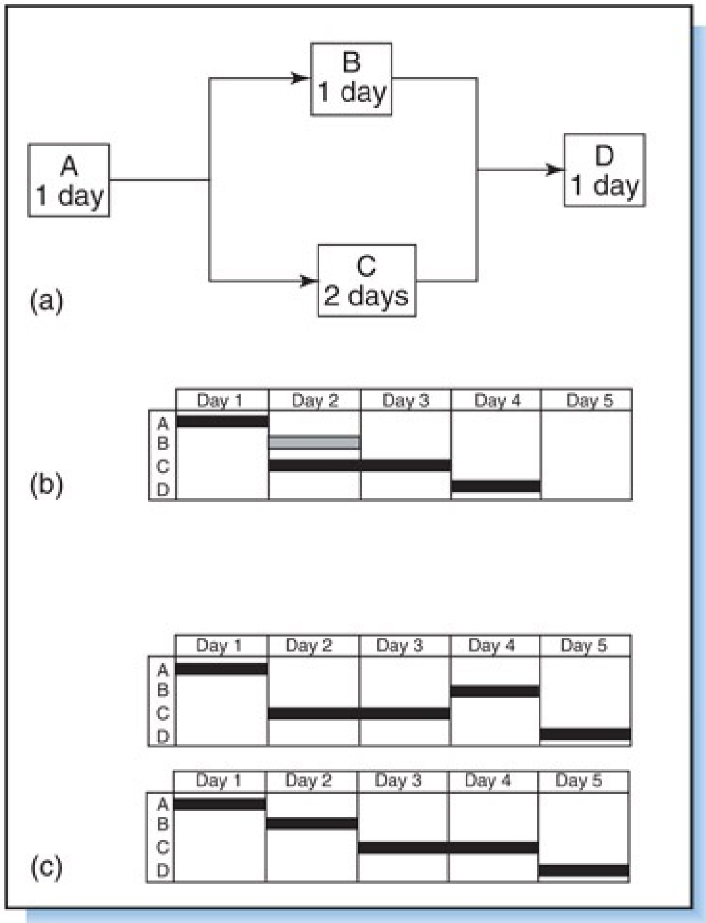


Figure 6.22 The effect of a constrained resource on schedule.

In general, projects tend to be either *resource-constrained* or *time-constrained*. In a resource-constrained project, the resources are limited in some way and the project completion date is determined by the availability of those resources. Figure 6.22 is an example. In a time-constrained project, the projectcompletion date is fixed and sufficient resources must be found to meet that date. A project that is both resource-constrained and time-constrained might not be able to meet the target completion date.

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 with specific skills), the workload is specified in terms of number of workers. Since 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 that not everyone is equal.

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, 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; loading, in contrast, is from the perspective of the *project*. It 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 amount 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 6.22 was such a case: Activities B and C require the same resource, but the resource cannot be used in both at the same time. Resources available in sufficient quantity do not pose an issue and can be ignored for scheduling purposes (air is an example—unless 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 leveling* in a *time-constrained project*. In this case, there is enough of the resource to complete the project on time; however, the quantity of the resource needed fluctuates throughout the project, making it difficult to manage the resource. The objective of resource leveling is to level the amount of the resource needed throughout the project. The second case is the situation mentioned before, the *resource-constrained project*—not having enough of a resource to do multiple activities at the same time.

Leveling 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 for a particular resource 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 gradual 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 are responsible for a stable, uniform 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 leveling: to alter the schedule of individual project activities such that the resultant workload for a required resource is somewhat uniform throughout the project.

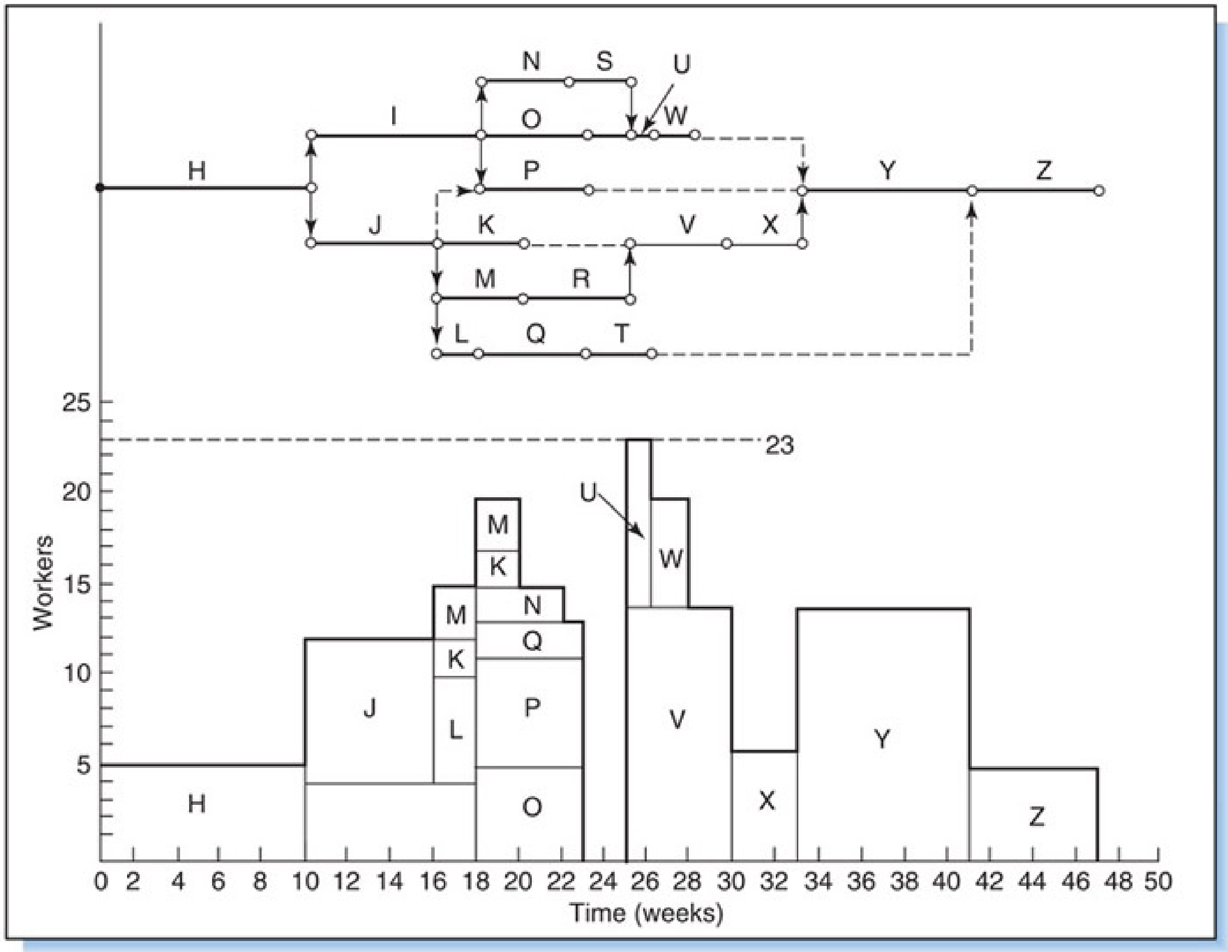


Figure 6.23 Schedule and corresponding worker loading for the LOGON project.

Figure 6.23 shows the loading of a resource for the LOGON project—the resource being a particular skill or trade (programmer, steel worker, etc.). The loading, bottom of Figure 6.23, is created from the schedule, top of Figure 6.23, and the weekly labor requirements in Table 6.5; it shows week by week the 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 stays at five workers (the weekly requirement for H). Over the next 6 weeks, activities I and J are scheduled, so the loading becomes 4 + 8 = 12, and so on.

Table 6.5 LOGON Project Weekly Labor Requirements

Activity	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Duration (weeks)	10	8	6	4	2	4	4	5	5	5	5	3	3	1	5	2	3	8	6
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

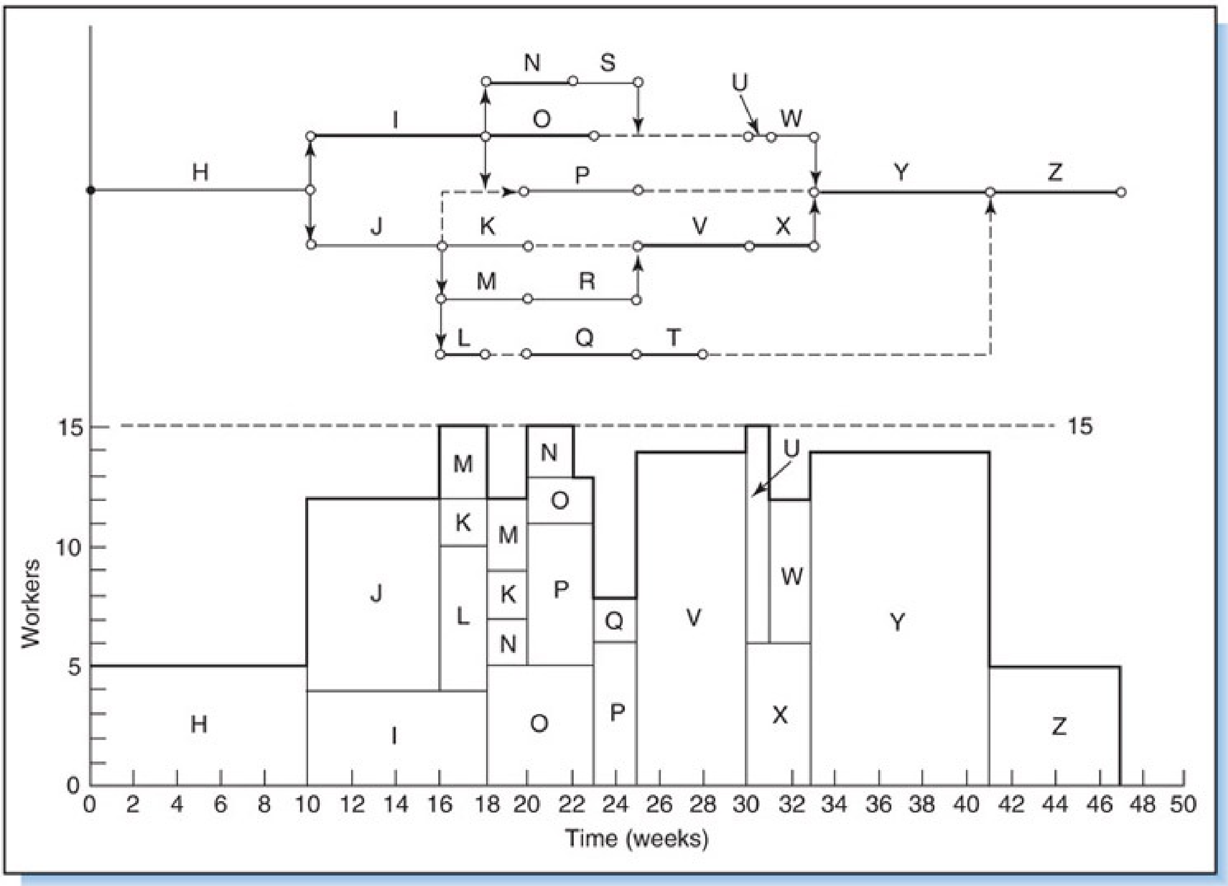


Figure 6.24 Smoothed worker loading for the LOGON project.

From Figure 6.23, you can see that the loading for the LOGON project might pose a problem because it fluctuates so much, varying from a maximum of 23 workers in Week 26 to zero workers in Weeks 24 and 25 (since 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 after their early times so as to reduce workload peaks and fill in workload valleys. For example, the somewhat smoothed workload in Figure 6.24 is achieved by delaying activities P (and hence Q) by 2 weeks and U (and hence W) by 5 weeks.

Although resource leveling is often necessary to reduce workload fluctuations, it potentially increases the risk of project delays because it reduces slack time. Less slack time means greater risk that an activity will not be completed by its scheduled finish date. In Figure 6.24 delaying activities U and W makes them critical (no slack remaining), so any delay in either will delay the project.

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 earlier, 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 on.

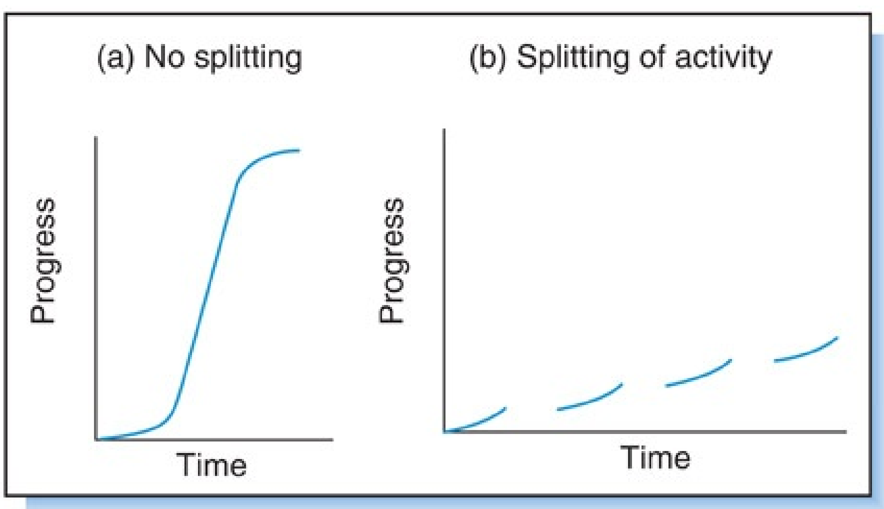


Figure 6.25 The effect of splitting of an activity upon duration.

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 6.25 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 gathers momentum. The sum of the durations of the pieces in (b) exceeds the duration in (a). The effect, known as *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.

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 6.26 where the design and build activities each proceed uninterrupted, although 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 the project duration and facilitates interaction between designers and builders.

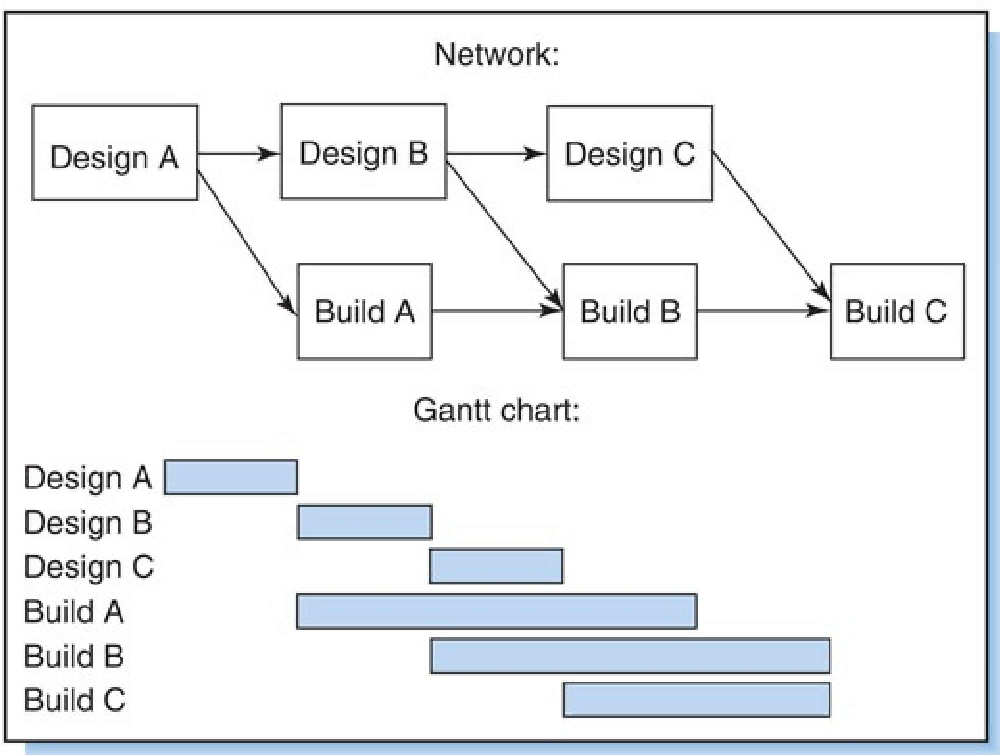


Figure 6.26 Multiple hand-over points of an uninterrupted activity.

Leveling Multiple Resources

Leveling is easy for a single resource but can be difficult for several simultaneous resources. Because work packages usually require resources from more than one functional unit or subcontractor, a schedule that provides a level loading for one unit may cause overloading or difficult-to-manage situations for others. For example, based on the weekly equipment requirements for LOGON shown in Table 6.5, the schedule that provides the somewhat-level *worker* loading in Figure 6.24 yields the erratic *equipment* loading shown in Figure 6.27. 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 6.23 that produces the erratic loading for workers yields a relatively level loading for equipment.

It is impossible to completely level 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, focus on leveling the "priority" resources—those where irregular loadings are the most costly to the organization or demoralizing to workers. The financial and social costs associated with hiring, overtime, and layoffs often dictate that "human resources"—the workers—be given the highest priority. Many project software packages perform scheduling analysis that permit simultaneous leveling of multiple resources.

Delaying activities is one method to level resources; others are to:

- Eliminate some work segments or activities (reduce project scope).
- Substitute resources.
- Substitute activities with less resource-consuming activities.

For example, when the most qualified workers are not available, either eliminate the work that requires their expertise or use less qualified workers. These options, however, might compromise the scope or quality of the work and increase the risk of the project not meeting requirements.

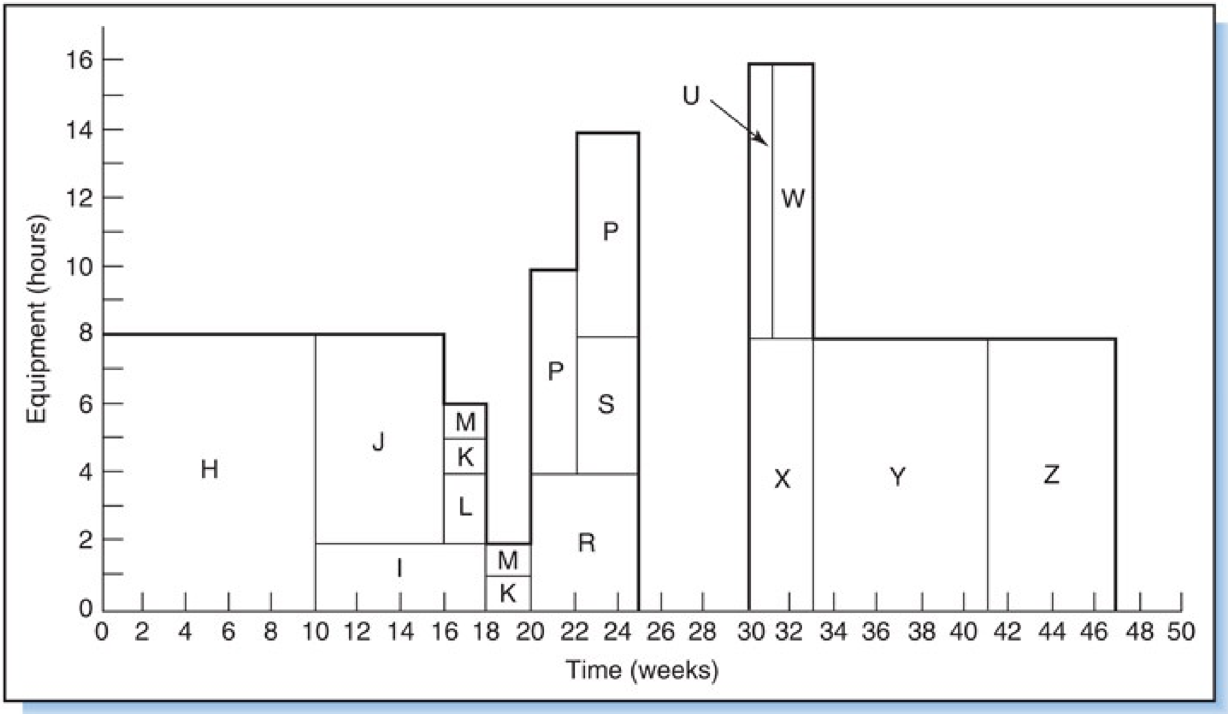


Figure 6.27 Equipment loading for the LOGON project.

Leveling a Resource-Constrained Project

What happens when the number of personnel, pieces of equipment, or available working capital restricts the schedule? 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 leveling 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 at the same time must be checked against the maximum. The problem is more than just leveling of resources; it involves rescheduling jobs or delaying them until the resources become available.

In the LOGON project, for example, suppose only 14 workers are available in any given week. The "leveled" schedule in Figure 6.24 results in a maximum loading of 15 workers. In this case it is not possible to reduce the maximum loading to less than 15 and still complete the project in 47 weeks. 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. With a problem like this something has to give, since it is not feasible to both satisfy the resource restriction *and* the project deadline of 47 weeks. Figure 6.28 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 complete because Activity X had to be delayed 3 weeks beyond its late start date.

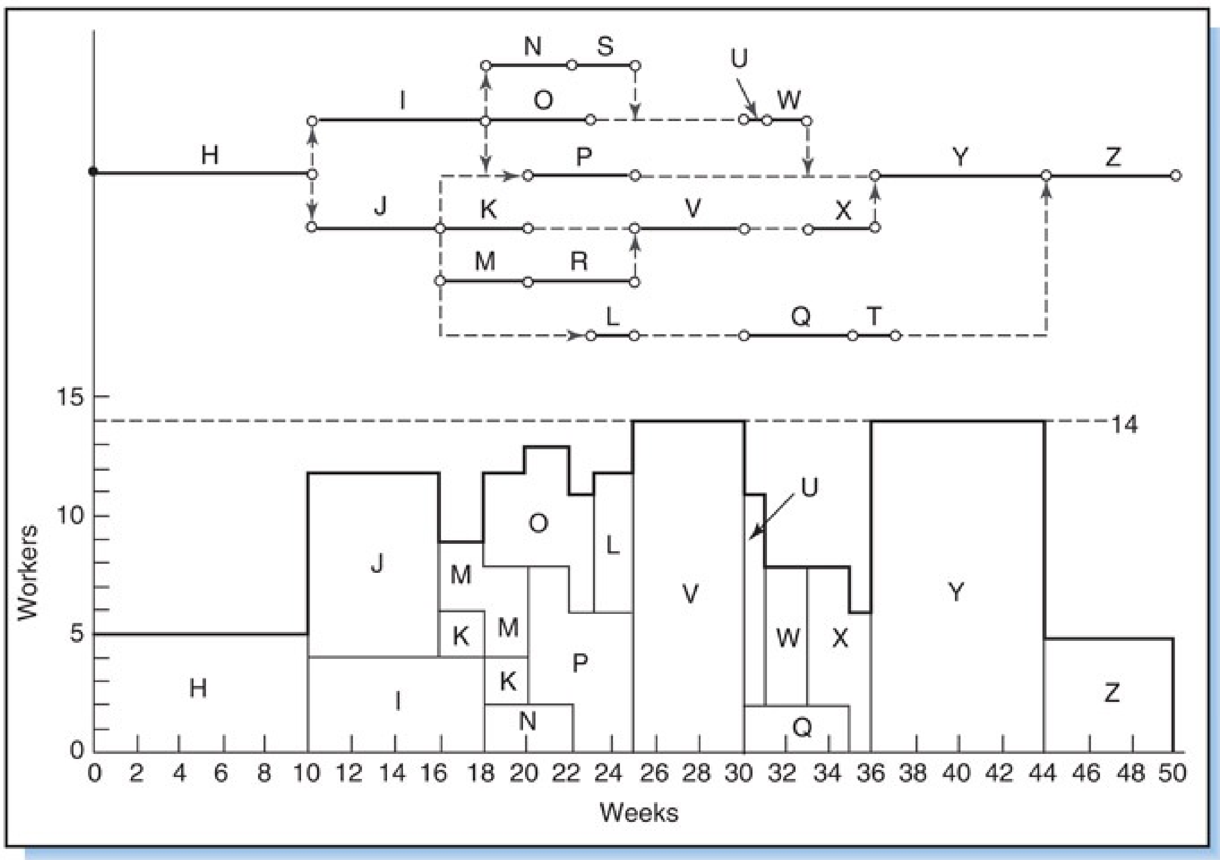


Figure 6.28 Schedule and corresponding worker loading for the LOGON project with 14-worker constraint.

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 important project resource is a technical inspector who has the skills to inspect and approve a variety of activities. Her work is exacting, however, which prevents her from working on more than one activity at a time. Suppose the activities in which she will be working are H, J, P, K, L, V, and X. These activities are highlighted in Figure 6.29. Because she can work on them only one at a time, the activities must be scheduled sequentially. Summing 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.

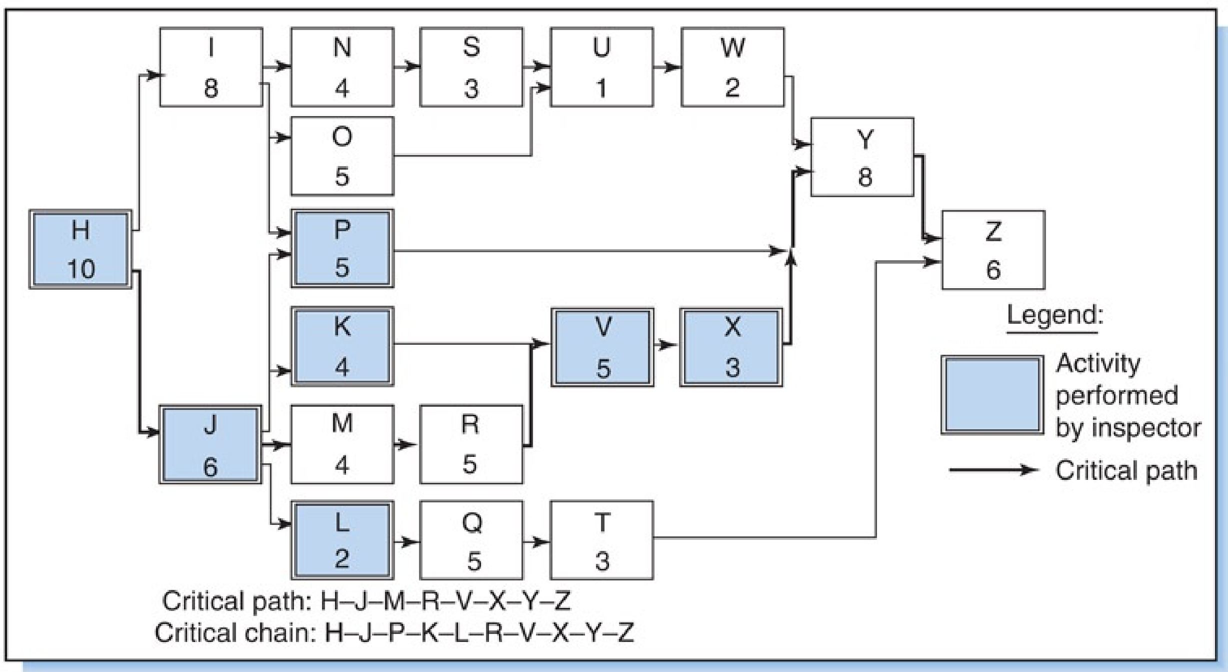


Figure 6.29 Activities in the LOGON project involving the resource of technical inspector.

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 6.22, 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 sets the project duration. This is further discussed in Chapter 7.

See Chapter 7

Scheduling with constrained resources involves decisions about which activities can be scheduled immediately and receive resources, and which should be delayed until resources are available. Project management software uses procedures based on simple rules (called *heuristics*) for scheduling with constrained resources; some of these are discussed in the next chapter.

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.

6.7 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 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. But this is a problem with scope planning and activity definition, which plagues every scheduling method, not just networks.

A related problem is that activities and durations in a schedule sometimes require periodic modification; this happens when there are too many activities in the network and not all are well defined. This problem can be addressed by initially creating a "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 5.

See Chapter 4 and 5

Another criticism relates to the fact that because it is sometimes difficult to demarcate one activity from the next, the boundary separating activities more or less arbitrary. This means that successors can sometimes be started before predecessors are finished, and the two "overlap" in the sequence. But again, this is not really a problem since PDM allows for overlap of activities, and hand-over points to treat activities as if they did overlap.

In summary, the shortcomings of networks are actually shortcomings in project definition. It can be argued (and innumerable project managers will attest) that network methods, though not perfect, offer a good approach for analyzing and creating project schedules.

6.8 Summary

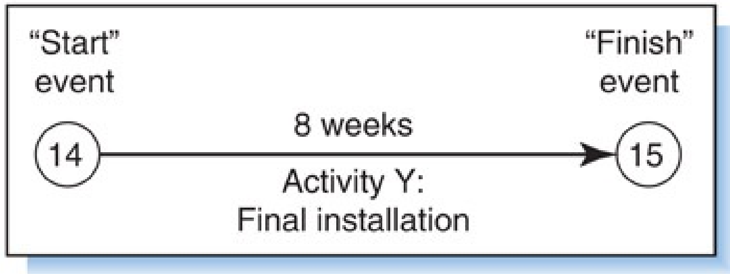
The advantage of networks is that they clearly display the interdependencies of project activities and show the scheduling impact that activities have on each other. This feature enables planners to determine critical activities and slack times, which is important for project planning and control. Knowledge 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 a variety of relationships between project activities to better reflect the realities of project work.

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 Symbols

T_e	Expected Project Duration: the expected length of the project.
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 of the 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.
FF = n	Finish-to-Finish: an activity can finish no later than n days after its immediate predecessor has finished.

Summary Illustration Problem



Activity	Time	ES	EF	LS	LF	Slack Total	Slack 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

The 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 6.30 shows the AOA representation for one activity and its events.

The AOA method represents each activity as a directed line segment (an *arrow*) between two nodes (or circles). As shown in Figure 6.30, 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 merely identifies the event. Each event must have its own unique number. In the example, the numbers 14 and 15 were chosen arbitrarily. Node 14 means "start Activity Y" and node 15 means "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 must point generally toward the right end of the network; the arrows cannot double back.[4]

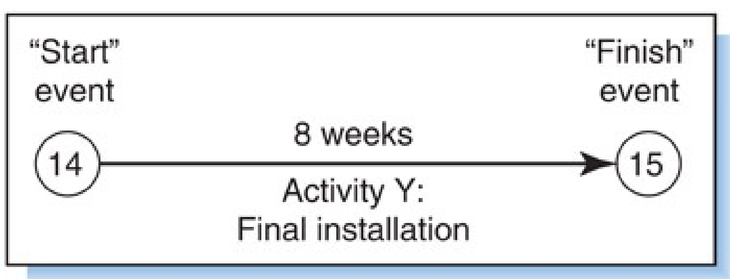


Figure 6.30 AOA representation for an activity and its start and finish events.

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 that it can be started only after *all* of its immediate predecessors have been 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. The following example demonstrates the need for dummy activities in an AOA network.[5]

An engineer decides to write a new computer program and to buy a new computer to run it on. The activities and their dependencies are illustrated in the AON network in Figure 6.31. The dependencies are:

1. Pay for the computer after buying the computer.
2. Install the computer program after writing the program *and* after buying the computer.

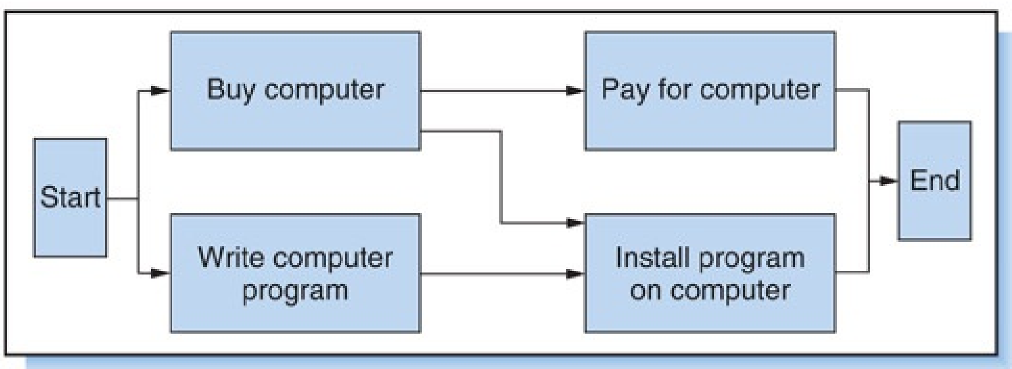


Figure 6.31 AON diagram.

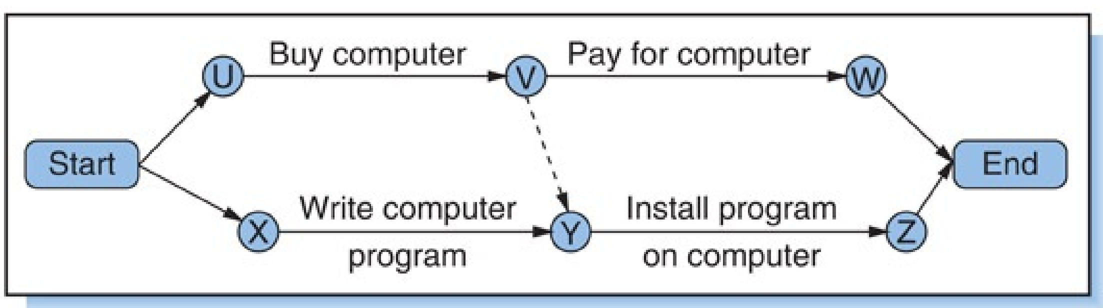


Figure 6.32Figure 6.31 converted to AOA diagram.

The AOA network for this project is shown in Figure 6.32. 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 "install program" to its two immediate predecessors, "buy computer" and "write computer program." Notice that the overall network has only one "Start" node and one "End" node.

AON versus AOA

Since AON networks do not require the use of dummies, they are easier to construct and interpret than AOA networks; as a consequence, 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, and some create both AOA and AON network diagrams. For a particular project, only one network method should be 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.

Some managers, however, argue that for practical purposes the method is incorrect. Project managers 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 successor activity *starts on the period following* the completion of its predecessors, not at the same time. Thus, the network would show the early start time of an activity as being the day (or week) later after the early finish day (or week) of its latest predecessor. Realistically this approach makes sense.

As an example, refer to Figure 6.33, which is Figure 6.8 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 + Duration − 1. This, EF = 1 + 10 −1 = 10. The ES for Activity H's successors, Activity I and Activity J, is on the *next* day, i.e., ES = 11.

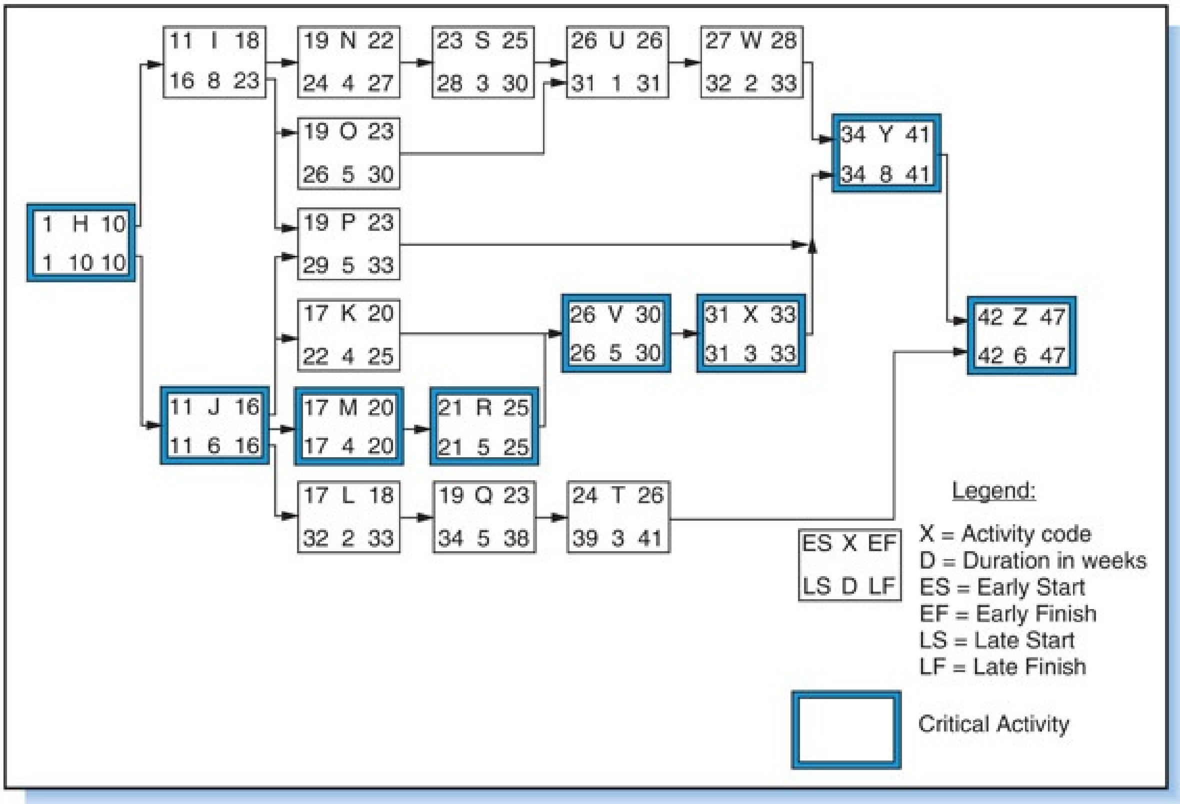


Figure 6.33 Figure 6.8 adjusted for "Day 1" assumption.

Using the assumptions of course affects the late times too. Making the backward pass through the network, LS = LF − Duration + 1. Thus, for Activity I with duration 8, if LF = 18 then LS = 18 − 8 + 1 = 11. The immediate predecessor of Activity I, Activity H, must finish the day before this, so LF = 10 for Activity H.

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:

Free slack for an activity = ES (earliest successor) − EF (activity) − 1

Project management scheduling software uses actual dates, not elapsed times, and the project start 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 be shown as the next period (day or week) after the finish dates of their successors.

Review Questions and Problems

1. What are the advantages of networks over Gantt charts?
2. Draw a network diagram of your college studies, 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 a letter to an old friend.
    b. Preparing a five-course meal (you specify the 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 another 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:

    1. Using your experience or imagination, create a WBS.

2. List the activities or work packages.
3. Show the immediate predecessors for each activity.
4. Draw the network diagram (using the AON scheme).

9. Draw the AON network diagrams for the following four projects:

1.	
Activity	Immediate Predecessor
A	–
B	A
C	A
D	B
E	D
F	D
G	D
H	E, F, G

2.	
Activity	Immediate Predecessor
A	–
B	A
C	A
D	B
E	B
F	C
G	D
H	D
I	G
J	E, F, H, I

3.	
Activity	Immediate Predecessor
A	–

B	A
C	–
D	–
E	D
F	B, C, E

4.	
Activity	**Immediate Predecessor**
A	–
B	–
C	–
D	C
E	A
F	B
G	E
H	F, G,J
I	A
J	D, I

10. Refer to Figure 6.1 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 to put on 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."

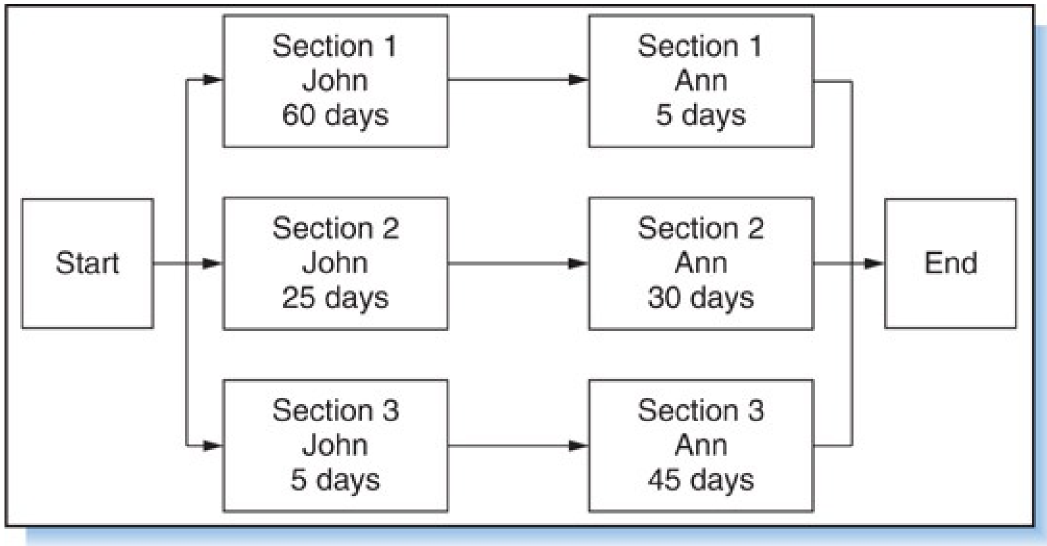
11. Eliminate redundant predecessors from the following lists so only immediate predecessors remain.

a.	Activity	Predecessor
	A	–
	B	–
	C	–
	D	B
	E	C
	F	A
	G	B, D, C, E
	H	A, B, C, D, E, F, G

b.	Activity	Predecessor
	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	Predecessor
	A	–
	B	–
	C	A
	D	A
	E	B
	F	B
	G	A, C
	H	A, B, D, E
	1	B, F
	J	C, D, E, F, G, H, I

12. Use Figure 6.5 (a) and (b) to draw Gantt charts for the ROSEBUD project.
13. Some projects have a fixed due date while others have to be finished as early as possible and the project manager only makes commitments on the completion date once she and her project management team have scheduled the project. Explain how the backward pass differs for these two project types.
14. Explain how it is possible that there can be slack on the critical path. 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, the risk of changes in the design, management focus, cash flow, and any other factor that 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 6.20, calculate ES, EF, LS, and LF for all activities.
19. To produce a manual, John has to write the text, after which Ann has to draw sketches and typeset the document. John can start with any section of the book (i.e., he does not have to start with Section 1). The work has to be done 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]

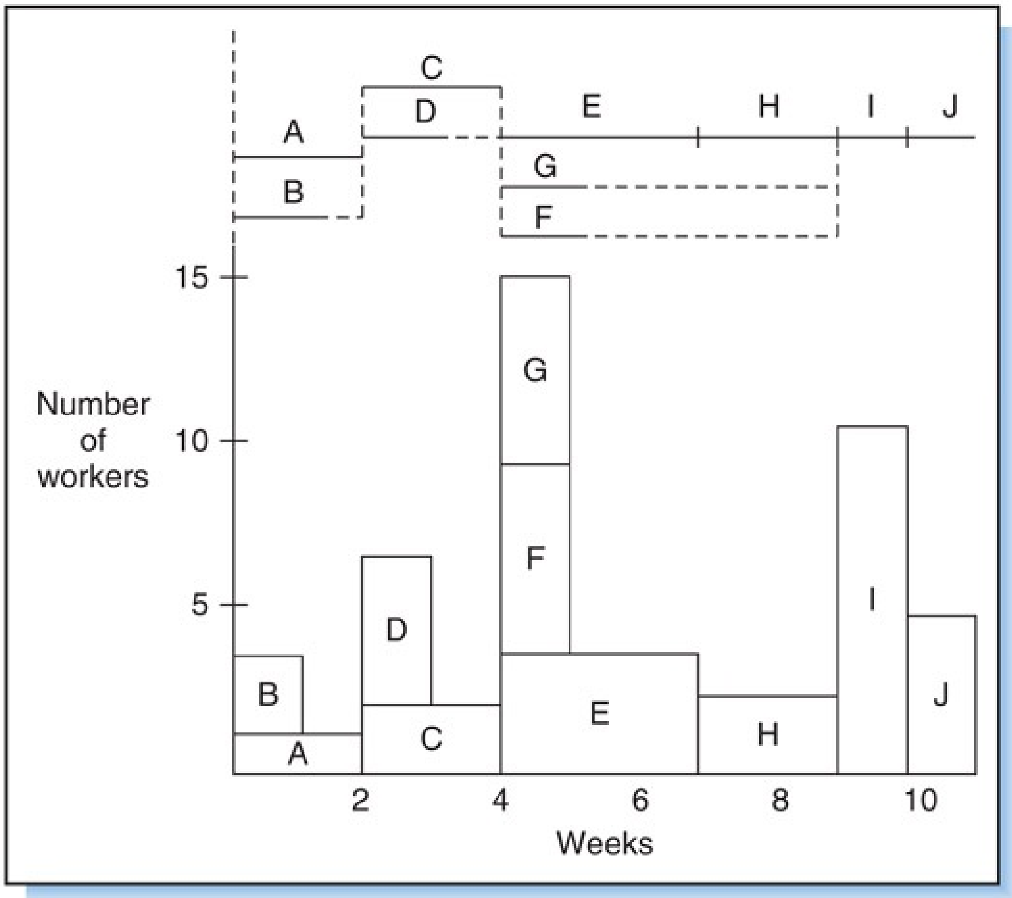
20. Why is leveling of resources preferred to large fluctuation of workload? What negative result could resource leveling cause?
21. Describe how resource leveling of a resource-constrained project differs from resource leveling in a time-constrained project.
22. The requirements for systems analysts 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. 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 6.23 (that produced an erratic loading for workers) yields a more balanced loading for equipment than the one shown in Figure 6.27.
26. Suppose in Figure 6.20 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. For each of the following predecessor tables:

   - Draw a corresponding AON network.

- Compute ES and EF for each activity.
- Compute LS and LF for each activity. Find the critical path.
- Determine the total slack and free slack.

a.	Activity	Predecessor	Duration
	A		6
	B		3
	C	A	9
	D	B	5
	E	B	4
	F	D	2
	G	E	8

b.	Activity	Predecessor	Duration
	A		3
	B	A	8
	C	B	9
	D	C	3
	E	B	2
	F	E, H	4
	G	A	6
	H	G	5
	J	D, F	1

c.	Activity	Predecessor	Duration
	A		9
	B	A	2
	C		8
	D	C	8
	E	B, D	7
	F	E	4
	G	C	4
	H	B, D, G	3

J		6
K	J	10
M	G, K	3
N	H, M	6

d.	Activity	Predecessor	Duration
	A		10
	B	A, E	9
	C	B, N	15
	D	C	7
	E		5
	F	A, E	6
	G	K, F	7
	H	G	12
	J		12
	K	E, J	4
	L	K, F	11
	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 system was used to create and maintain them? Who was responsible for system inputs and system operations? Describe the capabilities of the software system.
2. At what point in the project were networks created? When were they updated?
3. Was scheduling software used?
4. What was used first to develop the schedule: (a) a table such as Table 6.3, (b) a network diagram, or was (c) the Gantt chart drawn first? Comment on the method used.
5. Was all detail planning done upfront or was a phased approach followed?
6. How was the schedule reserve determined and included in the schedule?
7. Was the workload on resources made visible?
8. If the project was done within a matrix structure, how did communication between the functional and project managers take place?
9. Did the functional manger(s) take responsibility for workload on resources?
10. Was resource leveling done?
11. Were there any complaints about unrealistic workloads?

Case 6.1[9] Network Diagram for a Large Construction Project

The table below lists activities for constructing a bridge over an operational railway line, similar to the bridge described in Case 10.3.

Activity	Duration

No.	Activity Description	(Months)	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

(milestone)

1. Construct a network diagram for the project.
2. Do forward and backward pass calculations to indicate early and late start and finish times.
3. Indicate the critical path.
4. Indicate 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 6.2 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 6.34 shows part of the network and the number of days Bill estimated for each activity. What are the early and late start and finish times for all the activities? What is the earliest this portion of the project will be completed?

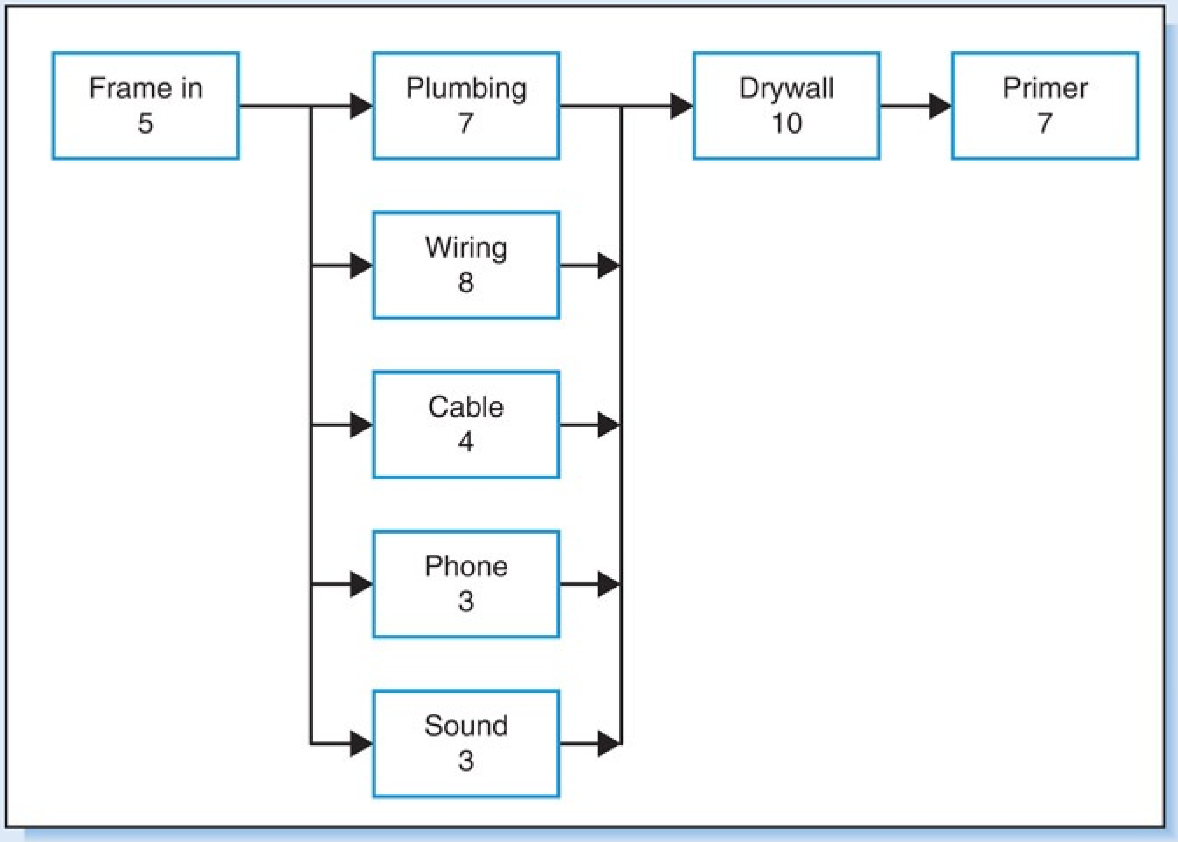


Figure 6.34 Partial network for hotel construction project.

Project schedulers are often faced with incorporating constraints that originate outside the project. 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?

Additionally, the owner of Melbourne Construction, Naomi Watts, wants to give the hotel owner a tour of the building, but not before 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 they can be? Bill is meeting with Naomi to discuss the situation.

Case 6.3 Melbourne Construction Company, B

One way to speed up a project is to speed up activities on the critical path (discussed in Chapter 7); another is to overlap activities. Refer to Case 6.2, Melbourne Construction Company, A: the network diagram in Figure 6.34 assumes finish-to-start relationships (or FS, meaning successors start only upon completion of predecessors) for the activities for one floor of the hotel. But each floor is large enough so that the crew for an activity can begin when the crew for its predecessor activities are only *partially* completed (this is called a start-to-start or SS relationship, meaning the start of an activity *lags* the start of its predecessors by some specified amount). In other words, activities in sequence can 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. It is also the case for Drywall installation, which is slower than its immediate predecessors, so Bill assigns an SS lag of 4 days between Drywall installation and its predecessors. This is shown in Figure 6.35.

See Chapter 7

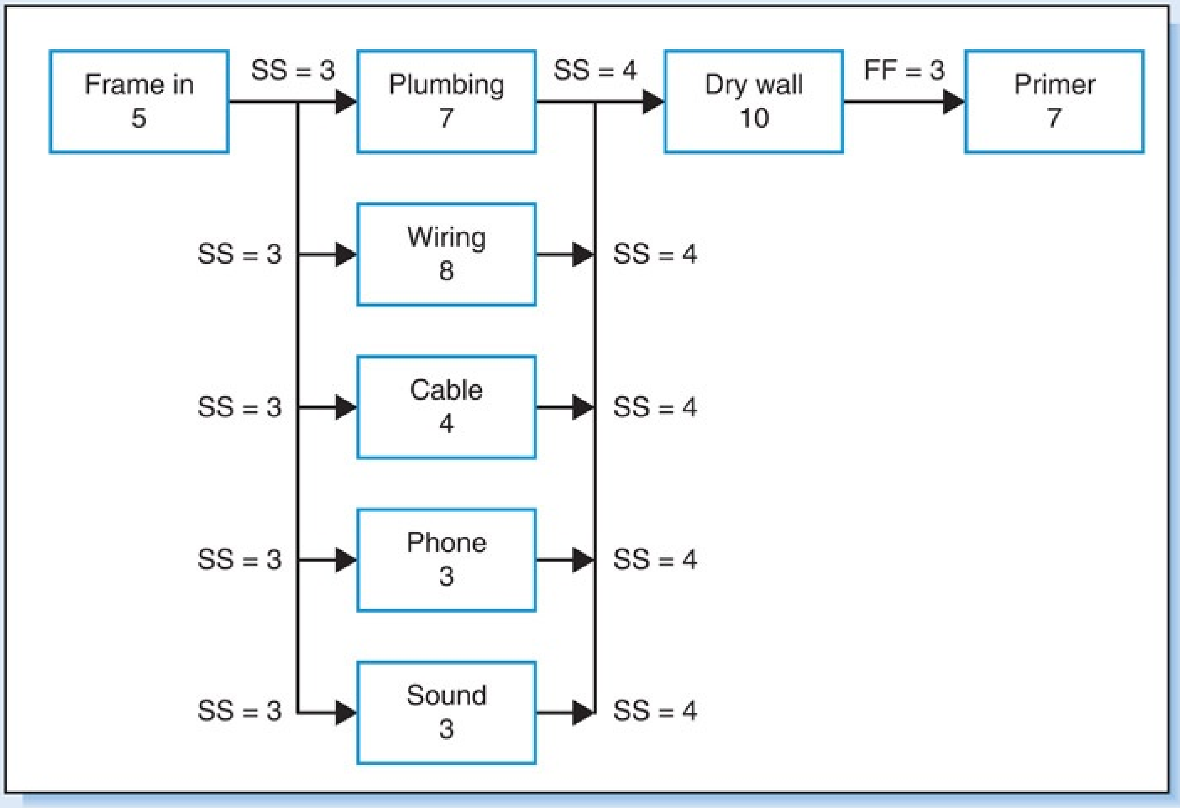


Figure 6.35 Network with lags inserted.

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 (i.e., Primer should finish no earlier than 3 days before Drywall finishes). This is also shown in Figure 6.35.

1. Compare the Gantt chart using the original FS relationship with the Gantt chart using the SS and FF lags. By how much do the lags speed up the project? Cable, Phone, and Sound take less time than Frame in; what potential problem might occur in overlapping them with Frame in?
2. Refer to Case 6.2, Melbourne Construction Company, A. 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 6.4 Melbourne Construction Company, C

Bill Asher, scheduler for Melbourne Construction Company, has created a network of the 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 6.36 shows the network of activities for the three floors and the number of days he 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 subcon-tractor moves to the next floor.

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 6.36 are based on Bill's estimates of total labor hours per activity and an 8-hour work day. 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 6.36 assume that subcontractors assign a "crew" of one worker to each activity. According to these time estimates, it should take 54 days to complete each floor. His boss, Naomi Watts, says that 54 days per floor is too long, and that to complete the project on time each floor should take no more than 35 days.

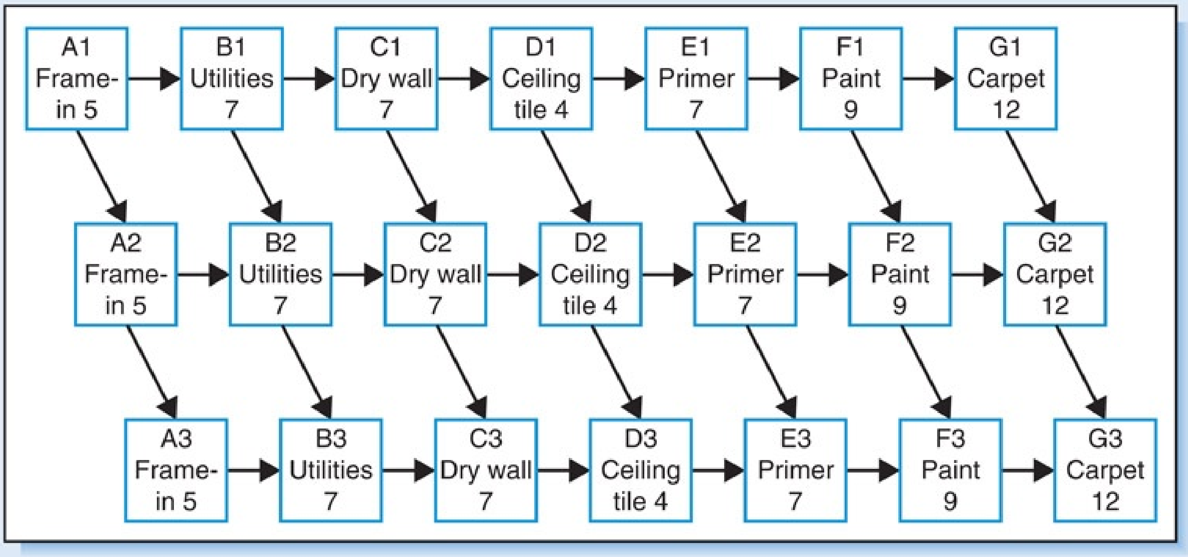


Figure 6.36 Network for three floors.

Looking at the seven activities per floor, Bill sees that the 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.

a. Bill's estimates assume one worker per activity. How many workers (what crew size) should the contractors assign to each activity such that each will take at most 5 days?
b. 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.

Endnotes

1. Duncan W.R. (ed.). *A Guide to the Project Management Body of Knowledge.* Newton Square, PA: Project Management Institute Standards Committee; 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.D. and Villoria R.L. *Network-based Management Systems (PERT/CPM).* New York, NY: John Wiley & Sons; 1967.

6. Steyn H. (ed.). *Project Management: A Multi-disciplinary Approach.* Pretoria: FPM Publishing; 2003. Reproduced with permission.

7. Ibid.

8. Ibid.

9. Ibid.

Chapter 7
Advanced Project Network Analysis and Scheduling

Look beneath the surface: never let a thing's intrinsic qualities or worth escape you

—Marcus Aurelius, Meditations

The scheduling methods discussed in Chapter 6 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 the methods for handling uncertainty in project completion dates, namely PERT and Critical Chain. It also covers methods for reducing the project duration, starting with CPM.

7.1 CPM and Time–Cost Tradeoff

The *critical path method* (CPM) is a way to reduce project duration by allocating resources among activities for the least cost. Developed in 1957 by DuPont Company, Remington Rand, and Mauchy Associates for DuPont plant construction, it is a mathematical procedure for estimating the tradeoff between project duration and project cost.[1]

Example 7.1: The House Built in Less Than 4 Hours[2]

With virtually unlimited resources and meticulous planning and control, a project can be done *very* fast. On March 13, 1999 the Manukau, New Zealand Chapter of Habitat for Humanity (a nonprofit organization dedicated to eliminating poverty housing) 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 7.1). 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, letter box, installed clothes line, 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.

Figure 7.1 The house built in less than 4 hours.

Photo courtesy of Habitat for Humanity.

What made the 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-fitting task was estimated to take 30 minutes, but took 1 hour; the 30-minute overrun was absorbed in the project buffer. *Finally*, the project made use of suitable technology, including prefabricated walls and components.

Time–Cost Relationship

CPM assumes that the time to perform a project activity varies depending on the amount of resources applied; as more resources (labor, equipment, etc.) 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 adding more labor or working overtime, but either way the cost goes up.

Ordinarily, work on any given activity in a project is performed at a *normal* (usual and customary) work pace. This is the "normal" point shown in Figure 7.2. Associated with this pace is the *normal time,* T_n, which is the time the activity will take under normal work conditions, and the *normal cost,* C_n, which is the cost of doing the activity in the normal time. (The normal pace is assumed to be the most efficient and thus *least costly* pace. Extending the time beyond the normal pace will not produce additional savings.)

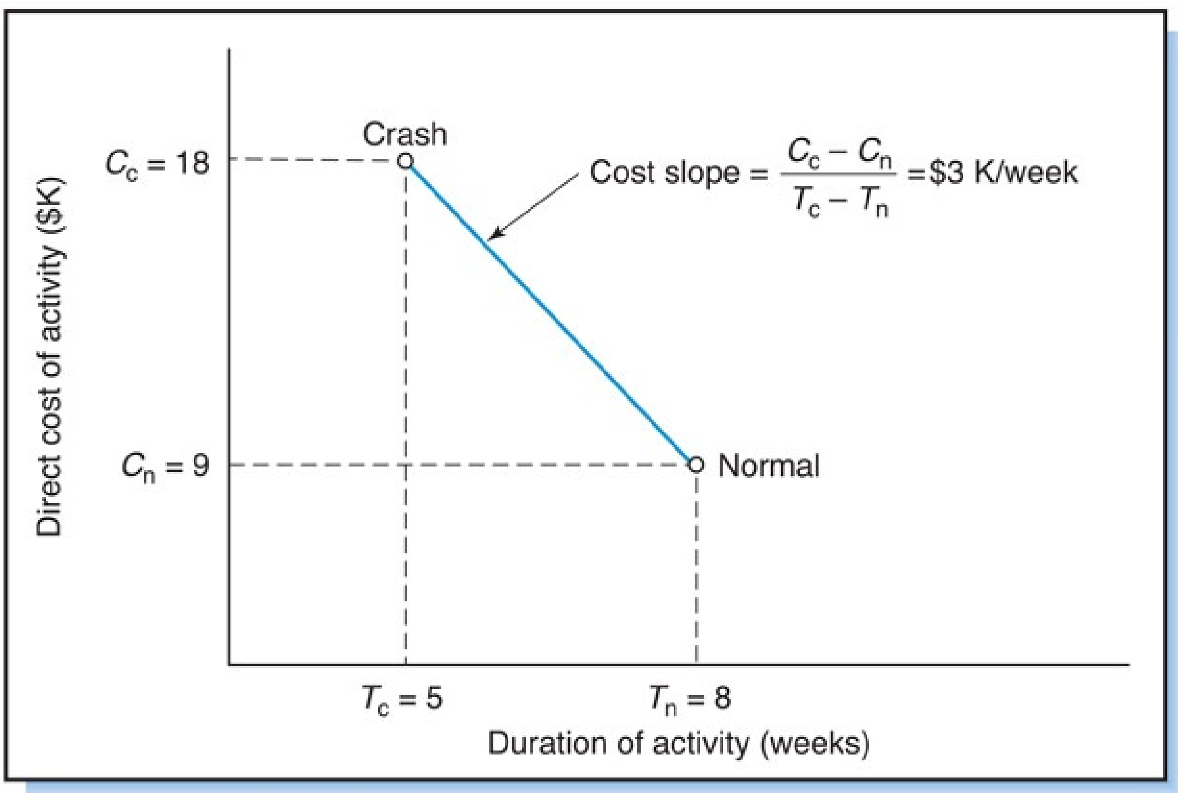


Figure 7.2 Time–cost relationship for an activity.

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 that the activity can be completed in the shortest possible time, the activity is said to be *crashed*. The crash condition (see Figure 7.2) represents not only the shortest duration, but the *most costly* as well. (For some activities, called *process limited*, there is no time–cost tradeoff since they require a specific time that remains constant regardless of resources. Fermenting wine or curing concrete are examples.)

As illustrated in Figure 7.2, the points for completing an activity under 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 line 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.

Using the formula, the cost slope for the activity in Figure 7.2 is $3K per week. Thus, 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 alter 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; completing it yet another week sooner (in 5 weeks) would increase the cost to $18K. According to Figure 7.2, this last step puts the activity at the crash point, the shortest duration for the activity.

Reducing Project Duration: Shorten the Critical Path

The cost-slope concept can be used to determine the least costly way to shorten a project. Figure 7.3 illustrates 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. Suppose we want to shorten the project duration. Recall from Chapter 6 that the *project duration is the length of the critical path*. In general, to shorten the project, the critical path must be shortened. 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*—either A, D, or G. Reducing an activity increases its cost, but because the reduction can be made *anywhere* on the critical path, the increase is minimized by selecting the activity with the smallest cost slope, which is Activity A. Reducing A by 1 week shortens the project duration to 21 weeks and adds $2K (the 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, an additional week can be cut from A to reduce the project duration to 20 weeks for a cost of $57K + $2K = $59K.

See Chapter 6

In general, each time an activity is shortened, it is necessary to check for any changes in the critical path. For example, as the top network in Figure 7.4 shows, shortening A uses up all of 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 because 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, as shown in Figure 7.4(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 as indicated in Figure 7.4(c). This adds \$5K per week, or 3 × \$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 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 7.1.

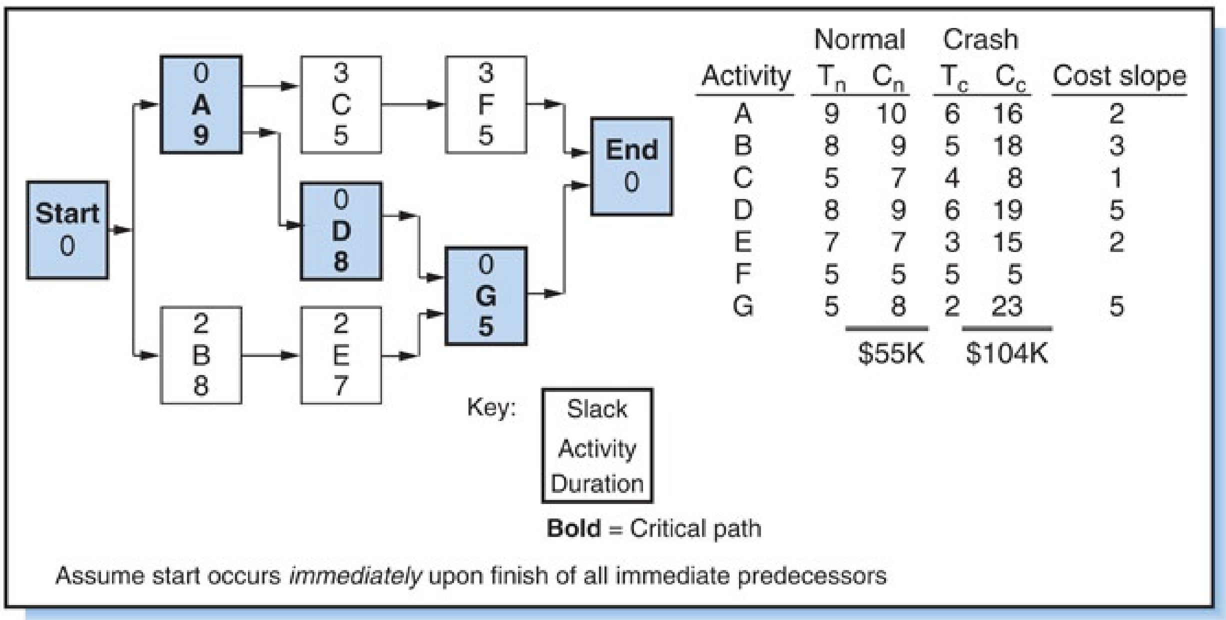


Figure 7.3 Time–cost tradeoff for example network.

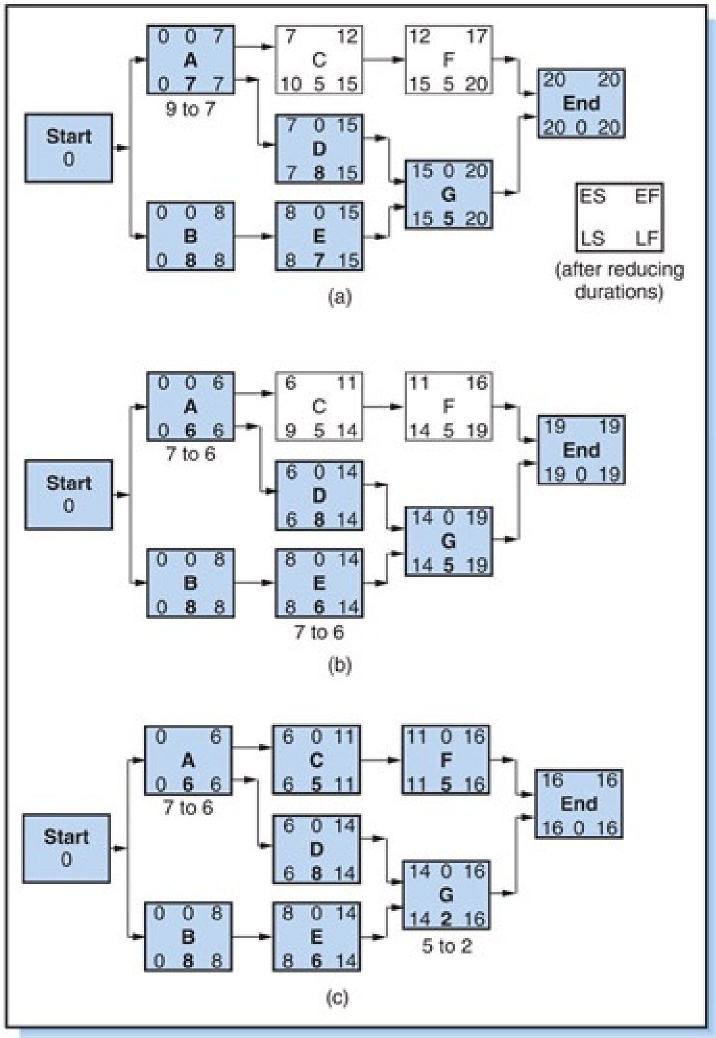


Figure 7.4 Reducing project duration.

Table 7.1 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 + $5 + $5 = $78
			$78 + $2 + $5 + $1 =

8	15	E ($2), D ($5), C ($1)	$86

* duration and cost using normal conditions

Crashing the Project: Shortest Project Duration

The time–cost procedure described determines which activities to speed up, step-by-step, so as 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 project duration* and avoid the intermediate steps, a simpler procedure is to simultaneously crash *all* activities at once. This, as Figure 7.5 shows, also yields the project duration of 15 weeks. However, the expense of crashing all activities, $104K (table in Figure 7.3) is artificially high because, as will be shown, not *all* activities need to be crashed to finish the project in the shortest time.

The project duration of 15 weeks is the time along 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 increase any noncritical activity by a certain amount without lengthening the project. In fact, the noncritical activities can be stretched until all the slack in the network is used up.

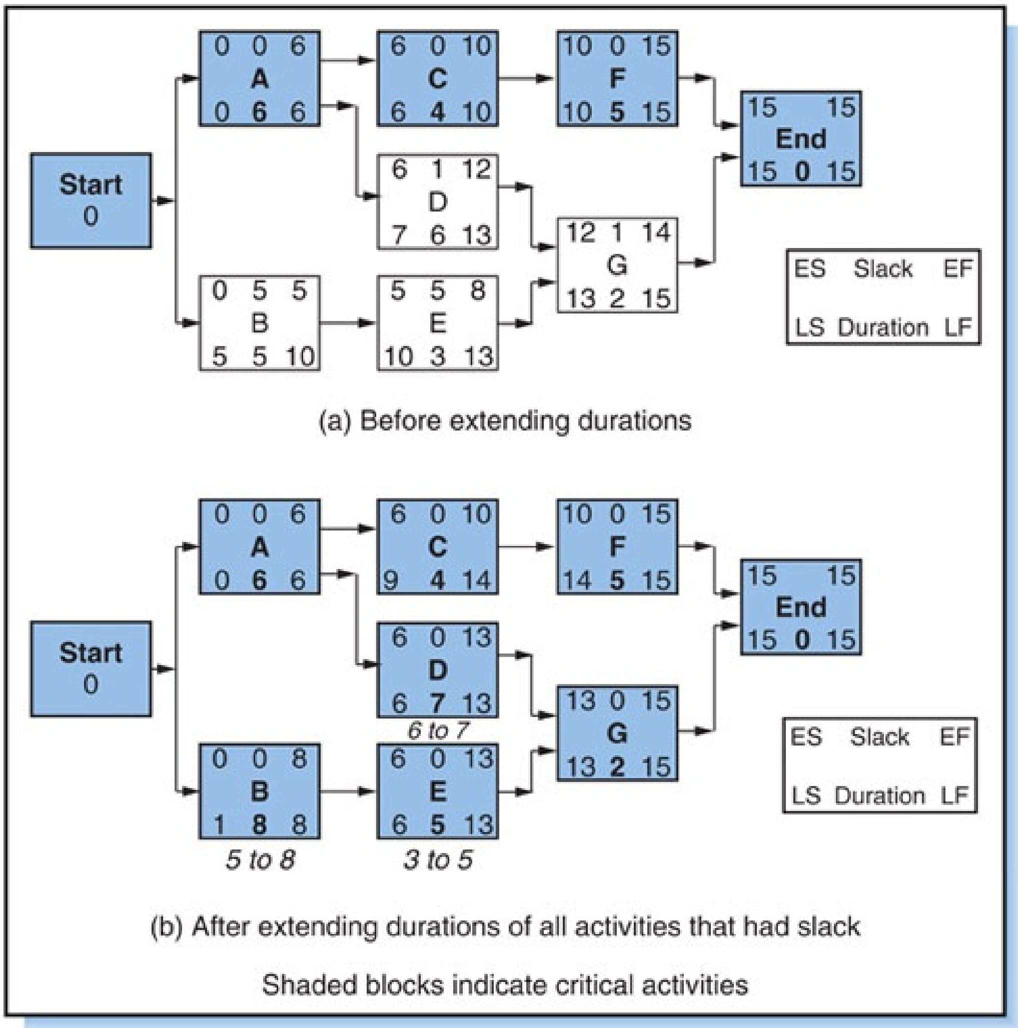


Figure 7.5 Example network using crash times.

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 7.5(a) that because Path B–E–G has a slack of 5 weeks, activities along this path can be stretched by up to 5 weeks without extending the project. Three weeks can be added to Activity B (bringing it to the normal duration of 8 weeks) without lengthening the project. Also, 2 weeks can be added to E and 1 week to D, both without changing the project duration.The final project cost is computed by subtracting the savings obtained in extending B by 3 weeks, E by 2 weeks, and D by 1 week from the initial crash cost.

$\$104K—3(\$3K)—2(\$2K)—1(\$5K) = \$86K$

In general, to obtain the shortest project duration (called "crashing the project"), first crash all activities, then extend the noncritical activities with the greatest cost slopes to use up available slack and obtain the greatest cost savings. An activity can be extended up to its normal duration, which is assumed to be its least-costly time (Figure 7.2).

Total Project Cost

The previous analysis dealt only with *direct costs*—costs immediately associated with individual activities 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 duration of the project. In other words, *indirect costs*, in contrast to direct costs, *decrease as the 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

$\text{Indirect cost} = \$10K + \$3K(T_e)$

where T_e is the expected project duration in weeks. Figure 7.6 shows this on the indirect cost line. It also shows the *total project cost*, which 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 gives the lowest total project cost. As Figure 7.6 shows, from a cost standpoint, 20 weeks is the "optimum" project duration.

In addition to direct and indirect costs, other costs that influence total project cost (and hence the optimum T_e) are *contractual incentives* such as *penalty charges* or *bonus payments*. As described in the Appendix to

Chapter 3, a penalty charge is a fee imposed on the contractor for not completing a project deliverable on time, and a bonus payment is a reward or cash inducement for completing it early.

See Appendix, Chapter 3

Suppose in the previous example the contract specified a target completetion by 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 7.7 shows the influence of these incentives on total project cost. Notice that even with incentives, the optimum duration (for the contractor) is at 19 or 20 weeks, not the contractual 18 weeks. This example reveals that a formal incentive agreement alone is not necessarily enough to influence performance. For the incentive to motivate the contractor it must have “teeth”; i.e., it must be of sufficient magnitude with respect to other project costs to affect contractor 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.

7.2 Variability of Activity Duration

Suppose you are driving to somewhere, and Figure 7.8 shows the the estimated time duration it will take you to get there. If everything goes well (no traffic or mechanical problems) you will get there in the shortest time—the "Optimistic Duration." Most likely, however, it will take you longer than that—the "Most Likely Duration" of 30 minutes. Of course, it could take longer than this—say, when traffic is

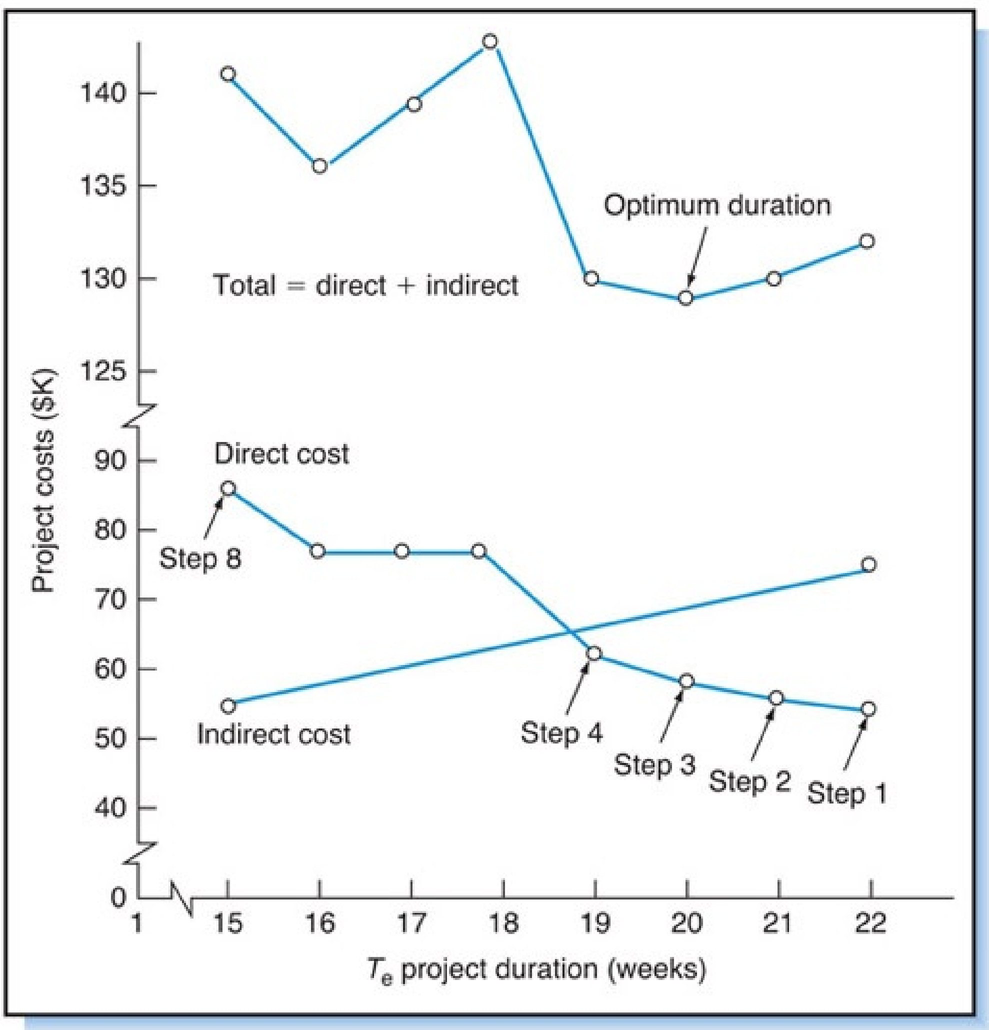


Figure 7.6 Total time–cost tradeoff for the project.

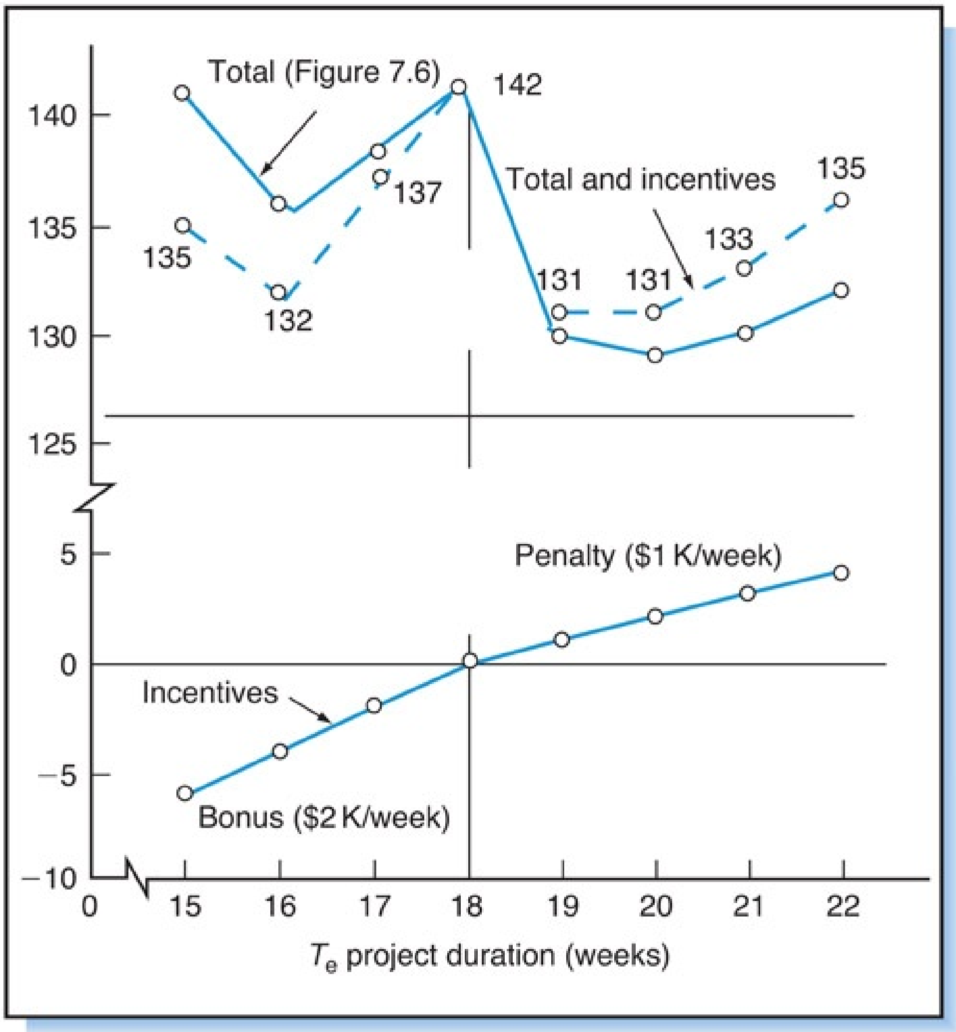


Figure 7.7 Time–cost tradeoff for the project with incentives.

congested or, worse, you get in an accident. Note in the figure 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 than the Most Likely time are greater than the chances of arriving earlier.

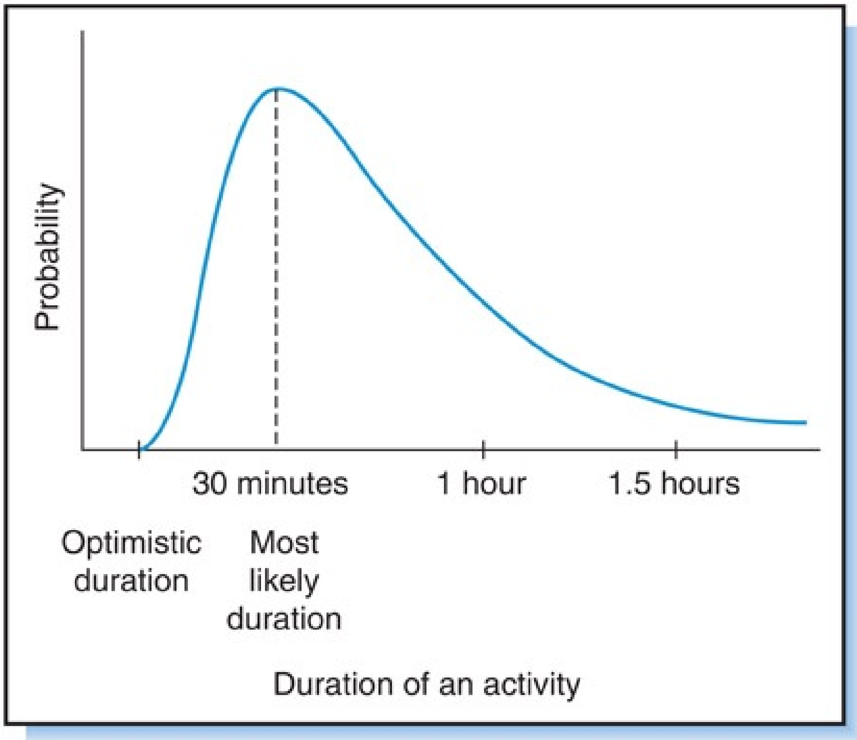


Figure 7.8 Variability of activity duration.

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 will take, how can you possibly say when the project will be completed?

The scheduling approaches discussed in Chapter 6 and the preceding section on time–cost tradeoff ignore variability and assume that activity durations are 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 6

Variability Effects on a Project Network

Figure 7.8 relates to a single activity. In a project some activities will be completed earlier than expected, others later. When activities are combined in a network, however, the early activities and late activities do not average out: in general, *it is only the late activities that impact the project completion.* This is one reason why projects tend to take longer than estimated.

Consider for example Activity A in Figure 7.9. 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 before the scheduled start date.

Consider a second example. Most project networks consist of several paths that merge together into a critical path. Figure 7.10(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 $0.5 \times 0.5 = 0.25$ or 25 percent. Figure 7.10(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 $(0.5)^5$ or about 3 percent. This effect is called *merge bias* or *merge-point bias.*

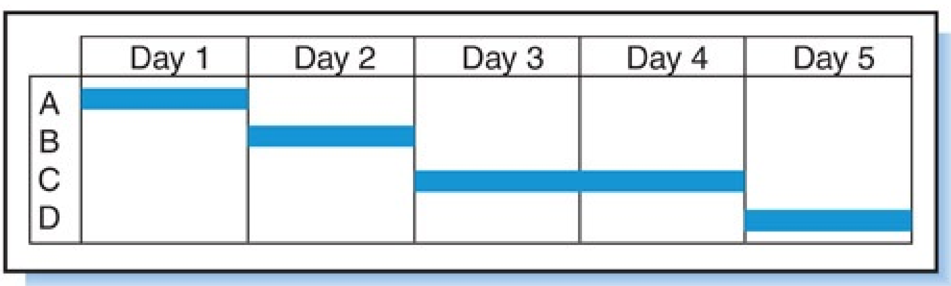


Figure 7.9 Activities delayed if Activity A is delayed.

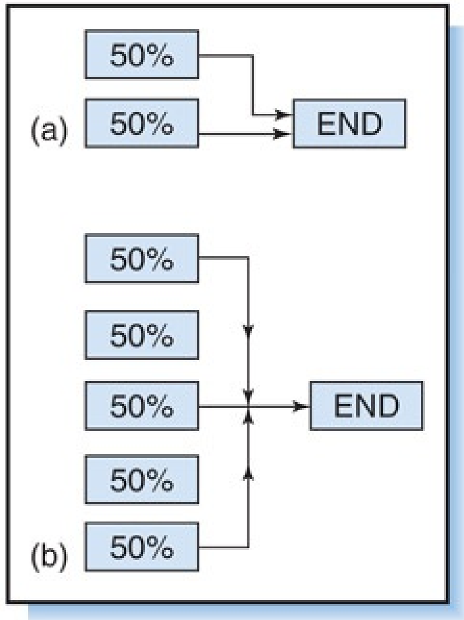


Figure 7.10 Activities delayed where paths merge. (a) Two paths merging, each with 50 percent chance of being on time; (b) Five paths merging, each with 50 percent chance of being on time.

Chapter 6 addressed the fact that the critical path is not necessarily stable but can change if noncritical activities take longer than planned or if critical activities take less time than planned. Either case can result in the project being delayed.

See Chapter 6

Several methods have been developed to help grapple with the uncertainty about when a project will be completed. These are addressed in the following sections, starting with PERT.

7.3 Pert

The PERT method was developed for application in projects with uncertain activity durations. It originated during the US Navy's Polaris Missile System program, an example of a complex research and development program with much uncertainty regarding the kind of research to be done, the stages of development needed, and how fast they can be completed. Projects like this are defined at the same time while technological developments are unfolding and before many of the problems in technology, materials, and processes have been identified. The project duration is uncertain and there is great risk that it will overrun the target completion time.

To provide greater certainty in estimating the duration of the Polaris 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 a tool not for scheduling, per se, but for *analyzing the project network* (and the schedules resulting from the network).

Three Time Estimates

The network methods discussed in Chapter 6 determine the critical path and slack times using *best estimates* for activity duration. PERT, however, addresses uncertainty in the durations by using three time estimates for activity duration —*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 the people who will perform or manage the activity.

See Chapter 6

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 project events will happen by a particular time.

As seen in Figure 7.11 the *optimistic time, a*, is the minimum time for an activity—the situation where everything goes well and there is little hope 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 only likely problems in work and not highly unlikely events such as natural disasters.

The three estimates in Figure 7.11 are related in the form of a *Beta* probability distribution with parameters *a* and *b* as the end points and *m*, the most frequent 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 7.8 had no end point on the right-hand side, the curve in Figure 7.11 precludes very unlikely events and has end point *b*.

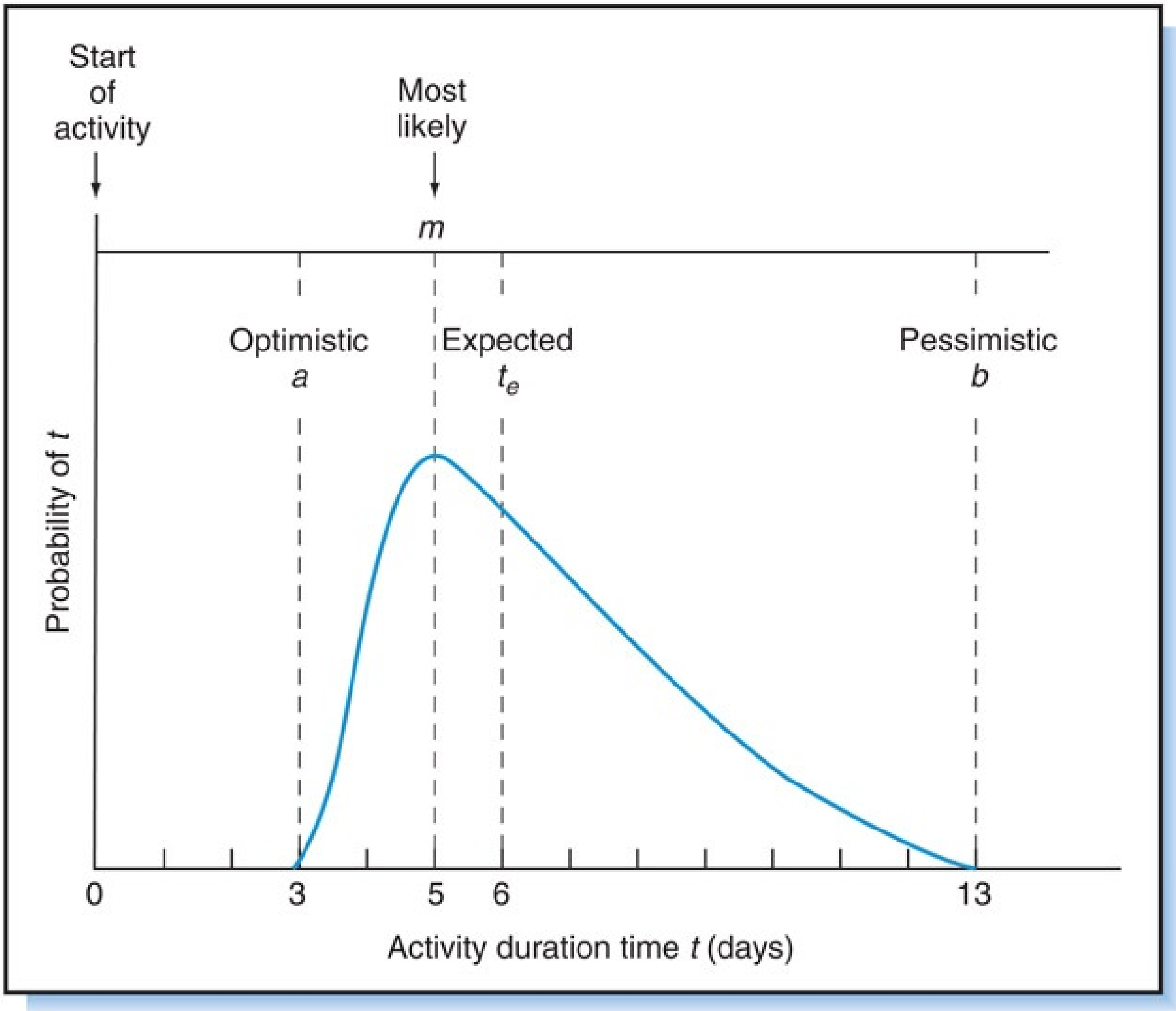


Figure 7.11 Estimating activity duration.

Based on this distribution and the three time estimates, 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 point on the distribution in Figure 7.11 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 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 a routine (repetitive) job, estimates of a and b are close to each other, V is small, and $t_e$is more likely.

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 6. 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.

PERT is a stochastic approach; hence the project duration is not considered a 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 activity durations, it follows that T_e is also an expected time. The project duration can be thought of as a probability distribution with an *average* of T_e. Thus 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 is computed as the sum of the variances of the activity durations along the critical path:

$$V_p = \sum_{CP} V$$

where V is the variance of an activity on the critical path.

These concepts are illustrated in the AOA network in Figure 7.12.

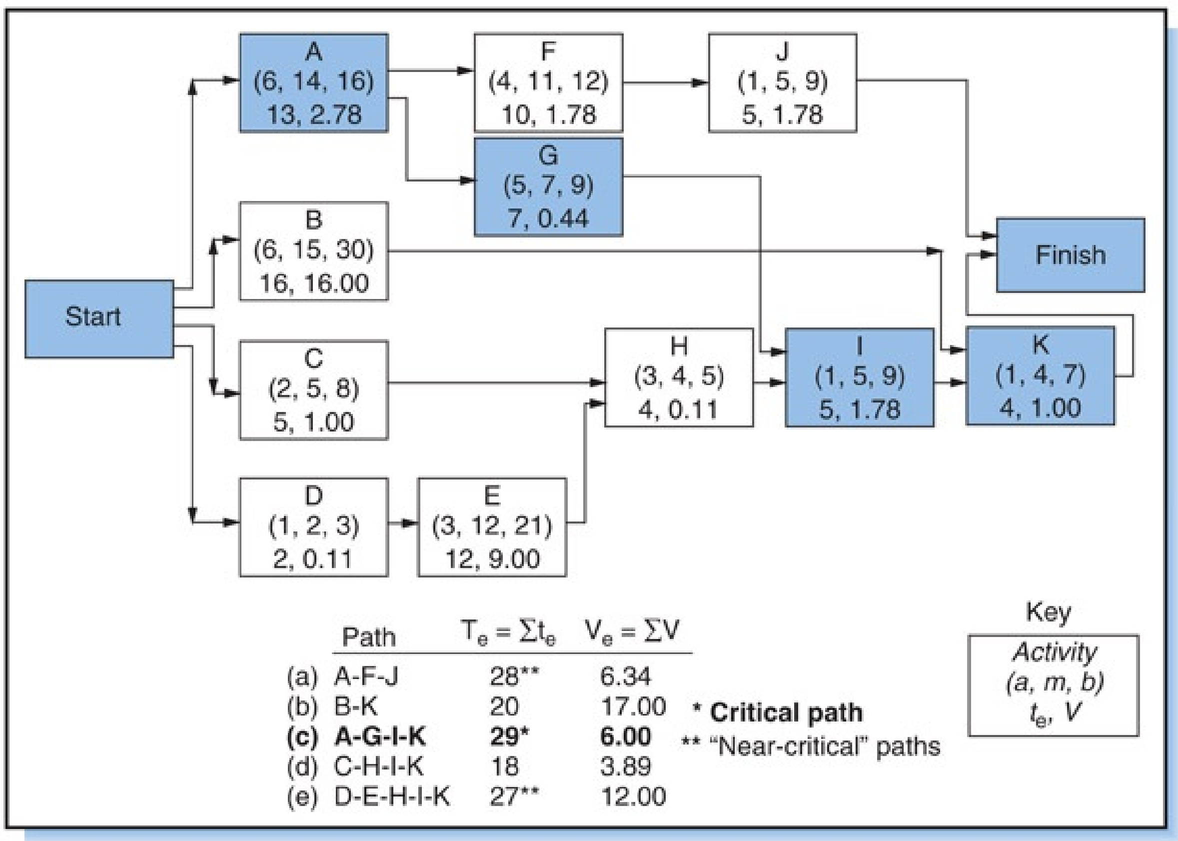


Figure 7.12 PERT network with expected activity durations and activity variances.

The distribution of project durations is assumed to be the normal, familiar bell-shaped curve. Given this assumption, it is easy to determine the probability of meeting any specified project target completion date T_S.

As examples, consider two questions about the project shown in Figure 7.12: (1) What is the probability of completing the project in 27 days? (2) If we want to be 95 percent sure about committing to project duration, what duration 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 question 1, use T_S = 27 days. From the network, the expected project duration, T_e, is computed as 29 days. Therefore

$$z = \frac{27 - 29}{\sqrt{6}} = -0.82$$

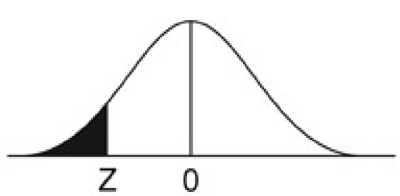
The probability of completing the project within 27 days is equal to the area under the normal curve to the left of z = −0.82. Referring to Table 7.2(a), the probability is about 21 percent.

To answer question 2 (duration with a 95 percent certainty): using Table 7.2(b), for a probability of 0.95 the z value is 1.6. As before, we calculate

$$1.6 = \frac{T_s - 29}{\sqrt{6}}, \text{so } T_s = 33 \text{ days}$$

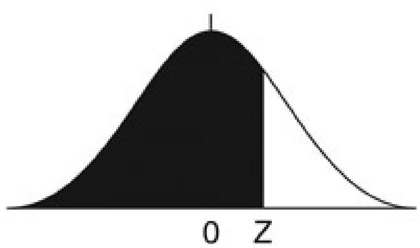
Table 7.2 Normal Distribution Function for Completing a Project by Time T_S

[a] Probability that project will be completed sooner than the expected duration (project will probably finish late if committed to the T_S date) duration.



Z Value	Probability	Z Value	Probability
0	.50	−1.2	.12
−0.1	.46	−1.3	.10
−0.2	.42	−1.4	.08
−0.3	.38	−1.5	.07
−0.4	.34	−1.6	.05
−0.5	.31	−1.7	.04
−0.6	.27	−1.8	.04
−0.7	.24	−1.9	.03
−0.8	.21	−2.0	.02
−0.9	.18	−2.1	.02
−1.0	.16	−2.2	.01
−1.1	.14	−2.3	.01

[b] Probability that project will be completed later than the expected duration (project will probably finish early if committed to the T_s date).



Z Value	Probability	Z Value	Probability
0.0	.50	1.2	.88
0.1	.54	1.3	.90
0.2	.58	1.4	.92
0.3	.61	1.5	.93
0.4	.66	1.6	.95
0.5	.69	1.7	.96
0.6	.72	1.8	.96
0.7	.76	1.9	.97
0.8	.79	2.0	.98

0.9	.82	2.1	.98
1.0	.83	2.2	.99
1.1	.86	2.3	.99

In other words, it is "highly likely" (95 percent probable) that the project will be completed within 33 days. Note that since we are working with values that are merely estimates, it does not make sense to compute 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 7.12 has two paths that are "near critical" in length. The variance of these paths is large enough that either could easily become critical by exceeding the 29 days of the original critical path. In fact, as you may wish to verify using the statistical procedure described previously, the probability of *not* completing Path (a) (A–F–J) and Path (e) (D–E–H–I–K) within 29 days is 34 percent and 29 percent, respectively. So there is more than a slight chance that these paths could become critical. The warning is: don't overemphasize the critical path and ignore the near-critical paths—paths that could themselves 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 66 percent and 71 percent, respectively, and the probability of completing (c), the critical path, is 50 percent. So the chance of completing all paths within 29 days is the product of the probabilities $1.0 \times 1.0 \times 0.66 \times 0.71 \times 0.5$, or less than 25 percent.

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. Possible 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 the successor when only some of the predecessor sub-activities have been completed.
2. Add resources to critical and near-critical activities by transferring resources from activities with large slack times.
3. Substitute time-consuming activities with ones that are less so, or delete activities that are not absolutely necessary.

Each of these has drawbacks. Fast-tracking increases the risks of making mistakes and having to repeat activities. Adding resources to speed up activities increases the cost, and transferring them between activities requires changing plans and schedules, increases administrative costs, and aggravates the functional managers who supply the resources. The final alternative, substitution or elimination of activities, jeopardizes project performance, especially when it equates to making "cuts" or using poorer quality materials or less-skilled labor.

Criticisms of PERT[6]

The PERT method has been criticized for its shortcomings. For example, it ignores human behavior and presumes that an activity will start as soon as its predecessor activities are completed—ignoring that resources might not be available or that people procrastinate. Also, it assumes that activity durations are independent, though often they are not. Whenever resources are transferred from one activity to another, then the durations of both activities are changed.

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 over 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.

Accuracy of estimates often depends on experience. Whenever a database can be formed based upon experience from similar activities from previous projects, a "history" can be developed for each kind of activity that can be used to estimate the durations for future similar activities. In fact, reliance on good *historical data* for estimating times makes the PERT method more appropriate for projects that are somewhat "repeatable" and less so for the first-of-a-kind projects. Because of this the PERT method tends to be used in construction and standardized engineering projects, but seldom elsewhere.

Some of PERT's shortcomings are addressed by simulation.

Monte Carlo Simulation of a PERT Network

Monte Carlo computer simulation is a procedure that takes into account 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; it also gives the probabilities of other paths becoming critical.[7]

Simulation allows the use of a variety of probability distributions besides Beta, including distributions based upon 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 independence of activity durations and normality of the project duration distribution. The following example from Evans and Olson illustrates usage of simulation to assess the likelihood of project completion time.[8]

Example 7.2: Simulation to Determine Project Duration

Refer to the project activities and time estimates in Table 7.3 and the project network in Figure 7.13.

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.

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 to Table 7.2, the probability is about 16 percent.

Table 7.3 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.

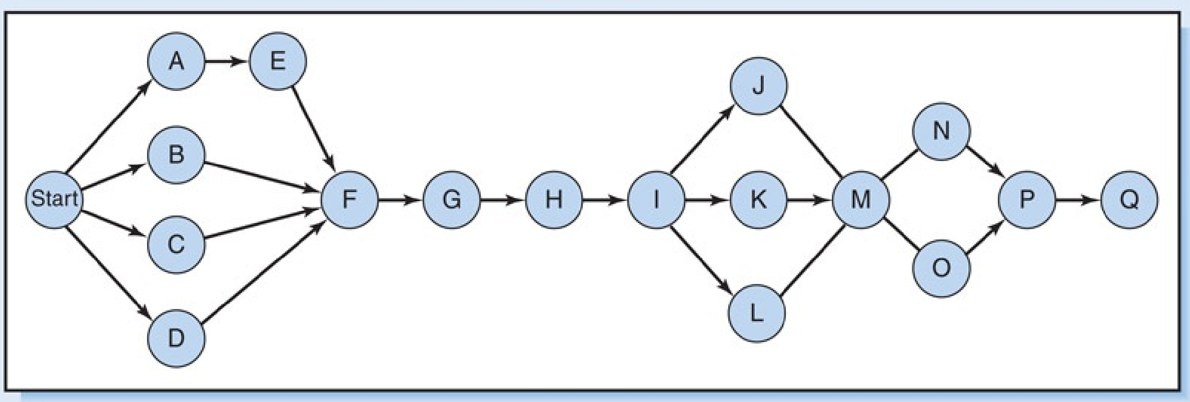


Figure 7.13 Project network.

Using the simulation program Crystal Ball to generate the completion times for 1,000 replications of the project yields the distribution in Figure

7.14. (Various other programs such as Risksim, @Risk, Arena, and Simul-8 can also be used.)[9]

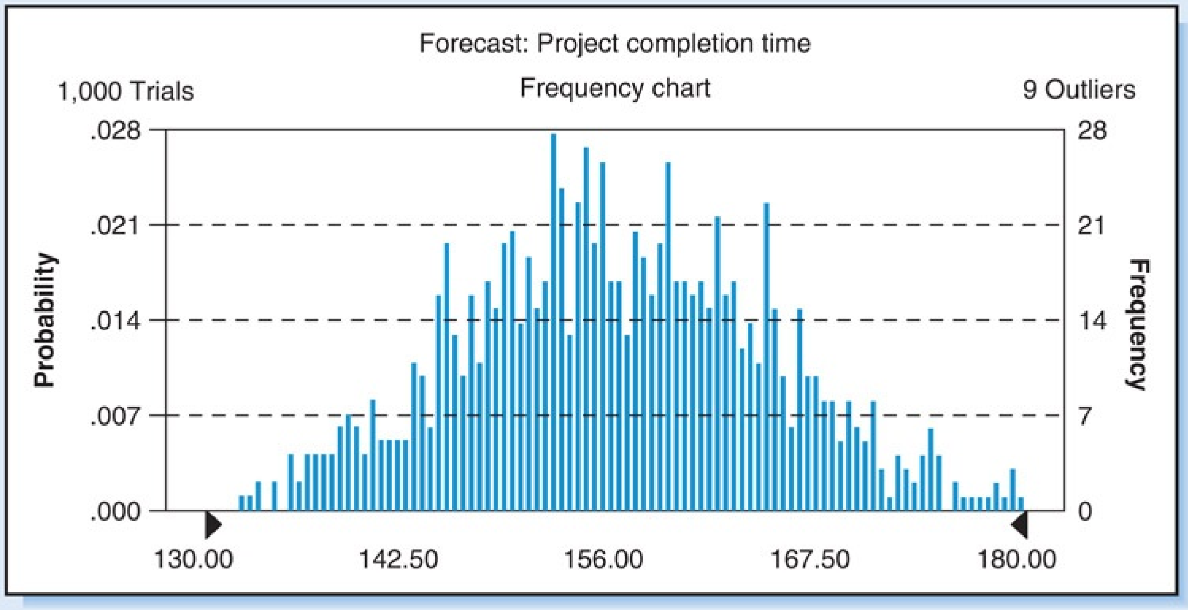


Figure 7.14 Crystal Ball simulation results for project completion times.

The simulation 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 7.14). 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.

Simulation provides more realistic results than PERT because it compensates for noncritical paths that can become critical. But like PERT, it is merely a method for analyzing schedules, not for creating them. 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 10.

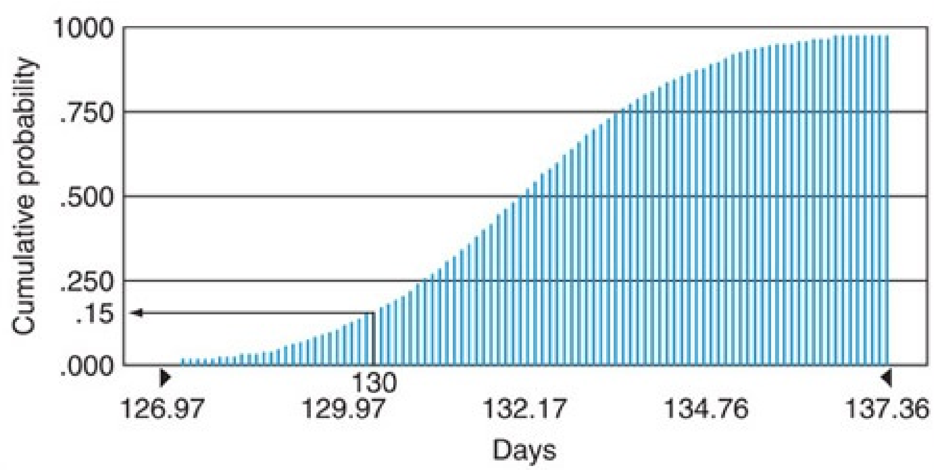
See Chapter 10

Why Projects are Often Late

The project manager might face considerable risk when committing to a due date based solely on the duration of the critical path. For example, a Monte Carlo simulation was used to calculate the probability of finishing a project given the critical-path activity durations shown in Figure 7.15. The most-likely critical path length is 130 days, but the simulation reveals the chance of finishing the project in that time is only 15 percent. Further, this simulation was applied to the critical path only and did not take into account noncritical paths that might become critical—which would reduce the probability even more. While individual *m* 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 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) 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 it also leads to underestimating project duration and cost. The Channel Tunnel ("Chunnel") 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. Plus, the project took nearly 18 months longer to complete than originally estimated.

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 7.15](#) Simulation results show low probability of finishing within critical-path time. Generated by means of Crystal Ball software, assuming triangular distributions.

7.4 Allocating Resources and Multiple Project Scheduling

Organizations in construction, consulting, systems development, and maintenance commonly use a pool of shared equipment and skilled workers from which all projects draw. This also happens in matrix organizations (Chapter 14) where projects share resources from the same functional departments. This section addresses the matter of scheduling multiple projects that share constrained resources.

See Chapter 14

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 their shared resources they are *inter*dependent.

As might be expected, the problem of scheduling multiple concurrent projects is analogous to scheduling multiple concurrent activities within a single project, but with modification to account for the economic, technical, and organizational issues that arise when dealing with multiple projects (see Chapter 18).

See Chapter 18

First, each project has its own target completion date, and all the projects must be scheduled to finish as close to those dates as possible to avoid deferred payments, penalty costs, or lost sales and revenues. Further, when projects are interdependent, delays in one project can have a ripple effect on others; the delay of a satellite development and launch project will cause the subsequent delay of a telecommunications 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 schedules 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 such 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, 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 $(n!)^{m}$. Optimization using normal polynomials is usually not feasible (the problem is "NP hard"). Heuristic methods, on the other hand, provide simple and acceptable solutions.

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.

Heuristics 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 receives the resource. The most common heuristic rules for determining scheduling priority are:

a. *As soon as possible*: activities that *can* be started sooner are given priority over (or scheduled ahead of) those that can be started later.
b. *As late as possible*: activities that can be finished later are given lower priority than those that must be finished earlier.
c. *Most resources*: activities requiring more resources are given priority over those requiring fewer resources.

d. *Shortest task time*: activities of shorter duration are given priority over those of longer duration (sometimes referred to as *shortest activity duration, shortest processing time*, or *shortest operating time*).
e. *Least slack*: activities with less slack are given priority over those with more slack; critical path activities thus have highest priority. (This rule is also referred to as *slack time remaining*.)
f. *First come first served*: activities that arrive earlier or require the resource earlier are given priority.
g. *Earliest due date*: when a resource is to be allocated to more than one project, the project with the earliest target completion date is given priority. Alternatively, the activity with the *earliest next operation* is given 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 employs some combination of these rules (e.g. using "shortest task time," then using "as soon as possible" as a tie breaker). Figure 7.16 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 Figure 7.16 shows, the rules yield different results, some better than others. In general, with the *as late as possible* rule, everything occurs at its latest date; the drawback of this is that a delay in any activity will delay 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 7.17 shows what happens when activities are scheduled (a) *longest-task first* versus (b) *shortest-task first.* 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 7.16 is 9 weeks, the critical path length. Given the 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.

7.5 Theory of Constraints and Critical Chain Method[10]

The theory of constraints (TOC) is a systems approach to improving the performance of business systems.[11] A premise of TOC is that every system has a goal and that often only one element of the system, called the *system constraint*, limits achieving that goal.

The application of TOC to project scheduling and control is called *critical chain project management* (CCPM) or the *critical chain method* (CCM). The constraint for an individual project is its *duration* or due date, and CCPM aims to reduce that duration and provide a more predictable project completion date.[12] 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 to multiple concurrent projects.[13]

Commitment to Due Dates

With traditional methods, people working in each project activity must *commit* to completing the activity by a target 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 pessimistic or "padded" time estimate. Although this behavior is quite normal, it results

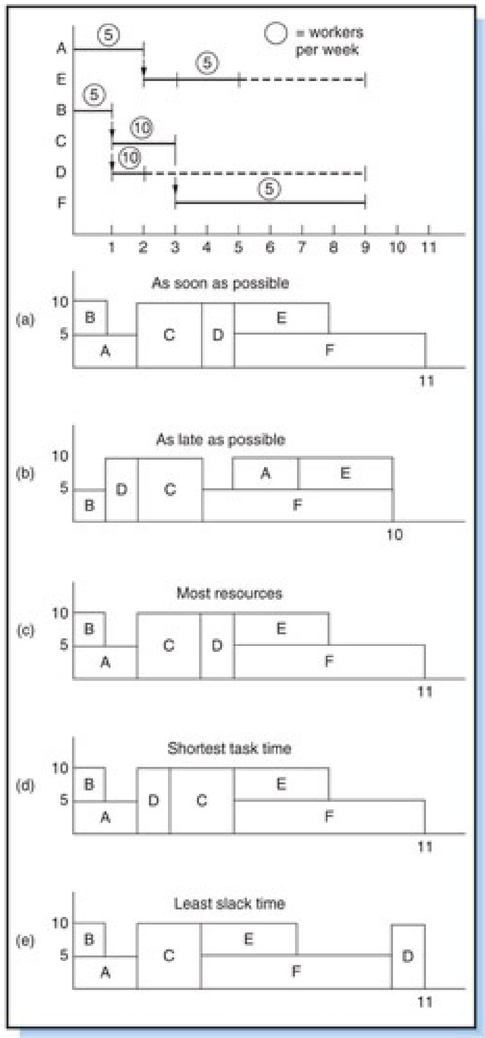


Figure 7.16 Results of several priority rules on project schedule and completion times.

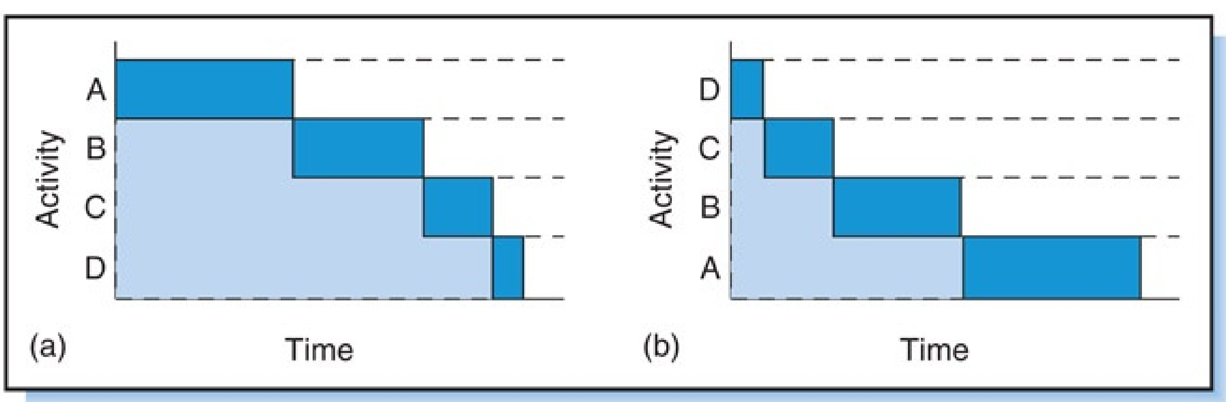


Figure 7.17 The shortest task time rule reduces waiting time. (a) Longest task first. (b) Shortest task first.

everywhere in inflated estimated activity durations and, hence, longer-duration projects. The CCPM approach to avoiding time padding is simply this: while the project manager obviously needs to commit to a project due date, 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 a target date. In requesting time estimates, everyone is asked to provide the most "realistic" time, meaning the time with a 90 percent chance of it being achieved. This time is then cut in half to obtain an "aggressive" estimate, which eliminates any padding for contingency. (It should be noted, however, that whenever project members provide "aggressive" (not padded) estimates, the manager does not cut them in half.) The realistic and aggressive estimates provide the basis for estimating the amount of "buffer" required to protect the project as a whole against delays.

Project Buffer and Feeding Buffers

To protect against delay, a *project buffer* (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. (As mentioned, the project manager commits to a date for the project, but people responsible for individual project activities do not commit to dates; they just try to complete their activities speedily.) But, you might ask, won't adding a time buffer lengthen the project duration? The answer is no, because the aggressive time estimates used for project activities are 50 percent shorter than the "realistic" but padded estimates. Dividing the realistic estimate by two results in two estimates: one, an aggressive estimate for the duration, the other an estimate of the "padding" that was included in the original, realistic estimate. Summing the padding estimates for all the activities to form the project buffer and putting the buffer at the end of the project will account for any activity delays.

To illustrate, consider the project shown in Table 7.4 and Figure 7.18. The critical path, P–Q–R–Z, is 32 weeks long. Now look at Figure 7.19 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 7.19 the presence of a *feeding buffer*, which is the number of weeks removed from the 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 noncritical path feeds into the critical path. The feeding buffer protects against delays in noncritical activities.

While project buffers and feeding buffers bear a resemblance to slack, they are *not* slack. Whereas with traditional methods activities are scheduled as early as possible and slack may be used to delay activities when necessary, in CCPM all activities are planned to start *as late as possible*, but with buffers. (As discussed later, however, during *execution*, critical activities are started *as soon as possible* to take advantage of gains from activities completed early.) The buffers "belong" to the project manager, and only she can allocate time from them; this will happen whenever an activity exceeds its planned duration.

Table 7.4 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

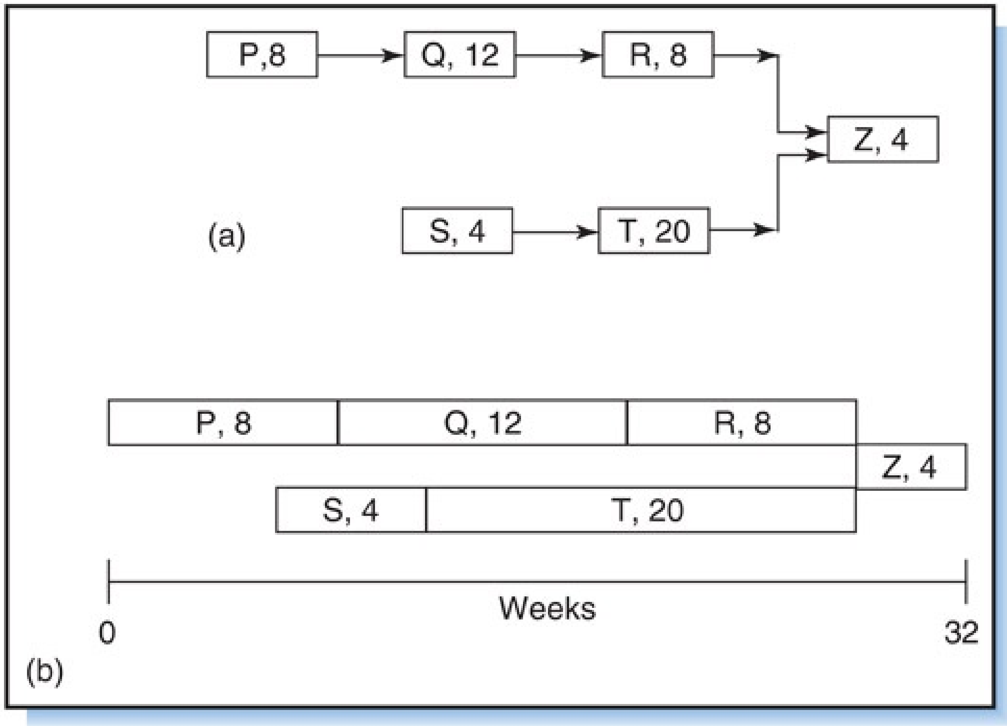


Figure 7.18 (a) Networks for activities in Table 7.4; (b) Time-scaled network for activities in Table 7.4.

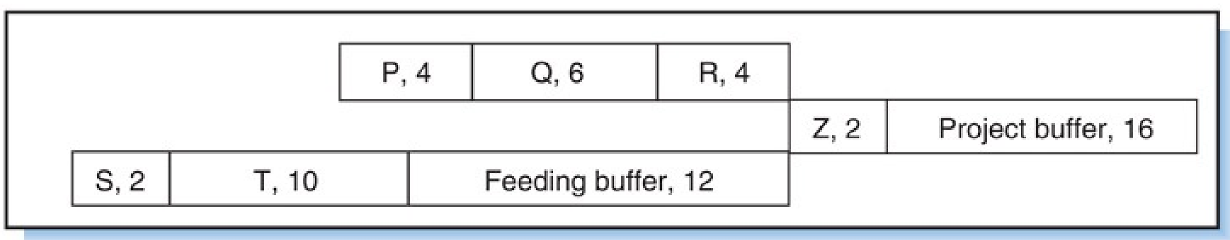


Figure 7.19 Schedule with contingency reserves allocated to the project manager.

But the size of the project buffer can be substantially less than the sum of the activity paddings that were removed. There are two reasons. First is the mathematical effect—well known in risk management—called *aggregation*, which says that the uncertainty of finishing a project on time is much less than the sum of the uncertainties of finishing individual tasks within that project on time. Because of the aggregation effect, the paddings removed from individual activity durations can be replaced by one project buffer that is much smaller than the sum of all the removed paddings.

The second reason why the project buffer size can be reduced stems from the obverse of Parkinson's Law, which states "work expands to fill the time available." Removing padding from each activity creates a sense of urgency; people have less time to do the work, hence tend to work faster than when they have more time. For these reasons, the project buffer size can be reduced by 50 percent. This is shown in Figure 7.20.

The project manager commits to completing the project on or before the date at the end of the project buffer, Week 28, although the project team works to complete the project on or before the date at the *start of* the project buffer, Week 20. Thus, there is a high likelihood that this project will be completed *in less than 28 weeks*. With the critical path method, there is a high likelihood that the project will be completed *after* the critical-path duration of *32 weeks*.

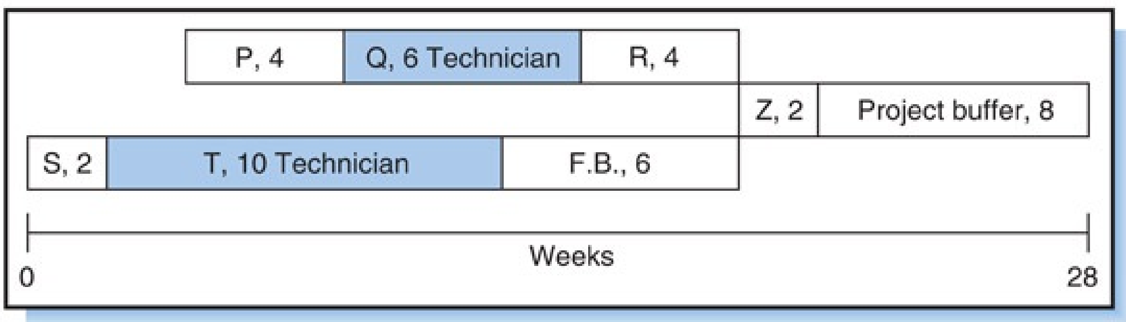


Figure 7.20 Schedule with buffer sizes reduced.

Critical Chain

Figure 7.20 reveals a potential problem, though: activities Q and T both use the same resource (the "technician"). Assume the technician can work on only one activity at a time. To allow for that, the schedule is adjusted as shown in Figure 7.21 (Putting Activity Q before T is another possible schedule). In the adjusted schedule, path S–T–Q–R–Z is called the *critical chain*, defined as the *path connecting activities that require the same resource*. The critical chain is not necessarily the longest path through the network, although whenever the length of the critical chain plus buffers exceeds the length of the critical path, the critical chain, *not* the critical path, determines the project duration.

Traditional network methods address resource-conflict problems by means of resource leveling, although the result will not necessarily be the same as with CCPM since they resolve these conflicts starting with an initial schedule and use available slack. In CCPM, resource conflicts are resolved during the initial scheduling process and resource dependencies are treated as a priori.

Note that the feeding buffer, F.B., is 4 weeks, not 2. The reason is because it follows only one activity, P, and hence the aggregation effect does not apply. Ultimately, the size of the buffer is at the manager's discretion and is whatever she decides.

If the schedule does not meet a predetermined due date or upper management wants the project completed sooner, additional resources must be added. But adding people and other resources is costly and often disruptive; hence CCPM attempts first to make full use of whatever resources are currently available.

Resource Buffers: Capitalizing on Finishing Work Earlier than Planned

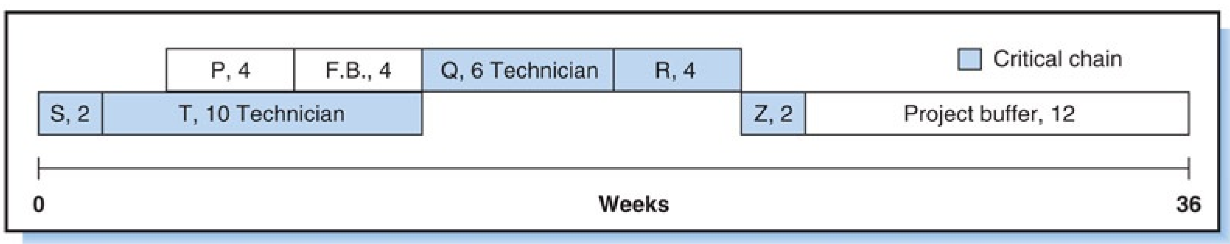


Figure 7.21 Schedule adjusted so that every resource performs only one task at a time.

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 (people and equipment) that have been prescheduled often are not available to start earlier because they are busy with something else. As a consequence, whenever delays occur, the project is delayed; whenever gains (predecessors finishing earlier than scheduled) occur, it makes no difference!

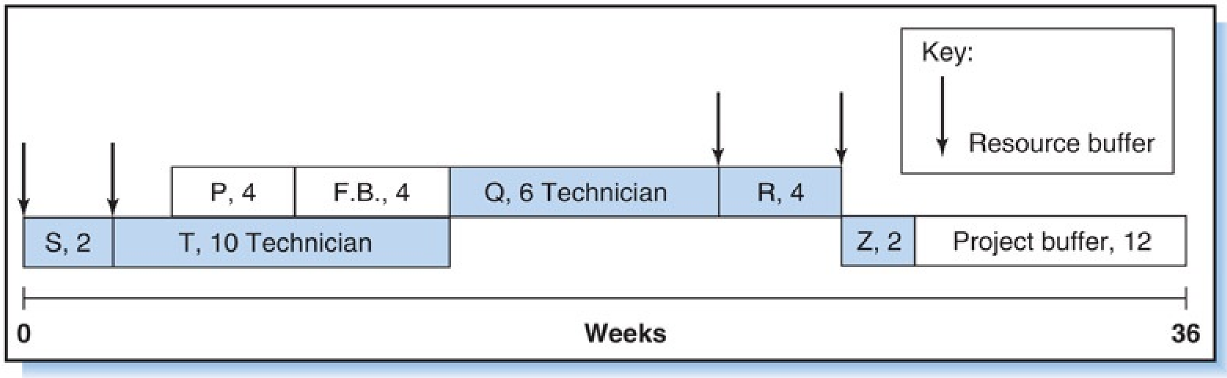


Figure 7.22 Resource buffers providing countdown on when to start critical activities.

In CCPM, the project team is able to capitalize on predecessors finishing early through the use of *resource buffers*. Unlike project and feeding buffers, resource buffers do not add time to the schedule. A resource buffer 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 marathon relay race, each runner is prepared to accept the baton from the previous runner, regardless when the latter 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 email or other messages to resources, counting down the time remaining before they must be ready to start an activity. The locations of resource buffers are illustrated in Figure 7.22.

Note that resource buffers are inserted only on the critical chain since feeding buffers are able to adequately deal with the 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 as to when she will finish Activity Q and must start Activity T.

Milestone Buffers

Sometimes milestone deadlines are set at intermediate times in the project, such as at the scheduled completion dates for project phases. In that case a *milestone buffer* is inserted before each milestone. When milestone buffers are used, the size of the project buffer is reduced; in effect, the project buffer is divided up among the milestone buffers. The different types of buffers are summarized in Table 7.5.

Sizing of Buffers

CCPM relies heavily on project and feeding buffers, so making them the right size is important. Goldratt suggests that activity durations be cut by 50 percent and that the project buffer be half the duration of the resulting longest path.[14] This method, which reduces project duration by 25 percent, was illustrated in Figure 7.19 and 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.[15] Newbold proposes the *square root of sum of squares* (SSQ) method, where the buffer size is set to the square root of the sum of squares of the differences between the low-risk duration and the mean duration for each task along the longest path leading to the buffer.[16] Others have suggested additional methods.[17]

Table 7.5 Summary of Buffer Types for a Single Project

Buffer Type	Function of the Buffer
Project buffer	Comprised of aggregated contingency reserves taken from activities on the critical chain; provides a contingency reserve between the earliest completion date possible and the committed 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 noncritical paths; stabilizes the critical chain by preventing noncritical activities from delaying critical activities.
Resource buffer	An early warning or "count down" to the start of a critical activity that ensures that resources are ready to do work on the critical chain as soon as all preceding activities have been completed.

Padding, Multi-tasking, and Procrastination

Projects take longer than necessary for many reasons. First, as already stated, people *pad* their time estimates, and the effect of this gets worse as each manager in the WBS adds 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.

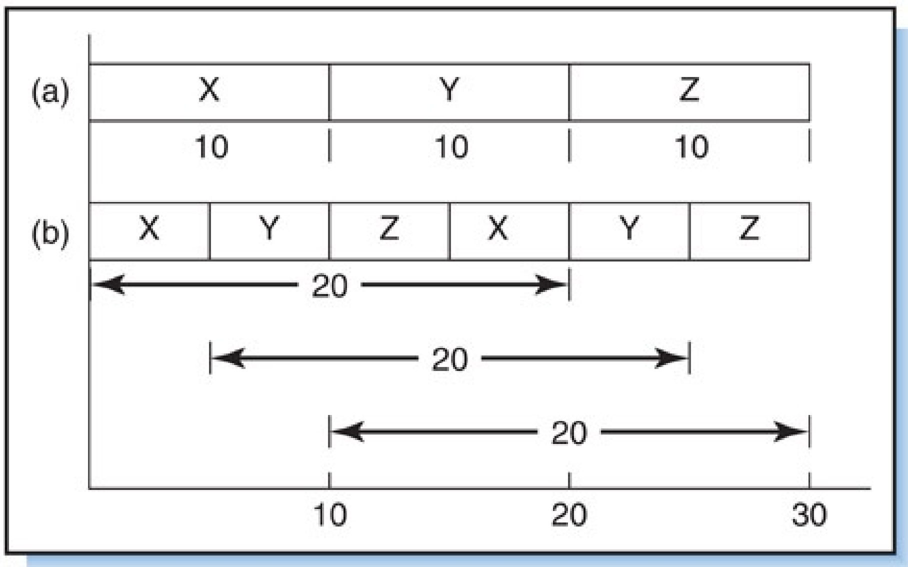


Figure 7.23 Effect of multi-tasking on elapsed and completion times.

Second, people *multi-task.* For example, 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 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 7.21(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 illustrated in Figure 7.23(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 the activities or projects are broken up and intermixed with other projects, the greater the elapsed time to finish any of them.

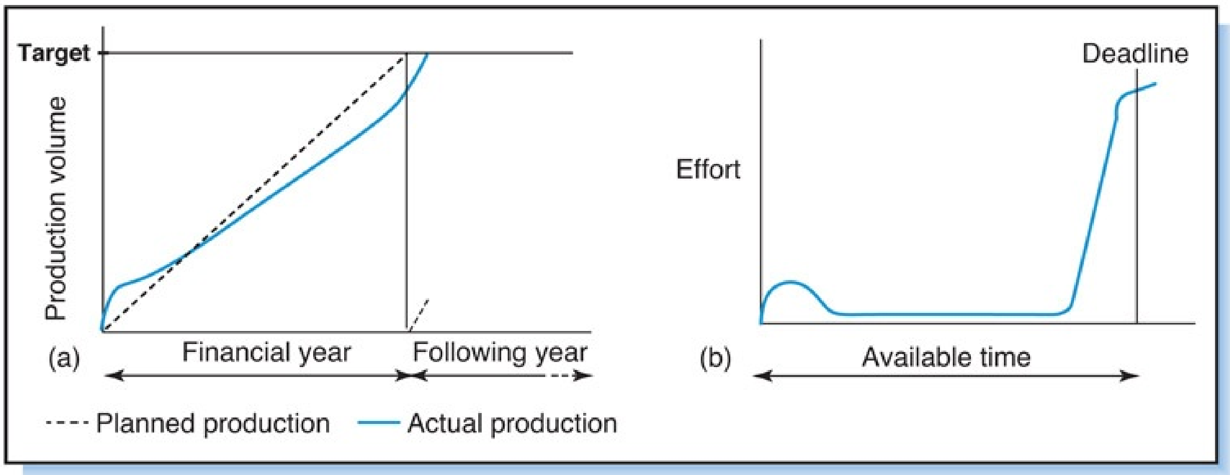


Figure 7.24 Students' syndrome (a) in a production and (b) in a project.

Compounding the effect is that multi-tasking precludes people from gaining the momentum they would have by focusing uninterrupted on only one task, as was illustrated in Figure 6.25. Multi-tasking should be avoided. By focusing on just one activity at a time, each activity is completed sooner, successor activities start earlier, and the project finishes sooner.

A third reason projects take longer than necessary is people *procrastinate* and waste available slack.[18] 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 people perceive slack, they 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. A similar effect happens in production and project environments, shown in Figure 7.24.

Shortening activity durations and scheduling activities as late as possible (with buffers) reduces the tendency to procrastinate.

Buffer Management for Project Control

Every activity on a 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. The amount of buffer "consumed" is determined each day by simply asking people working on activities the expected remaining duration and calculating to what extent that duration overruns the aggressive estimate and, hence, consumes (penetrates) the buffer. As long as 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. Whenever someone is required to work on more than one activity, the most-consumed buffer indicates which activity has highest priority. Buffer management is further discussed in Chapter 11.

See Chapter 11

Challenge of CCPM: Changing Behavior

A belief in most project organizations is that because the project manager has to commit to the due date, everyone in the project must also commit to due dates. In CCPM, the premise is that only the project manager needs to commit to a completion date; everyone else works toward realistic (relatively "aggressive") estimates. While getting everyone to accept this premise in small projects can be relatively easy, such is not the case for major projects. Since most people are accustomed to working toward deadlines, this requires no less than a cultural and behavioral change at all levels of the organization, including top management. Senior managers and customers who do not understand the principles of CCPM will try to work around them.

Software Support for CCPM[19]

Many project management software packages include provision for CCPM, and although many others require add-ons to make them compatible with CCPM. For example, Sciforma™ and Concerto™ software fully support CCPM, but MSProject supports it only with the Prochain™ add-on.

7.6 TOC Method for Allocating Resources to Multiple Projects[20]

As much as 90 percent (by value) of all projects are carried out in multi-project environments.[21] The theory of constraints (TOC) provides a five-step procedure for scheduling the start of new projects so as to maximize the number of projects an organization can handle concurrently.

Step 1: Identify the Constraint

What prevents a company from doing more projects? If the company has few projects on the order books, the constraint might be limited market share or the size of the market. But if it has more demand for projects than the capacity to do them, the constraint is anything that reduces the rate at which projects are completed (flow rate) to less than the maximum; 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 ahead of which queues of work pile up. Such a resource, the system constraint, is called the "drum resource" because it sets the tone or pace of everything flowing through the process. The TOC philosophy emphasizes the importance of keeping the constrained (drum) resource busy, preventing starvation and blockage from reducing the flow. To ensure this, *drum buffers* are used. Application of drum buffer to project management is described elsewhere.

Experience with implementing the theory of constraints, 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 their *execution*. Typical 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.[22] For *planning* a set of projects, *a rule* regarding the scheduling of this activity may be used as proxy for the constraint of the set of projects. 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 7.25, the shaded activities represent final integration in nine projects. To ensure there are always 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 on *all* resources through subordination of individual project schedules as discussed in Step 3 below. For *executing* the set of projects, the typical constraint is the *time* available for *managers to support* the projects.

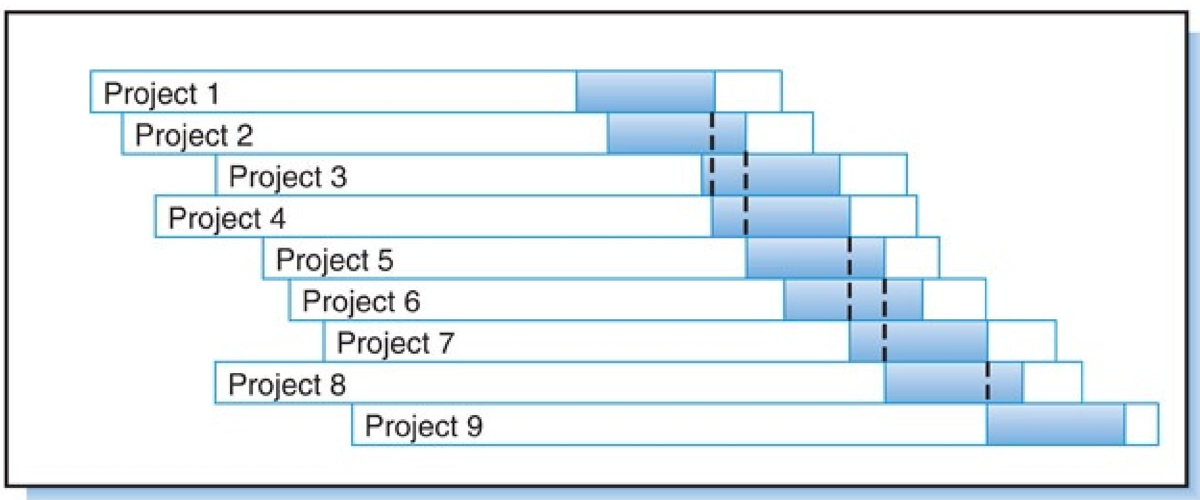


Figure 7.25 Only three projects in final integration at any point in time.

Step 2: Decide how to Exploit (utilize) the constraint

The rule "only three projects in final integration" enables the company to exploit the final integration constraint for project planning purposes. Combining the scheduled date for when a project should start final integration with the project's duration determines when the project should be initiated. 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., the rate at which projects are completed), and ensures that commitments to customers are met.

If the constraint for *executing* (as opposed to releasing) the set of projects is time available for managers to support the projects, then measures must be taken to assure such time is available. If lack of management time constrains the flow of projects, then this constraint has to be removed.

Step 3: Subordinate everything else to decisions regarding the constraint

During scheduling, the release of projects (authorization) is based on the load on the constraint. If the company decides that it should have only three projects in the final integration phase, the proxy constraint is the rule "three projects in final integration." Each new project would be slotted to start final integration at whatever time is necessary to maintain that three-project maximum. The schedules of individual projects are therefore subordinated to the schedule of the multi-project constraint—i.e., the rule regarding the number of projects in the integration phase. The project at 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 capacity so as to increase the number of projects in final integration from three to four. It typically implies costly measures such as building new facilities and hiring and training additional people. Steps 2 and 3 therefore ensure that existing capacity is utilized effectively before money is devoted to acquiring additional resources.

If the constraint is management time available for project support, then the management system should be simplified to improve its effectiveness in performing support functions.

In multi-project environments, resource buffers have less utility. The resources are dedicated to individual projects; they do not become available when a task is completed. They simply begin work on the next task, even if they have to wait a while before starting. Actually, it is desirable that they are idle between projects: to maximize project flow, resources should have to wait for work; work should not wait for resources.

In this way it is possible that all activities (including those on noncritical paths) can start as soon as predecessor activities are completed. Monitoring project buffers is all that is necessary to keep projects on track (as illustrated in Chapter 11, Example 11.3). Tracking project buffers simplifies the tracking and control process during project execution, relieves managers’ workloads, and increases the time available to support other projects. In turn, this elevates the constraint for project execution—the time managers have available to make better project decisions.

See Chapter 11

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:

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 indicated by buffer status (Chapter 11, Example 11.3) and (b) minimize buffer consumption by doing all tasks as soon as possible.[23]

How good is the TOC heuristic for planning compared to traditional priority rules? The answer is somewhat equivocal. For some simple projects the results are the same. One exploratory experiment where TOC was compared with the oft-used least slack rule showed TOC gave better results,[24] but another yielded poorer results.[25] 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.

7.7 Discussion and Summary

This chapter has covered project scheduling methods that address time constraints, resource constraints, uncertainty about activity and project durations, and multiple projects sharing resources. The methods offer ways to accelerate projects and reduce uncertainty about completion dates, although, unlike simpler techniques such as Gantt charts and critical path networks, they have gained lower acceptance and are applied mainly in relatively sophisticated industries. All the methods have limitations, yet all have merits.

- CPM enables managers to determine the least costly way of reducing project duration to complete the 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.
- The critical chain method, CCPM, based on the theory of constraints (TOC), 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 without warning, but buffers in CCPM provide relative stability to the critical chain—the path connecting activities that require the same constrained resource. CCPM offers a practical and relatively simple way to schedule projects, but it requires a shift in human behavior since only the project manager and nobody else is required to commit 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.

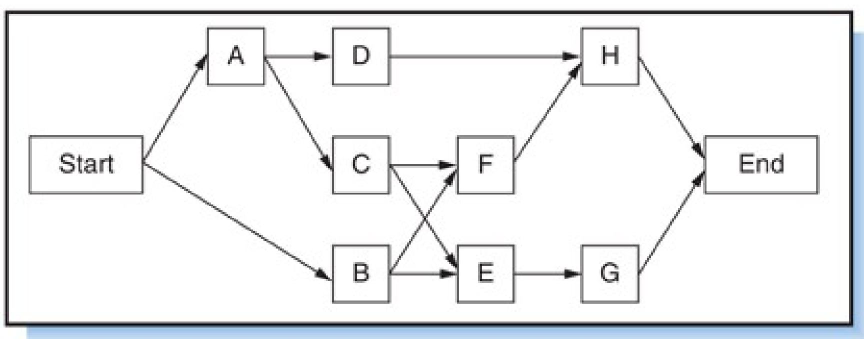
- All the above methods are supported by commercial software, which simplify their application and eliminate computational difficulties. But as with all management methods, appropriate application of the techniques assumes a sound understanding of the principles that underlie them and management acceptance.

Summary List of Symbols

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 would 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 T_n	Normal C_n	Crash T_c	Crash C_c	Cost Slope
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

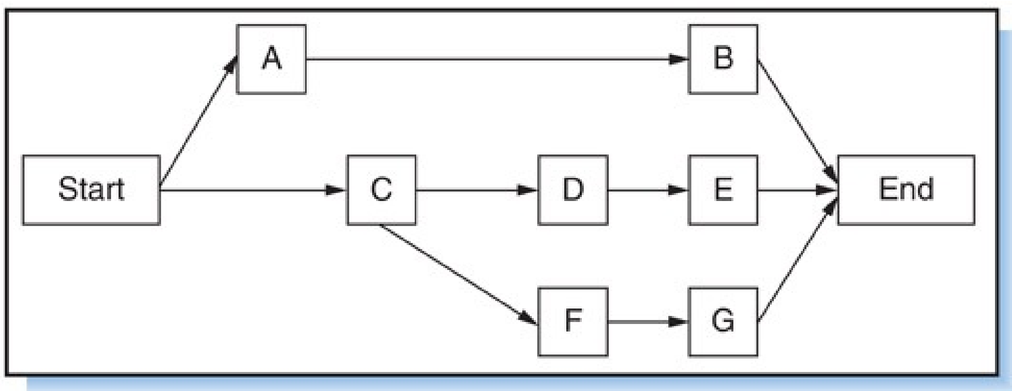
a. Verify that the normal duration is 22 days and that the direct cost is $3,050.

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	

a. What is the earliest the project can be completed under normal conditions? What is the direct cost?

b. What is the least costly way to reduce the project duration by 2 days? What is the project cost?

c. 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

a. 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?
b. 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 finds that the delivery time for a specific item is never delivered in less than 5 days. The worst case scenario is that it takes 30 days for the item to arrive. A delivery lead time of 10 days is the most frequent.

a. Calculate the expected delivery time
b. What estimate would you give for its variance?
c. What factors would you take into account when deciding the amount of time to allow of the item for 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.

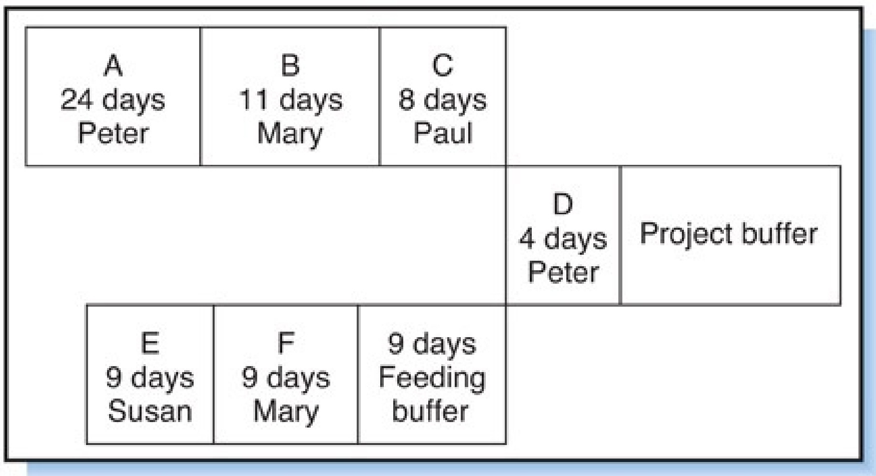
a. What is $P(T_e < 23)$?
b. What is $P(T_e < 32)$?
c. For what T_s is the probability 95 percent that the project will be completed?

14. For the network in Figure 7.12, what is the probability of completing each of the five paths within 30 days? What is the probability of completing them *all* 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 has to do the two tasks, indicate two possible critical chains.
b. Reschedule the work and indicate 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 6.

See Chapter 6

a. Indicate the critical chain on the diagram.
b. Assume the schedule uses durations from which any contingency (padding) has been removed. Insert a project buffer and feeding buffers as required.

19. Refer to Figure 7.22. Scheduling Activity T before Activity Q would also have been a way to resolve the resource contingency. Explain why this

alternative was not selected.

20. Consider the data about project activities given in the table below.

    a. 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.
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. Revisit Question 18:

    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 7.6) to the problem 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 put 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 which 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 7.1 Bridgecon Contractors

Bridgecon Construction Company specializes in the detailed, design, and construction of steel and concrete bridges. The first phase of the company's project management methodology, Initiation, includes identification of project opportunities and assessment of each project's 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; this marked the end of the first phase.

The project next enters the second phase, Estimating, which includes site visits by the estimating team, review of available resources and skills, detailed risk assessment, and preparing a preliminary plan for detail design, procurement, logistics, and construction. The phase includes activities A and B in Table 7.6, which are necessary to prepare the bid for building the bridge. The phase concludes with a presentation to the customer, the rail authority.

The project manager and the estimating team meet with the architect and structural engineers who produced the concept design to acquaint themselves with the design. They then meet with subcontractors who they might choose to construct the pilings and fabricate steel components. Following these meetings the “Initial Duration Estimate” and “Initial Cost Estimate” are completed, as listed in Table 7.6.

The RFP for building the bridge says acceptance of the plan by the rail authority is one criterion for selecting a contractor. Early on it becomes evident that starting with activity D and until the completion of activity S, operation of one of the railway lines under the bridge will be impaired, although an informal discussion with the rail authority indicates that that might be acceptable. During a subsequent meeting, however, the rail authority expresses concern that the impairment will last 17 weeks and requests 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 is approached. This subcontractor says it will be able to halve the time of activity H for a total cost of $960,000.
-

  Two ways are 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 (and their equipment) could be temporarily reallocated from another project. Adding this team and temporary workers to the original team could lead to completing the work in 1 week.

Table 7.6 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	1320
N	Transport structural steel components and	1	M	433

	erect on-site			
P	Erect pylons and fi ll with concrete	2	J	840
Q	Construct main span deck on pre-cast concrete beams	3	H, K, N, P	2800
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	
				10621

The manager of the other project estimates that the reallocation would cause him to forfeit an incentive fee of $150,000 for finishing his project early. The managers of the two projects agree 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 can be reduced to 3 weeks but would increase the activity's cost to $730,000. It could be reduced to 2 weeks but would cost $820,000.

- The duration of activity Q can be reduced to 2 weeks, but the activity would cost $2,929,000.

Questions

1. Compile a list showing the reduced periods for impairment of 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 7.2 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 to complete it in that time, but nonetheless it discusses the possibility with Iron Butterfly Company's project manager, Frank Wesley. Reviewing the network diagram (below, Figure 7.26), Frank checks the feasibility of this and then asks his managers and technical staff to give him three time estimates for every activity in the project. The estimates are given below in Table 7.7.

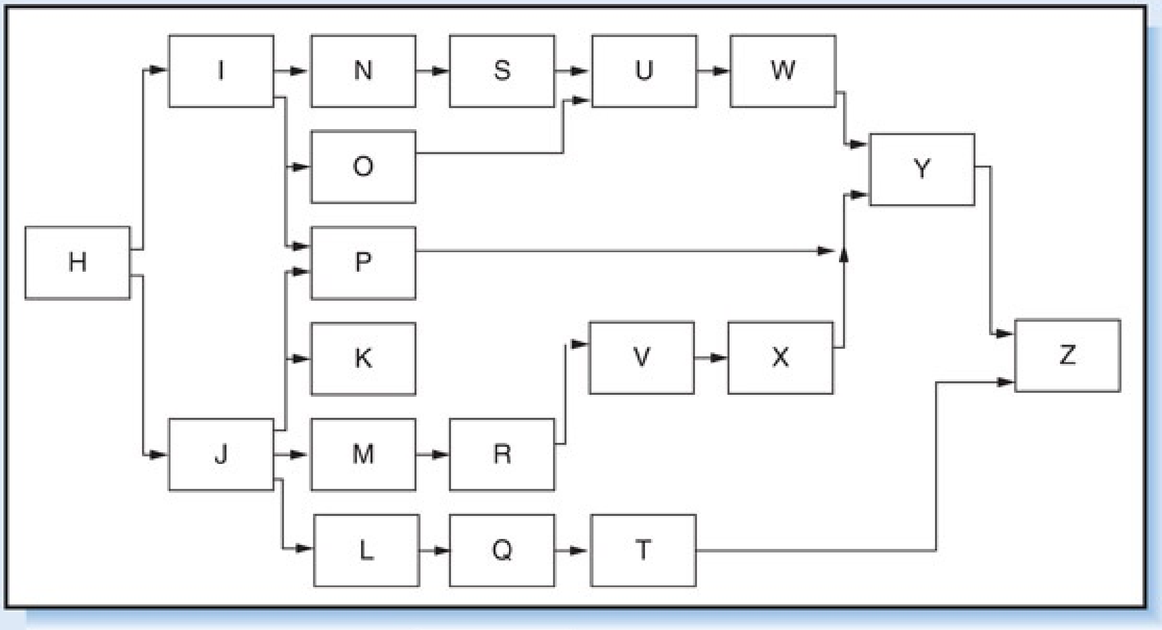


Figure 7.26 LOGON project.

Questions

1. What is the probability of finishing within 40 weeks?
2. Do you foresee any significant risk of a delay that the calculations for (1) above do not take into account?
3. Determine the most likely project duration.

Table 7.7 **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

Case 7.3 Papua Petera Village Project

The Papua Petroleum Company is building a village to support workers for an oil field in Sumatra. The principle work of the project involves five work packages: build road, clear site, erect buildings, erect power generation plant, and build water purification system. Figure 7.27 is a high-level network diagram for the 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 excludes costs for materials, supplies, components, and systems that are fixed regardless of work times.

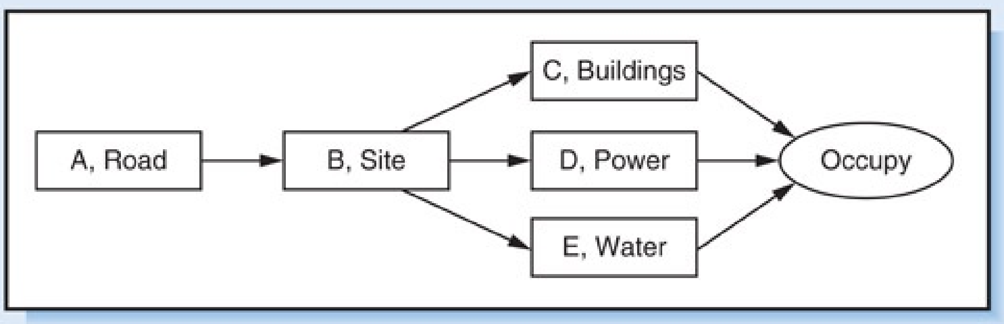


Figure 7.27 Papua Petera Village Project.

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 is able to 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. Additonally 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.

Endnotes

1. CPM 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.

2. 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.

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, no. 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* 5, no. 1, November 1974; and Vazsonyi A. L'Historie de la grandeur et de la decadence de la methode PERT. *Management Science* 16, no. 8, April 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, no. 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. Crystal Ball is a registered trademark of Decisioneering, Inc.; for information see decisioneering.com. RiskSim is a trademark of Treeplan.com; see www.treeplan.com. For @Risk, see www.palisade.com. For Arena, see www.rockwellautomation.com. For Simul8, see www.simul8.com

10. Viljoen P. Goldratt Schools, Personal communication, Pretoria, SA, May 2007, May 2010, and April 2015.

11. Goldratt E. *What is this Thing Called Theory of Constraints and How Should it be Implemented?* New York: North River Press, Inc. 1990.

12. Pittman P. *Project Management: A More Effective Methodology for the Planning and Control of Projects.* Georgia: PhD dissertation, University of Georgia; 1994; Goldratt E.M. *Critical Chain.* Great Barrington, MA: North River Press; 1997.

13. Walker E. *Planning and Controlling Multiple, Simultaneous, Independent Projects in a Resource Constrained Environment,* Georgia: PhD dissertation, University of Georgia; 1998. A TOC method for allocating resources to multiple projects was developed in this study and subsequently has been developed further.

14. Goldratt E. *Critical Chain.* Great Barrington, MA: North River Press; 1997, p. 156.

15. Herroelen W. and Leus R. On the merits and pitfalls of critical chain scheduling. *Journal of Operations Management* 2001; (7): 559–577; 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*; 2008; 19(1): 73–88.

16. Newbold R. *Project Management in the Fast Lane – Applying the Theory of Constraints.* New York: St. Lucie Press; 1988.

17. Tukel I., Rom W. and Eksioglu S.D. An investigation of buffer sizing techniques in critical chain scheduling. *European Journal of Operational Research* 2006; 172: 401–416; also see Trietsch D. The effect of systemic errors on optimal project buffers. *International Journal of Project Management*, 2005; 23: 267–274; Shou Y. and Yeo K. Estimation of project buffers in Critical Chain Project Management. *Proceedings of the IEEE international conference on management of innovation and technology*, 2000: 162–167.

18. Goldratt, *Critical Chain.*

19. For Prochain, see www.prochain.com; for Sciforma, see www.sciforma.com; for Concerto, see www.realization.com.

20. Viljoen P. Goldratt Schools, Personal communication, Pretoria, SA, May 2007, May 2010, and April 2015.

21. Turner J.R. *The Handbook of Project-Based Management.* London, UK: McGraw-Hill; 1993.

22. 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.

23. Adapted from training material of Realization. See www.realization.com.

24. Dass S. and Steyn H. An exploratory assessment of project duration in multiple-project schedules where re-sources are allocated by the theory of constraints method. *SA Journal of Industrial Engineering,* 2006; 17(1): 39–54.

25. 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* 2004; 35(2): 39–50.

Chapter 8
Cost Estimating and Budgeting

Cost estimates, budgets, WBSs, schedules, quality, and risk are interrelated. Ideally, cost estimates are based upon elements of the WBS and are prepared for each work package. When the cost cannot be estimated because an activity is too complex, the activity is broken down further until it can. When the work is ill-defined or uncertain, the estimate is initially based upon judgment and is later revised as information becomes available. Project schedules dictate the need for resources and the rate of expenditures, but the converse is also true: constraints on resources and working capital dictate the schedules. Imposing practical constraints on costs is necessary to create realistic project budgets; failing to do so results in projects being completed at exorbitant expense, or terminated prematurely due to lack of funds. Both occurrences are relatively commonplace.

Cost estimating, budgeting, and control sometimes are thought to be the exclusive concerns of cost specialists, planners, and accountants, but in projects they should be everyone's concern. Project participants who are closest to the sources of costs—engineers, scientists, systems specialists, architects, or others—should be involved in the estimating and budgeting process. Commonly, however, these same people are disdainful of budgets and ignorant about how they work or why they are necessary.

The project manager, of course, must also be involved. Although she does not need to be a financial wizard, she does 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 most technical projects the *cost engineer* reviews the deliverables and requirements, assesses the project from both cost and technical points of view, and advises the project manager. Cost engineering is discussed later.

8.1 Cost Estimates

The cost estimate can seal the project's financial fate. When project costs are overestimated, the contractor risks losing the job to a lower bidding competitor. Worse is when they are underestimated. A $50,000 fixed price bid might win the contract, but obviously the contractor will lose money if the project ends up costing $80,000. 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, relatively 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 soon afterward.

But a very low bid can also signify that the contractor cut corners in the estimate, forgot to include things, or was just sloppy. The consequences for both client and contractor can be disastrous, ranging from suffering a loss to bankruptcy. Cost estimates are used to develop budgets. After the project begins actual costs are compared to estimated, budgeted costs as one measure of the project's performance. Without good estimates it is impossible to evaluate work efficiency or to determine in advance how much the project will cost on completion.

8.2 Cost Escalation

Estimating costs accurately can be difficult because the estimation process begins in project conception and before much is known about the project. The less well-defined the project, the greater the chances that actual costs will substantially differ from estimated 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*. Some escalation is normal and up to 20 percent is common. Usually, the larger and more complex the project, the greater the potential for escalation. The costs of cutting-edge technology and research projects frequently escalate upwards of several hundred percent. The Concorde supersonic airliner exceeded the original estimate by a factor of five, nuclear power plants often exceed estimates by a factor of two or three, and NASA spacecraft sometimes by a factor of four to five.

Figure 8.1 shows a plot of percent cost overrun versus year of decision to build for 111 transportation-related projects spanning approximately 80 years.[1] The study reporting these findings also looked at 246 other projects and got a similar picture. Clearly, overruns have been and remain common. How does that happen? There are many reasons.

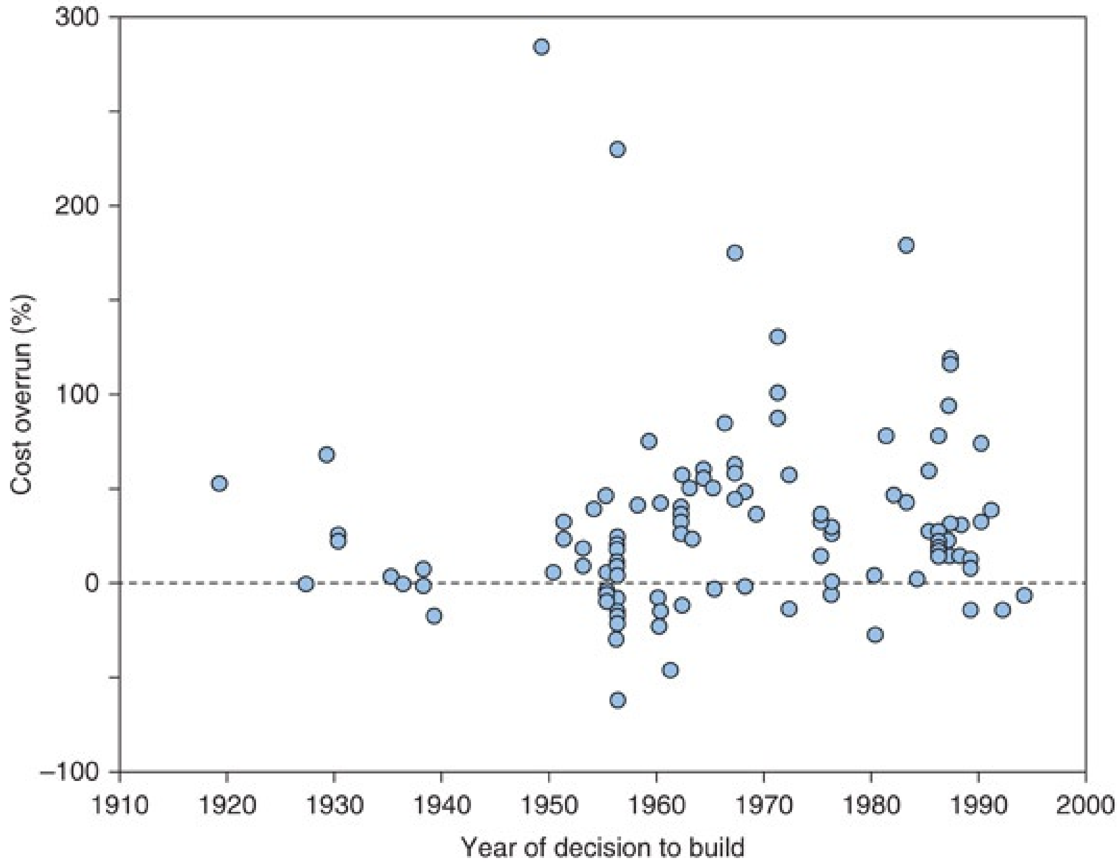


Figure 8.1 Projects versus percent cost overrun.

Uncertainty and Lack of Accurate Information

Much of the information needed for accurate estimates is simply not available when costs are first estimated. 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, when the design is finalized and work activities are well-defined (during the definition phase) can material and labor costs be accurately determined. In most research and development projects the activities are unpredictable, of uncertain duration, or must be repeated. Nonetheless, management should strive for the clearest, most definitive scope of work, project objectives, and requirements, because without them, obtaining accurate cost estimates is near impossible.

During the project, whenever change to product designs or project schedules are necessary due to product concept changes, developmental barriers, strikes, legal entanglements, or skyrocketing wages and material costs, the project cost estimate should also be updated so as to serve as a valid baseline for tracking and controlling project costs.

To allow for uncertainty, an amount called a *contingency fund* or *budget reserve* is added to the original estimate.[2] This is the budget equivalent of the *schedule reserve* or *buffer* mentioned in previous chapters. The contingency amount is proportional to the uncertainty of the work; 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 (discussed later in this chapter) is intended primarily to offset small overruns arising from estimating errors, omissions, and minor design changes and schedule slippages. 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 will tend to rise to expend whatever remains in the fund. Contingencies are discussed later.

Changes in Requirements or Design

Cost escalation also occurs due to scope creep—discretionary, nonessential changes made to system requirements and plans. These changes come from *a change in mind*, not from oversights, mistakes, or mandates that would make them imperative. The routine tendency is for users and contractors alike to want to modify systems and procedures—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 to comply. Any change, however, no matter how small, causes escalation; it usually involves a combination of redesign or 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. Formal *change control* procedures to reduce the number of changes and contain escalation are described in Chapter 11.

See Chapter 11

Economic and Social Factors

Even with good initial estimates and few changes, cost escalation occurs because of social and economic 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 earned is delayed. Rarely can such problems be anticipated and their impacts incorporated into the contingency fund.

One economic factor that influences cost escalation is *inflation*. The contractor might offset this factor by inflating the price of the project, although ability to do that is often precluded by the actions of competitors or federal restrictions on price increases. Some protection from inflation may be gained by including 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 protection 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 are reimbursed for those effects until later.

Cost estimates are typically based upon prices at the time of estimating. Thus, whenever inflation rates become known the estimates should be adjusted so as to provide a valid baseline from which to later identify cost variances and take corrective action. In long-term projects, future wage rates should be forecasted; this is done by starting with estimated wage costs in current dollars and then applying inflation rates over the project's duration.

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, resulting in a cost or price escalation. This topic is discussed in Chapter 19.

See Chapter 19

Inefficiency, Poor Communication, and Lack of Control

Cost escalation also results from work inefficiency, poor management and planning, poor communication, lack of supervision, 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, tracking and monitoring of activities and tight control improve efficiency and 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 where they have little experience. 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 the estimate, they "pad" it. The more the estimator's ego is involved in the job, the less reliable the estimate.

Companies avoid the problem by employing professional cost estimators; these are not the same people who will do the work. Remember the earlier contention about involving project participants in planning the project? Experienced workers are usually better at estimating time and materials than they are costs. So the doers (those who do the work) should define the work and estimate the needed resources and time, but the professionals should estimate the cost. People often confuse the 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 honest prediction of what it *will* take. A cost estimate should never be based upon a previously set target or goal; thus, estimators must be positioned organizationally so they will not be coerced into providing the numbers someone wants.[4]

Project Contract

Chapter 3 describes the relative merits of different forms of contracts; at least two of these, fixed-price and cost-plus, are relevant to project cost escalation. A fixed-price agreement incents the contractor to control costs so as to keep the accrued costs below the contracted price. In contrast, a strictly cost-plus contract offers little incentive to control costs. In fact, when profit is computed as a percentage of costs (rare these days), the contractor is motivated to "allow" costs to escalate and tack on all kinds of fees. Other forms of agreements such as incentive contracts permit cost increases, but encourage cost control and provide motivation to minimize escalation.

See Chapter 3

Information and Assumptions

Information and assumptions from which estimates are made should always be cross-checked. Are the assumptions of the customer and contractor regarding costs correct? Does everyone agree on the work, materials, and other elements to be 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? Like the scope statement (and perhaps in reference to it), the cost estimate should explicitly identify all the cost elements used to produce the estimate.

Bias and Ambition[5]

Finally, it is human nature for the champions of projects to be optimistic about their projects. In fact, without champions most 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 know that if a project is important enough, sufficient funding to complete it will materialize, no matter the size of the overrun.

Example 8.1: Escalation of the Bandra-Worli Sea Link Project

> 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 headlines from local news media refer to the Bandra-Worli Sea Link (BWSL) roadway and cable-stayed bridge in Mumbai—India's equivalent to San Francisco's Golden Gate Bridge and a good example of megaproject woes. The 8-km bridge and its approaches bend 200 meters into the Arabian Sea to connect downtown Mumbai with its western suburbs. The bridge reduced travel time by half, to about 30 minutes.

The project was approved in early 1999 following 7 years of study; it was supposed to start in May, cost 650 crore (US $120M), and finish by mid 2001. But work did not begin until December, and already the estimated completion date had slipped to mid 2002. Then came the monsoons, which brought the project to a near halt in 2000 and 2001. 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 came the announcement that project costs had increased by 50 crores; 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, 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. By June 2005 the completion date was reset to September 2006 and the project cost to 1,306 crore. 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 cost: 1,600 crore (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 150 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).

8.3 Cost Estimating and the Systems Development Cycle[6]

Project cost estimating happens throughout all phases of the project life cycle.

The first estimate is made during project conception. Since very little hard cost information is available at that time, the estimate is the least reliable that it will ever be. Uncertainty about the project cost and duration may be large, as illustrated by the largest "region of time–cost uncertainty" in Figure 8.2. How much the project will *really* cost and how long it will *really* take are very much open to question. The project is compared to other, similar projects, and an estimate is made based upon standards of what it should take—labor time, materials, and equipment—to do the job. When there are no similar projects or 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 cost estimates often dictates that contracts be of the cost-plus kind. As more aspects of the system and project are defined, the material costs, labor times, and labor rates can be nailed down, and cost estimates become more reliable. When the system and project are fairly routine, the estimates are somewhat more reliable and contractors are willing to accept incentive-type or fixed-price contracts. In fact, the awarding of contracts is sometimes put off until designs are somewhat complete and more accurate cost 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. Contractors required to bid before they can attain reliable estimates must include a contingency amount in the estimate to cover the uncertainty.

As the project moves into the middle and later phases, when work is being completed and funds expended, cost estimates become more certain. The shrinking time–cost uncertainty regions in Figure 8.2 illustrate this. As the uncertainty decreases, the amount in the contingency fund is reduced. A contingency starting out at 15 percent of the base project estimate might be decreased halfway through the project to 8 percent, then to 3 percent at the three-fourths mark, and then to 1 percent at final installation to cover minor corrections at sign-off.

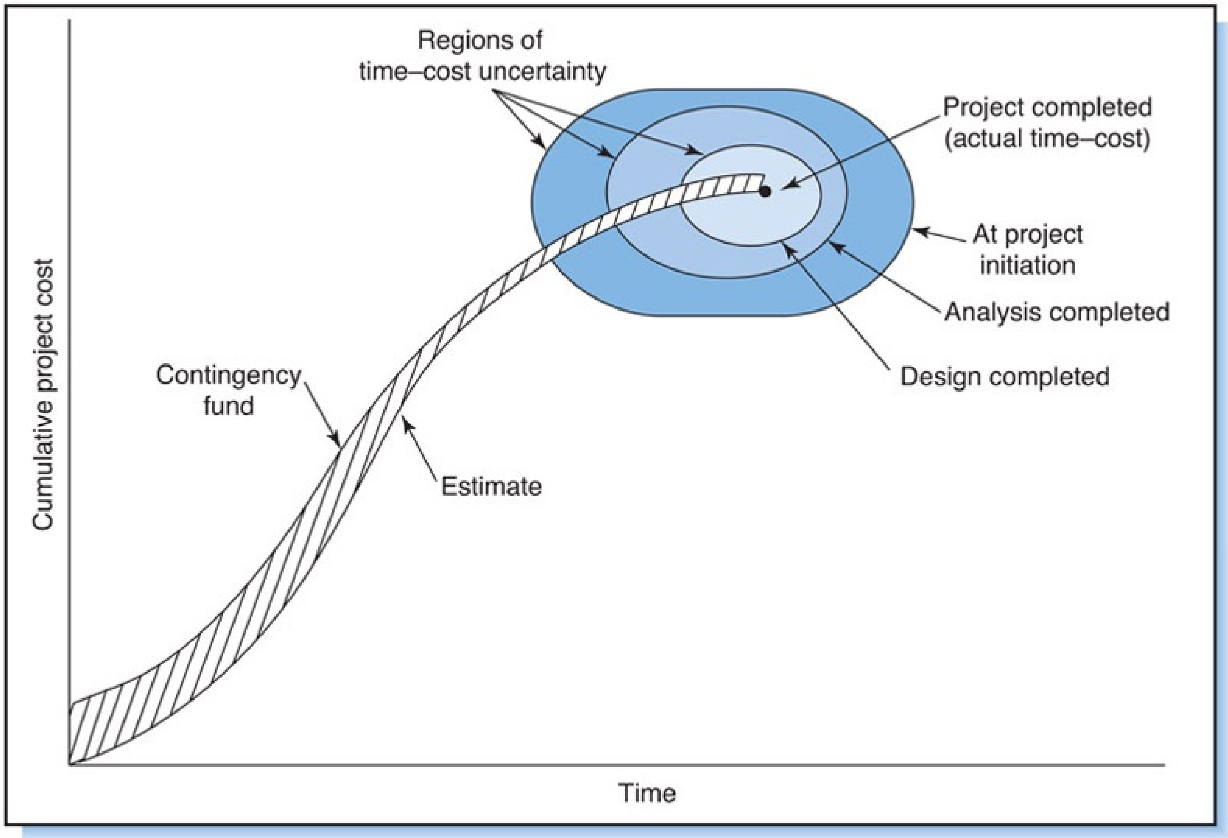


Figure 8.2 Time–cost graph showing cumulative project cost and regions of time–cost uncertainty.

As discussed in Chapter 4, usually a detailed plan is developed only for the most-immediate, upcoming phase of the project (phased or rolling wave plan), and that plan will include a cost estimate and cost commitments for the phase. At the same time, every attempt is made to look beyond that phase and to develop a realistic cost estimate for the entire project.

See Chapter 4

Once developed and approved, the estimate for the project becomes the budget and the baseline against which project progress and performance will be measured. It is thus bad practice to keep changing the estimate during the project because that destroys the purpose of a baseline—to measure progress and control costs. Sometimes, however, escalation factors render the baseline estimate obsolete and mandate periodic revisions.

8.4 Cost Estimating Process

Estimate versus Target or Goal

Sometimes the word "estimate" is confused with "target" and "goal." It shouldn't be. Whereas an estimate is an attempted *realistic assessment* based upon known facts about the work, required resources, constraints, and the environment, a target or goal is a *desired* outcome. Other than by chance, the estimate will *not* be the same as the target or goal. That said, once computed the estimate can be compared to a target value or goal, and the work activities, resources, and schedules revised to bring the estimate closer to the target. Never should the estimate be a simple plug-in of the target value.

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 cost $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 *neither is accurate* 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 so-called *three-point estimate,* which combines optimistic (a), pessimistic (b), and most likely (m) cost estimates to arrive at an expected cost estimate—analogous to the PERT approach for computing expected time:

$$C_E = \frac{a + 4m + b}{6}$$

Classifying Work Activities and Costs

The cost estimating process begins by breaking the project down into work phases 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 phases or packages and receive specific assignments.

The team looks for tasks in the project that are similar to existing designs and standard practices and can readily be adopted. Work is classified as *developmental* or as an adaptation of existing, *off-the-shelf* (OTS) designs, techniques, or procedures. Developmental work involves uncertainty in design, testing, and fabrication, so cost estimating is more difficult compared to estimating for OTS items or duplicated work, which is straightforward and uses known prices or records of material and labor costs for similar systems or activities. Overruns for developmental work are common, especially due to inaccurate labor estimates. It is thus often beneficial to make use of existing designs and technology as much as possible.

Estimated costs are classified as *recurring* and *nonrecurring*. Recurring costs happen more than once and are associated with activities periodically repeated 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.

In a pure project form of organization the project manager delegates responsibility for estimating to others, combines their estimates, and presents the final figures to management. In a matrix organization, estimating is the joint responsibility of the project and functional managers; the project manager coordinates the effort and accumulates results.

Although this typifies the estimating process, the actual approach will depend on the information available and the required accuracy of the estimate. Most estimates are made using variants of four methods: expert judgment, analogy, parametric, and cost engineering.

Expert Judgment

An *expert judgment* is an estimate provided by an expert—someone who from breadth of experience or expertise is able to provide a ballpark estimate. It is a "seat of the pants" estimate used whenever lack of information precludes more-rigorous cost analysis. Expert opinion 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.

Analogous Estimate

An *analogous estimate* is developed by reviewing costs from previous, similar projects. Often such an estimate is handy as a relatively fast reality check for estimates derived from detailed planning. The method can be used at any level: overall project cost can be estimated from the cost of an analogous project; work package cost can be estimated from analogous work packages; and activity cost can be estimated from analogous activities. The cost for a similar project or work package is 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 analogy project cost 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 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 costs from the analogy must be "scaled" up. But twice the size does not mean twice the cost, and the size-cost relationship must be uniquely determined.

Example 8.2: 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:

Cost (proposed) = Cost (analogy)[Capacity (proposed)/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.[7]

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 extant information about prior projects. Companies that are serious about using the method gather cost documentation and retain it on a database that classifies costs according to type of project, work package, activity, and so on. When a new project is proposed, the database is accessed for cost 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 8.3: A Case of Costly Mistaken Analogy[8]

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 actually 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 shut down until the leak is fixed. In a nuclear plant the water cannot be turned off or the plant shut down; the reactor continues to generate heat and if not cooled will melt, cause pipes to rupture, and radiation to disperse. 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

A *parametric estimate* is derived from an empirical, mathematical relationship. The method can be used with an analogy project (case in Example 8.3) to scale costs up or down, or it can be applied directly without an analogy project when 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.

Example 8.4: 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) 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. A bottom-up approach, it provides the most accurate estimates of all the methods, but it is also the most time-consuming. The method requires detailed work-definition information that, often, is not available early in the project. It first divides the project into activities or work packages (e.g., from the WBS), then divides each of these into cost categories. For small projects like Example 8.5 the approach is simple and straightforward.

Example 8.5: Cost Engineering Estimate for a Small Project

The project manager for the DMB project is preparing a project cost estimate. He begins by breaking the project into eight work packages and creating a simple schedule. Three labor grades will be working on the project; for each work package he estimates the needed number of labor hours per week for each grade. Hours per week per labor grade are represented inside the shaded boxes in Figure 8.3.

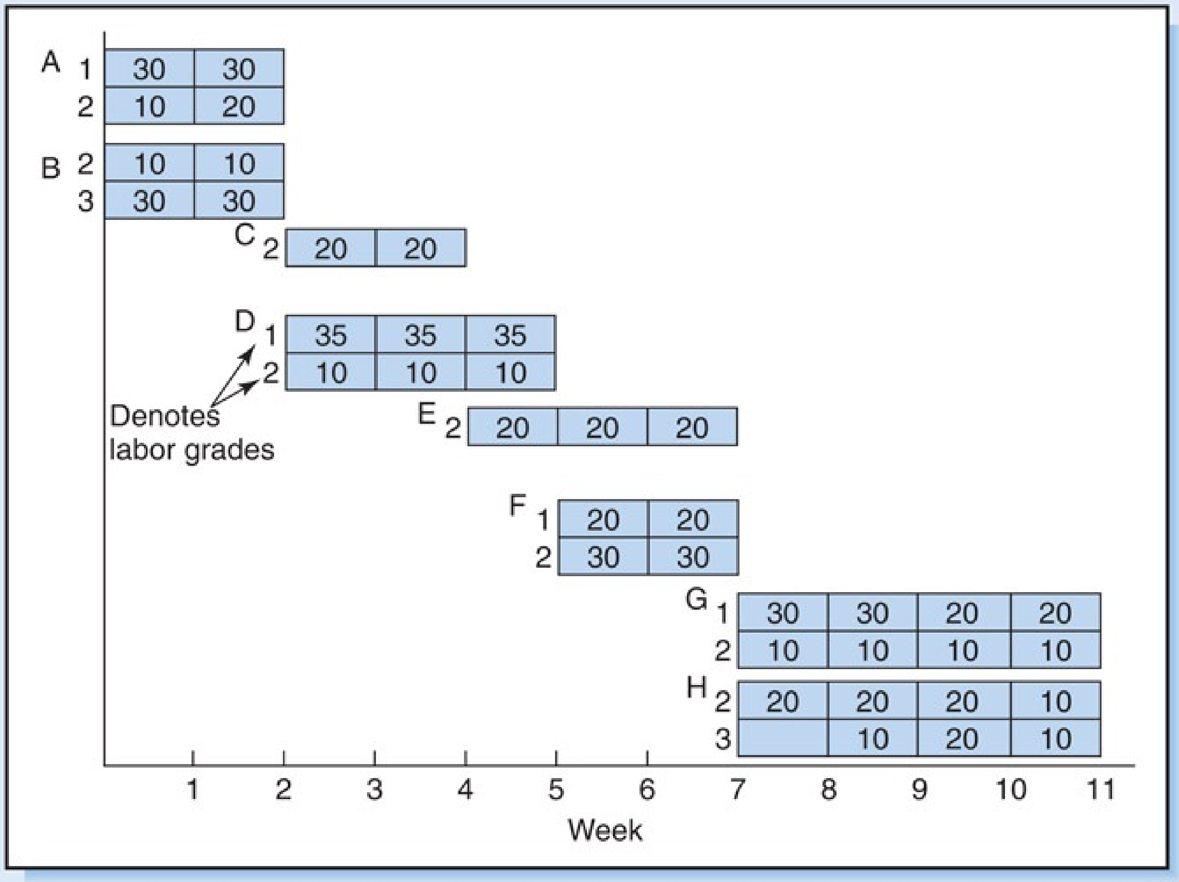


Figure 8.3 Schedule showing hours allocated to work packages by labor grade.

For each work package he also estimates the cost of material, equipment, supplies, subcontracting, freight charges, travel, and other non-labor costs. Table 8.1 summarizes the labor hours and non-labor costs. The sum of the non-labor costs is $26,500.

Table 8.1 Labor Hours and Non-Labor Costs

Work Package	Hours By Labor Grade 1	2	3	Non-Labor Costs 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

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 (the overhead amount is *added* to the labor cost; determining overhead rates is discussed later). Therefore, labor-related costs are:

Grade 1: 305($10)(100% + 90%)	=	$5,795
Grade 2: 350($12)(100% + 100%)	=	$8,400
Grade 3: 100($15)(100% + 120%)	=	$3,300
	$	17,495

The preliminary estimate for labor and non-labor cost is $17,495 + $26,500 = $43,995. Suppose the company routinely adds 10 percent to all projects to cover general and administrative expenses; this puts the cost at $43,995(1.1) = $48,395. If the contingency amount is also 10 percent, the total cost estimate for the DMB project is $48,395(1.1) = $53,235.

At the work package or lower level, detailed estimates are sometimes derived with the aid of *standards manuals* and *tables.* Standards manuals contain time and cost information about labor and materials to perform 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 an estimate of the required number of boxes, times the labor standard per box, and then multiplies that by the hourly labor rate. For software development the industry standard is 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 Example 8.5 although more involved. First, the manager of each work package breaks the work package down into "basic" areas of work such as "engineering" and "fabrication." Supervisors in each basic area then estimate the hours and materials needed to do the work. The engineering supervisor might further divide work into the tasks of structural analysis, computer analysis, layout and installation drawings, and manuals, then for each task develop an estimate of the duration and the labor grade or skill level involved. Similarly, the fabrication supervisor might subdivide the work by 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, reference documents, and rules of thumb ("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 time and material estimates to be sure that no costs were overlooked or duplicated, everyone understood what they were estimating, correct estimating methods were used, and allowances were made for risk and uncertainty.[9] The estimates are then aggregated as shown in Figure 8.4 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 a company-wide overhead (to cover general company expenses) to arrive at a cost estimate for the total project. The accumulation of work package estimates (upward arrows in Figure 8.4) to derive the project estimate is called the "bottom-up" approach.

Contingency Amount

Contingency amounts are added to estimates to offset uncertainty. As mentioned, 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 the project as a whole. *Activity contingency* is an amount estimated to account for "known unknowns" in an activity or work package, i.e., sources of cost increases that could or likely will occur; they include scrap and waste, design changes, increases in the scope, size, or function of the end-item, and delays due to weather. Later, when the project budget is established, this 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 total project cost, the result is the *base estimate* for the project cost.

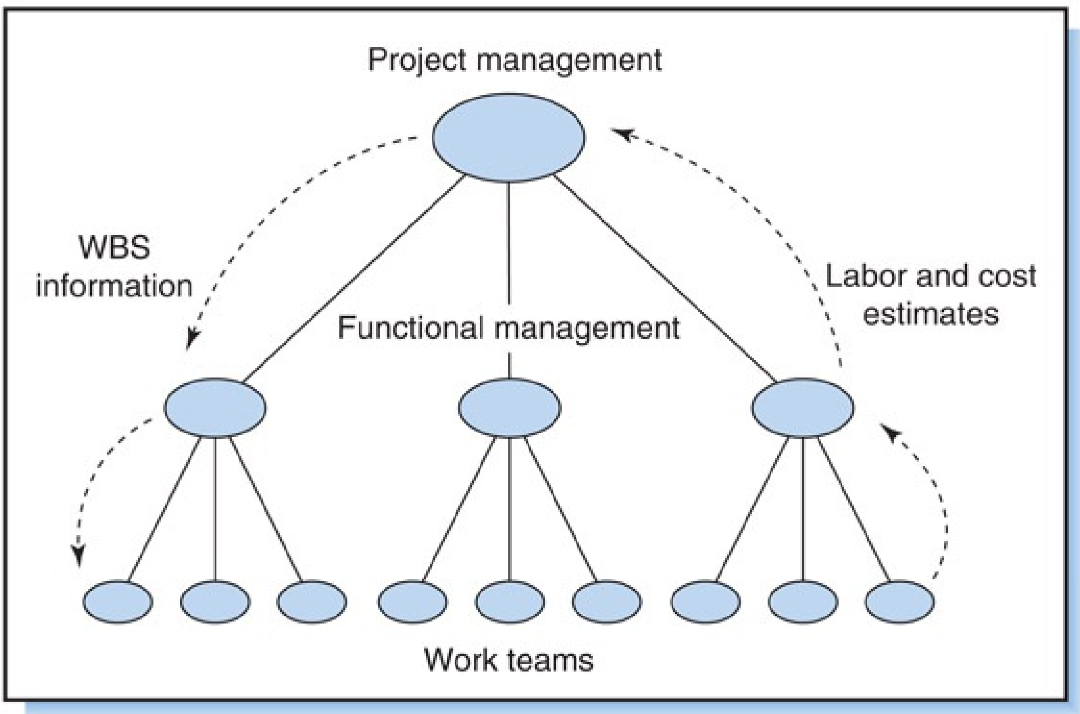


Figure 8.4 The estimating process.

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 the contingency; in larger projects, the program or senior management does. Adding the project contingency to the base estimate gives the *final cost estimate.* This is the *most likely cost* for the project.

Besides the activity and project contingencies, the *corporation* might set aside an allowance to cover overruns. This amount, the *overrun allowance,* is added to the most likely cost to yield a cost where, as a rule, the probability of exceeding it is less than 10 percent. The overrun allowance is controlled by program or corporate managers and ordinarily is not available to the project manager without approval.

Top-Down versus Bottom-Up

In general, estimating can be done in one of two ways: top-down and bottom-up. Top-down refers to estimating cost 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 cost by looking at elements of the project—individual work packages and end-item components. Costs for each work package or end-item element are estimated separately and then aggregated to derive the total project cost. Example 8.5 is a bottom-up approach; Example 8.2 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 estimates than the top-down method but is more time-consuming and requires more data.

Use of top-down or bottom-up corresponds somewhat to the project life cycle. In project initiation, the cost estimate might consist of no more than a top-down "ballpark" number to suggest the order of magnitude of the project cost. The estimate gives the customer or contractor a rough idea of the size of the cost, but the method involves little effort and, consequently, the estimate is low accuracy; little confidence is attached to it.

As the proposal is being prepared in the conception phase, the cost estimate is often based upon the top-down method of looking at previous but analogous projects and compensating for differences between, and lessons learned from, them. The accuracy of the estimate depends upon the validity of the analogies and the ability to distinguish differences and areas of uncertainty.

In the definition phase (and sometimes also in conception, depending upon when reliable data is available), the cost estimate is often prepared using the bottom-up approach. This method provides a fairly accurate estimate as well as the cost figures necessary to establish the project budget and control accounts, discussed later.

Reconciling Estimates

The project manager submits the final cost estimate to company management along with forecasts showing the effects of potential escalation factors such as inflation and risks. Management compares the estimate against the *gross estimate*, the goal or target set by management or the customer, and either accepts it or mandates a revision. If the gross estimate is larger, the project manager reviews the work package estimates for possible oversights or over-optimism. If the final estimate is larger, the project manager reviews the work package estimates for incorrect assumptions, padding, and other sources of excess cost.

Reducing Costs

What happens if the final cost estimate must be reduced? Everyone in the project will want to retain their share of the budget and will resist funding or staff reductions. Non-managers especially (engineers, scientists, systems analysts), often unaware of the constraints, will resist cuts. Through diplomacy and negotiation, the project manager tries to convince everyone to look for ways to *reduce costs*. Failing that, she must convince them to accept any *imposed* reductions.

To reconcile differences between gross and final estimates, managers sometimes exercise an across-the-board cut on *all* estimates. This is poor practice because it fails to account for judgmental errors or excessive costs on the part of just a few units. It also unfairly penalizes managers who tried to produce fair estimates and were honest enough not to pad them. Such indiscriminate, across-the-board cuts induce everyone later to pad estimates for their own protection.

Suppose you are the project manager and it is clear that top management's target budget is simply too small to do the project. There are two courses of action: either undertake the project and attempt wholeheartedly to meet the budget, or hand it over to another manager. If you decide on the former, you should document and report your disagreement to top management; later, ways might be found to reduce costs and complete the project within budget. If the contract is cost-plus, the risk is low since additional costs will be reimbursed. If the contract is fixed-price and the budget is so underfunded as to likely require cutting corners or stalling the project, then you should suggest the project be cancelled or that another person be appointed project manager. Not only is this good business practice, it is the ethical thing to do.

8.5 Elements of Estimates and Budgets

Budgets and cost estimates both state the cost of doing something. The difference is that the estimate comes first and is the basis for the budget. An estimate might have to be refined many times, but once approved it becomes the budget and the amount for which the organization and work units then commit to 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 fiscal operating budgets are similar; the difference is that the former covers the 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 Expense[10]

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 needed in each labor grade, and the number of hours or days for each. 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 8.5 shows the wage rates for three labor grades and the associated labor hours, time-phased over a 6-month period.

When the wage rate is expected to change over the course of the work, a weighted average wage rate is used. In Figure 8.5, suppose the initial rate for an assistant is expected to rise from \$20 to \$25 in months 3, 4, and 5. Instead of \$8,000, the labor cost for an assistant would be 100(\$20) + 100(\$25) + 100(\$25) +100(\$25) = \$9,500. The average wage rate would thus be \$9,500/400 hours =

$23.75/hour.

Project: CASTLE — Date: April 1, 1592

Department: Excavating — Work package: Moat

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								
Assistant	$20/hour		100	100	100	100		400	8,000
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 non-labor direct costs.

Figure 8.5 Typical 6-month budget for a work package.

Direct Non-Labor Expense

Direct non-labor expense is the total expense of non-labor charges applied directly to the activity. It includes subcontractors, consultants, travel, telephone, computer time, material costs, purchased parts, and freight. This expense is represented in Figure 8.5 by the line "other direct cost." Material costs include allotments for waste and spoilage and should reflect anticipated price increases. Material costs and freight charges sometimes appear as separate line items called *direct materials* and *overhead on materials,* respectively; computer time and consultants 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 incurred only for a specific project or work package. Not included are the 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 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 to apply is 180,000/150,000 = 1.20. Thus, for every $1.00 charged to direct labor, $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. The department's overhead cost is not allocated to projects 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 expenses, marketing and promotion, salaries and expenses of top management, and employee benefits. These costs might not be tied to any specific project, so they are allocated across all projects, to certain projects, or to parts of certain projects. For example, corporate-level overhead would be allocated across all projects, project management overhead on a per-project basis, and departmental overhead to specific project segments to which a department contributes. 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.

The actual manner in which indirect costs are apportioned varies in practice. The example for the SETI Company in Table 8.2 shows three methods for distributing indirect costs between two projects, MARS and PLUTO.[11] Notice that although company-wide expenses remain the same, the cost of each project differs depending on the method of allocating indirect costs.

Clients want to know the allocation method used by their contractors, and contractors should know the allocation method used by subcontractors. For example, Method I in Table 8.2 is good for the client 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 (somewhat labor intensive).

Table 8.2 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	40

		Indirect Total	160
Project Costs	**MARS Project**	**PLUTO Project**	**Total**
Direct labor (DL)	50	100	150
Direct nonlabor(DNL)	40	10	50
	90	110	200
		Direct Total	200
		Direct and Indirect Total	360

Some methods for apportioning indirect costs:

1. Total indirect proportionate to total direct costs

	MARS Project	PLUTO Project	Total
DL and DNL	90	110	200
OH and G&A	72	88	160
	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 (DLand DNL]	18	22	40
	148	212	360

III. OH proportionate to direct labor only; G&A proportionate to DL and OH and DNL

	MARS	PLUTO	Total
DLand OH and DNL	130	190	320
G&A	16.25	23.75	40
	146.25	213.75	360

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. These appear in the budget, as shown inFigure 8.5. 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.

Profit and Total Billing

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, determined, in part, by the kind of contract as discussed in the Appendix to Chapter 3. 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. They usually do not appear on budgets for lower-level work elements.

See Chapter 3

8.6 Project Cost Accounting Systems

A project might consist of hundreds or thousands of elements—workers, materials, and facilities, all of which must be estimated, budgeted, and controlled. To expedite the process, reduce confusion, and improve accuracy, you need a system to help compute estimates, create, store and process budgets, and track costs. 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). While the main focus of the PCAS is on project costs, the system also assists tracking and controlling schedules and work progress. When the PCAS is combined with other project planning, control, and reporting functions, the whole system is referred to as the *project management information system (PMIS).*

The PCAS is used throughout the project life cycle. During project conception and definition it accumulates work package costs estimates to produce the project cost estimate. This estimate later becomes the basis upon which the project and work package budgets are created.

During project execution, the PCAS accumulates, credits, and reports project and work package expenditures. It creates time-phased budgets (for example, Figure 8.5), which help managers monitor costs and verify that the work has been completed and charged. The system also enables budget revisions. The PCAS functions are summarized in Figure 8.6.

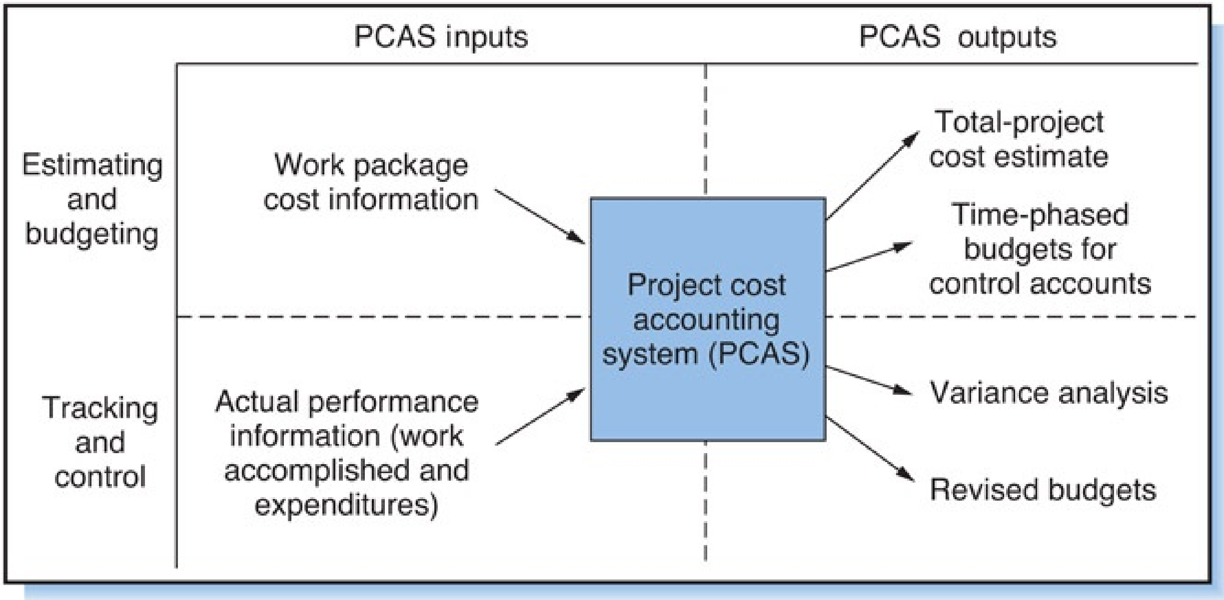


Figure 8.6 Elements of a project cost accounting system.

Example 8.6: A Pmis for Estimating Labor Requirements and Costs[12]

Sigma Associates is a large architectural/engineering firm that developed its own PMIS to assist in estimating, planning and scheduling.

At Sigma, the project manager begins each potential project by creating a WBS to identify the main work packages. Using a menu in the PMIS, she reviews the history of similar work 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 6).

See Chapter 6

The labor requirements estimate is given to the comptroller who, through the PMIS, 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 PMIS 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 high-quality standards but is more profitable.

Time-Phased Budgets

The project manager needs a way to track and control where expenses are accruing, how well the project is progressing, and where problems are developing. One way is with a *time-phased budget*—a method that consolidates the project budget and the project schedule to show the distribution of budgeted costs according to the project schedule. Figure 8.5 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, allowing the project manager throughout the duration of the project to compare budgeted costs and actual expenditures month-by-month.

For projects where a substantial amount of the costs originate from 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.

8.7 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 8.5, is used to compare actual costs with budgeted costs throughout the project.

On larger projects, however, a single, project-wide budget is too insensitive; once the project is underway and expenses accrue, it would be difficult to quickly locate the sources of cost overruns. 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 also are established for project costs not readily attributable to any specific work package. For example, for monies allocated for items, materials, or equipment that can be used by anyone or any work package, or for jobs such as administration, supervision, or inspection that apply across the whole project, separate control accounts are established. These accounts are usually set up for the duration of the project or are extended period-by-period as needed or as funds are appropriated.

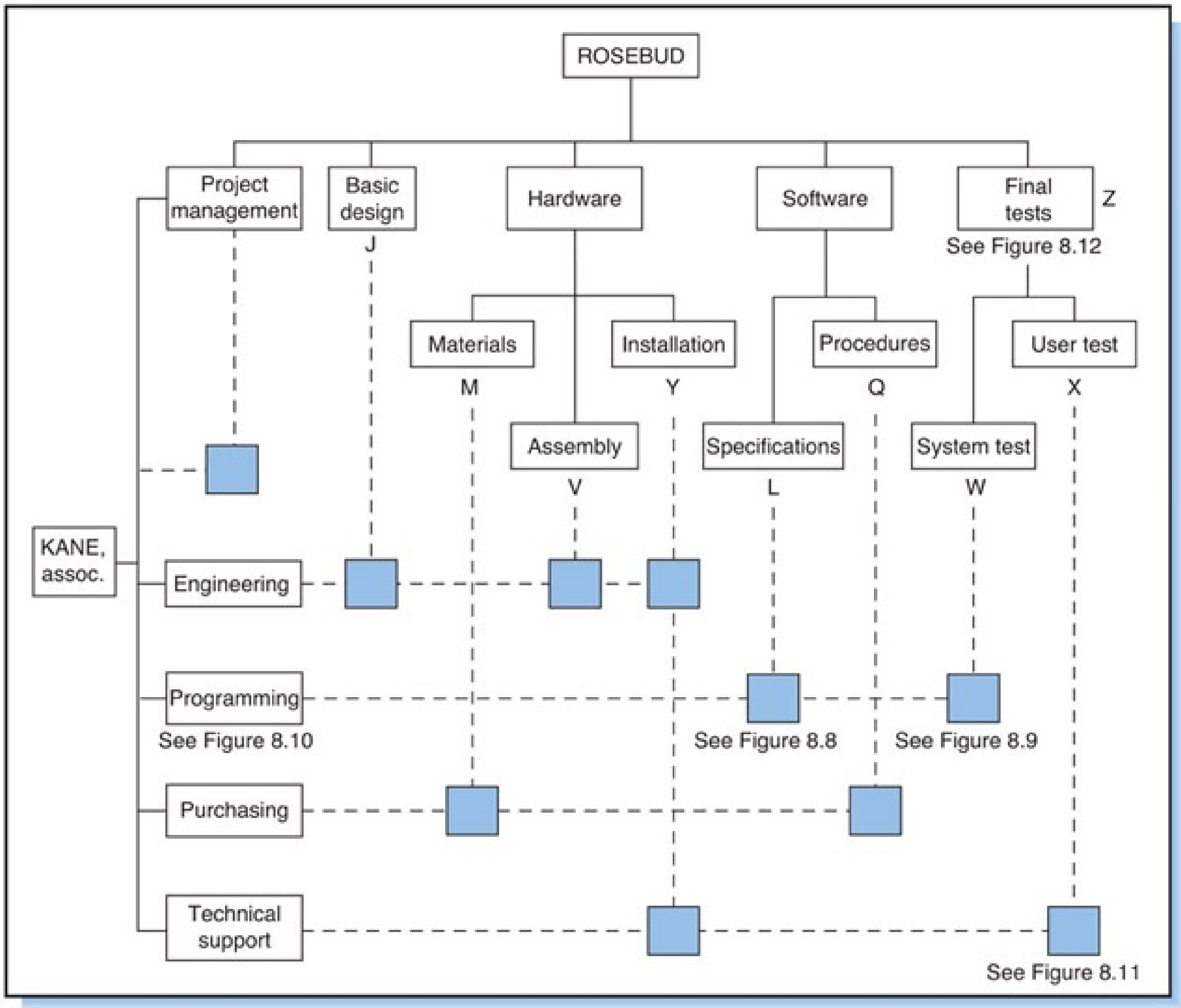


Figure 8.7 Integration of WBS and organization structure showing control accounts. (See Figures 8.8 through 8.12 for details.)

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. As an example consider the Robotics Self-Budgeting (ROSEBUD) project described in Chapter 6. Figure 8.7 shows the WBS for the project and the

See Chapter 6

organization chart for the contractor, KANE & Associates. The shaded boxes represent locations of control accounts; notice that each represents all or part of a work package for which a functional area is responsible. For the same project, Figures 8.8 and 8.9 show, respectively, the time-phased budget portions of the control accounts for the programming department's contributions to work packages L and W.

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 8.7, the ten shaded boxes represent departments/work packages for which initial cost estimates were made and, subsequently, budgets and control accounts were established.

8.8 Cost Summaries[13]

With the control account structure shown in Figure 8.7, 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. For example, consolidating accounts in Figure 8.7 horizontally results in a control account for each functional department. Figure 8.10 shows this with the time-phased budget for the programming department, which sums the budgets for work packages L and W (Figures 8.8 and 8.9).

Control accounts also can be consolidated vertically through the WBS. This would be useful for tracking and controlling individual work packages, clusters of work packages, or the project as a whole. Figure 8.12 is the time-phased budget for final tests, which is the sum of the budgets for work packages W (Figure 8.9) and X (Figure 8.11).

Project ROSEBUD Date ________

Department Programming Work package L—Software specifications

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 8.8 Budget for programming department for Work Package L.

Project ROSEBUD Date

Department Programming Work package W—System test

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 8.9 Budget for programming department for Work Package W.

Project: ROSEBUD   Date:

Department: Programming   Work package: ALL

Charge	Rate	Months+ 1	2	3	4	5	6	Totals 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 8.10 Budget for programming department.

Project: ROSEBUD   Date:

Department: Technical support   Work package: X—User test

Charge	Rate	Months+ 1	2	3	4	5	6	Totals 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 8.11 Budget for Work Package X.

Project: ROSEBUD

Date:

Department: Technical support; Programming

Work package: (W + X) Final Tests

Charge	Rate	Months+						Totals	
		1	2	3	4	5	6	Hours	Cost
Direct labor									
Professional	$35/hour						30	30	1,050
Associate	$30/hour						90	90	2,700
Assistant	$20/hour								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 8.12 Budget summary for final tests.

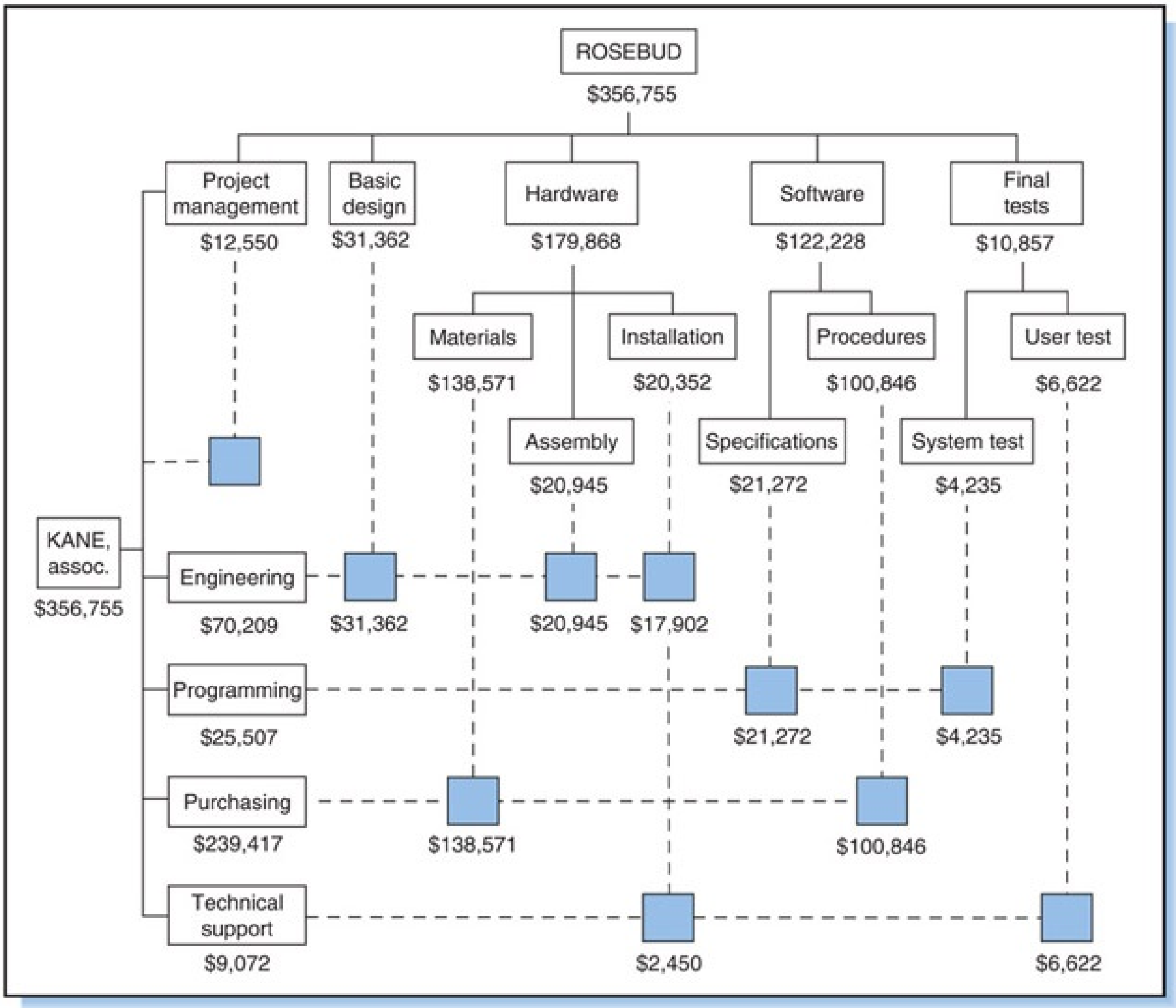


Figure 8.13 Aggregation of control account information by project and organization.

The PCAS and control-account structure permit costs to be summarized in a variety of ways: Figure 8.13 shows budgeted amounts aggregated vertically and horizontally, and Table 8.3 shows budgeted cost types summarized for the five departments and nine work packages in the ROSEBUD project. Thus, through the PCAS and account structure, it is easy to identify cost deviations from budget at the project level or company level, and to readily trace such deviations to the work packages and departments that caused them. Chapter 11 will describe this.

See Chapter 11

8.9 Cost Schedules and Forecasts[14]

Questions arise during project planning about how expenditures vary throughout the project, which periods have the heaviest cash requirements, and how expenditures compare to income. To answer these and other such questions, it helps to anticipate the estimated "pattern of expenditures" as derived from work package cost estimates and the project schedule. Following are examples.

Table 8.3 Cost Summary for ROSEBUD Project

	Labor ($)				Overhead ($)						
	Engineering	Programming	Purchasing	Technical Support	Engineering	Programming	Purchasing	Technical Support	Materials	General and Administrative	Total Cost
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 8.8 to see, for example, how costs in this row were developed

Cost Analysis with Early and Late Start Times

A simplifying assumption used in cost estimating is that costs in each work package are distributed uniformly. For example, a 2-week, $22,000 work package is assumed to have expenditures of $11,000 per week. With this assumption, it is easy to create a *cost schedule* that shows the cost *each week* of the entire project.

As an example look at Figure 8.14—the time-based network for the LOGON project using early start times. Then look at Table 8.4, which lists the work packages for LOGON and associated time and cost information. The weekly direct cost for each activity is the total cost divided by the time; e.g. for Activity H, it is $100K/10 weeks = $10K/week. Using the time-based network, the cost for the project *each week* can be computed by adding the costs for all activities scheduled in the week. The procedure is the same as described in the Chapter 6 for determining the resource loading. According to Figure 8.14, during the first 10 weeks only Activity H is scheduled, so the project weekly cost stays at $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 J—are scheduled, so the weekly cost is their sum, $8K + $4K + $18K + $21K = $51K. These weekly costs, shown in the third column in Table 8.5, represent the *cost schedule* for the project. The fourth column, cumulative expense, represents the forecasted total project cost as of any given week. These costs are graphed in Figure 8.15.

See Chapter 6

Using the same procedure, project cost schedules and forecasts can be prepared based on *late* start times. Figure 8.16 is the time-based network for LOGON using late start times. The last two columns of Table 8.5 are the late-start weekly and cumulative costs.

Given the early and late cost figures in Table 8.5 it is possible to analyze the effects of delaying activities on project costs. By comparing weekly costs from early start times with those from late start times in Figure 8.17, the influence of schedule changes on project costs is readily apparent. The shaded region in the top figure—the *feasible budget region*, which is 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 from delaying activities.

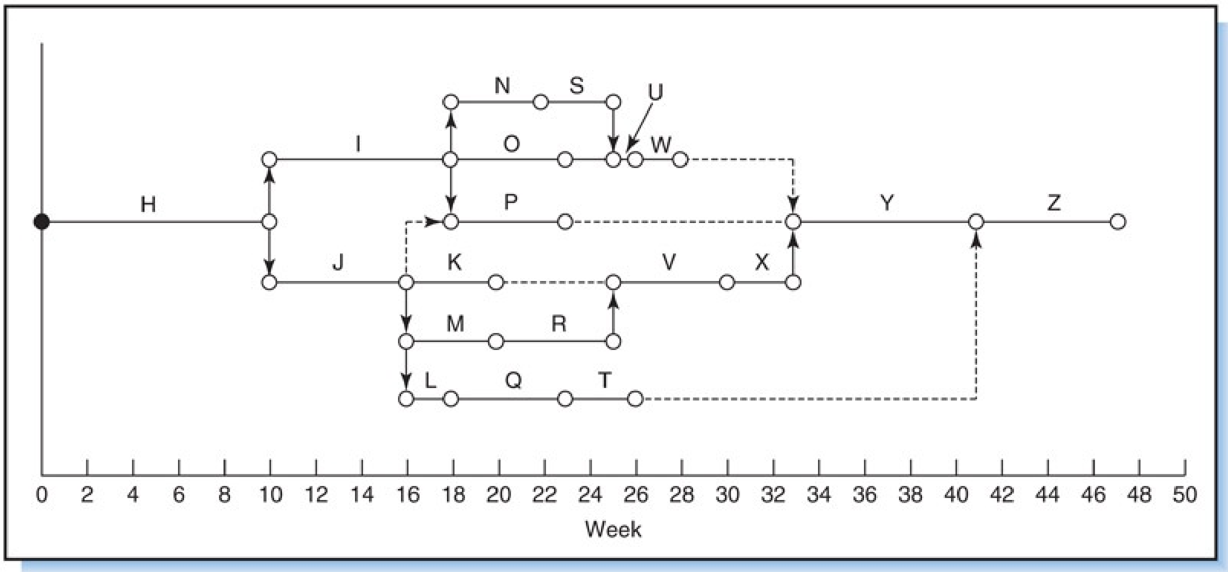


Figure 8.14 Time-based network for the LOGON project using early start times.

Table 8.4 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
1	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	CM
0	5	50	10	5
P	5	60	12	6
Q	5	80	16	CM
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			

Table 8.5 **LOGON Project Weekly Expense Using Early Start Times ($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
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

When funding restrictions constrain project expenditures, cost schedules reveal the places where the schedule must be changed. For example, Figure 8.17 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 would result in a more “level” cost profile and peak expense of only $54,000. The method for leveling resources discussed in Chapter 6 is applicable to leveling project costs; costs are treated as just another resource.

See Chapter 6

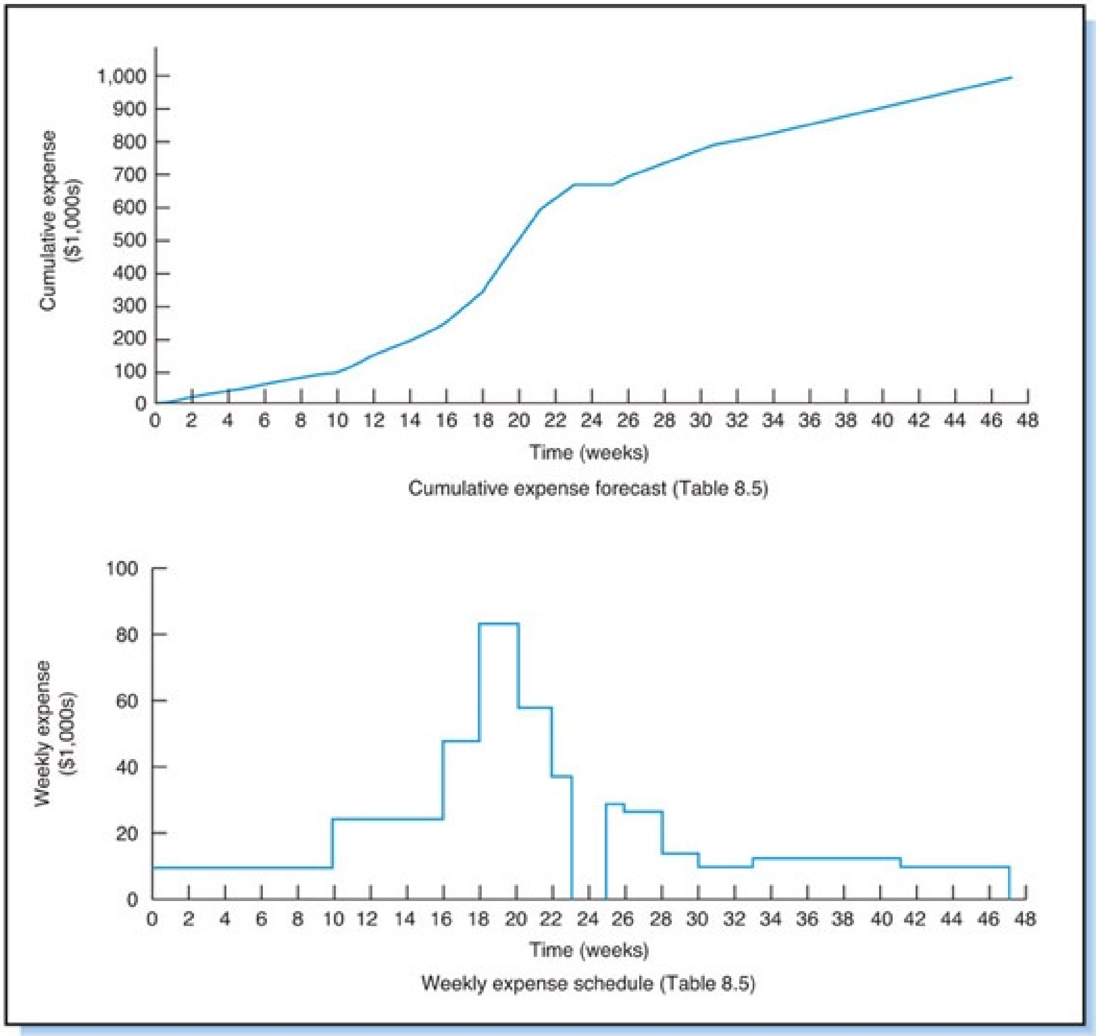


Figure 8.15 Planned weekly expenses and cumulative expenses for the LOGON project.

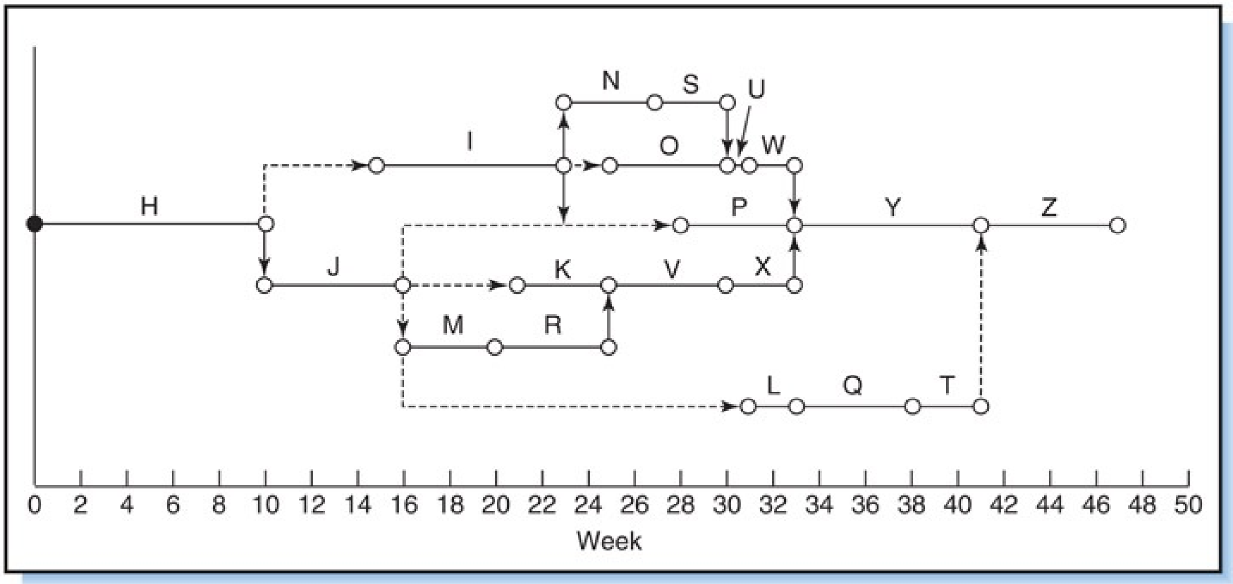


Figure 8.16 Time-based network for the LOGON project using late start times.

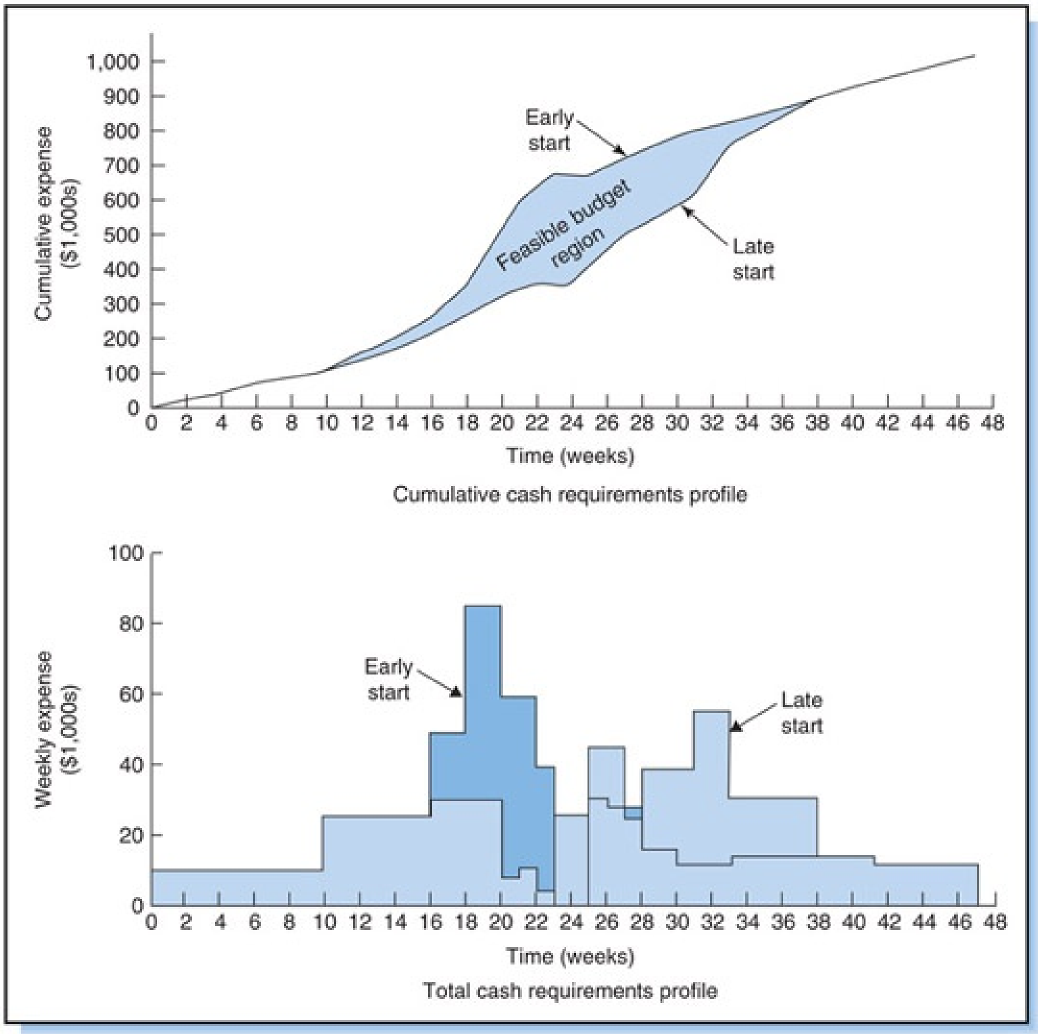


Figure 8.17 Comparison of expenses, early versus late start times.

Effect of Late Start Time 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 substantial savings in the present worth of project costs. For example, suppose the duration of the LOGON is 47 *months* (instead of 47 weeks as used so far). If the annual 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 8.5 (again, assuming the weeks shown to be months instead) and discounting them back to time zero. Now, when the late start times 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 date? Not necessarily. Remember, delaying activities uses up slack time and leaves less time to deal with problems that could delay the project. Thus, whether or not to delay work should also depend 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. Activities that are less familiar, such as research and development work, should be started earlier to retain slack that might be needed to absorb unanticipated delays. This assumes the critical path method; CCPM uses resource buffers, which preclude the need for slack. Also, whether or not to delay work should depend on the schedule of customer payments. If payments are tied to project milestones, then work tied to those milestones should not be delayed. Other factors concerning which activities can be delayed are discussed in Chapter 6.

See Chapter 6

Cash Flow

A problem the project manager often faces 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.[15] The project manager must do a juggling act, balancing income from the client (typically 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 are required.

Figure 8.18 shows an example of forecasted cash flow. All contractually agreed-to sources of income over the life of the project are compared to all foreseeable expenditures, direct and indirect, as well as any penalty costs or scheduled payments—should the project be 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 is 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 his costs. The drawback, however, is that, should the project encounter serious problems, an unscrupulous contractor, having already received several payments, can simply walk away from the job 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 until all defects discovered by the customer have been rectified.

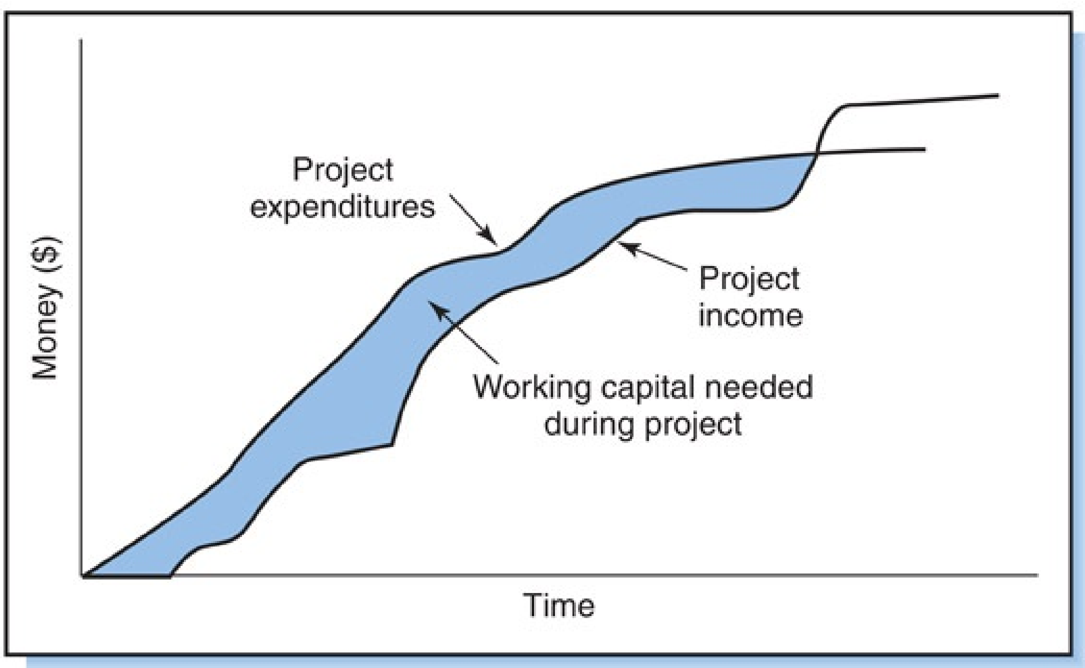


Figure 8.18 Balancing project income and expenditures.

8.10 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 a system represent but the tip of the cost iceberg and that costs to operate (e.g. fuel consumption) and maintain (e.g. parts replacement) it are usually far greater. Whereas the emphasis in this chapter is on *project costs,* i.e., costs incurred during the *project* phases of definition and execution, LCC include costs for the remainder of the system life cycle—the operation phase, eventual disposal of the end-item, and sometimes the conception phase too (initiation and feasibility).

Anticipating LCC is necessary because costs influence many 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 construction costs and plant features, the one with the lowest operating costs will likely win.

The LCC similarly affect decisions regarding development projects, and economic analysis in feasibility studies (Chapter 3) should consider *all costs* for acquisition, operation, maintenance, and disposal of the system 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 high, as were costs for operation and maintenance. At issue were whether enough people would pay the high ticket prices necessary for the airlines to make a profit, and whether enough airlines would purchase the aircraft for the manufacturers to make money. Ultimately, many felt the answer was no on both counts. The US Congress cancelled subsidies for developing the aircraft, and the program dissolved. Meantime the Europeans decided differently and went on to manufacture the Concorde, only 14 of which went into service. Concorde flew for nearly 27 years and the last one was retired in 2003. The LCC were never recouped; had not the governments of Great Britain and France provided subsidies, the airlines and manufacturers would have lost money.

See Chapter 3

Key design decisions affecting the operation and maintenance of a system are made early in the project life cycle—in conception and definition. A product with a high development cost and purchase price becomes more appealing if it can be designed to have a relatively low operating cost. For example, a more fuel-efficient vehicle might be higher priced than less efficient vehicles, but customers readily pay the premium knowing that over the vehicle's life they will recoup it through fuel savings and lower pollution. Of course, estimating LCC involves making assumptions about technology, market, and product demand, and relies on historical costs of similar systems and projects; still, it is a sensible way to assess projects, especially when there is a choice between alternative designs or proposals.

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: it determines how soon the end-item will start to generate revenues, gain market share, and accrue profits or other benefits. The higher costs of speeding up the project are compared to the benefits gained from an early 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 require shut-down and rehabilitation 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 8.7: Life Cycle Costs for an Operational Fleet of Spaceships

This illustration extends on previous SpaceShip One examples, but the numbers are purely hypothetical. Having gained 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 5 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 are made 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 yr x $2] + [5 yr x 80 flights x $0.4]} = $260 million
Total revenues ($ million) = (5 yr x 80 flights x 4 passengers x $0.190) = $304 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, might be somewhat certain, although passenger demand might be fairly uncertain.

The LCC model plays an important role in system design and development. Based on the model, sensitivity analysis can be performed to see what happens when costs increase or decrease to show best case, most likely, and worst case scenarios. The model can also 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 is also 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 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 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.* Concurrent engineering is discussed in Chapter 14.

See Chapter 14

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 in Phases A and B of the project is illustrated in Figure 8.19, which shows the percent of life cycle costs *committed to* versus stage of the project. For example, the figure shows that about 80 percent of a product's LCC is determined by decisions made in the project's concept and design stages, which is well before the product is manufactured and used. This means that whatever the total product LCC, 80 percent is based upon choices made in the first two stages of the project.[16] Unless those decisions correctly account for what will happen in the later stages of production and operation, the result will be a protracted systems development period, delayed launch of the product, and high production and operating costs.

8.11 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.

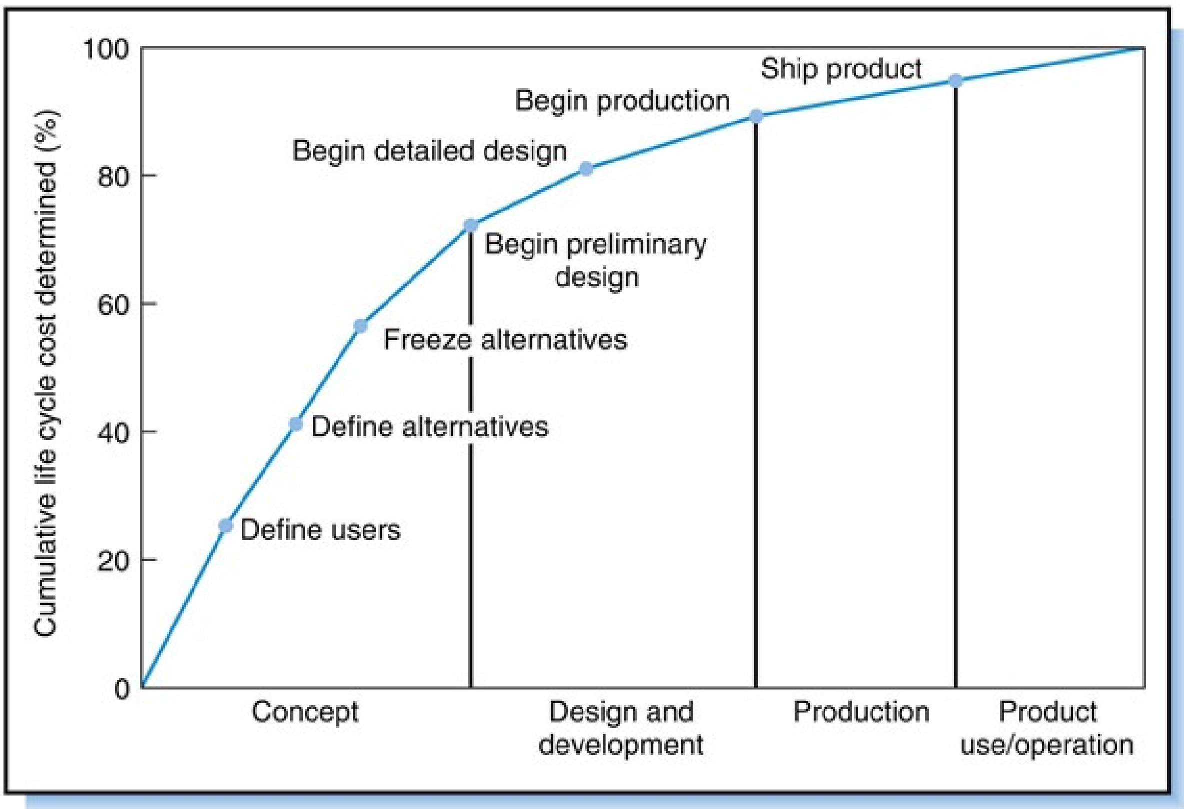


Figure 8.19 Percent of life cycle cost set during stages of the systems development life cycle.

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. Estimate accuracy is partly a function of the stage in the project life cycle during which the estimates are prepared; the further along the cycle, the easier it is to produceaccurate estimates. However, accurate estimates are needed early in the project, and this accuracy can be improved by clearly defining project scope and objectives, and subdividing the project into small tasks and work packages. In general, the smaller the work element being estimated and more standardized the work, the greater the accuracy of the estimate. The aggregate of cost estimates for all the work elements plus overhead costs becomes the cost estimate for the overall project. 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 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.

Review Questions and Problems

1. 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?
2. Define cost escalation. What are major sources of cost escalation?
3. What is the purpose of a contingency fund (management reserve)? How is the contingency fund used and controlled?
4. Describe what the term “phased (rolling wave) project planning” means.
5. How do changes in requirements cause cost escalation?
6. How does the type of contractual agreement influence the potential for cost escalation?
7. What is the relationship between phases of the project life cycle and cost escalation?
8. What are life cycle costs and how are they different from project costs?
9. 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?
10. Explain the difference between accuracy and precision. Give two examples that illustrate the difference.
11. 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.

12. 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?
13. What is the role of the functional units and subcontractors in cost

estimating?

14. Describe the different kinds of contingency amounts and the purposes each serves.
15. Describe the PCAS. What is its purpose and how is it used in project planning?
16. What is a time-phased budget? What is the difference between a budget and a cost estimate?
17. Distinguish recurring costs from nonrecurring costs.
18. What are six cost elements shared by most estimates and budgets?
19. How are direct labor expenses determined?
20. What expenses are included under direct non-labor?
21. How is the overhead rate determined?
22. What is a control account and what kinds of information does it contain? How does a control account fit into the structure of the PCAS?
23. How are control accounts aggregated horizontally and vertically? Why are they aggregated like this?
24. How are time-based forecasts prepared and how are they used?
25. What are the reasons for investigating the influence of schedules on project costs? What is the feasible budget region?
26. 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?
27. Refer to Case 5.1, the Barrage Construction Company, in Chapter 5. 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 5, Figure 5.19, for Sean's WBS.

See Chapter 5

28. The example in Table 8.2 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:

	MARS	Direct Non-Labor PLUTO		G&A
Materials	30	5	Freight	8
Other	10	5	Other	32
	40	10		40

Assuming all remaining costs shown in Table 8.2 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.

29. Chapter 7 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.

See Chapter 7

Suppose that in the MARS project in Table 8.2, 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?

30. Use the table below and the network in Figure 8.20 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

a. Compute the ESs and LSs for the project. Assume T_S is the same as the earliest project completion date.
b. Construct a time-based network for the project such as Figure 8.14 (use early start times).
c. Construct two diagrams similar to those in Figure 8.15 showing the weekly and cumulative project expenses.

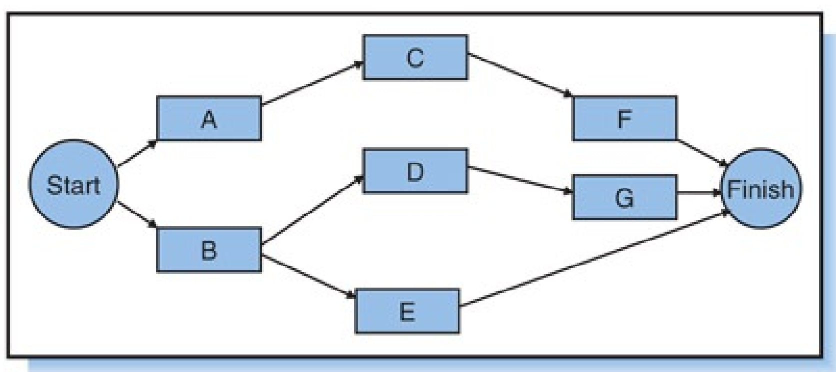


Figure 8.20 ARGOT project network.

31. Using the data in problem 30, repeat Steps b and c using late start times. Then identify the feasible budget region using the cumulative curves.
32. Explain retention money and performance guarantee.

Questions About the Study Project

1. How were project costs estimated? Who was involved? Describe the process.
2. When did estimating take place? How were estimates checked and accumulated? How were they related to the WBS?
3. What, if any, were the principle causes of cost escalation in the project?
4. 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?
5. How often and when were cost estimates revised during the project?
6. How were overhead costs determined? What basis was used for establishing overhead cost rates?
7. How were estimates tallied to arrive at a total project cost estimate? Who did this?
8. 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?
9. Describe the process of creating the project budget. Show a sample budget (or portion thereof).
10. How were management and supervisory costs handled in the budget?
11. 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?
12. What kinds of costs summaries were prepared? Who were they sent to? How were they used? Show some examples.
13. Did the PCAS produce time-phased cost schedules and forecasts? Show some examples. How were they used by the project manager?

14. Were life cycle costs a consideration in the project? Explain.

Case 8.1 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 commercial spacecraft (SpaceShipTwo, or SS2), launch aircraft (WhiteKnightTwo, or WK2), and support equipment. 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 8.7 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 8.7 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 8.2 Estimated Costs for the Chunnel Project[17]

Before construction began on the English Channel Tunnel (Chunnel) 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 Chunnel, 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 Chunnel 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 Chunnel 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. Why might the estimates for the Chunnel 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 Chunnel.

Case 8.3 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 8.21).

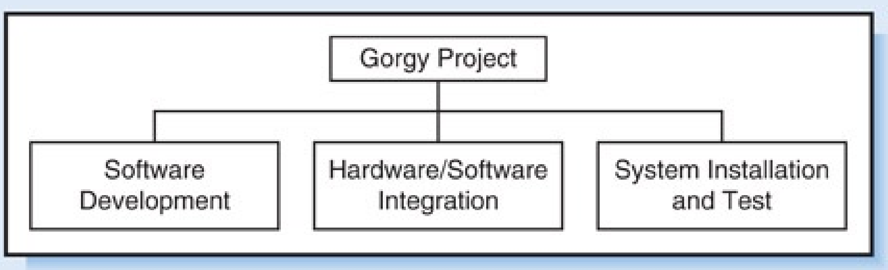


Figure 8.21 Gorgy Project.

She estimates the costs for the three work packages as follows:

Development	$2 million
Integration	$1 million

Installation/Test	$1.5 million
Project	$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 Ghophal, 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 based it on the number provided in the RFP for what the development portion of the project should cost based on the customer’s experience from 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/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."

Shireen suspects that Fiona's cost estimation approach is rather simplistic and leaves ample room for error. List at least four inadequacies in her approach and places where the estimates can go wrong.

Case 8.4 Melbourne Construction Company, D

Bill Asher, the estimator for Melbourne Construction Company, is currently 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 m^2). 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 for the footings specified for the hotel, the standard is 0.066 labor hours per SFCA.[18]

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. Adding workers will not necessarily shorten task times and might even increase them.)

Endnotes

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. Harrison, *Advanced Project Management,* pp. 154–161.

7. Dingle J. *Project Management: Orientation for Decision Makers*. London: Arnold/John Wiley & Sons; 1997, p. 105.

8. Pool R. *Beyond Engineering: How Society Shapes Technology*. New York, NY: Oxford University Press; 1997; Heppenheimer T.A. Nuclear power. *Invention and Technology* Fall 2002; 18(2): 46–56.

9. 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.

10. See ibid., Chapter 14, for discussion of labor costing in projects.

11. This example is derived from a similar one in Rosenau M. *Successful Project Management.* Belmont, CA: Lifetime Learning; 1981, pp. 89–91.

12. This example is derived from Wilson T. and Stone D. Project management for an architectural firm. *Management Accounting*; October 1980, 25–46.

13. The kinds of cost summaries used often depend on the kind available in the software, though many software packages permit customizing of reports.

14. Wiest J. and Levy F. *A Management Guide to PERT/CPM*, 2nd edn. Upper Saddle River, NJ: Prentice Hall;

1977, pp. 90–94.

15. 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."

16. Smith P. and Reinertsen D. *Developing Products in Half the Time*. New York: Van Nostrand Reinhold; 1991, pp. 224–225.

17. Fetherston D. *The Chunnel*. New York, NY: Times Books; 1997, pp. 141–142.

18. *RS Means Labor Rates for the Construction Industry*, 41st edn. Norwell, MA: RS Means; 2013.

Chapter 9
Project Quality Management

I have offended God and mankind because my work didn't reach the quality it should have.

—Leonardo da Vinci

Besides meeting the budget and schedule, project success depends on how well a project meets performance requirements. Performance requirements generally are based upon project stakeholders' needs and expectations about the functioning and performance of the project end-item or deliverables. A "high-quality" project is one that meets performance requirements, satisfies the needs and expectations of all key stakeholders, and causes no harm elsewhere.

9.1 The Concept of Quality

In the 1950s *quality* was viewed as the process of inspecting products that had already been produced and to separate 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 in *processes* to ensure things are done right the first time, every time, and a culture where everybody is quality-focused.

But in the competitive pursuit, 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 lowering costs and shortening 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 the joint in a rocket motor and ignite the main fuel tank shortly after launch, causing an explosion and killing the seven astronauts onboard. While engineers had previously warned about the risk of this happening, the launch proceeded as scheduled because of a promise to politicians; for the sake of meeting a schedule, quality was compromised.

The London Tower Bridge, Figure 9.1, offers a contrast.[2] It opened in 1894, 4 years late and costing nearly twice the estimated £585,000. But more than 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 and pollution—problems its original designers never considered. In terms of time and cost the project was a failure; in terms of quality it has been a raving success.

Figure 9.1 London Tower Bridge.

Source: iStock.

What is Quality?

Quality is meeting specifications or requirements—but it means more than that. While meeting project specifications will usually prevent a customer from taking a contractor to court, it alone cannot ensure the customer will be satisfied with the end result or the contractor will receive gratitude or win repeat business.

Ideally a project aims *beyond* specifications and tries to fulfill customer expectations—including those not articulated; it aims at delighting the customer. Project managers sometimes assume, wrongly, that customer needs, expectations, and requirements are readily evident or will require little effort to research and specify.

Fitness for Purpose

The term *quality* implies that a product or deliverable is *fit for the intended purpose*; this can involve a wide range of criteria such as performance, safety, reliability, ease of handling, maintainability, logistical support, and no harmful environmental impacts. Beyond fitness, however, the customer will also consider 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. The project manager must seek to balance the multiple aspects of a product and define 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 or remove 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 malfunction or failure 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 try to eliminate every defect.[4] Even the most minor defect in 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. 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 to do that 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 appropriate—hence best-quality—coal for the purpose; 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 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—some that could be described as fads. The most lasting and popular movement since the 1980s is *total quality management* (TQM). TQM is a set of techniques and more—it is a mindset, 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 in this and later chapters.

Complementing TQM is the management philosophy of *lean production* (LP). 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 originated at Toyota and have been successfully adopted around the world. In some industries (e.g. autos and electronics) virtually all the big players have adopted lean production. In project environments, LP methods are being applied to product development and construction. Some of these methods are described in Chapter 13.

See Chapter 13

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 another 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 called DMADV—Define, Measure, Analyze, Design, and Verify. In projects, the Six Sigma approach 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.

Quality of the end-item or product starts with clearly defined system requirements or specifications 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 the intended purpose and any negotiated compromises. Comprehensive specifications for the deliverable should be included in the project scope definition.

Project quality management includes management processes as well as techniques to reduce the risk that products or deliverables will not meet requirements. The following sections discuss these processes and techniques.

9.2 Project Quality Management Processes

Project quality management has three processes: quality planning, quality assurance, and quality control (Figure 9.2). *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 nonconformities and quality planning and assurance as the "healthy lifestyle" to prevent nonconformities in the first place.

As shown in Figure 9.2, project quality control links quality planning and quality assurance to ensure that quality assurance happens according to 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 on future projects.

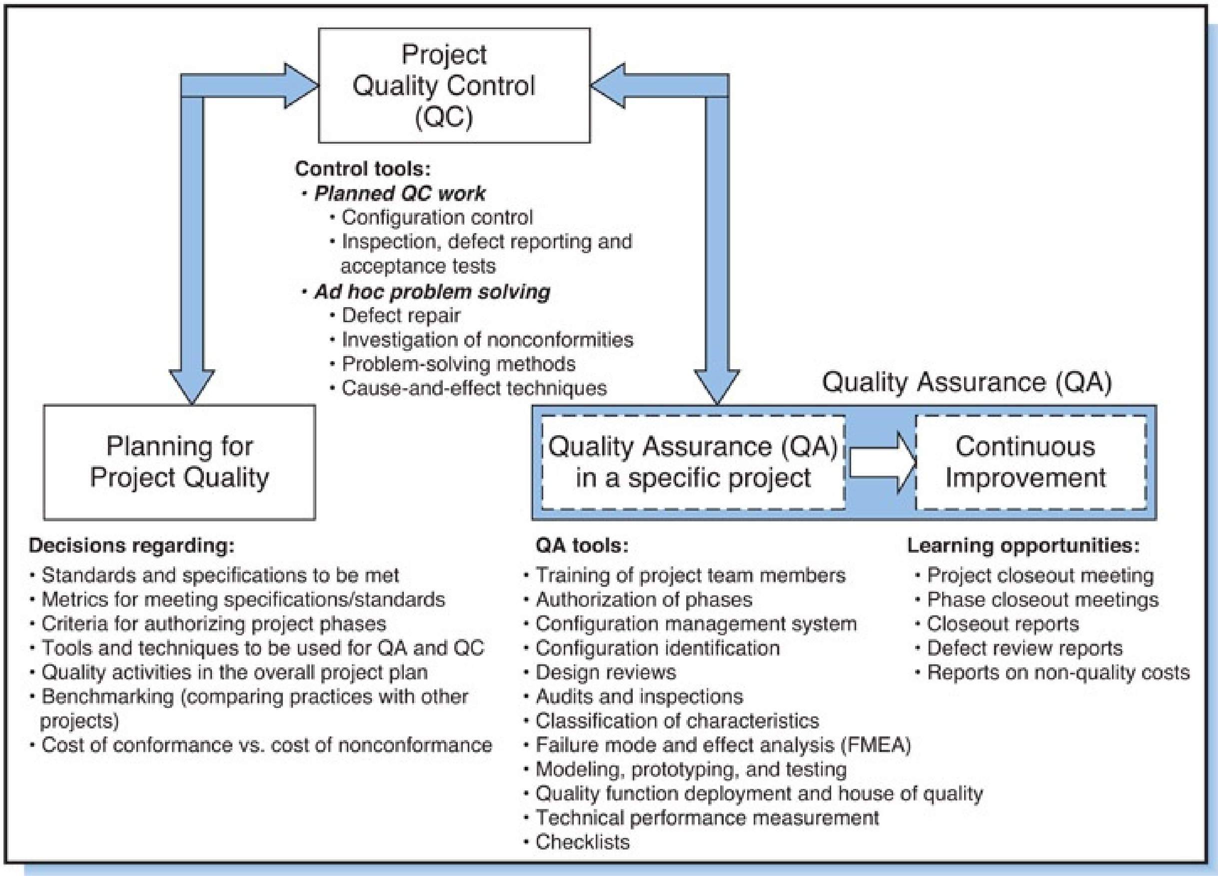


Figure 9.2 The project quality management process.

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 plan as part of the execution plan for each project.

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 that already exist such as the ISO 9001 standard (a quality management system).[7] For design and development projects, this standard prescribes that an organization shall 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.

Costs of Quality

Since quality is always related to value for the money spent, quality planning should consider the costs and benefits of quality activities. A *cost–benefit analysis* is performed to evaluate the proposed quality activities. Money spent on quality assurance and control should be justified in terms of expected savings or benefits from fewer or eliminated nonconformities.

The costs of quality are classified as prevention, appraisal and control *(costs of conformance*), and internal failure and external failure *(costs of nonconformance*):

1. Prevention: costs of training, design reviews, and activities aimed at preventing errors; includes cost of quality planning.
2. Appraisal and control: costs of evaluating products and processes, including product reviews, audits, tests, and inspections.
3. Internal failure: costs associated with nonconformities discovered by the producer; includes costs for scrap, rework, and retest.
4. External failure: costs incurred as a result of product failures after delivery to the customer; includes costs for replacements, warranty repairs, liability, lost sales, and damaged reputation.

While the 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.[8] 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 (prevention, appraisal, and control) are among the many the project manager must justify to management and the customer, and include in the project plan and budget.

Project Quality Management Plan

The project quality management plan (quality plan) is an important component of the project execution plan discussed in Chapter 5. 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.

See Chapter 5

Identifying, scheduling, budgeting, and assigning responsibility for quality assurance and control activities is done utilizing the same principles and methods as for other project activities, discussed in Chapters 4 through 8. The plan addresses the quality management approach of the project, indicates 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

See Chapters 4–8

management system and classification of characteristics procedures – both discussed later) and how they would be implemented in the project.

If not covered sufficiently in a project management methodology, the plan should indicate how each of the project phases would be initiated and authorized, and how phases and the entire project would be closed out.

The plan should address all relevant elements indicated in Figure 9.2, including how the project team will ensure that the quality requirements as stated in the specifications and standards would be met. This can typically include:

- 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 would take place and test

specifications, procedures and reports
- Any required design reviews, the purpose of each, people involved, and outputs required
- Audits
- Checklists
- Techniques that would be used, e.g. FMEA or control charts.

The plan can also indicate how non-conformances, customer complaints, and corrective actions would be handled. It should clearly indicate the person primarily responsible for each task and the roles and responsibilities of others involved. The responsibility matrix discussed in Chapter 5 can be used for this purpose.

See Chapter 5

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 9.2, 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 everyone participating is heard. The discussion is formally documented and a list of lessons learned 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 12; they play an important role in knowledge management, discussed in Chapter 17.

See Chapters 12 and 17

Quality Control

Quality control is the ongoing process of monitoring and assessing work, and taking corrective action so as to achieve the planned quality outcomes (requirements and specifications). It also verifies that quality assurance activities are being performed as specified in the quality plan. Quality control is one aspect of *project control*—a topic of Chapter 11 but included here for continuity.

See Chapter 11

Quality control can be contrasted to *scope verification*: whereas scope verification refers to the acceptability of project deliverables *by the customer*, quality control refers to conformance to specifications as set *by the contractor*. Scope verification includes verifying the *acceptability* of specifications and standards, but 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 acceptance tests before handover of deliverables 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 that is discovered only *after* the item has been 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 an alternative material to be used. A third form of deviation from specification is a *modification*; this is a change to specification that is considered permanent.

Control activities as illustrated in Figure 9.2 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 has to work 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 lowest price supplier 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 this 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 9.1: 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. 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.

9.3 Techniques for Quality Assurance in System Development

This section further explains the items in Figure 9.2. In phased project planning, authorization of a phase implies that plans for the phase have satisfied pre-specified criteria, including that the plans include sufficient measures for quality assurance. System developers employ a variety of such measures, as discussed in this section.

Configuration Management[10]

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 technical 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 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. Policies and procedures that form the configuration management system for a project should be included in 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 2, any subsystem, component,

See Chapter 2

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.

The master copy (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 in its operation and maintenance. Where multiple units are produced (e.g. cars, airplanes, appliances) and 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 and is illustrated in the following example.

See Chapter 4

Example 9.2: Traceability and the Apollo Spacecraft[11]

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 extruded, 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.

Configuration control is based on the following principles:

1. Any organization or individual may request a modification, waiver, or deviation.
2. The proposed change and its motivation are documented. Standard documents exist for this purpose: for modifications, the document is called a *change proposal, change request, change order*, or *variation order*.
3. The impact of the proposed change on system performance, safety, and the environment is evaluated; so is its impact on other hardware items, software, manuals, and methods of manufacturing or construction and maintenance.
4. The change is assessed for feasibility, which includes estimating the resources needed to implement the change and the impact of the change on schedules.
5. The group responsible for approving or rejecting the change is the *configuration board* (CB) or a *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.
6. 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, and manuals

and other documentation.
7. The implemented change is monitored and controlled to ensure it complies with the approved change proposal.

Change requests are sometimes classified as Class I or Class II. Class I requests can be approved by the contractor or the developer; Class II must be approved by the customer.

Configuration control is an aspect of project control and, in particular, change control, both discussed in Chapter 11.

See Chapter 11

Design Reviews

The project manager must ensure that the proposed design is acceptable in all respects; such 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 review*s, described in Chapter 12) provide confirmation of design assumptions (e.g. load conditions), data used in the design process, and 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:

See Chapter 12

1. omissions or errors
2. compliance to regulations, codes, specifications, and standards
3. cost of ownership
4. safety and product liability
5. reliability
6. availability
7. ability to be constructed or manufactured (manufacturability)

8. shelf life
9. operability
10. maintainability
11. intellectual property rights
12. ergonomics.

The reviews involve representatives from all disciplines, functions, users who will be connected to the deliverable throughout its life cycle, and, often, outside designers and subject matter experts. (This relates to concurrent engineering, discussed in Chapters 4 and 14.) For example, the design review for a chemical plant, mine, or factory would include representatives from:

See Chapter 4 and Chapter 14

- The organization that will operate the facility
- The technical support area that will be maintaining the facility
- The construction company
- 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; 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, but for the design of a complex system, it would be necessary to convene several reviews 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 design 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 over-seen 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.

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. In recent years the Internet has become an effective medium for conducting design reviews.[12]

Informal Design Reviews

Although formal reviews are necessary, the project manager should encourage informal design reviews, which are informal discussions among designers and between designers and other stakeholders. Good suggestions can originate anywhere, but it is up to the designer to decide whether or not to use them. Draft designs, reports, and other deliverables should be shared regularly (and, ideally, voluntarily) among peer designers and others for informal review. In a healthy quality culture, teams use *brainstorming* to evaluate and edit not only designs, but also reports and deliverables of all kinds. 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 separated from poor ones.

Example 9.3: Formal and Internal Reviews in the Mars Pathfinder Project[13]

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 Path-finder (see Example 11.4, Chapter 11) project the review board comprised 25 consultants and seasoned managers from NASA and Jet Propulsion Laboratory (JPL, the site responsible for most of the Path-finder design work), none of whom was associated with the project.

See Chapter 11

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 overruns 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 is laborious and 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. Mature designers appreciate the design review process in terms of the networking experience, innovative ideas gained from others, and reduced risks, but less mature designers tend to feel insulted or intimidated by the process. It is human nature for people to feel less than enthusiastic about others' ideas and to resist suggested changes to their own. 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.

Audits

Unlike design reviews, which relate specifically to the design of a product, audits are broad in scope and include a variety of investigations and inquiries. The purpose of audits is to verify that *management processes* comply with prescribed processes, procedures, and specifications regarding, for example, system engineering procedures, configuration management systems, warehousing and inventory control systems, and safety procedures. They are also used to verify that *technical processes* such as welding adhere to prescribed procedures, and to determine *project status* based upon careful examination of certain critical aspects of the work. 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, they focus strictly on verifying that processes are being implemented as required. The auditor can be an internal staff 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 9.4: 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 deliverable 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 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 most impact 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*, and *incidental* (alternatively, *critical, major A, major B*, and *minor*). 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 characteristics classified as *incidental* 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 weld 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 in particular 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 to classify the brake's components as critical as well. Failure mode and effect analysis (FMEA) sometimes plays a role in the classification process.

Failure Mode and Effect Analysis

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 (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 can likely occur are also listed.
3. Assign a *probability* 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 of the effects.
6. Compute the *criticality* of each failure mode. Criticality is a function of both the probability of the failure and the seriousness 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.

Table 9.1 illustrates: In the columns "Sev" (severity), "Prob" (probability), and "Det" (detectability—would the failure be difficult to detect?) each failure mode is rated 1 to 10. RPN (risk priority number) is the criticality of the failure mode, computed as:

RPN = Sev × Prob × Det

Items are then prioritized by RPN with highest RPN first.

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 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 10.

See Chapter 10

Modeling and Prototyping

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 9.2 shows the kinds of models built and tested in the project phases and the kinds of risks they eliminate. Projects for the development of large processing plants often use models in a similar fashion (Table 9.3); models for such projects usually start out as laboratory equipment but grow in sophistication and capacity to enable a pilot operation that leads to a demonstration plant that closely replicates the proposed facility.

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 9.5: Modeling the Form and Fit of Boeing 777 Components[14]

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

Table 9.1 FMEA Table

System ____________
Subsystem ____________
Component ____________
Design Lead ____________
Core Team ____________

Potential
Failure Mode and Effects Analysis
(Design FMEA)
Key Date ____________

FMEA Number ____________
Prepared By ____________
FMEA Date ____________
Revision Date ____________
Page ______ of ______

Item/ Function	Potential Failure Mode(s)	Potential Effect(s) of Failure	S E V	Potential Cause(s)/ Mechanism(s) of Failure	P R O B	Current Design Controls	D E T	R P N	Recommended Action(s) Completion Date	Responsibility and Target Actions Taken	Action Results				
												New Sev	New Occ	New Det	New RPN

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

Table 9.2 Phases for Development of Products

Project	Objectives Relating

Phase	Model Built and Tested	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
Validation	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 9.3 **Phases for Development of Process 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

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: tests conducted by the *contractor* to make sure that the system design (1) meets system requirements and (2) is being followed by the producer or builder, and (3) tests conducted by the *customer* and others to ensure the system meets user requirements and other contractual agreements.[15]

The first category of tests is aimed at verifying 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 so as to catch problems as early as possible. Of course, even if the design is perfect, if the builders cut corners on materials and procedures or do not conform to the design, the system will be inadequate; hence, the second category of tests is necessary to verify that the builders are correctly following 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 the user personnel who will operate the system, may expose design deficiencies that project designers and engineers 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 formed from all the subsystems is tested to ensure it meets the user's performance requirements.

Tests are performed against earlier developed system objectives, systems specifications, and normal user requirements. Sometimes, in addition, they are performed *in excess* of specifications for normal conditions to determine the actual capacity or point of failure of the system. 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. In *fatigue tests* the system is subjected to an increasing load or repeated cycles until it fails; this is done to determine the system's ultimate capacity. 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 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 9.6: Team Inspection Process[16]

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 member of the team with a copy of the document or code and any supporting documentation at least 2 days before the meeting.

Each inspection meeting lasts for about 2 hours during which an average team can inspect 10–15 pages of text or 400 lines of code. 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.

9.4 Techniques for Quality Control

Quality control involves performing the tasks defined in the quality management plan and taking any 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 subjected to nondestructive testing.

Although testing the end results from 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 would 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 after production begins. 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 of Quality Control

In the 1980s, Kaoru Ishikawa of Tokyo University defined the basic tools of quality control.[17] The 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 in a production environment (Kawasaki Steel Works), the tools discussed below are nonetheless applicable to projects.[18]

Run Chart

A run chart is a graph of observed results plotted versus time to reveal possible trends or anomalies. The plot of schedule performance index versus cost performance index illustrated in Figure 11.12 is a form of run chart that tracks project performance and shows if it is improving or worsening in terms of schedule and cost.

Control Chart

Control charts are widely used for tracking and controlling repetitive processes and detecting process changes. For projects that include the development of production processes, one deliverable would be specifying the relevant charts for controlling the quality of the process. Readers involved in projects aimed at the delivery of repetitive operating systems should refer to books on statistical control techniques such as *Juran's Quality Control Handbook*.[19]

Pareto Diagram

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 situations, including those relating to quality. Quality consultant Dr. Joseph Juran in the late 1940s posited that the large majority of defects result from a relative few causes; thus, for economic reasons it makes sense to identify the vital few causes of most defects and to direct the most effort at removing them.

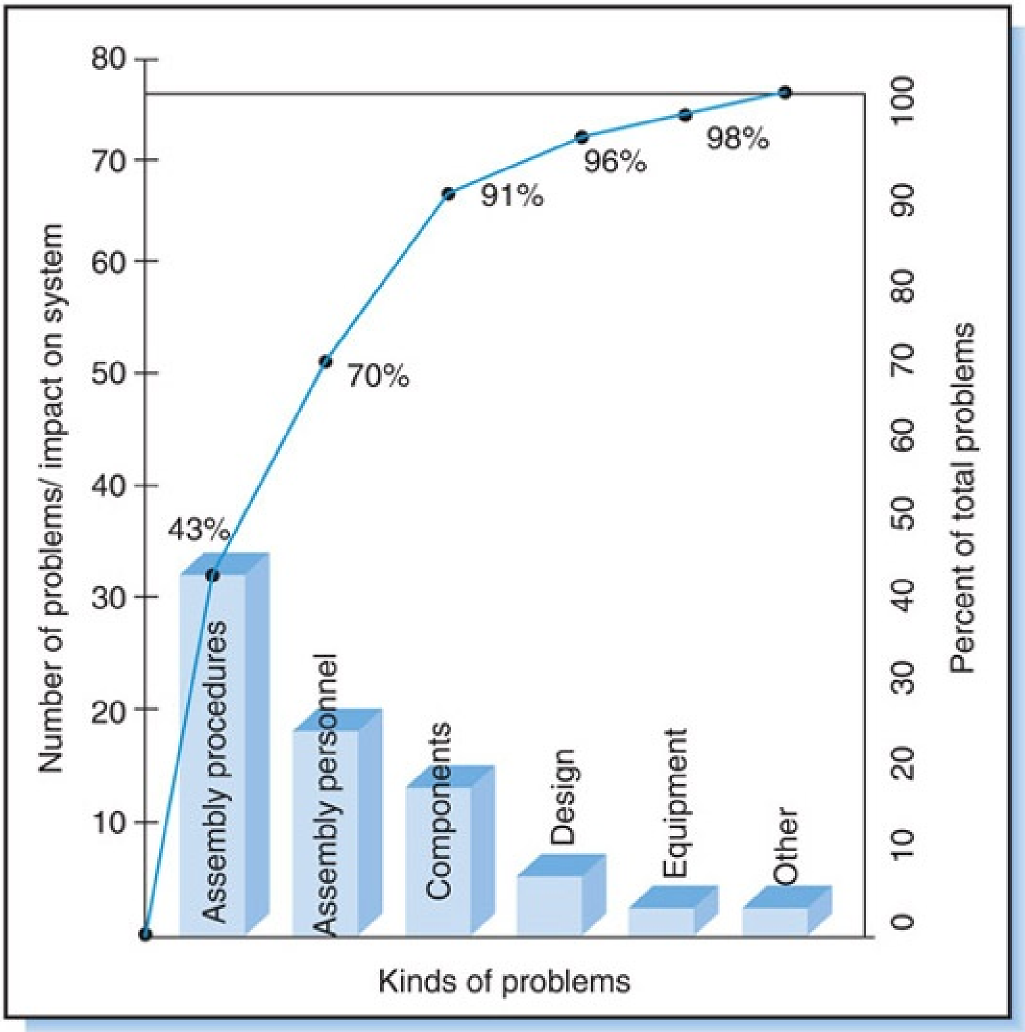


Figure 9.3 Pareto diagram.

Figure 9.3 is a Pareto diagram: the histogram on the bottom of the diagram shows 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 kind of problem accounts for43 percent of all problems; the first and second combined account for 70 percent. Thus, resolving just the first two kinds of problems would eliminate 70 percent of the problems.

Cause-and-Effect Diagram

Problems are often best addressed through the collective experience of project teams. 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* or *Ishikawa* diagram), which is a scheme for arranging the causes for a specified effect in a logical way. Figure 9.4 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" 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 the potential *causes,* and (2) given a cause, to identify the outcomes (*effects*) that might ensue. Identifying the causes is an obvious first step to resolving problems. CE diagrams are also used in risk analysis, and an example is given in Chapter 10.

See Chapter 10

Other Tools for Quality Assurance and Control

Many planning and control methods described elsewhere in this book also apply to quality assurance and control. For example, much quality assurance effort in a product design project is directed at keeping the project team focused on customer requirements and preventing distortion or misinterpretation of those requirements as the project moves between stages, departments, and people. Quality function deployment (QFD), discussed in Chapter 4, serves just such a purpose. Technical Performance Measurement (TPM), discussed in Chapter 11, can also be considered a tool for quality assurance.

See Chapters 4 and 11

Checklists—for preparing plans and doing inspections, testing, and design reviews—and FMEA are also quality tools; they help prevent important issues from being overlooked. Of course, a disadvantage of checklists is that people tend to overly rely on them and ignore things *not* on the list. So, the last item on every checklist should be “Now, list all the possible important things not already on this list and check them too!”

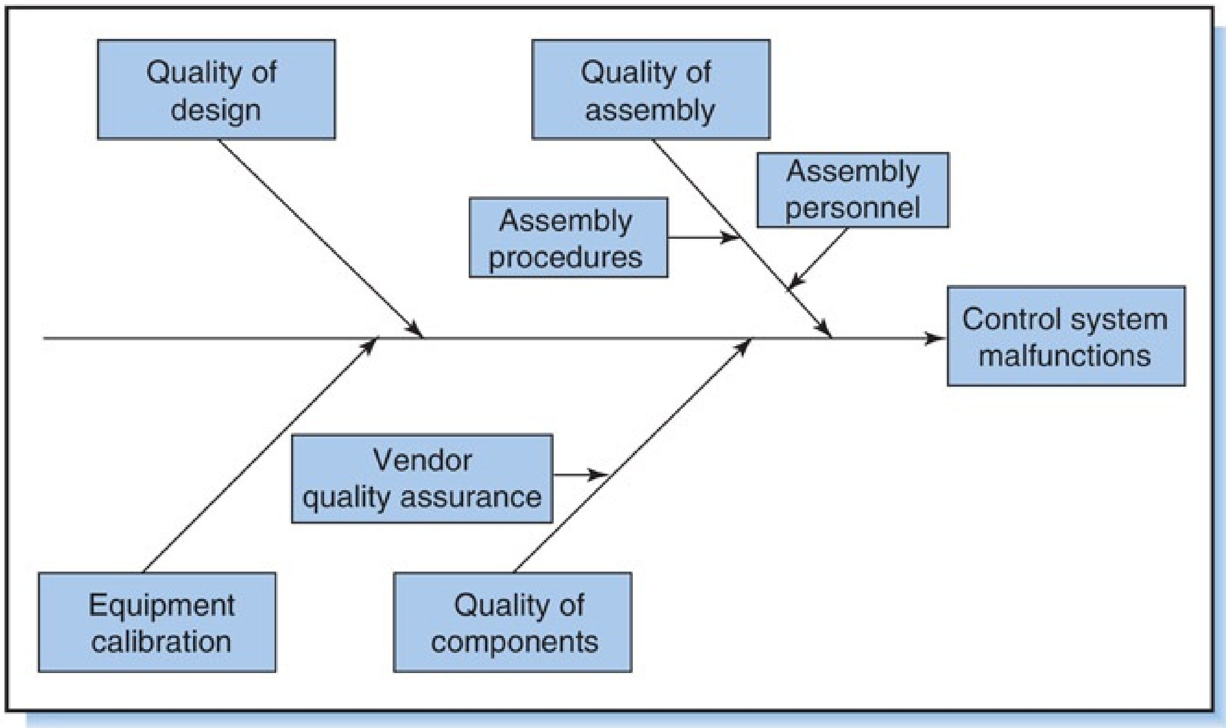


Figure 9.4 Cause-and effect (fishbone or Ishikawa) diagram.

9.5 Summary

Project schedules, budgets, and quality management address the three dimensions of project goals: to finish on time and on budget, and satisfy requirements. Project quality accounts for an end-item's compliance to specifications and fitness for the purpose, and customer expectations. It does not necessarily imply the highest grade, most product features, or even zero defects; what it does imply is whatever is considered "best" in terms of customer expectations about the end-item's intended use.

Quality management includes three processes: *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 problems solving.

Project quality management has benefited from the quality philosophies of TQM and Six Sigma, both 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, as well as experimenting with models and prototypes. Many of the techniques apply also 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 indicating lectures, exams etc. 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.

   (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 10.
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.

See Chapter 10

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 section on the Project Quality Management Plan that are 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.

Case 9.1 Ceiling Panel Collapse in the Big Dig Project

(For more about the Big Dig Project—Boston's Central Artery/Tunnel Project, see Chapter 15, Example 15.4 and Case 15.3.)

See Chapter 15

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; 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 contended, too few anchors were used. “They didn’t have enough to carry the load. There was no room for error.” He added, however, 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 a single 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 to handle 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 that 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,” said one US Representative. 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 CE (fishbone, Ishikawa) diagram to illustrate possible causes and effects. Include the possible causes mentioned in the case. The diagram should have been developed before construction, therefore also indicate other possible failure modes and other 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 9.2 Fifa 2010 World Cup South Africa™[26]

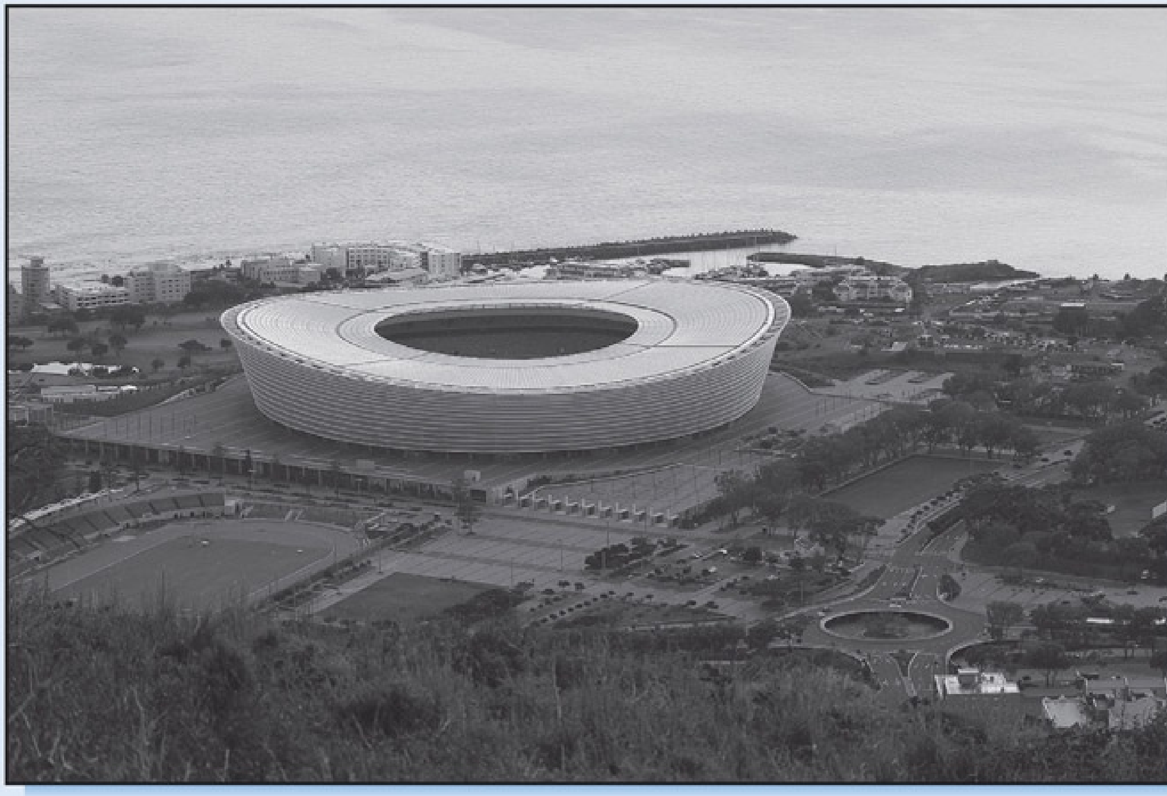


Figure 9.5 Cape Town Stadium used for 2010 FIFA World Cup.

Source: iStock.

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). A centerpiece stadium for the games is the newly constructed Cape Town Stadium shown in Figure 9.5. The requirements for the once-offFIFA matches typically far 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 9.4. 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 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, electric power, backup electricity, water supply and waste water 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 of the relevant stakeholders were required to attend an on-site inspection and to 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?

Table 9.4 Main Stakeholders in FIFA 2010 and Their Roles

Stakeholders	Roles
FIFA (International Federation of Association Football)	Main customer
(French: *Fédération Internationale de Football Association*)	
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

4. Comment on the problem of people from different organizations working in the same space at the same time.

Case 9.3 Airbag Adversity

Insufficient quality management during product development, launch, and production can lead to subsequent costly projects and programs 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 such as 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 a total of 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) law suits 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.

Endnotes

1. Carruthers M. *Principles of Management for Quality Projects.* London: International Thompson Press; 1999.

2. Ibid.

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.* Boca Raton, FL: CRC/Productivity Press, 2011.

7. International Systems Organization, *ISO 9001, Quality Management Systems—Requirements.* Geneva, 2015.

8. Crosby, op. cit.

9. Kransdorff A. The role of the post-project analysis. *The Learning Organization* 3(1); 1996: 11–15.

10. 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, 2003.

11. Gray M. *Angle of Attack: Harrison Storms and the Race to the Moon.* New York: W.W. Norton; 1997, pp. 170–171.

12. East E., Kirby J. and Perez G. Improved design review through web collaboration. *Journal of Management in Engineering.* April 2004: 51–55.

13. 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.

14. http://www.boeing.com/commercial/777family/pf/pf_computing.html (accessed in August 2006).

15. Where multiple units of a system are produced, the first and second groups of tests are performed on a prototype. When the system is single-copy, the second group of tests occurs during construction, and the design is verified afterward.

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. Juran J. and Gryna F. *Juran's Quality Control Handbook*, 4th edn. New York: McGraw-Hill; 1988.

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 May 20 2007.

26. Personal communication, Eugene van Vuuren. Member of LOC Technical Team. Nov. 2010.

Chapter 10
Project Risk Management

Life "looks just a little more mathematical and regular than it is; its exactitude is obvious, but its inexactitude is hidden; its wildness lies in wait."

—G. K. Chesterton[1]

Every project is risky, meaning project outcomes won't turn out as planned. The project could significantly overrun cost or schedule targets, or the end-item may fall short of requirements. Project outcomes result from many things, including some that are rather unpredictable and over which project managers have little control. 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 emerging problems.[2]

10.1 Risk Concepts

Risk is a function of the uniqueness of a project and the experience of the project team. 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 avoid 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 problematical event will occur.
2. The *impact* of the event if it does occur.

Risk is a joint function of the two,

Risk = f(likelihood, impact)

A project will ordinarily be considered "risky" whenever at least one—the likelihood or the impact—is large. For example, a project will be considered risky when the potential impact is human fatality or massive financial loss, even if the likelihood is small. Risk can also mean *opportunities,* e.g. the potential for additional rewards, savings, or benefits. Typically, however, risk management focuses on the risk of failure.

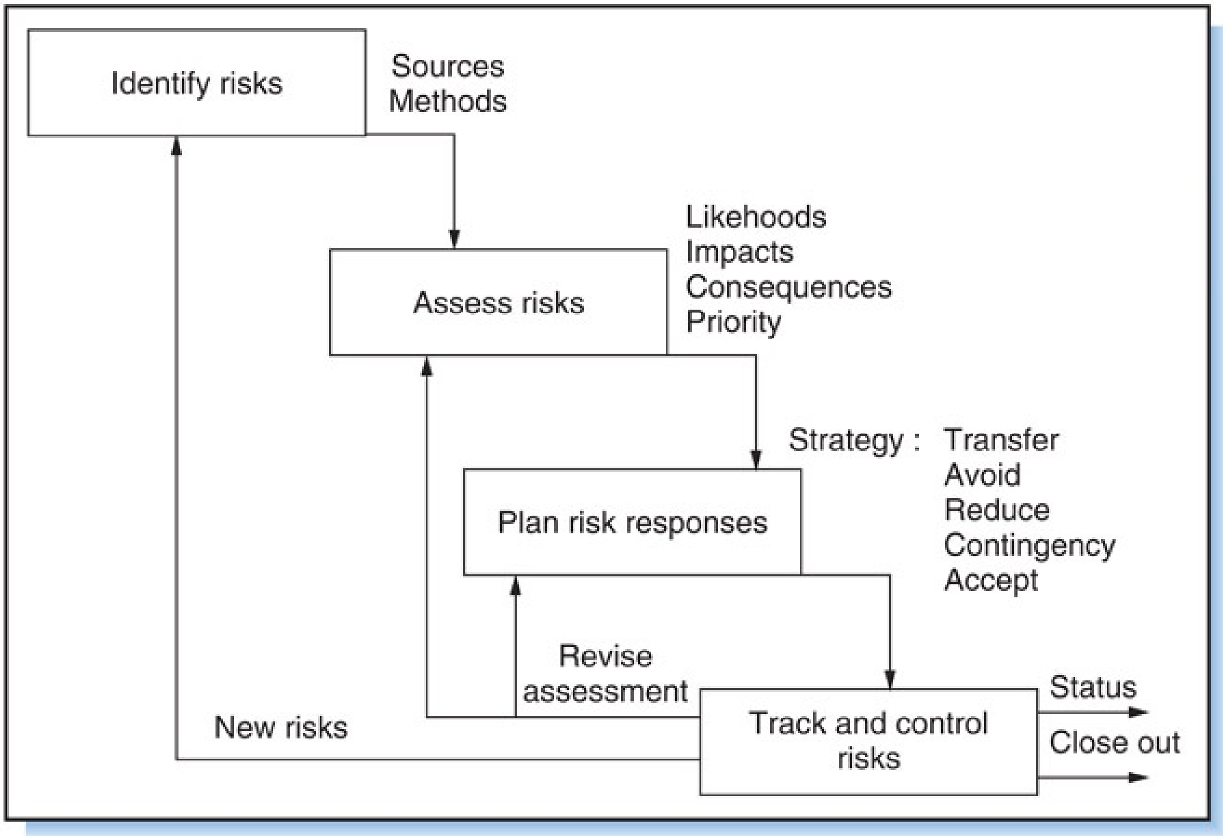


Figure 10.1 Risk management elements and process.

Many 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 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 risk management. The process and elements of risk management are shown in Figure 10.1.

10.2 Risk Identification

You can only manage things you are aware of. Thus, risk management begins with identifying the risks and predicting their consequences.

Risk 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 illustrated in Chapter 9, Table 9.1). In product development projects, the risk of failure tends to be high in the early stage of preliminary design, but diminishes later. Some risks remain throughout, such as loss of funding or management commitment.

See Chapter 9

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 make the project difficult or destined to fail. Factors that contribute to high risk include:

- Using an unusual approach.
- Attempting to develop a system while furthering technology at the same time.
- Developing and testing new equipment, systems, or procedures.
- Operating in an unpredictable or variable environment.

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 often 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.[3] Comparing and selecting projects based upon criteria such as risk is discussed in Chapter 18.

See Chapter 18

Risk Sources

Any uncertain factor that can influence the outcome of a project is a *risk source* or *risk hazard*. Identifying risk sources involves learning as much as possible about things you know could go wrong and the outcome for each, as well as trying to identify things you don't already know—the "unknown unknowns."

Risk sources in projects can be classified as internal risks and external risks.

Internal Sources

These are sources of risk that originate inside the project and over which the project managers and stake-holders have some measure of control. They fall into three main categories: market risk, assumptions risk, and technical risk.

Market risk is the risk of not fulfilling market needs 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 the customer; thoroughly defining needs and requirements early in the project; closely monitoring trends and developments among markets, customers, and competitors; and updating requirements as needed 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 not meeting time, cost, or technical requirements, or resulting in unanticipated and harmful side effects.

Technical risk is the risk of encountering technical problems with the end-item or project activities. (Sometimes these risks are listed in special categories —*schedule risks* being those that would cause delays, *cost risks* those that would lead to overruns, and so on.) Technical risk is high in projects that involve new and untried technical applications, but is low in projects that involve familiar activities and technologies.

One approach to expressing technical risk is to rate the project end-item or primary process as being high, medium, or low according to the following features:[4]

- *Maturity*. How experienced or knowledgeable the project team is in the project technology. An end-item or process that takes advantage of existing experience and knowledge is less risky than one that is innovative, untried, or cutting edge.
- *Complexity*. How many steps, elements, or components the product or process contains 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.
- *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/or 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 (e.g. the discrete-staged approach is less risky than fast-tracking).

A sub-category of technical risks are health, safety, and environmental risks;

these include all hazards to project workers, the larger society, and the ecology as a consequence of the project. These risks stem from short-term hazards due to working conditions and procedures, and materials used in the project, and from long-term post-project hazards from the functioning, operation, or mere existence of the project end-item.

External Sources

These are risk sources that originate from outside the project and over which project managers often have limited or no ability to control them. 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
- physical environment (weather, terrain)
- labor availability (strikes and walkouts)
- material or labor resources (shortages)
- customer or subcontractor control over project work and resources.

Another important source of risk is stakeholders. By definition, whether situated internal or external to the project, stakeholders are affected by the project and, like the risk sources listed above, many of them are able to influence project outcomes—both positively and negatively. Identifying and working with stakeholders is discussed in Chapters 15 and 17.

See Chapter 15 and 17

Identification Techniques

Project risk sources (hereafter just called "risks") are identified in many ways; principle among them are project analogy, checklists, WBS analysis, process flowcharts, project networks, cause-effect diagramming, brainstorming, and the Delphi technique.

Project Analogy

The *project analogy* method involves scrutinizing the records, post completion summary reports, and project team members' recollections of earlier analogous projects to identify risks in upcoming projects. The more complete, accurate, and well-catalogued the documentation of past projects and the better peoples' memories, the more useful are these as sources 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 are being assessed. *Knowledge management* methods, described in Chapter 17, promote learning from past projects that can help anticipate risks in new ones.

See Chapter 17

Checklists

Documentation from prior projects is also used to create *checklists* of risk sources in projects. A checklist is originally created from the experiences from past projects and is updated as new experience is gained from recent projects. Risk checklists can pertain to the project as a whole or to specific phases, work packages, or tasks within the project.

To illustrate, the checklist in Table 10.1 shows the risk severity associated with three categories of risk sources: (1) status of implementation plan, (2) number of module interfaces, and (3) percentage of components requiring testing. Suppose, for example, an upcoming project will use a standard plan, have eight module interfaces, and test 16 percent of the system components. According to the checklist, the project will be rated as low, low, and medium, respectively, for the three risk sources.

The more experience a company or manager gains with projects, the more they learn about the risks, and the more comprehensive they can make the checklists. As experience grows with completed projects, the checklists are expanded and updated. 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 variant of the checklist is the *risk matrix*, a table wherein the columns are the project phases and the rows the sources of risks. The cells of the matrix indicate the presence, absence, or severity of specified risks for all phases of the project.

Table 10.1 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

A disadvantage of risk checklists is that people might look at only the risks listed and not consider any *not* on the list. Checklists therefore need to be supplemented by other methods.

Work Breakdown Structure (WBS)

Risks can be identified using the WBS. Each work package is scrutinized for potential technical hurdles or problems with management, customers, suppliers, equipment, or resource availability. It 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*. A flowchart illustrates the steps, procedures, and flows between tasks and activities in a process. Examination of a flowchart can pinpoint potential trouble spots and areas of risk.

FMEA and HAZOP

The Failure Mode and Effects Analysis (FMEA) method (see Chapter 9) can be used to identify conditions leading to system failure and, thus, subjecting the project, people, and the environment to risk. Another method called HAZOP—Hazard and Operability Study—is a rigorous investigation of a system to assess what happens when it starts up, shuts down, or encounters problems. The focus of the study is on the system design and possible errors, omissions, or inherent hazards. Both methods are widely used in technical projects; HAZOP is commonly used in process industries and infrastructure projects.

See Chapter 9

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* (Chapters 6 and 7). For example, risk sometimes increases at merge points in the network where work performed by different teams comes together and must be integrated; sometimes only then do problems become evident, such as subsystems produced by two teams not matching up or functioning correctly. The risk of project delay from this so-called "merge-point bias" is discussed in Chapter 7.

See Chapter 6 and Chapter 7

Brainstorming and Cause-and-Effect Diagram

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 in the project, and record them on a *cause-and-effect (CE) diagram* as shown in Figure 10.2. Brainstorming and CE diagrams are used in two ways: (1) given an identified, potential outcome (*effect*), to identify the potential *causes* (sources); (2) given a risk source (*cause*), to identify the outcomes that might ensue (*effects*). Figure 10.2 illustrates the first use: it shows potential sources leading to the effect of "completion delay."

The diagram in Figure 10.2 is divided into the generic risk categories of software, hardware, and so on (other categories are possible). Each category is subdivided into more fundamental sources of risk. In Figure 10.2, for example, the category "staff" includes the risk of "staff shortage," which could be caused by the inability to hire and train additional staff. Analysis techniques related to CE are further discussed in Chapter 9.

See Chapter 9

To encourage original thinking and generation of a comprehensive list of possible risks, the risks should *not* be assessed during brainstorming. Any early mention that a risk is "unrealistic" or "impossible" might lead to some very important risks being discarded. Hence, no risks should be assessed until the risk list has been compiled.

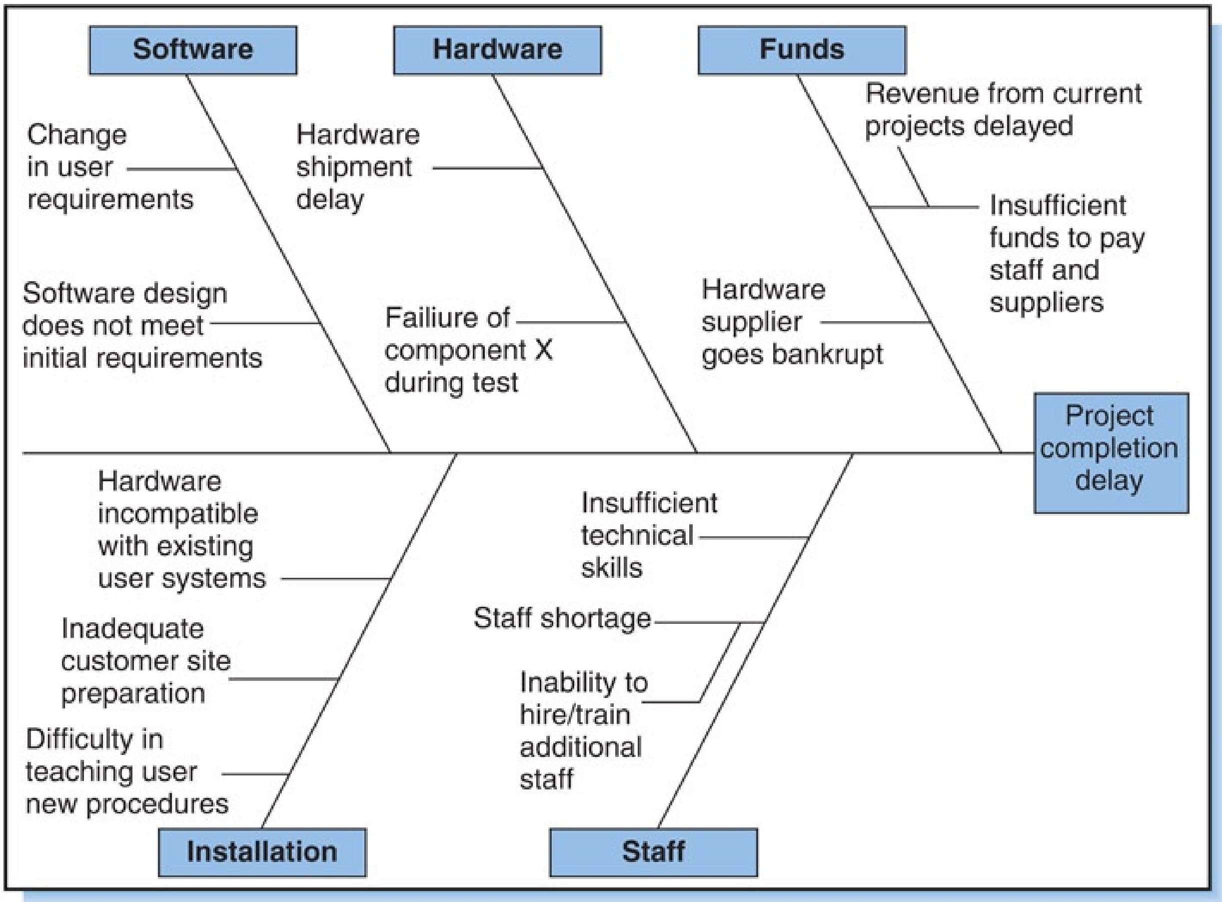


Figure 10.2 Cause-and-effect diagram.

Risks related to the project end-item may also be discovered during formal design reviews, which are discussed in Chapter 9.

See Chapter 9

Delphi Technique

The term *Delphi* refers to a group survey technique for combining the opinions of several people to develop a single judgment. The technique comprises a series of structured questions and feedback reports. Each respondent is given a series of questions (e.g. what are the five most significant risks in this project?), for which he writes his opinions and reasons. The responses of everyone surveyed are summarized in one report that is given to everyone. Seeing others' opinions, respondents have the opportunity to modify their own opinions. Because the 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 don't, they must also explain why. The process continues until the group reaches a collective opinion. Studies have proven the technique to be an effective way of reaching consensus.[5]

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."

10.3 Risk Assessment

Risks are ubiquitous, but it is only the notable or significant ones that 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 (risk tolerant) because they are ignorant of the risks or of their consequences.

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.[6] It can be expressed as a numerical value between 1.0 (certain to happen) and 0 (impossible) or as a qualitative rating such as high, medium, or low. Numerical values and qualitative ratings are sometimes used interchangeably. Table 10.2 shows an example: when, for instance, someone says, "the likelihood of this or that risk is low," that means the probability of it happening, according to the table, is 20 percent or less.

But Table 10.2 is an illustration only. The association between qualitative ratings and particular numerical values is subjective and depends on the experience of the project team and the risk tolerance of stakeholders. For example, Table 10.2 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. Often, people have difficulty agreeing on the qualitative rating for a given numerical likelihood value and vice versa, even when using the same information or experience; this is described later in Example 10.2.

Table 10.3 is a checklist for five potential sources of failure in computer systems projects and associated numerical likelihoods.[7] For example, 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. To repeat, the 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 10.3, the CLF can be computed as a weighted average,

$$CLF = (W1)\, M_H + (W2)\, C_H + (W3)\, M_S + (W4)\, C_S + (W5) D$$

where W1, W2, W3, W4, and W5 each have values 0 through 1.0 and sum to 1.0.

Table 10.2 Risk Likelihood: Qualitative Ratings for Quantitative Values

Qualitative	Numerical
Low	0–0.20
Medium	0.21–0.50
High	0.51–1.00

Table 10.3 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	Performance dependent on existing system
0.7 (significant)	Complex design; existing technology	New, but similar to 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

* M_S, failure likelihood due to immaturity of software; C_S, failure likelihood due to complexity of software; M_H, failure likelihood due to immaturity of hardware; C_H, failure likelihood due to complexity of hardware; 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.

Example 10.1: 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 10.3, $M_H = 0.1$, $C_H = 0.3$, $M_S = 0.5$, $C_S = 0.3$, and $D = 0.5$. If all sources are rated equally at 0.2, then

$$\text{CLF} = (0.2)0.1 + (0.2)0.3 + (0.2)\,0.5 + (0.2)0.3 + (0.2)0.5 = 0.34$$

The application of this CLF will be discussed shortly.

Note that the computation in equation (10.1) 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 10.3.[8] To construct the diagram, start with a list of previously identified risks (e.g. from Figure 10.2) and draw them as shown in Figure 10.3. Then look at each risk and ask whether it is influenced by, or has influence on, any of the other risks. If so, draw arrows between the related risks to indicate 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 small, about 15 or fewer. Risks with the most connections are the most important. In Figure 10.3, these would be risks I.2, S.1, and S.2; each is influenced by other risks, which increases the failure likelihood.

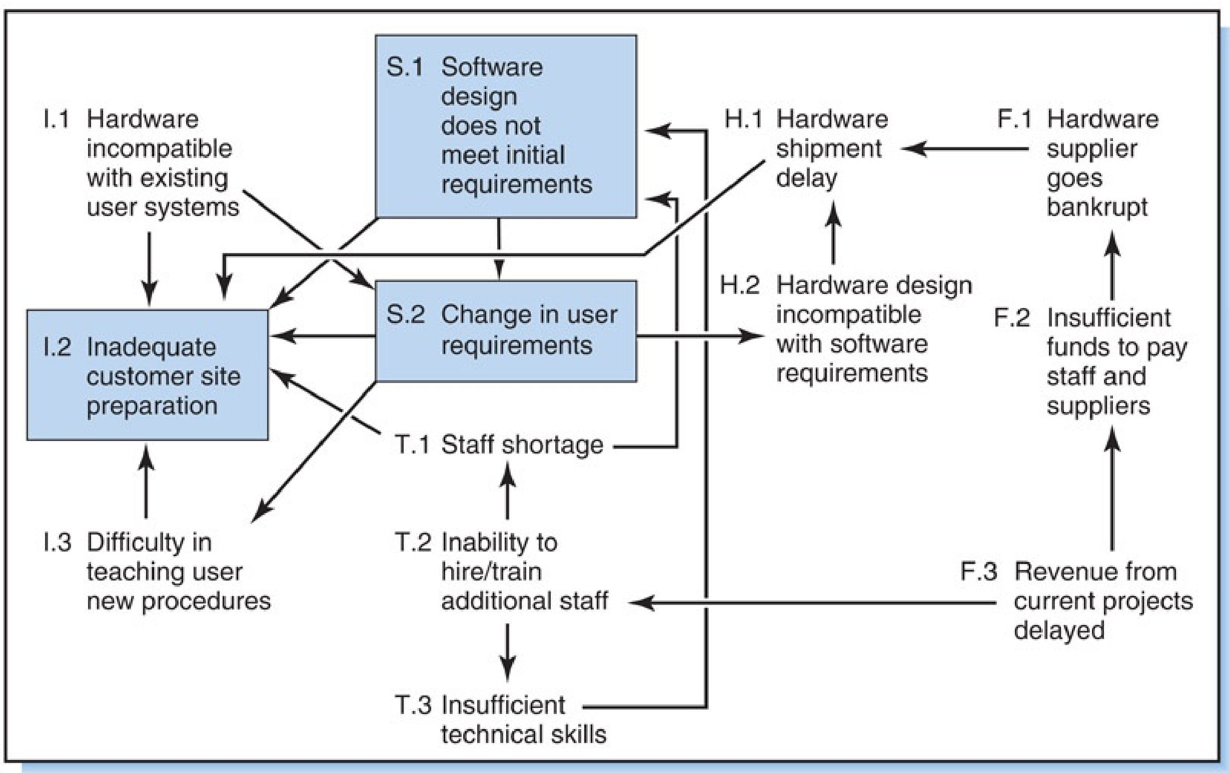


Figure 10.3 Influence diagram.

Risk likelihood also is affected by the future: *ceteris paribus*, activities planned further in the future are more risky (have greater likelihood of failure) than those closer at hand.[9] This is because activities farther in the future have greater chances of being influenced by unknowns. After the project enters execution and moves toward completion, the likelihood of failure diminishes. But there is a tradeoff: risk diminishes as the project progresses, but the *stakes* in the project—the amount of human and financial capital sunk into it—increase. This means that the loss incurred from failure later in the project will be much greater than the loss if incurred earlier.

Risk Impact

What would happen if a risk hazard materialized? 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, publicity, pollution, and so on. For example, the impact of insufficient resources might be failure to meet schedule or user requirements.

Risk impact can be expressed as a qualitative rating such as high, medium, or low based upon a manager's or expert's judgment about the impact. For example, a risk leading to a schedule delay of 1 month might be considered "medium impact," whereas a delay of 3 or more months might be deemed "high impact."

Table 10.4 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 possibly not achievable	>50% increase	Large (unacceptable)

Adapted from Roetzheim, W. *Structured Computer Project Management.* Upper Saddle River, NJ: Prentice Hall; 1988, 23–26.

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 10.4, 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.[10]

The assigned risk impact values are largely subjective—even when derived from empirical data.

Example 10.2: Estimating Risk Likelihood and Risk Impact in New Technologies

Risk assessment in new technologies is, well, difficult. The risk of a serious problem can stem from a chain of events (e.g. a machine malfunctions, a sensor does not detect it, an operator takes the wrong action), and to assign the probability of the risk requires identifying all the events in the chain, estimating the probability of each, and combining the probabilities together. Managers and designers can try to think of every event, but they can never be sure that they haven't missed some. When a project involves new technologies, the estimates are largely guesses. In 1974 MIT released a report stating that the likelihood of a 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 later a reactor at Three Mile Island suffered a partial meltdown and released radioactivity into the atmosphere.[11]

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.[12] 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.[13] 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 weighted average,

$$CIF = (W1)TI + (W2)CI + (W3)SI$$

where W1, W2, and W3 have values 0 through 1.0, and together sum to 1.0. CIF will range from 0.0 to 1.0, where 0 means "no impact" and 1.0 means "the most severe impact."

Example 10.3: Computation of CIF

A particular failure in the ROSEBUD project to meet certain technical goals is expected to minimally impact technical performance and be corrected within 2 months at a cost of 20 percent. Therefore, from Table 10.4:

TI = 0.1, SI = 0.5, CI = 0.5

Suppose the most important criteria are technical performance, followed by the schedule, then cost, and the weights assigned to the criteria are 0.5, 0.3, and 0.2, respectively. Therefore, from equation (10.2):

$$CIF = (0.5)(0.1) + (0.3)(0.5) + (0.2)(0.5) = 0.22$$

Equation (10.2) 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 CLF 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 a new technology poses a risk of not meeting performance requirements. The plan is to try out the technology, but to abandon it and use a proven approach if early tests reveal poor performance. 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 special 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 possible ways to reduce or prevent risk hazards, as discussed later.[14]

Risk Consequence

Earlier, the notion of risk was defined as being a function of risk likelihood *and* risk impact; the combined consideration of both is referred to as the *risk consequence* or *risk exposure*.

The most common way, mathematically, to express risk consequence is,

$$\text{Risk consequence} = (\text{Likelihood}) \times (\text{Impact})$$

Using the previously computed likelihood (CLF) of 0.34 (Example 10.1) and impact (CIF) of 0.22 (Example 10.3), the risk consequence rating, RCR, is

$$RCR = (CLR) \times (CIR) \quad (10.4) = (0.34) \times (0.22) = 0.078$$

RCR ranges in value from 0 to 1.0, and a very small RCR such as 0.0748 might be judged as “inconsequential.” Assessing values of RCR as being high, medium, or low consequence is subjective, and the principle use of RCR is to compare and prioritize risks—to separate those that can likely be ignored (small RCR, low consequence) from those that must be heeded (large RCR, high consequence).

The 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 delaying the project by 4 months and increasing the cost by $300,000. The risk consequences for time and cost are thus

Risk consequence time (RT) = (4 months)(0.40) = 1.6 months = 6.4 weeks 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 a large number of 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., RC = $5), but would you still play it with a 10^{-6} percent chance of losing $5,000,000 (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 10.4. 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 18; a quick comparison of Figure 10.4 and Figure 18.5 reveals the similarly.

See Chapters 9 and 18

The method is also similar to the Failure Mode and Effect Analysis (FMEA) technique discussed in Chapter 9. Both methods identify and analyze the consequences of risk, although FMEA is directed specifically at risks in technical systems.

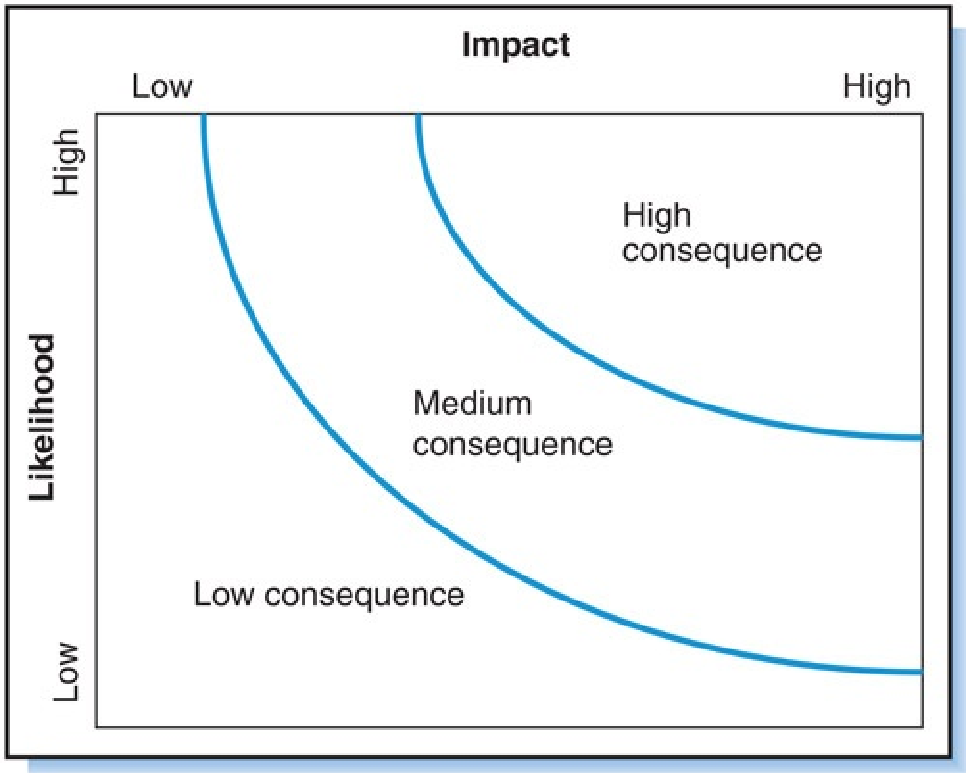


Figure 10.4 Risk consequence as a function of likelihood and impact.

PERT

The PERT and Monte-Carlo simulation methods discussed in Chapter 7 can be used to account for risk in project scheduling and to estimate additional time needed to compensate for risks in meeting project deadlines.

See Chapter 7

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 especially between *m* and *b*. For an 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* to account for greater risk will result in a larger value of t_e. This logically allows more time to complete the activity and compensate for any 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

Projects are subject to numerous risks, yet only relatively few might be important enough to merit attention. To decide upon which risks to focus, management might specify a level of expected risk consequence and address only risks at that level or higher. For example, if the level were set at a consequence of 2 or more days of delay in the project, then only risks S1, F1, T1, T3, I1, I2, and I3 in Table 10.5 would be addressed.

Based upon the computed risk consequences, the risk sources can be listed on a *risk register* or *risk log,* and those with moderate-to-high consequences given a careful look. Project team members, managers, subcontractors, and customers review the sources and prepare appropriate responses to them. Table 10.6 illustrates an example risk register showing rank-ordered risk sources and mitigation measures.

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 a low priority.[15]

In a complex system with a large number of relationships where joint failures in several 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 not the same as 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.[16] Similarly, the nuclear meltdown 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 more throughout Europe and Asia.[17] The lesson: any risk with a severe impact should never be ignored, no matter how small the likelihood.

Table 10.5 Risk Likelihood, Risk Impact, and Expected Value Consequence

		Likelihood (L) (%)	Impact (I) Days Late	Consequence: L × I (Days)
	Software			
S1	Design does not meet initial requirements	20	10	2
S2	Change in user requirements	30	5	1.5
	Hardware			
H1	Hardware shipment delay	en	5	0.25
H2	Hardware incompatible with software	5	10	0.5

Funds				
F1	Hardware supplier goes bankrupt	5	40	2
F2	Insufficient funds for staff and suppliers	5	15	0.75
Staff				
T1	Staff shortage	10	20	2
T2	Inability to hire/train additional staff	15	10	1.5
T3	Insufficient technical skills	10	30	3
Installation				
I1	Hardware incompatible with user systems	5	60	3
I2	Inadequate customer site preparation	20	10	2

10.4 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, accept, or contingency plan for it.

Transfer Risk

Risk can be transferred between customers, contractors, and other parties using contractual incentives, warranties, penalties, or insurance.

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 as a consequence of the project.
- Damage to materials while in transit or in storage.

Table 10.6 Risk Registry

Risk ID Number	Risk Source or Condition (Ranked, Highest Consequence Rating 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 perform to 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 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 of process. Dedicate proper staffing. Training early in project.

- Breakdown or damage of equipment.
- Theft of equipment and materials.
- Sickness or injury of workers, managers, and staff.
- Fluctuations in exchange rates or "forward cover" (see Chapter 19).

See Chapter 19

Subcontract Work

Risks arise from uncertainty about how to approach a problem or situation. One way to avoid such risk is to hire contractors that specialize in handling those 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 those components. This relieves the manufacturer of the financial risk associated with the tooling and equipment to produce these components. But as mentioned, transfer of one kind of risk often means inheriting another kind. For example, in subcontracting work for the components, the manufacturer now must rely on outsiders, which increases the risks associated with quality control and scheduling. But such risks often can be reduced through careful management of the subcontractors.

Contract Type

Risk can be transferred or allocated through the use of an appropriate contract type, as discussed in the Appendix to Chapter 3. When the statement of work is clear and involves little uncertainty, the contractor will readily accept a *fixed price* contract. An example would be the building of a wall according to a well-defined drawing and specifications, in which case the contractor perceives little risk. However when the scope of the work is unclear and the potential for change is great, the contractor is less likely to commit to a fixed price and take on the risk of an overrun. In such cases the contractor would find a *cost-plus* contract more appropriate since it covers all expenses incurred during the project.

See Chapter 3

Whereas in a fixed-price contract the contractor assumes most of the risk for cost overruns, in a fixed-price with incentive fee contract the contractor accepts roughly 60 percent of the risk, and the customer 40 percent. In a cost plus incentive fee contract, the contractor assumes about 40 percent, the customer 60 percent. In a cost plus fixed fee (CPFF) contract the customer assumes most or all of the risk of an overrun.

In large projects, a variety of contracts are used depending on the risk associated with individual work packages or deliverables. In the Chunnel project, the most uncertain part was tunneling under the English Channel; thus, that part of the work was contracted on a CPFF 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 a cost plus percentage fee contract.[18]

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 has to pay for it. To avoid losses, a contractor might feel pressured to cut corners, which of course increases the customer's risk of receiving a sub-par quality end-item. To lessen such risks, the contract must stipulate strict quality inspections and penalties.

Risk Responsibility

The individual or group held responsible for each particular risk in a project should be specified. Risks may be transferred, but they can never be completely "offloaded." A warranty or guarantee 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 as long as the item is on the ship; as soon as it is hoisted over the rail of the ship the risk is transferred to the buyer.

A party willing to accept responsibility for high risk in a project will usually demand a high level of *authority* over the project. A customer agreeing to accept the risk of poor 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 usually also insist on *compensation* to cover the risks. The CPFF contract illustrates: 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 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. Many risk factors can be avoided, but not all, especially in complex or leading-edge projects. Research and new product development projects are inherently risky but offer potential for huge benefits later on. Because the size of the risk is often proportional to the potential payoff, rather than avoiding risk it is better to try to reduce risk to an acceptable level.

Reduce Risk

Among the ways to reduce technical risk (its likelihood, impact, or both) are to:[19]

- 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, system 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 the risk. For example, *decoupling of activities* and subsystems, i.e., making them independent of one another, prevents a failure in any 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.[20] A design margin is a quantified value that serves as a safety buffer held in reserve and allocated by management. 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:

Target value = Requirement + 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 10.4: Design Margin Application for the Spaceship

The weight requirement for the spaceship navigation system is 90 lbs. 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 pound that is fixed and cannot be reduced. But A and B are being newly developed, and their design goals have been set at 50 pounds for A and 30 pounds 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 pounds, 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 margin values 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 encourage designers to exceed requirements—e.g. to design systems that weigh less than required. Of course, the margins must be carefully set so as to reduce the risks while not increasing the costs.

Design margins focus on risks associated with meeting technical requirements. Among ways to reduce risks associated with meeting *schedules* are:[21]

- Create a master project schedule and strive to adhere to it.
- Schedule the most risky 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.
- Provide project and feeding buffers (contingency reserves), as discussed in Chapter 7.

See Chapter 7

To reduce the risk associated with meeting *budget* or *cost* targets:[22]

- Identify and monitor the key cost drivers.
- Use low-cost design alternative reviews and assessments.
- 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.

The last way is especially powerful for reducing risk. *Breadboards* and *prototypes*, i.e., test mockups and models, enable ideas to be tested experimentally so designs can be corrected early in the project.[23] This greatly reduces the need for later design changes, which can be costly. The following illustrates other ways to reduce risk.

Example 10.5: Managing Schedule and Cost Risk at Vancouver Airport[24]

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 of 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 new 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.*[25] The project team might be aware of many things that could be done beyond the stated requirements, but in most cases these will consume additional resources and add time and cost. Unless the customer approves the added time and cost, these things should be avoided.

Contingency Planning

Contingency planning implies anticipating whatever risks might arise and then preparing a course of action to cope with them. The initial project plan is followed, yet throughout execution the risks are closely monitored. Should a risk materialize as indicated by a trigger symptom, the contingency action is adopted. The contingency can be a post-hoc remedial action to compensate for a risk impact, an action undertaken in parallel with the original plan, or a preventive action initiated by a trigger symptom to mitigate the risk impact. Multiple contingency plans can be developed based upon "what-if" scenarios for the 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 10.4, 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). Sometimes nothing can be done to avoid, reduce, or transfer a risk, in which case the risk must be accepted, regardless of the consequence. Fortunately such situations are rare.

Responding to a risk sometimes creates a new, *secondary risk* (see Example 11.1 in the next chapter). When planning risk responses, the project management team should check for such risks before implementing the plan.

See Chapter 11

10.5 Risk Monitoring and Response

Identified risks are documented and listed in the *risk log* or *risk register* and rank ordered, greatest risk consequence first. For risks with the most serious consequences, mitigation plans are prepared and strategies adopted (transfer, reduce, avoid, or contingency); for those of little or no consequence, nothing is done (accept).

The project should be *continuously monitored* for symptoms of previously identified risks as well as newly emerging risks (not previously identified). Known risks may take a long time before they 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 organize a meeting to pick a solution. Sometimes the response is to do nothing; however, nothing 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 is 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.

10.6 Project Management *Is* Risk Management

Risk management supplements and is a part of other project management practices such as requirements and work definition, scheduling, budgeting, configuration management, change control, and performance tracking and control. With all of these, managers identify and assess the risks so they can proactively reduce them or plan for the consequences. If, for example, a project must be completed in 9 months but is estimated to take closer to 12, management can take a multitude of steps to increase the likelihood of it 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 as part of the master execution plan alongside the schedule, budget, quality management plan, change control and configuration management plan, communications plan, and so on. During project execution, risk tracking is incorporated as a measure in the project tracking and control process. Ideally, many project team members and other stakeholders are involved in risk identification, response planning, and risk tracking.

Of course, not all projects *need* comprehensive risk management. On small projects, a small, well-paid and motivated staff can usually overcome difficulties associated with the risks and, if not, the consequences are usually small anyway. In larger projects, however, where the stakes and risks of failure are high, risk management is especially important. These projects require awareness and respect for all the significant risks—safety, legal, social, and political, as well as technical and financial.

Risk Management Principles

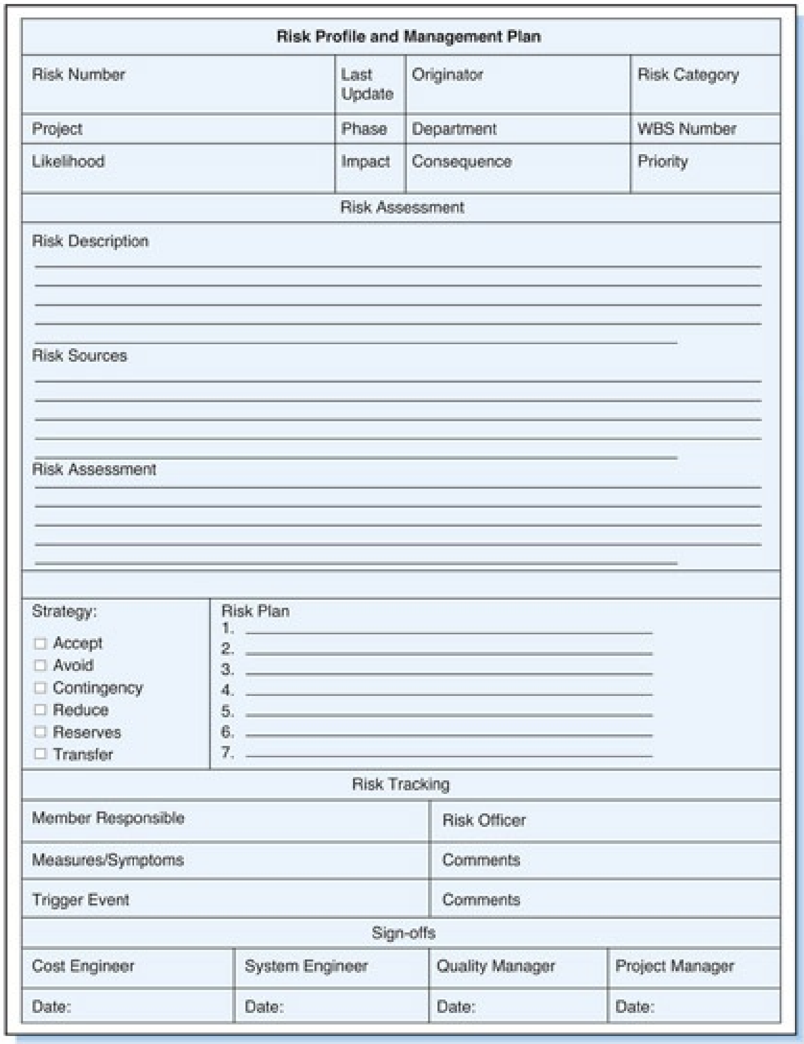
Every project for which non-trivial risks are known or suspected should have a risk management plan. The plan should specify for a particular project procedures for identifying and assessing risks, person(s) involved in the risk management process and specific responsibilities, methods for assessing 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, such as the following:[26]

- Create a *risk 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 symptoms (trigger events) that would indicate when the risk is materializing. In general, high-risk sources should have lots of eyes watching closely, and contingency plans should be updated to reflect project progress and emerging risks. Figure 10.5 illustrates the template for a risk profile—a summary of everything known about a risk. This document would be retained in a binder or library, updated as necessary until the risk is believed to no longer exist and is "closed out."
- Appoint a *risk officer* to the project, someone whose principle responsibility is the project's risk management. This should not be the same person as the project manager; he should *not* be a can-do person but, instead, a devil's advocate, identifying and tracking all the reasons why something might not work—even when everyone else believes it will.
- Include in the budget and schedule a calculated *risk reserve*—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 (described in 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 exact amounts held in the reserves strictly confidential (else a project will tend to consume whatever amount is available), although others should know *there is a reserve available* (otherwise they will build in their own *secret* reserves).
- Establish *communication channels* (perhaps anonymous) within the project team to ensure any bad 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, detailed 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 future, similar projects and identifying possible risks.

Expect the Unexpected

Having identified myriad risk hazards and consequences and prepared all kinds of controls and safeguards, people can be led to believe that everything that possibly could go wrong has been anticipated and covered; 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



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 10.5 Document for the profile and response plan of an identified risk.

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.[27]

Example 10.6: Managing Risks as They Arise—Development of the F117 Stealth Fighter[28]

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 risk because many lessons had to be learned during the program and significant challenges had to be overcome. But the program managers *expected* challenges throughout the program, from early design and test, through evaluation and final deployment. To handle the risks, numerous decisions were made on the spot between program managers for Lockheed (contractor) and the Air Force (customer). The program was set up for rapid deployment of resources to solve problems *as they arose*. Managers from the customer and the contractor worked closely to minimize bureaucratic delay. Schedules were optimistic and based on assumptions that everything would work, however everyone throughout the management chain *knew the risks* and the challenges to overcome, so problems never came as a surprise or threatened program support. This is a good example of *managing* risk as opposed to *avoiding* risk.

Risk Management Caveats

For all the good it can provide, risk management can *create* 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 actually 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 peoples' 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 cannot be instructed to analyze situations that are beyond human imagination. Experience provides but 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 disastrous, 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."[29] The appropriate risk management strategy for most projects is to try to accommodate and mitigate risk, not to avoid or eliminate it.

10.7 Summary

Project risk management involves identifying the risks, assessing them, and planning the taking of 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 or market requirements, technical problems arising in the work, weather, labor and supplier problems, competitors' actions, and changes imposed by outsiders. Such risk hazards are identified using a variety of methods and draw from experience with past projects and scrutiny of planned projects.

Of innumerable risks in projects, only the important ones need 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 effect of the risk; 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 very small.

Risk response planning addresses the ways identified risks will be handled. Some risks can be transferred to other parties or spread among many stakeholders or subcontractors. Some can be avoided; some should be eliminated. But sometimes 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 position responsible for identifying and tracking the risks, and a budget and schedule reserve. 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; 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 18.

See Chapter 18

Appendix: Risk Analysis Methods

Four common methods for risk analysis 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 and schedules.

An expected value is the average or mean outcome of numerous repeated circumstances. For risk assessment, expected value represents the average outcome of a project if it were repeated many times, accounting for the possible occurrence of risk. Mathematically it is the weighted average of all the possible outcomes, where the weights are the likelihoods of the possible outcomes, that is

$$\text{Expected value} = \Sigma[(\text{Outcomes}) \times (\text{Likelihoods})]$$

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

$$\text{RT} = (\text{Corrective time}) \times (\text{Likelihood})$$

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

$$\text{RC} = (\text{Corrective cost}) \times (\text{Likelihood})$$

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 as a whole is 0.3, and, if the risk materializes, the project would be delayed 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 nil. Hence

RT = (5)(0.3) + (0)(0.7) = 1.5 weeks,

and

RC = ($10,000)(0.3) + (0)(0.7) = $3,000

These figures, RT and RC, are the *schedule reserve* and *project contingency* (budget reserve), respectively, mentioned in Chapters 7 and 8.

See Chapter 7 and Chapter 8

Accounting for the risk time, the *expected project completion time, ET*, is

ET = BTE + RT = 26 + 1.5 = 27.5 weeks.

Accounting for the risk cost, the *expected project completion cost, EC*, is

EC = BCE + RC = 71,000 + 3,000 = $74,000

When the corrective time and cost cannot be estimated, then ET and EC are computed as

$$\text{ET} = \text{BTE}(1 + \text{likelihood}) = 26(1.3) = 33.8 \text{ weeks}$$

$$\text{EC} = \text{BCE}\ (1 + \text{likelihood}) = \$71{,}000(1.3) = \$92{,}300$$

These examples account for risk factors that affect the project as a *whole*. Another way to determine risk consequence is to first disaggregate the project into work packages or phases and then *for each* estimate 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 (10.6) through (10.9) because risks so pinpointed to individual tasks or phases can be more accurately assessed, and the necessary corrective actions and associated time and costs for particular tasks are easier to identify.

Say a project has eight work packages; the following table lists cost information and EC for each, where EC is computed as

EC = BCE + [(corrective cost) × (likelihood)]

As shown in the table, the EC for the *project* is $84,850.

Now, for the same eight work-package project, the next table gives time information, where ET is computed as

ET = BTE + [(Corrective time) × (Likelihood)]

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

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

Suppose the project network is as shown in Figure 10.6. Not considering the risk time, the critical path would be J–M–V–Y–W–X, which gives a project BTE of 26 weeks. Accounting for risk consequences, the critical path does not change but the duration increases to 27.9 weeks. This is the project ET.[30]

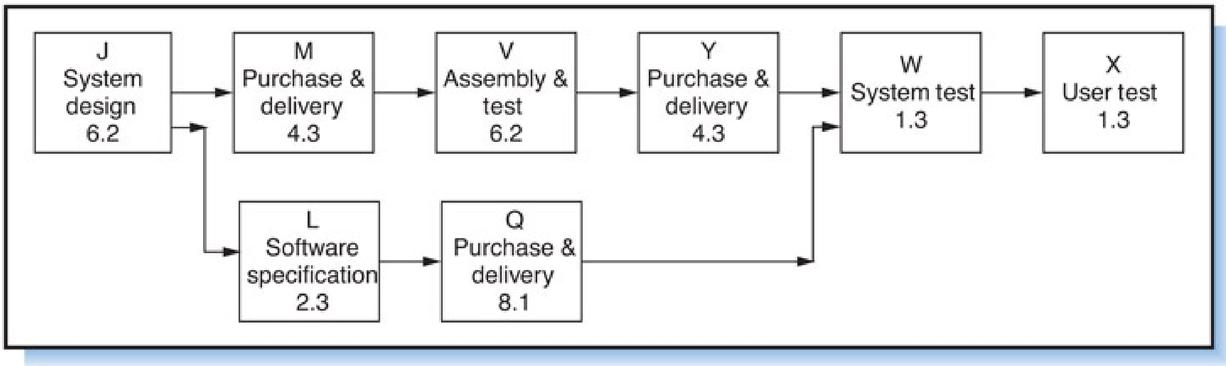


Figure 10.6 Project network, accounting for risk time.

Although activities on critical and near-critical paths should be carefully monitored, in general, *all* activities with high-risk consequences (high likelihood and/or high impact) should also be carefully 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 the equivalent to the long-run averages, which result from repeating something many times; this is questionable in projects, since seldom are project activities identically repeated.

Decision Trees[31]

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 consequence.

One application of decision trees is to weigh the cost of potential 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 failure likelihood to 0.1.
> Strategy 2 will cost $1 million and will reduce the failure likelihood to 0.4.

The decision tree and resultant expected project costs are shown in Figure 10.7. 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 project selection, discussed in Chapter 18.

See Chapter 18

Uncertainty and Payoff Tables

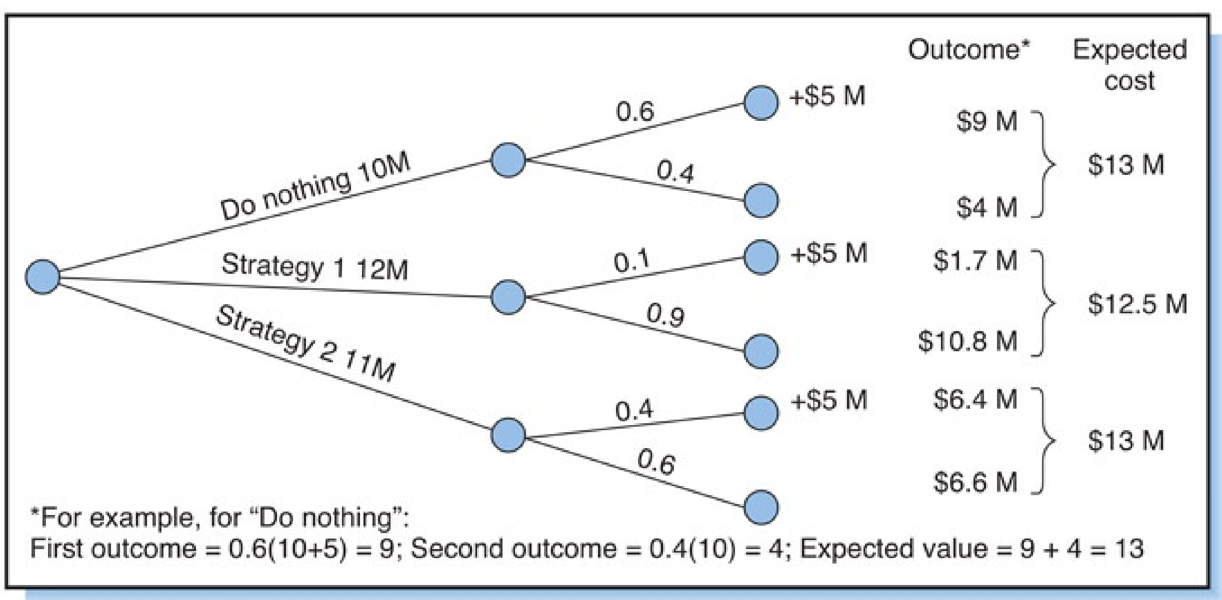


Figure 10.7 Decision tree.

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 what might occur. To determine the best strategy under uncertainty, begin by identifyingpossible 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 indicate 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 particular performance features of the product. The development effort can be directed in any of three possible directions, referred to as strategies A, B, and C, each of which will provide the product with 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 products enter 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 (shown in Table 10.7).

The question: Which strategy should be adopted? The answer: It 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 of Nature N1. Thus, optimistic project sponsors will adopt Strategy C. 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.

Table 10.7 Payoff Table

	States of Nature		
Strategy	N1	N2	N3
A	60	30	−20
B	60	50	60
C	90	70	40

Table 10.8 Regret Table

	States of Nature		
Strategy	N1	N2	N3
A	30	40	80
B	30	20	0
C	0	0	20

Any choice of strategy other than the best one 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 10.8. For example, given the payoffs in Table 10.7, 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, pick the smallest of these, $20 million, which occurs for Strategy C. Thus, Strategy C is the best choice in terms of minimizing 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 back to the payoff table, Table 10.7, 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

$$1/3(60) + 1/3(30) + 1/3(-20) = 23.33, \text{ or } \$23.33 \text{ million}$$

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 point to Strategy C. This in itself might convince decision makers that Strategy C is most appropriate.

Simulation

Application of simulation to project management, illustrated in Chapter 7, gives the probability distribution of 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 7, example 7.2, indicated the project would be completed in 147 days, the simulated completion time distribution (Figure 7.14) 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 date by which the project must be completed, simulation can be used to estimate the likelihood of failure and, hence, determine whether to prepare contingency plans or change the project requirements, activities, or network.

See Chapter 7

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 involves 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 10.3, 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 10.3 and 10.4 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. Do equations (10.1), (10.2), and (10.3) present good ways for rating the overall likelihood, impact, and consequences of risk? Discuss pros and cons of using these equations.

13. Discuss briefly each of the following ways to handle risk: transfer risk, avoid risk, reduce risk, contingency plan, 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 will 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 that Subsystems Y and Z can be designed to meet their respective design targets, can the output requirement for the overall system 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 10.8 below is the network for the Largesse Hydro Project:

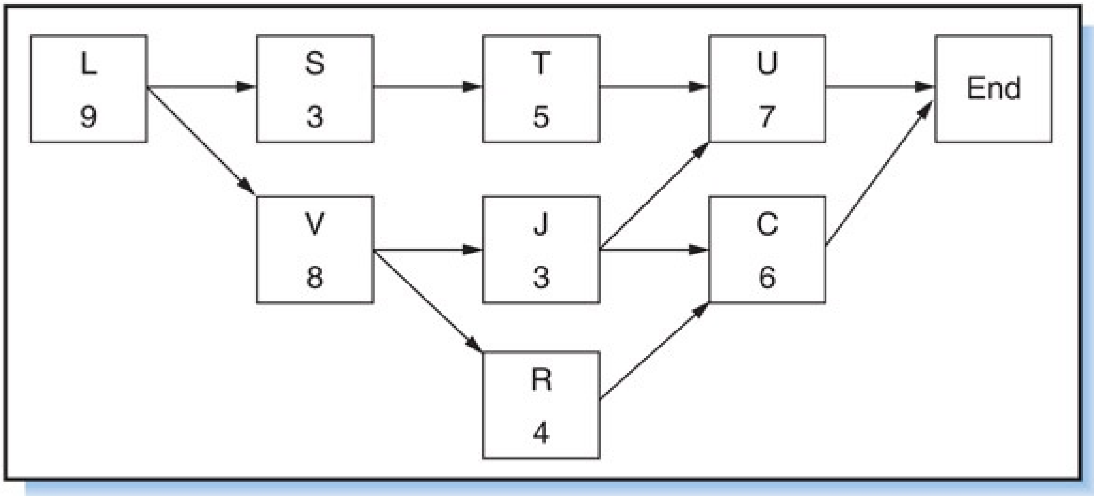


Figure 10.8 Largesse Hydro Project.

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	Corrective 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 is 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 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 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. Set up a decision tree to show 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 has been 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" increases the project price and decreases 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 (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. 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:

	Creative Robotics Workload

Payoffs: Strategy	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 similar to the 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 is in contrast to 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 risks identified?
7. How were risks dealt with (through risk transfer, acceptance, avoidance, reduction, etc.)?
8. Discuss the use of contingency plans and budget and schedule reserves to cover risks.
9. What risks materialized during the project and how were they handled?

Case 10.1 The Sydney Opera House[32]

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 a magnificent structure (Figure 10.9), but it was nightmare to design and build.

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 prior 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; some experts questioned that it could be built at all. Interestingly, because the design was *so* unique, some people thought it would also be inexpensive to build. The initial cost was estimated at $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 strong winds blowing up the harbor. Thus, they would have to be carefully designed and constructed to prevent the building from blowing away!

Figure 10.9 Sydney Opera House.

Source: iStock.

Government managers worried that people scrutinizing the design might raise questions about potential problems and stall the project. They thus quickly moved ahead and divided thework into three main contracts: the foundation and building except the roof, the roof, and the 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.

Case 10.2 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 three customer focus groups. The department is excited about the new concept and presents it to top management, who approves it for further study. Lisa Denney, senior director of new 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.

After a few weeks 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 even manufacture the product on its own. Lisa checks out several design/development firms, asking one, Margo-Spinner Works Company, MSW, to review the product concept. MSW assures Lisa that although the technology is new to them, it is well within their capability. Lisa reports this to top 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 be given *complete* control of the development work. Lisa feels uncomfortable with the proposition, but knows of no other design company 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 other parties did that served to increase or decrease the risks.

Case 10.3 The Nelson Mandela Bridge[33]

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 10.10) has been acclaimed for its functionality and beauty.

Lack of space for the support pylons (towers) between the railway lines dictated that the bridge design would have a long span. This resulted in a structure with the bridge deck supported by stay cables from pylons of unequal height. The pylons on the northern side are 48 meters high and those in the southern side are 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 the 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 being near continuous rail operations. The pumping method posed the risk of the stone aggregate and cement in the concrete mixture segregating 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.

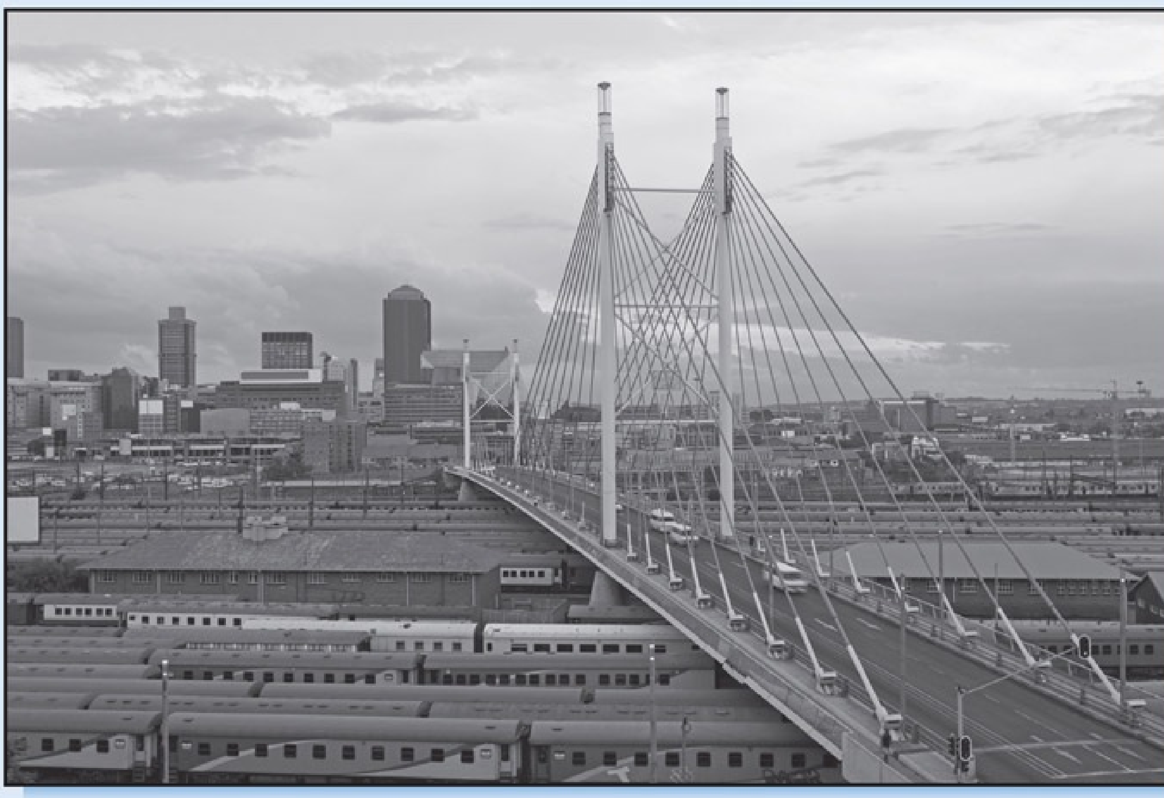


Figure 10.10 Nelson Mandela Bridge, Johannesburg.

Source: iStock.

The concrete had to be transported by trucks to the site, which risked interrupting the concrete supply owing to traffic congestion in the city.

Despite working over a busy yard with trains running back and forth, no serious accident occurred at any time in the 420,000 labor-hours project. The pump never failed and construction finished on time. The stay cables—totaling 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 completion of the bridge, some felt that the costs incurred to reduce the risks had been excessive; others held that the risks were too high and not enough had been done to reduce them.

Questions

1. How would you have identified the risks? (Refer also to methods in [Chapter 9](#).)

   See [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.

Possible Risk Event	Plans to Address Risk				
	Accept	Avoid	Reduce	Transfer	Contingency Plans and/or Contingency Reserves
Failure to make an acceptable profit					
Not finishing the construction by Nelson Mandela's 85th birthday					
Interference with rail activities					
Geological structures necessitating expensive foundations					
The concrete mixture segregating when pumped into the columns					
A pump failure while concrete is being pumped					
Interrupted supply of concrete due to trucks transporting concrete delayed in traffic					

3. State whether the risks listed in the table above are internal or external.
4. Describe how you would determine the expected values of the risks

listed in the table.

5. Compile a complete list of information that you would require in order to make an assessment of 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 different 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.
11. How could a manager protect himself against the risk of making a decision that might later have negative implications?

Endnotes

1. Quoted in Bernstein P. *Against the Gods: The Remarkable Story of Risk*. New York: John Wiley & Sons; 1996, p. 331.

2. 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.

3. See Argus R. and Gunderson N. *Planning, Performing, and Controlling Projects*. Upper Saddle River, NJ: Prentice Hall; 1997, pp. 22–23.

4. Adapted from Michaels J. *Technical Risk Management*. Upper Saddle River, NJ: Prentice Hall; 1996, pp. 208–250.

5. Turoff M. and Linstone H. (eds). *The Delphi Method: Techniques and Applications*, 2002, http://is.njit.edu/pubs/delphibook/

6. 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 reliable because humans often can do a pretty good job of assimilating lots of factors.

7. 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*.

8. See Dingle J. *Project Management: Orientation for Decision Makers*. London: Arnold; 1997.

9. See Gilbreath R. *Winning at Project Management: What Works, What Fails, and Why*. New York: John

Wiley & Sons; 1986.

10. Roetzheim, *Structured Computer Project Management*, pp. 23–26.

11. Pool R. *Beyond Engineering: How Society Shapes Technology*. New York: Oxford University Press; 1997, pp. 197–202

12. Kotulak R. Key differences seen in Columbia, Challenger disasters. *Chicago Tribune*; Feb. 2, 2003, Section 1, p. 5.

13. Pool, *Beyond Engineering*, pp. 207–214

14. Michaels, *Technical Risk Management*, p. 40.

15. 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.

16. Mitroff I. and Linstone H. *The Unbounded Mind*. New York: Oxford; 1993, pp. 111–135.

17. Ibid.

18. Anbari F. (ed.). *Case studies in Project Management*: *The Chunnel Project*. Newton Square, PA: Project Management Institute; 2005.

19. Eisner H. *Computer-Aided Systems Engineering*. Upper Saddle River, NJ: Prentice Hall; 1988, p. 335.

20. See Grady J. *System Requirements Analysis*. New York: McGraw-Hill; 1993, pp. 106–111.

21. Eisner, *Computer-Aided Systems Engineering*, p. 336.

22. Ibid.

23. 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 feasibility of a design concept. Breadboards, prototypes, and modeling are discussed in Chapters 2 and 9.

See Chapter 2 and Chapter 9

24. Wakabayashi H. and Cowan B. Vancouver International Airport expansion. *PM Network*; September, 1998, pp. 39–44.

25. Whitten N. Meet minimum requirements: anything more is too much. *PM Network*; September 1998, p. 19.

26. 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.

27. Dorner D. *The Logic of Failure*. Reading, MA: Addison-Wesley; 1997, p. 163.

28. 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.

29. Ibid., pp. 186–190.

30. For other approaches to risk time analysis, see Michaels, *Technical Risk Management*.

31. 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.

32. Adapted from Kharbanda O. and Pinto J. *What Made Gertie Gallop: Learning from Project Failures*. New York: Van Nostrand Reinhold; 1996, pp. 177–191.

33. *Source*: Kromhout F. Divisional Director, Bridges, BKS (Pty) Ltd, Pretoria.

Chapter 11
Project Execution, Monitoring, and Control

The topics of the preceding six chapters fall largely under the realm of planning and are initially addressed in the conception and definition phases of the project life cycle. By the end of the definition phase the project manager and team will have prepared a full set of requirements and specifications and a somewhat complete plan for the most immediate stages of the project, as well as an outline for the remaining stages, which constitute the *execution phase* or "delivery" part of the project life cycle. What happens in the execution phase, and what most outsiders see when they look at a project, is the first topic of this chapter.

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. Keeping the project moving and on target, tracking progress, and overcoming obstacles is the purpose of *project control,* which is the other topic of this chapter.

11.1 Phase C: Execution

The execution phase typically includes the stages of *detail design, production/build*, and *implementation* (Figure 11.1), though, in actuality, the stages differ depending on the purpose of the project. In hardware development projects the stages are typically design, development, and production; in construction projects they are design and build; and in consulting they are background research, report compilation, and presentation. Many companies have customized project methodologies with their own unique project phases and stages; these are discussed in Chapter 17. All projects that produce a physical end-item—a product, building, or system—also include an implementation stage wherein the end-item is handed over to the user. This chapter looks at the stages of design and production/build; the next chapter covers the implementation and project closeout.

See Chapter 17

11.2 Detail Design Stage

In the detail design stage, system specifications are converted into plans, sketches, or drawings. The outputs of this stage are pictorial forms—blueprints, flow charts, or schematic diagrams—or models showing the end-item system's components, dimensions, relationships, and configuration.[1]

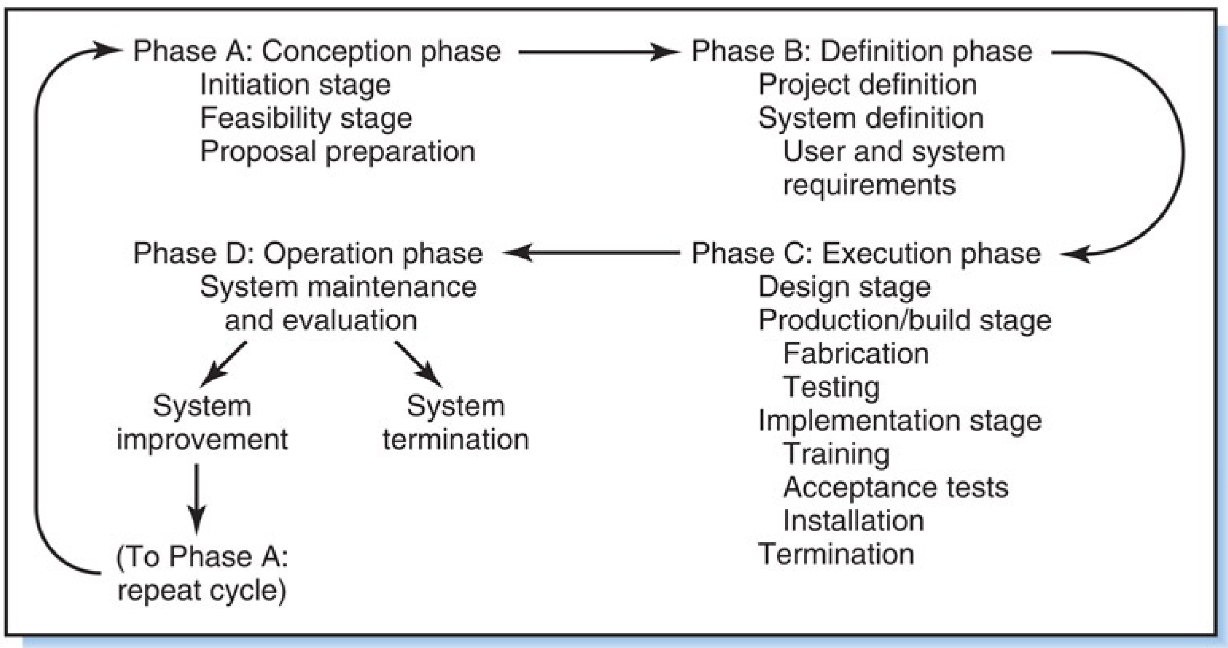


Figure 11.1 Phase and stages systems development life cycle. Phases A, B, C = project life cycle.

During the design stage, the system is broken into tiers of subsystems, components, and parts. Various design possibilities for elements at each tier are reviewed for compatibility with each other and elements at higher-level tiers, and ability to meet system specifications and cost, schedule, and performance requirements. The breakdown into tiers and components uses earlier-mentioned tools (block diagramming, Chapter 2; requirements breakdown structure, Chapter 3; and work breakdown structure, Chapter 5).

See Chapters 2, 3 and 5

The design process is composed of two interrelated activities. First is preparation of a *functional* design that shows the system components and their relationships. The purpose of this design activity is to determine the *logical*, functional elements of the system and how they should be interconnected to achieve the system's objectives. This is the thrust of systems engineering (Chapter 2) and system definition/FEL-3 (Chapter 4).

See Chapter 2 and Chapter 4

Second is preparation of a *physical* design that shows what the actual system and its components will look like—their sizes, shapes, and relative positioning. This design activity results in engineering, manufacturing, architectural, and other types of drawings and models that show the details necessary to later fabricate, assemble, and maintain the system. This design activity sometimes reveals places where the functional design is impractical or infeasible because of assembly, maintenance, or appearance considerations, in which case it 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 have a ripple effect on the others. For instance, one subsystem might have to be enlarged, which robs space from another subsystem that then has to be moved to somewhere else, which displaces still another subsystem, and so on. Uncontrolled, the result is a never-ending series of redesign iterations, as illustrated in Example 11.1.

Example 11.1: Design Complexity in the Chunnel[2]

One of the mandated requirements for the English Channel Tunnel

(Chunnel) 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 inside. 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 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 changed, too.

Design and production/build do not always occur as discrete, sequential stages, but rather overlap. The building or constructing of a portion of the system commences as soon as *some of* the design is completed, then the building of another part begins when some more of the design is completed, and so on. In other words, the system is built while it is still being designed—a practice referred to as *fast-tracking* or *design-build*. 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 1 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 system or building will have been fabricated and might have to be rebuilt—increasing costs and schedules. The usual sequential or "slow-tracking" method takes longer but allows more time to discover and resolve design problems before construction begins.

Interaction Design[3]

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, cell phones, and entertainment systems—all of 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 ever more features, a process that leads to "bloatware." Compare, for instance, all the things you presumably *could do* with word-processing and spreadsheet software with the few features you actually use. Not only do such products contain too many features, they inter-mix never-used features with often-used ones, making the whole 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 generators are complex but they are used by trained personnel, not the average person. Commercial products (camera, car console, cell phone, etc.) 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"—aspects of design that address how product functions and the user interact. Whenever a programmer adds a pet function to a product, or a marketing manager insists on another product feature, they are packing in features they want, but ignoring the impact on the average end-user.

The project manager and systems engineer must retain control over the design process and particularly the interaction design. This starts with knowing the end-users and their wants, aptitudes, and skill levels, incorporating these into user requirements, and thereafter considering the end-user in every decision that will influence the function and operation of the product.

Controlling Design

Project reviews, discussed in Chapters 9 and 12, are scheduled at key milestones. Ideally, they are attended and headed by objective outside experts to ensure that the functional design satisfies requirements and the final design meets the users' personal needs and budget.

See Chapter 9 and Chapter 12

Throughout the design stage changes to earlier designs might be necessary due to new technology, technical problems, or new requirements. Since these inevitably require alterations to work activities, project management is responsible for monitoring the changes, determining their impacts on plans, schedules, and budgets, and relaying the impacts to stakeholders for approval of changes. This is all handled through a *change control system*, as described later.

Design changes tend to increase project costs, but as shown in Figure 11.2, design costs are typically but a small fraction of production costs. Design decisions impact life cycle costs (Figure 8.19), so they must bot be hurried. Consequently, prolonging this stage to get the design right the first time is usually far less costly than changing the design or fixing design-related problems later in the project. But the design stage cannot be allowed to continue indefinitely and sometimes project management must impose a freeze date after which no discretionary changes are allowed.

Project management is also responsible for ensuring that design efforts are adequately documented. Everyone must know the design's features, configuration, strengths, and limitations, and documentation is the precursor for system production, operation, and maintenance.

Planning for Production/Build and Later Stages

In the design stage, the project manager is already looking ahead and planning for the production stage. This planning addresses all aspects of production—tools, equipment, and materials needed, assembly procedures, testing, packaging, and so on—and includes a detailed production/build schedule. If the design documentation (specs, drawings, etc.) is not fully completed, the production plan might have to be prepared in phases.

Important to note is that the plan for the production/build stage must account for all the systems, activities, and resources necessary to *produce, operate*, and *maintain* the end-item system; this includes "side-items," so-called to distinguish them from the main contract end-item of the project. Side-items are no less important than the main end-item; in fact, without them it would be impossible to produce, operate, or maintain the end-item. Although side-items are usually developed and produced by others, the manager of the overall project is responsible for ensuring they have all been identified, that contractors are developing and producing them, and they will be ready for usage when the main end-item has been completed. Side-items are discussed in Chapter 12.

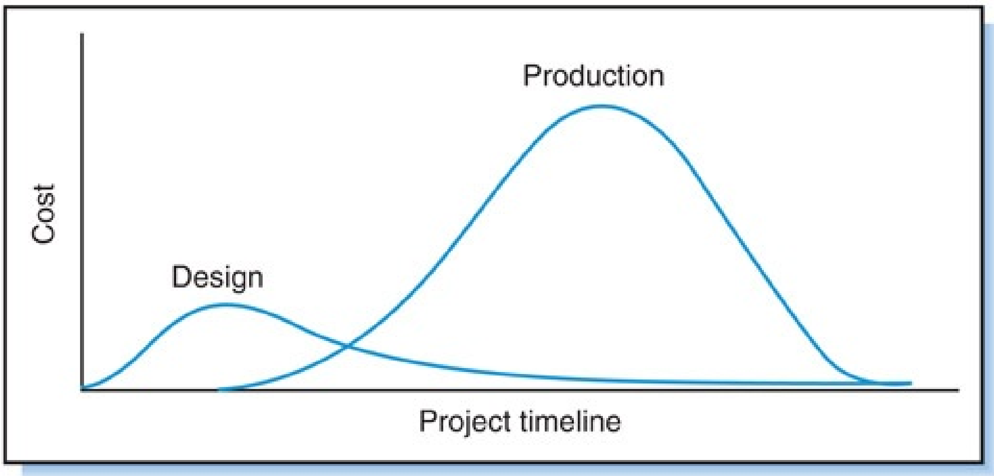


Figure 11.2 Relative costs for design and production.

11.3 Production/Build Stage

Detailed designs in hand, the contractor is ready for production/build. For mass-produced items, this means the system is ready to manufacture; for one-of-a-kind items, it means the system is ready for fabrication/construction. The main activities in this stage are system fabrication, testing, and planning for implementation.

System Fabrication

System fabrication begins as soon as design work has been sufficiently completed. Components prepared by the contractors and its 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 the budgets and schedules. During this stage the project manager and manufacturing or construction manager share responsibility for the principal management tasks of releasing work orders; monitoring, inspecting, and documenting progress; comparing planned versus actual results; and taking corrective action.

During system fabrication, work quality is constantly being assessed. As with most tasks in the production/build stage, quality control is not, per se, the responsibility of the project manager; nonetheless, because the project manager *is* responsible for the quality of the final system, she must ensure that other managers in the production/build stage have implemented a quality plan (discussed in Chapter 9) to achieve the project's quality objectives.

See Chapter 9

Throughout the production/build stage, numerous tests might be performed on components, subsystems, and the final end-item to ensure everything conforms to requirements. Aspects of these tests are discussed in Chapter 9.

The project manager oversees preparation of test plans and schedules and includes them in the production/build plan, and she ensures the necessary resources are available to perform tests and that test results are documented and filed for later reference.

Planning for Implementation

With phased project planning, details of the project plan are filled in as information becomes available; during each project stage, the detailed plan for the next stage is prepared. Planning for implementation should start early in the project—in the definition phase, however not until the production/build stage can implementation planning proceed in detail.

Implementation is the process of turning the system over to the user. The two prime activities in implementation are installing the system in the user's environment and training the user to operate the system. The project manager must develop the plan in advance so that the implementation stage can begin upon or before completion of the production/build stage. The plan must ensure that needed side-items will be available in time for user training, system installation, and operation. It should address the strategy for replacing the existing system with the new one and include:

1. The approach for converting from the old system to the new system.
2. Sequencing and scheduling of implementation activities.
3. Acceptance criteria for the new system.
4. The approach to phasing out the old system and reassigning personnel.

An initial implementation plan might have been developed as part of the project execution plan; now a more detailed plan is prepared with the participation of the customer to address the above points. As this plan is being prepared, the contractor accumulates materials to train the user in system operation and maintenance. For complex systems, these materials 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. These are among the "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 indefinitely; to prevent this, user requirements defined early in the project should include conditions or criteria for customer acceptance of the system. This is further discussed in the next chapter.

11.4 Monitoring and Control Process

Project monitoring and control—the process of keeping the project moving in the direction as laid out by the execution plan—happens throughout the project, from initiation to closeout. The process is pertinent to any situation where work must achieve certain goals or conform to a plan. Since the bulk of project work happens in the execution phase, it is here that the bulk of project monitoring and control also happen.

In simplistic terms, the project control process involves assessing progress against planned objectives or performance 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 actual temperature coincides with the desired temperature (i.e., variance becomes zero).

Virtually every monitoring and control process follows the same steps of (1) setting the performance standard, (2) comparing actual performance to the standard, and (3) taking corrective action to remove any variance.

In projects, set *performance standards* happens largely in the definition phase and early in the execution phase. The standards are user requirements, technical specifications, budgeted costs, schedules, and resource requirements.

The next step—compare *actual performance* to the standards—happens during the execution phase. Budgets, schedules, and performance specifications are monitored and compared to actual expenditures, test results, work completed, and other performance measures.

Last, take *corrective action* occurs whenever actual performance significantly differs from planned performance: either something is done to meet the planned results and standards, or the planned results and standards are revised. In the latter case, the contractor must work with the customer to change the objectives, revise the requirements, and modify the plan. There should be no surprises, and any *revisions should be reported to all relevant stakeholders*.

Worth repeating is that to keep the project aligned with standards (requirements, schedules, and budgets, etc.) there must first be a plan! In other words, *the precursor to project control is project definition*: without clear and complete requirements and a good project plan, there can be no project control.

Project Monitoring

Project monitoring refers to tracking the project, assessing how well it is doing, and forecasting how it will do in the future. Project monitoring involves data collection and interpretation, and information reporting.

The data collected must relate to project performance standards, i.e., to project plans, deliverables, schedules, budgets, and requirements. Typical data sources include material purchase invoices, worker time cards, 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 scrutinize, too little will not adequately reflect project status and allow problems to go unchecked. The data must be analyzed, and the results reported quickly and frequently enough so managers can quickly spot deviations from the plan and take corrective action.[4]

How frequently should data be collected, assessed, and reported? A good rule of thumb is to assess work progress every week. For small projects this ensures that even work packages lasting only a few weeks will be checked at least twice. For work packages lasting several months, assessment every 2 to 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 burdensome. The frequency also depends on the people doing the work (competent, motivated people can be monitored less than less competent or less motivated people), and the level of the work monitored (e.g. the program level can be monitored less often than the work package level).

Internal and External Monitoring and Control

Monitoring and controlling a project happens both internally and externally. *Internal control* refers to the *contractor's* procedures for monitoring work, reporting status, and taking action. *External control* refers to additional procedures imposed by others, including the customer. Military and government contracts, for example, impose external control by stipulating:

1. Frequent contractor reporting of schedules, cost, and technical performance.
2. Inspections of work by government program managers.
3. Inspection of the contractor's books and records by government auditors.
4. Strict contractor-imposed terms on allowable project costs, pricing policies, etc.

External control can be a source of annoyance to the contractor since it involves managers overlooking managers and adds to bureaucratic and administrative costs. Nonetheless, it is sometimes necessary to protect the customer's interests, especially in cost-plus projects. Ideally, the contractor and customer are able to work together amicably to establish compatible plans and work monitoring methods.

Traditional Cost Control

In non-project situations, work performance is measured with *variance analysis*, which compares the amount spent with the amount budgeted. In project situations, simple cost variance analysis is inadequate.

Example 11.2: Cost Variance Analysis

Consider the following weekly status report for the work package "software development":

Budgeted cost for period	Actual cost for period	Period variance
= $12,000	= $14,000	= $2,000
Cumulative budget to date	Cumulative actual cost to date	Cumulative variance
= $25,000	= $29,000	= $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 really over budget.

Suppose the $25,000 was the amount budgeted for completing 50 percent of software development. If 50 percent of the work had actually been completed as intended, then the project would, in fact, be over budget, and something would have to be done to reduce or eliminate the $4,000 overrun. Now suppose only 30 percent of the work had been completed; in that case the project would be clearly over budget (and behind schedule too), and further cost overruns could be expected just to get caught up. As a third possibility, suppose that 70 percent of the work had been completed; this is substantially more work than what was scheduled. Because of that the project might not be over budget, and could even be under budget for the amount of work performed.

The point of the example is that to be able to assess project status, cost information isn't enough; you also need information about *work progress*—information about the percentage of work completed, milestones achieved, and so on, all discussed later.

Cost-Accounting Systems for Project Control

In the early 1960s the US government developed a PERT-based scheduling and cost-accounting system called *PERT/Cost.* The system became mandatory for all military and R&D contracts with the US Department of Defense and NASA (DOD/NASA). Any contractor wanting to work for DOD/NASA had to use the system and produce the necessary reports. PERT/Cost was an improvement over traditional cost-accounting techniques and spurred the development of other even-more sophisticated systems to track work and report progress and costs. It was the original network-based project cost accounting system—the PCA system mentioned in Chapter 8 and modern-day Earned Value Management (EVM) systems discussed later in the chapter.

See Chapter 8

Today most PCASs integrate information about work packages, budgets, and schedules into a unified project control package. They permit the causes of cost and scheduling overruns to be pinpointed among numerous work packages or budgets. Two common features of these systems, to be described later, are use of *work packages* and *control accounts* as the basic data collection point for project control, and use of *earned value* to measure project performance.

11.5 Work Packages and Control Accounts

Earlier chapters described the role of work packages and control accounts in project planning and budgeting; not coincidentally, they are also key elements of project control. Each control account consists of one or more work packages; each work package is like a contract for a specific job with requirements, a work description, budget, schedule, and so on. Thus, each work package and control account is a focal point for data collection, work progress evaluation, problem assessment, and corrective action.

Large projects may be composed of hundreds of work packages, making it potentially difficult to identify the ones causing cost or schedule overruns. An advantage of a PCAS is that it can readily sort through all of the work packages to locate sources of problems. Although the individual work package remains the central element for control, a PCAS can consolidate and report information for *any* level of the project, from the individual control-account or work-package level up to the project level. Because higher-level accounts in the control-account structure are built up through the WBS and organizational hierarchies, variances in costs and schedules at any project level can be traced down through the structure to identify work packages causing the variances.

Work Authorization

Part of the project control process is *work authorization* or start-stop control: all project work is started only after formal authorization and is stopped only upon review and acceptance. This applies to both the project as a whole and to each of the work packages. At the project level, the project manager is authorized to begin the project upon acceptance by the customer, program manager, and/or top management of the project plan. The project manager then authorizes the managers of sub-projects to begin, who authorize managers and supervisors of work packages at the next level lower, as shown in Figure 11.3. The process is a continuation of the initiation and authorization process described in Chapter 3 and shown in Figure 3.9.

See Chapter 3

The same process also applies to authorizing *phases* of a project: after each phase the customer and other stakeholders evaluate the *results* of the phase, the plan for the next phase, and the risks, and, if all are acceptable, they authorize the next phase (see "phased project planning" discussed in Chapter 4 and "gating process" in Chapter 17). Sometimes in contracted projects, if the results of the phase are judged unacceptable, the contractor is paid the amount owed and the project is terminated.

See Chapters 4 and 17

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 specifies the contractual requirements and gives the project manager the go-ahead. The comptroller or project 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," or "test order," depending on the kind of work). As the scheduled start date for the task draws near, the project manager or project office releases the authorization document to the contractor or department to perform the work. For simple projects or simple activities, verbal authorization might suffice.

Collecting Cost, Schedule, and Work Progress Data

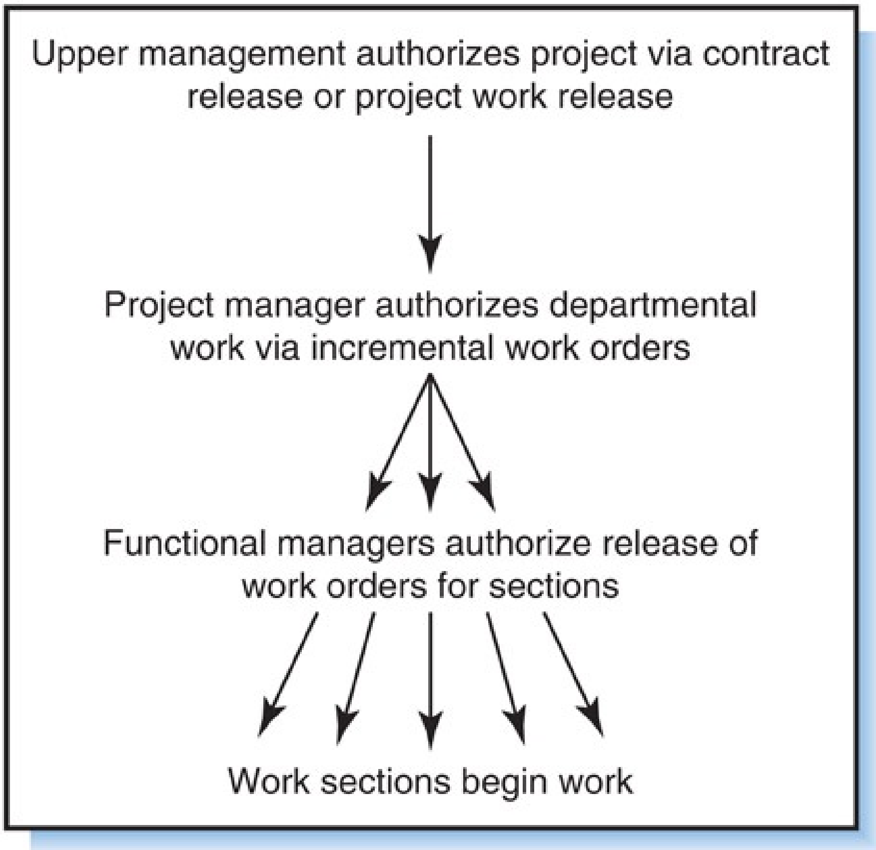


Figure 11.3 Project work authorization process.

See Chapter 8 and Chapter 12

After work begins, data about *actual* costs and work progress on the work package is periodically collected and entered into the PCAS or PMIS (discussed in Chapters 8 and 12). The PCAS generates performance reports periodically or as needed for each work package, department, and the entire project.

Assessing the impact of work progress on schedules is the responsibility of the functional manager or team supervisor in charge of the work package and task. Each week the supervisor tallies the labor hours for each task as indicated on time cards. She notes tasks completed and tasks still “open,” and estimates the time still needed to complete open tasks; from the latter she estimates the “percent complete,” or percentage progress for the task. Progress is recorded on a Gantt chart showing completed and open tasks. Figure 11.4 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 whole bar indicates 100 percent complete; highlighting a quarter of the bar, for example, means 25 percent complete. Work packages K, L, M, N, O, P, and Q are all open (underway but not yet completed); the first three and Q are behind schedule; O is ahead.

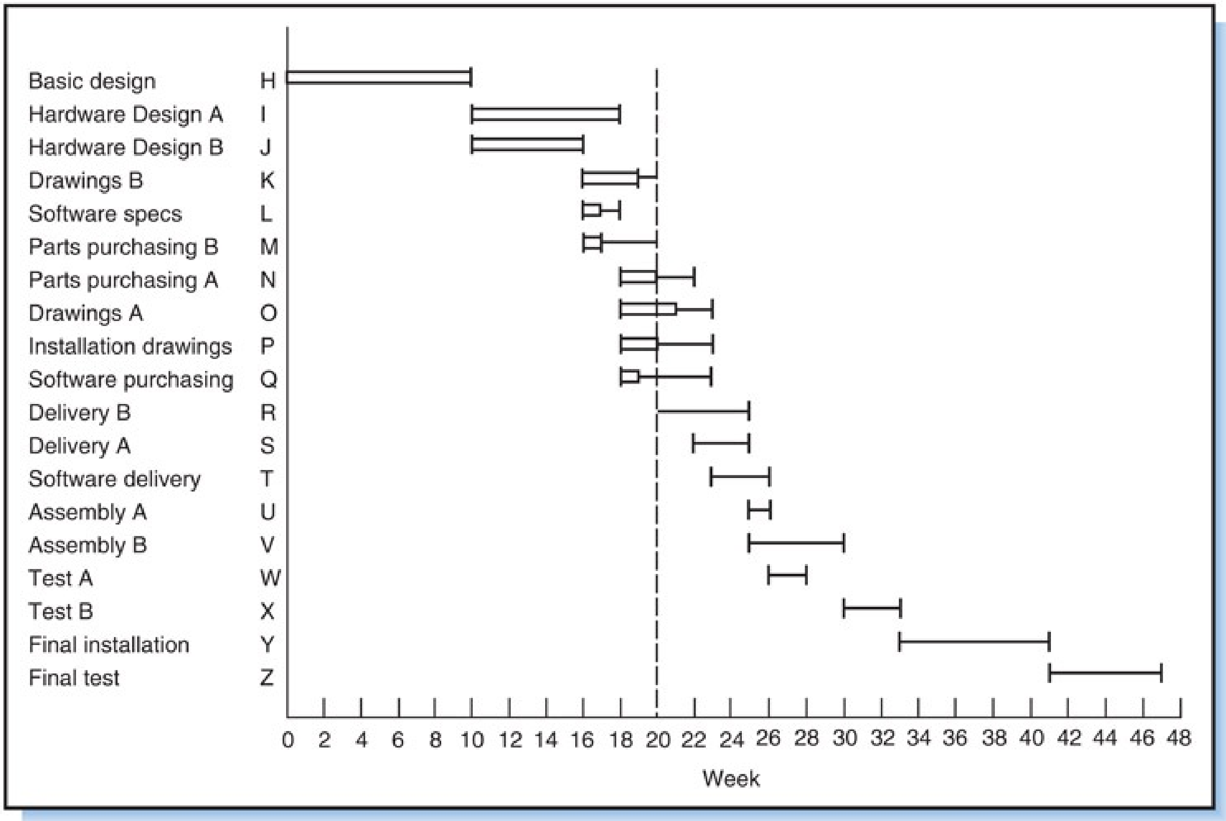


Figure 11.4 Gantt chart showing work status as of Week 20.

How is work progress and percent complete measured? Costs and time elapsed are easy to measure, but neither says much about the actual work progress so project managers must often rely on subjective measures. In a survey of ways to measure ongoing project performance, Thompson identified the following.[5]

1. *Supervision*. Managers and supervisors assess progress by direct observation, asking questions, and reviewing written reports and project documentation.
2. *Milestones*. These are easily measured end-points of tasks, or transition

points between tasks; for example, completion of drawings, reports, design documents, or solution-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 objectively measure technical progress at intermediate stages of the project.
4. *Reviews*. Described in Chapter 12, these are meetings with managers and technical personnel to review the progress of a design or system against the plan.

See Chapter 12

5. *Outside experts*. The project manager or other stakeholder invites experts to serve on a panel. The panel assesses project status by observing work in progress, talking to project personnel, and reviewing documentation.
6. *Status of design documentation*. Experienced project managers can determine when a design is nearly finished by the "completeness" of documents such as drawings, schematics, functional diagrams, manuals, and test procedures.
7. *Resources utilized*. A request for or change in resources may reflect progress; for example, tasks nearing completion often require special testing or implementation facilities, personnel, and equipment.
8. *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.
9. *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.
10. *Changes, bugs, and rework*. Because, ordinarily, the number of change requests (discussed later), bugs, problems, and so forth, should decrease as a project gets closer to completion, a sustained high number may indicate lack of progress. This is discussed later under "issues tracking."

While measures like these are used to assess in-progress work and reveal emerging problems, also needed are the risk management measures discussed in Chapter 10 to *anticipate* problems.

See Chapter 10

Each week the work package supervisor tallies current expenses. Labor hours reported on time cards 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 (a frequent source of cost overruns) are also included. Any estimated changes to budgets or schedules for remaining work are documented by the work package supervisor and submitted to the project manager for approval. The supervisor forwards a report to the project manager showing costs of all work completed in prior periods plus work accomplished in the current period. Once the project manager validates this information, she enters it 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 cost to complete the project; as described later, these provide forecasts of the completion date and project cost at completion. When a task or work package is completed, its budget is closed to prevent any additional, unauthorized billing.

11.6 Project Monitoring and Control Emphasis

Project monitoring and control addresses five areas: scope, quality, schedule, cost, and procurement.

Scope Control

Projects have a natural tendency to grow over time because of changes and additions to the scope, a phenomenon called "scope creep." Changes or additions to the scope reflect changes in requirements and work definition, which are usually accompanied by increases in time and cost. The aim of scope control is to identify where requirements or work changes are occurring, ensure the changes are necessary and beneficial, restrain or delimit the changes wherever possible, and manage implementation of the changes. Scope control is implemented through the *change control system* and *configuration management*, described later.

Quality Control

Quality control is managing the work to achieve the desired requirements and specifications in the end-item, and taking preventive measures to reduce or eliminate errors and mistakes in the work process. Discussed in Chapter 9, project quality control starts with the *quality management plan* that states conditions or stipulations about what is necessary in each work package to ensure quality results. It also specifies the tests, inspections, and reviews to assess progress toward meeting requirements. In technical projects, progress is tracked using a methodology called *technical performance measurement* (TPM), discussed later in the chapter.

See Chapter 9

The project manager can appoint a *quality-improvement team*—a small group responsible for identifying and eliminating the sources of both unique and repetitive quality problems. On a small project, one cross-functional team might serve this function; on a large project, several specialized teams might be needed to address problems with particular phases or technologies.

Schedule Control

The intent of schedule control is to keep the project on schedule. Even the most carefully planned projects fall behind for reasons beyond anyone's control, including, for example, necessary scope changes, weather problems, and material shortages. Schedule overruns also happen due to multi-tasking, procrastination, and incorrect time estimates, as discussed in Chapter 7. Following are some guidelines for reducing schedule variability and schedule overruns.

See Chapter 7

Time Buffers and Fever Charts

Described in Chapter 7, a time buffer is a schedule reserve, an amount of time included in the expected project duration to account for uncertainty. To implement a time buffer, the finish date, computed by using tight durations, is increased by the buffer amount. If the late finish date is July 31 and the time buffer is 4 weeks, the target finish date is set for August 31.

Time buffers are an aspect of the critical chain project management (CCPM), which prescribes locating buffers at the end of the critical chain for the project and the ends of all sub paths feeding into the critical chain. Once a project is underway, the amount of buffer "consumed" is tracked. Each time a task in the critical chain or a feeding chain is delayed, it "consumes" the time buffer. The more of the buffer consumed, the more likely the buffer will be exhausted and the target finish date overrun. Hence, the project should be managed so as to minimize buffer consumption.

Buffer consumption is tracked and controlled with a "fever chart," a graph that shows the percentage of project buffer consumed versus the percentage of the critical chain completed (Figure 11.5). Early on, a "healthy project" will have consumed little or none of its buffer, but as the project progresses the percentage of buffer consumed can be expected to increase and the plot on the graph to rise diagonally. Monitoring the graph enables the project manager to somewhat 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. For a healthy project the slope of the line is shallow and much of the project buffer remains unconsumed by the end of the project. Completing a project with buffer remaining is equivalent to completing the project *ahead* of the target finish date. Thus, to complete the project early, the project must be managed so as to minimize buffer consumption; the more buffer remaining, the further ahead is the project.

Yellow on the chart indicates potential for the project to overrun its target date; red means strong potential. The fever chart is updated weekly and quick action is taken whenever the plot veers into the yellow or red zones. Tasks consuming the buffer are identified so managers can divert more resources to them, decouple them from the critical chain, or take other measures. Fever charts are also created for feeding buffers to monitor the risk of delaying the critical chain.

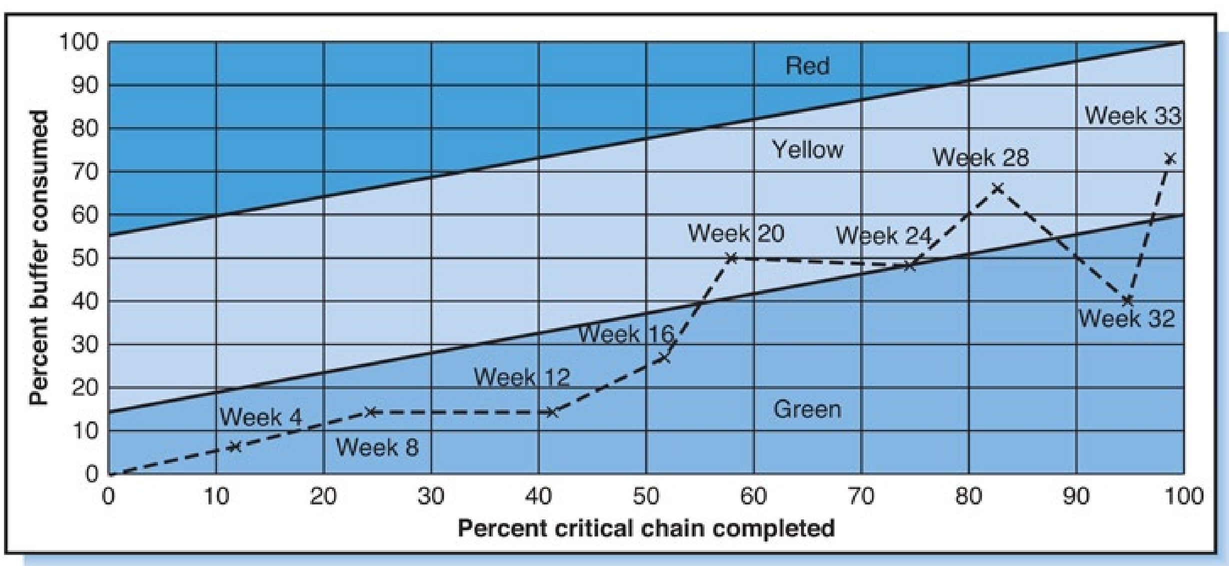


Figure 11.5 Fever chart: Percent buffer consumed versus percent critical chain completed. The fever chart is

divided into green, yellow, and red regions to denote project status.

Example 11.3: Fever Chart Computation

Points on the fever chart are computed as follows; each week:

1. Estimate percent complete 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: Elapsed weeks to-date − CC weeks-completed = Weeks buffer consumed. Compute: Weeks buffer consumed/Project buffer length = Percent buffer consumed (y-axis)

The above procedure can be modified using days. Results would be the same though more precise.

As an example of the computation consider the project in Figure 11.6.

Suppose the data points in the fever diagram in Figure 11.5 were derived from the project status as assessed every 4 weeks (status should be assessed every week; 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 11.1.

Per the table, last line, the project is completed in 33 weeks, which is the amount by which the project finished ahead of target (36 − 33 = 3).

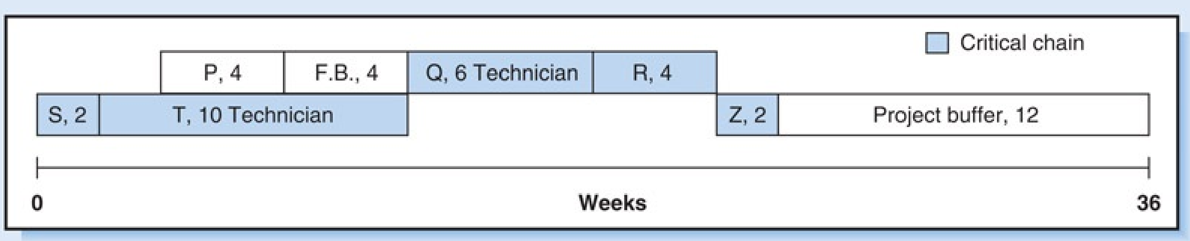


Figure 11.6 Critical chain = 24 weeks, project buffer = 12 weeks.

Table 11.1

Week	Tasks & Percent Completed	Weeks Completed	Percent CC Completed	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 = 5	41.7%
33	S, T, Q, R, Z 100%	22 + 2 = 24	100%	33 − 24 = 9	75%

The fever chart is one way to manage time buffers; the following example shows another.

Example 11.4: Doling Out the Reserves: The Mars Pathfinder Project[6]

The goal of the Pathfinder project was to land on Mars a skateboard-sized, self-propelled, six-wheel rover that would move over the terrain and send back photos and scientific data (Figure 11.7). The project's budget reserve was $40 million—about 30 percent of the total budget (a large percentage, but common in risky technological projects), and its schedule reserve was 20 weeks, about 13 percent of the project's 37-month design, build, and test schedule.

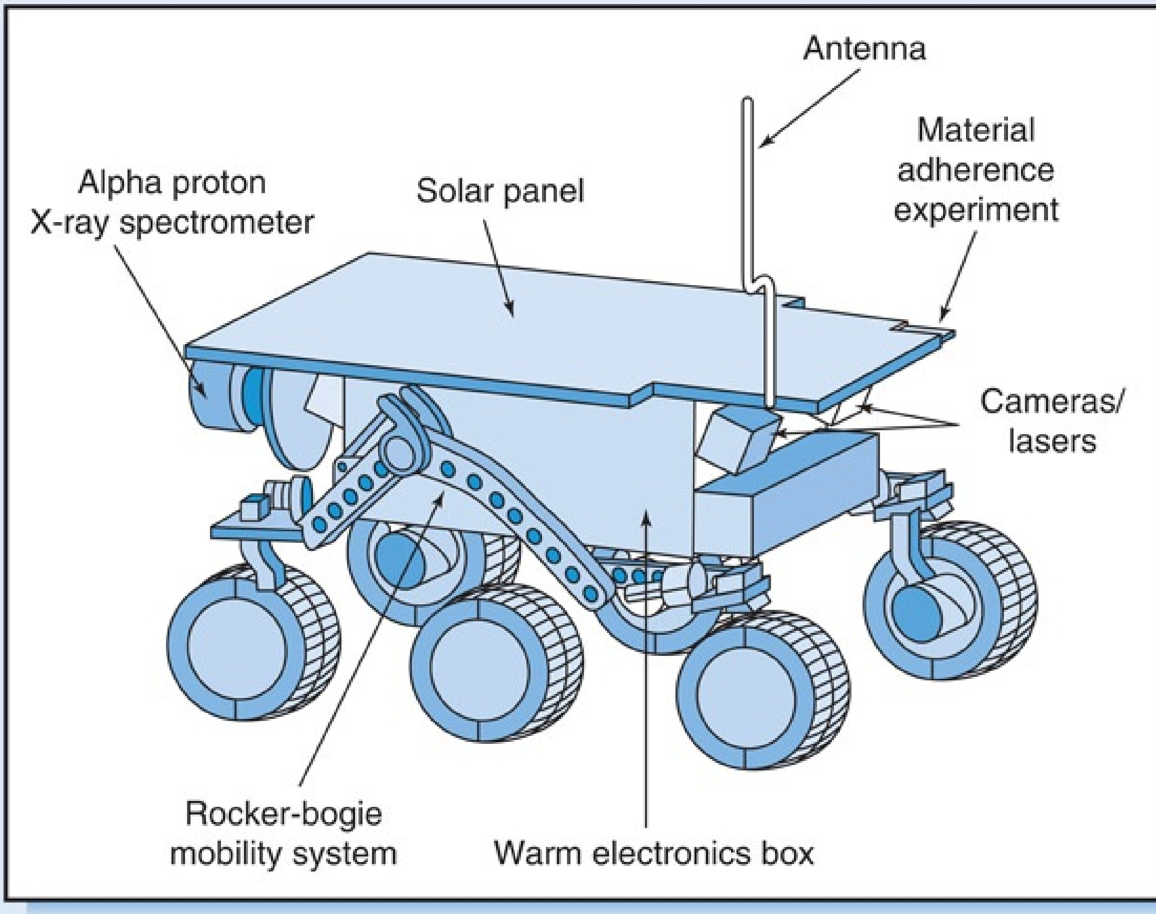


Figure 11.7 The Mars Pathfinder rover.

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 amount of the schedule reserves available for use in *each period* of the project. For example, *none* of it was to be used (no slippage allowed) for the start of system assembly and test. If problems then arose, the guideline was to commit whatever budget reserves necessary to keep the project on schedule. (Time was a strategic issue: 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.

Frequently Report Activity Status

Tasks or work packages on the critical path or critical chain should be ready to start at the earliest possible time, but for that to happen the project team needs to know the status of each task's predecessors. Especially in time-sensitive projects like Pathfinder, every activity should provide its successor activities with daily status reports stating the expected days remaining to complete and the earliest date when successors should expect to begin work. The mandate is that *as soon as* immediate predecessors of a critical activity are completed, the team assigned to the next activity will begin work, even if it has to stop working on something else. In CCPM terminology, the countdown to begin work on the critical chain early is called a *resource buffer*.

Publicize Consequences of Delays and Benefits of Early Finish

Everyone—team members, subcontractors, and suppliers—should know the consequences of a schedule overrun and the benefits of finishing early. The project contract might offer incentive payments for early completion, or budget extra money for bonuses to workers and subcontractors who finish early. The next example describes other measures to keep projects on schedule.

Example 11.5: Meeting Launch Deadlines at Microsoft[7]

Microsoft meets product launch dates by utilizing visual freeze, internal target ship dates, and time buffers. A "visual freeze" 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 dates" that pressure developers to decide which product features must absolutely be included and which may be forgone; otherwise they tend to keep adding product features 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 puts time buffers in project schedules. The buffers, which can range from 20 to 50 percent of the total schedule time, are used exclusively to cover uncertainties and not routine John, not clear to me tasks. The project team strives to meet the internal ship date, which is the launch date announced to the public *minus* the time buffer.

Procurement Control

The main contractor is responsible for the quality, schedule, and cost of all items procured for the project. Often, the project manager will visit and inspect each supplier's facilities to make sure they are capable of meeting requirements. After the project is underway, she monitors each supplier's progress by visiting the supplier's site and requiring frequent status and, sometimes, expense reports from the supplier. The project manager does whatever is necessary to prompt or assist suppliers when problems arise. For all major outsourced items and services, contingency plans should be prepared, including possible contractual provision to transfer work to other suppliers in case the original suppliers encounter serious or unrecoverable problems. These contingencies are addressed in the procurement plan and the project risk plan.

Cost Control

The purpose of cost control is to track variances in expenditures versus budgets, eliminate unauthorized or inappropriate expenditures, and minimize or contain cost changes. It identifies where and why variances have occurred, and when changes to cost baselines are necessary.

Cost control happens at both the work-package level and the project level, using the cost account structure and PCAS described in Chapter 8. Through the PCAS, actual expenditures are tallied, validated, accumulated, and compared to budgeted costs. Using the methods described next, the project manager periodically reviews actual and budgeted costs, assesses the work completed, and estimates the project's completion cost and completion date.

See Chapter 8

11.7 Performance Analysis and Earned Value Management

The performance of the project or any portion of it can be assessed with three variables: BCWS, ACWP, and BCWP. These are industry-standard acronyms, but to save ink we will use the abbreviated terms PV, AC, and EV.

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*. For example, in Chapter 8, Table 8.5 and Figure 8.15 show the cumulative and weekly expenses for the LOGON project. These amounts represent PV. In Week 20, for example, to-date PV is $512,000 and weekly PV is $83,000.

See Chapter 8

2. *AC* is the *actual cost of the work performed* (or ACWP)—the actual expenditure as of a given time period.
3. *EV* is the *earned value* (or *budgeted cost of the work performed*, BCWP)—the *value* of the work performed so far (fully and partially completed work packages) according to the *original budget*.[8] Thus,
   - For a completed work task, EV is the same as the PV for that task.
   - For a partially completed work task, EV is computed as the estimated *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.)

Application of these variables to track and assess project performance is called earned value management, or EVM. The following example illustrates.

Example 11.6: EV versus PV in the Parmete Company

The Parmete Company has a $200,000 fixed-cost contract to install 1,000 parking meters. The contract calls for removing old parking meters from their stands and replacing them with new ones at a cost of $200 per meter.

Parmete estimates that 25 meters can be installed each day. At 25 meters per day and $200 per meter the project should finish in 40 working days with a total PV of $200,000. Also on that basis, the planned value of the work scheduled (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 meters ($200 times 25). For example, as of Day 18,

$$PV = 18 \text{ days} \times (25 \text{ meters}) \times (\$200) = \$90{,}000$$

That is to say, as of the eighteenth day of work on the project, the project schedule and budget specify that $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 meters *actually* installed to-date, times the $200 budgeted for each meter. Suppose, for example, that as of the eighteenth day 400 meters had been installed; thus,

$$EV = (400 \text{ meters}) \times (\$200) = \$80{,}000$$

In other words, as of the eighteenth day, $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 the value of work that should have been but was not done. It represents 50 parking meters, or 2 days' worth of work, meaning that as of the eighteenth day 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 only 16 days' worth of work progress. This is represented on the graph for PV and EV in Figure 11.8.

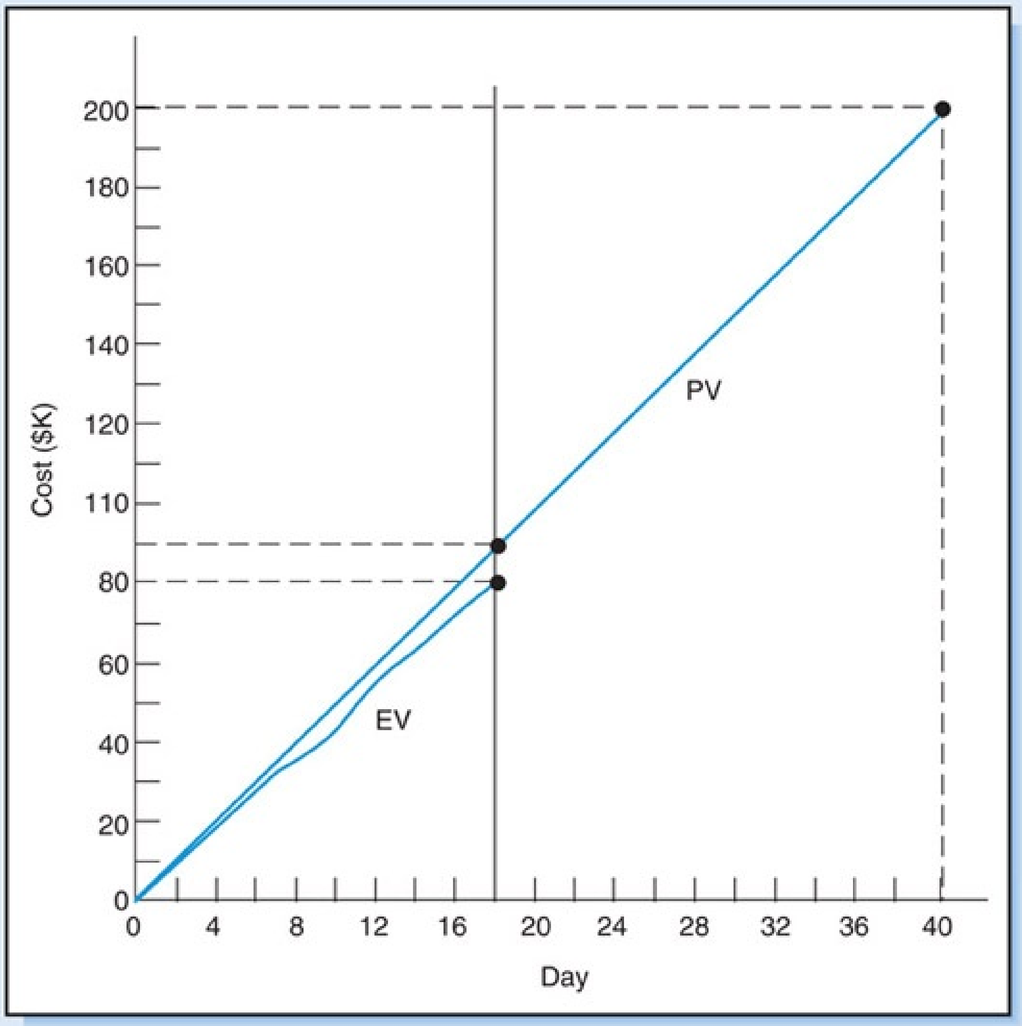


Figure 11.8 Graph of PV and EV.

Besides completed tasks, the EV must also reflect tasks started but only partially completed (open tasks). For example, suppose before quitting at the end of the eighteenth day the meter installer had just enough time to remove an old meter but not to install a new one: the work on that task was 50 percent completed. If this were meter no. 401, then EV for the eighteenth day 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 on day 18 is $80,100, which represents slightly more than 16 days [$80,100/(25 × $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 11.9.

1. *Schedule variance*: SV = EV − PV = − $83,000.
2. *Time variance*: TV = SD − BCSP [where SD is the "status date" (here Week 20); for BCSP (budgeted cost, scheduled performance), refer to Figure 11.8 at EV for Week 20, then see the week where PV equals this EV (about Week 19)]; thus, TV = (20 − 19) = 1 week.
3. *Cost variance*: CV = EV − AC = − $101,000.

Negative SV indicates that the project is behind schedule; positive SV suggests it is ahead of schedule. An SV for Week 20 of −$83,000 means the project is behind schedule. TV shows approximately how much behind schedule, in this case about 1 week because $429,000 worth of work has been completed (EV), which is roughly the value of work (PV) that was supposed to be completed about one week earlier.

Negative CV indicates that the project is overspending for the work completed; positive CV indicates it is under spending. CV of −$101,000 indicates that LOGON is overspending.

Work Package Analysis and Performance Indices

Assessing project status requires knowing the performances of all work packages. With information from the PCAS, a figure like Figure 11.9 can be prepared for every work package and control account.

Refer back to Figure 11.4, the status of the LOGON project as of Week 20. It shows that activities H, I, and J have been completed and are closed, and activities K through Q are "open" and underway. Although this gives a general *overview* of work package and project status, to determine the origins of sizes of any delays or overruns each work package must be assessed in detail; this is done by computing two *performance indices* for each activity:

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, respectively; values less than 1.0 indicate the opposite.

Table 11.2 shows performance information for all LOGON activities as of Week 20. The indices CPI and SPI show trouble spots and their relative magnitude: L, M, and Q have fallen the most behind

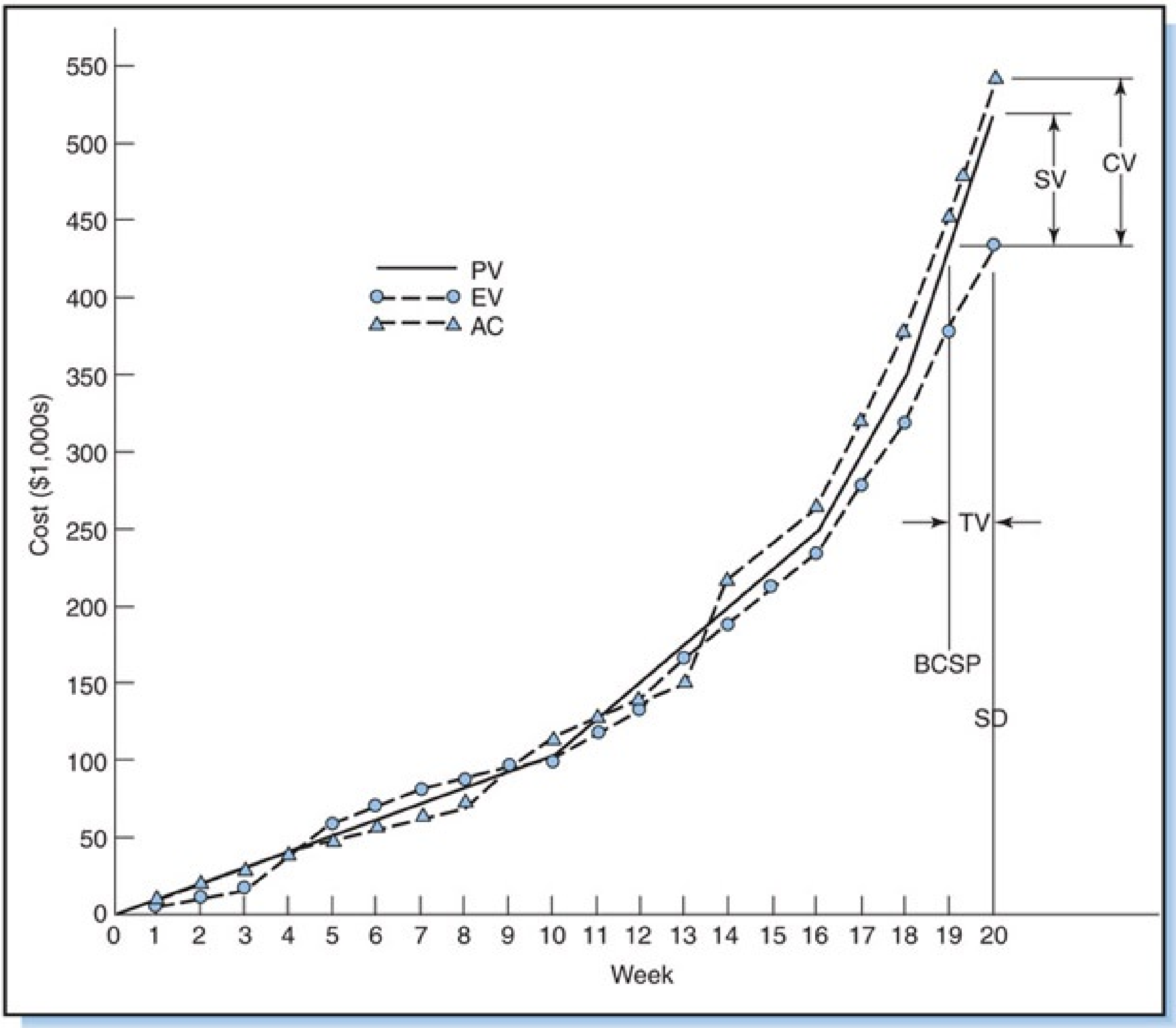


Figure 11.9 LOGON project status as of Week 20.

Table 11.2 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

schedule (they have the smallest SPIs), and L and M have the greatest cost overruns relative to their sizes (they have the smallest CPIs). The overall project is "somewhat" behind schedule and over cost (SPI = 0.83; CPI = 0.81).

Focusing on either *only* the project level or *only* the work package level to assess project status can be misleading, and the project manager should scan both levels, back and forth. If she looks only at the project level, good performance of some activities may hide poor performance in others. If she focuses only on individual work packages, she can easily overlook the cumulative effect from slightly poor performance in many activities: small cost overruns in many individual work packages can add up to a large overrun for the project. For example, SV in Figure 11.9 (−$83,000) suggests that the entire project is behind schedule (TV = −1 day), yet scrutinizing Figure 11.4 reveals that only one of the behind-schedule work packages, Activity M, is on the critical path (see Chapter 6, Figure 6.8). Since activity M appears to be about 3 weeks behind schedule, the project must also be 3 weeks behind schedule—*not* 1 week as estimated by the project-level TV.

See Chapter 6

The importance of monitoring performance at the work package level is further illustrated by an example from the ROSEBUD project. Figure 11.10 is the cost report for Work Package L for Month 2. (The numbers in the PV columns are derived from the Month 2 column in the budget plan in Figure 8.8, Chapter 8.) Current period and cumulative numbers are the same because Work Package L begins in Month 2.

See Chapter 8

The performance indices for ROSEBUD Work Package L are;

SPI = EV/PV = 0.80
CPI = EV/AC = 0.74

indicating both schedule and cost overruns as of Month 2. Suppose the project manager investigates the costs for Work Package L and discovers the following:

Project: ROSEBUD — Date: Month 2
Department: Programming — Work Package: L Software specifications

Charge	Current period PV	Current period EV	Current period AC	Current period SV	Current period CV	Cumulative to date PV	Cumulative to date EV	Cumulative to date AC	Cumulative to date SV	Cumulative to date CV
Direct labor										
Professional										
Associate										
Assistant										
Direct labor cost	6,050	4,840	6,050	−1,210	−1,210	6,050	4,840	6,050	−1,210	−1,210
Labor overhead	4,538	3,630	5,445	−908	−1,815	4,538	3,630	5,445	−908	−1,815
Other direct cost										
Total direct cost	10,588	8,470	11,495	−2,118	−3,025	10,588	8,470	11,495	−2,118	−3,025
General/administrative	1,059	847	1,150	−212	−303	1,059	847	1,150	−212	−303
Total costs	11,647	9,317	12,645	−2,330	−3,328	11,647	9,317	12,645	−2,330	−3,328

Note: EV is for 80 percent of work scheduled and labor overhead is increased to 90 percent of labor cost.
SPI: EV/PV = 0.80 CPI EV/AC = 0.74

Figure 11.10 Cost chart for ROSEBUD project as of Month 2.

First, although AC = PV for direct labor, PV reflects the estimate that only 80 percent of work scheduled for the period was performed (EV = PV × SPI = 6050 × 0.80 = 4850). Second, although AC = PV for direct labor, the AC and PV for labor *overhead* are different due to a rate increase from 75 percent to 90 percent during

Month 2. Whereas PV would reflect the old rate (0.75 × 6050 = 4538), AC would reflect the new (0.9 × 6050 = 5445). The point: the fact that total AC > total PV in this case has no bearing on the actual performance of the work package but stems from a change in the overhead rate, something over which the project manager has no control.

Project: ROSEBUD Date: Month 3

Department: Programming Work Package: L Software specifications

Charge	Current period PV	Current period EV	Current period AC	Current period SV	Current period CV	Cumulative to date PV	Cumulative to date EV	Cumulative to date AC	Cumulative to date SV	Cumulative to date CV
Direct labor										
Professional										
Associate										
Assistant										
Direct labour cost	5,000	6,050	5,000	1,050	1,050	11,050	11,050	11,050	0	0
Labor overhead	3,750	4,538	4,500	788	38	8,288	8,288	9,945	0	1,657
Other direct cost										
Total direct cost	8,750	10,588	9,500	1,838	1,088	19,338	19,338	20,995	0	1,657
General/administrative	875	1,059	950	184	108	193	193	2,100	0	166
Total costs	9,625	11,647	10,450	2,022	1,196	21,272	21,272	23,095	0	1,823

Note: EV is for 121 percent of work scheduled, but for cumulative it is 100 percent (made up for delay in Period 2). $1,823 CV reflects increase in overhead rate.

Figure 11.11 Cost chart for ROSEBUD project as of Month 3.

Now look at Figure 11.11, the cost report for the same work package but for Month 3. The performance indices for cumulative figures are:

SPI = EV/PV = 1.00
CPI = EV/AC = 0.92

Notice first that direct labor AC = PV for the month, but more work was performed than planned for the month (EV > PV), which made up for the work deficit in Month 2 and resulted in the task being completed on schedule (indicated by SPI = 1.00). The work package has negative CV—caused by the increase in the labor overhead rate from 75 percent to 90 percent. The point? Of the numerous factors that affect project work progress and costs, some are beyond the project manager's control. To determine the sources of variances and places where the project manager can or must act requires scrutiny of costs and performance at the work package level. Project level analysis is simply inadequate.

Monitoring Performance Indexes and Variances

Using project-level CPI and SPI, project stakeholders can get a quick "ballpark" estimate of the project's performance to-date. Although the estimate can be somewhat inaccurate, it allows the manager to track broad trends in project performance. The plot of SPI against CPI in Figure 11.12 is an example: LOGON started out in the marginal and poor regions, briefly recovered, and then drifted disturbingly back to and *remained* in the poor region. The project manager should be talking to team leaders and functional managers to identify the detailed causes of the problems.

Seldom do actual and planned performance measures coincide; as a result, nonzero variances are more the rule than the exception. This leads to the question: What amount of variance is necessary to mandate taking action?

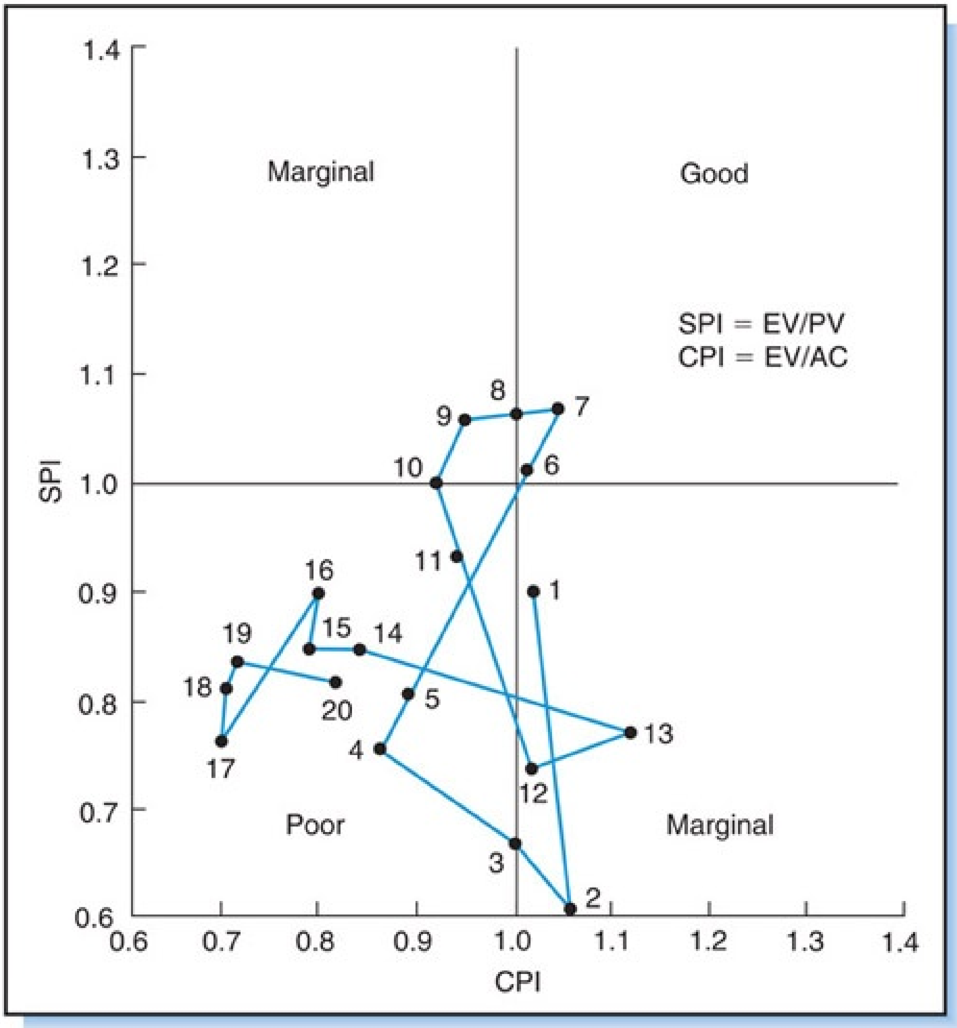


Figure 11.12 LOGON project cost/schedule performance plotted for Months 1 through 20.

Table 11.3 Example of Variance Boundaries

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 11.3 shows "acceptable" variance limits for different project levels—work package, department, and project. Only when a variance exceeds a limit are corrective measures considered. Sometimes the limits are allowed to vary. In

research projects (Pathfinder, for example), they start out somewhat large but are reduced as the project moves forward. This coincides with project risk, which starts out large but diminishes as the project progresses.

Upper and lower variance limits set on costs and schedules help identify places 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 corner-cutting and shoddy workmanship. For technical performance, both upper and lower variance limits are set on technical requirements. Lower limits are necessary to insure requirements are met; upper limits are necessary to prevent excessive or unnecessary development work.

Updating Time Estimates

Following each progress review it might be necessary to update scheduled completion dates of tasks or work packages. In general,

Forecast finish date for a task = Start date + Time remaining

where *time remaining* is determined in two ways. The first is to consider 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

$$\text{Percent progess per day} = \frac{\text{Percent of task completed so far}}{\text{Days worked on task so far}}$$

The other way is to simply accept the opinion of a reputable source ("it'll take another 5 days to finish the job"). This way often yields a more accurate estimate than the other because it accounts for any recent changes in the rate of work progress.

Example 11.7: Revising the 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 remaining part 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.

Estimate at Completion

Periodically the project manager prepares a *to-complete* forecast, which is an estimate of the time and cost needed to complete the project. This forecast plus the current status of the project provide a forecast of the date and cost of the project *at completion*. Estimates for the cost remaining to complete the project (the *to-complete* cost) and the project final cost (the *at completion* cost) are computed as follows:

ETC (Estimated cost To Complete project) = (BAC − EV)/CPI

where BAC is the *Budgeted* cost At Completion for the project (= total PV at target completion).

EAC (Estimated cost At Completion) = AC + ETC

The following two examples illustrate.

Example 11.8: Forecasting ETC and EAC for ROSEBUD Project

Figure 11.13 shows the ROSEBUD project Gantt chart with percent complete and EVM metrics for Week 13. Given this information, how much will the project likely cost to complete and how much will it 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:

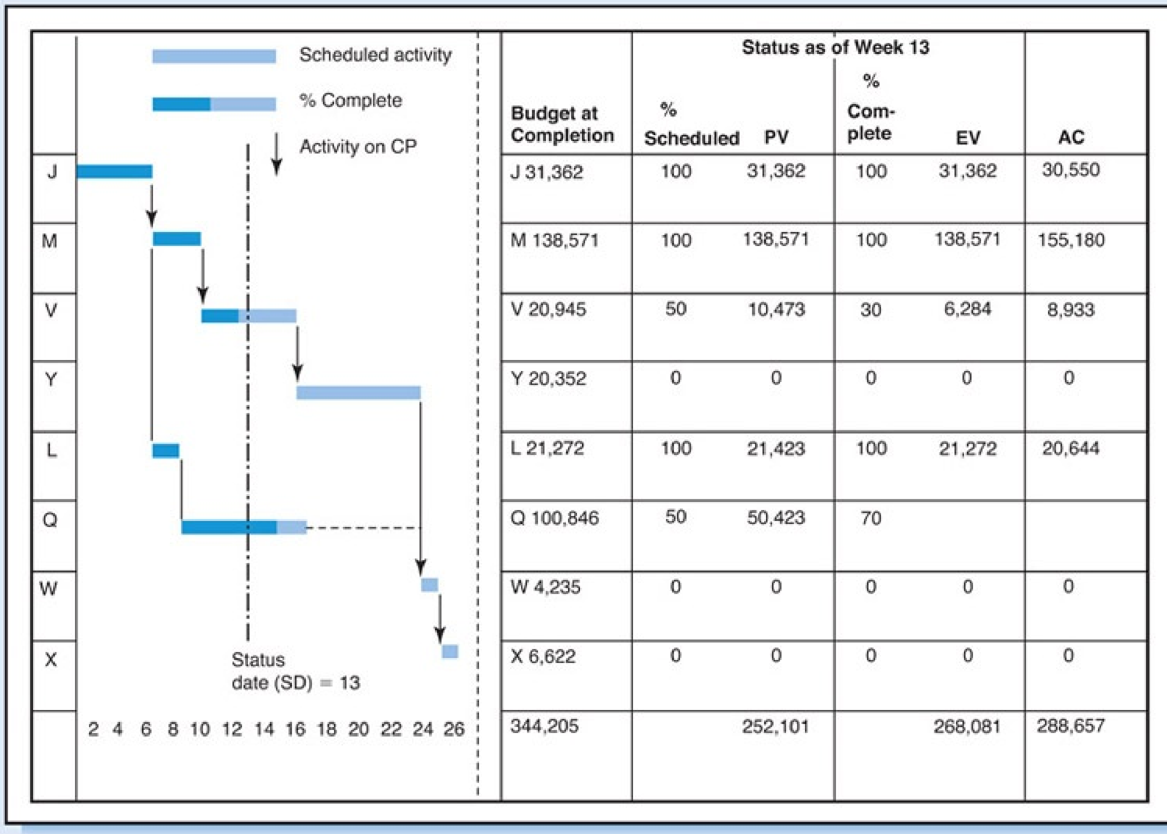
ETC = 76,124/0.9287 = $81, 968.

Since $288,657 has already been spent, the project estimated at-completion cost is:

EAC = AC + ETC = 288,657 + 81, 968 = $370,625.

This is an overrun of $370,625 – $344,205 = $26,420, or 7.7 percent.

Notice that according to EV, the project is slightly ahead of schedule (EV = 268,081) > (PV = 252,101), although more likely it is *behind* schedule because Activity V is on the critical path and appears roughly 1 week behind schedule according to the Gantt chart.



	Status as of Week 13				
Budget at Completion	% Scheduled	PV	% Com-plete	EV	AC
J 31,362	100	31,362	100	31,362	30,550
M 138,571	100	138,571	100	138,571	155,180
V 20,945	50	10,473	30	6,284	8,933
Y 20,352	0	0	0	0	0
L 21,272	100	21,423	100	21,272	20,644
Q 100,846	50	50,423	70		
W 4,235	0	0	0	0	0
X 6,622	0	0	0	0	0
344,205		252,101		268,081	288,657

Figure 11.13 ROSEBUD Project Status as of Week 13.

Example 11.9: Forecasting ETC and EAC for LOGON Project

From discussion earlier in the chapter, for the LOGON project at Week 20,

CPI = 429,000/530,000 = 0.81, thus,
ETC = (990,000 − 429,000)/0.81 = $692,593, and
EAC = 530,000 + 692,593 = $1,222,593.

Lacking other information, the revised project completion date is estimated as shown in Figure 11.14 by extending the EV line, keeping it parallel to the PV line, until it reaches the level of BCAC, $990,000. The *horizontal* distance between the PV line and the EV line at BCAC is roughly the schedule overrun (negative time variance) for the project; on Figure 11.14 this is roughly 3 weeks, so the estimated completion for the project is Week 50 instead of Week 47.

This revised completion date remains to be verified since an actual delay will depend on whether behind-schedule activities are on the critical path. From an earlier discussion we know that Activity M is on the critical path and is 3 weeks behind schedule, so the LOGON project is likely also 3 weeks behind.

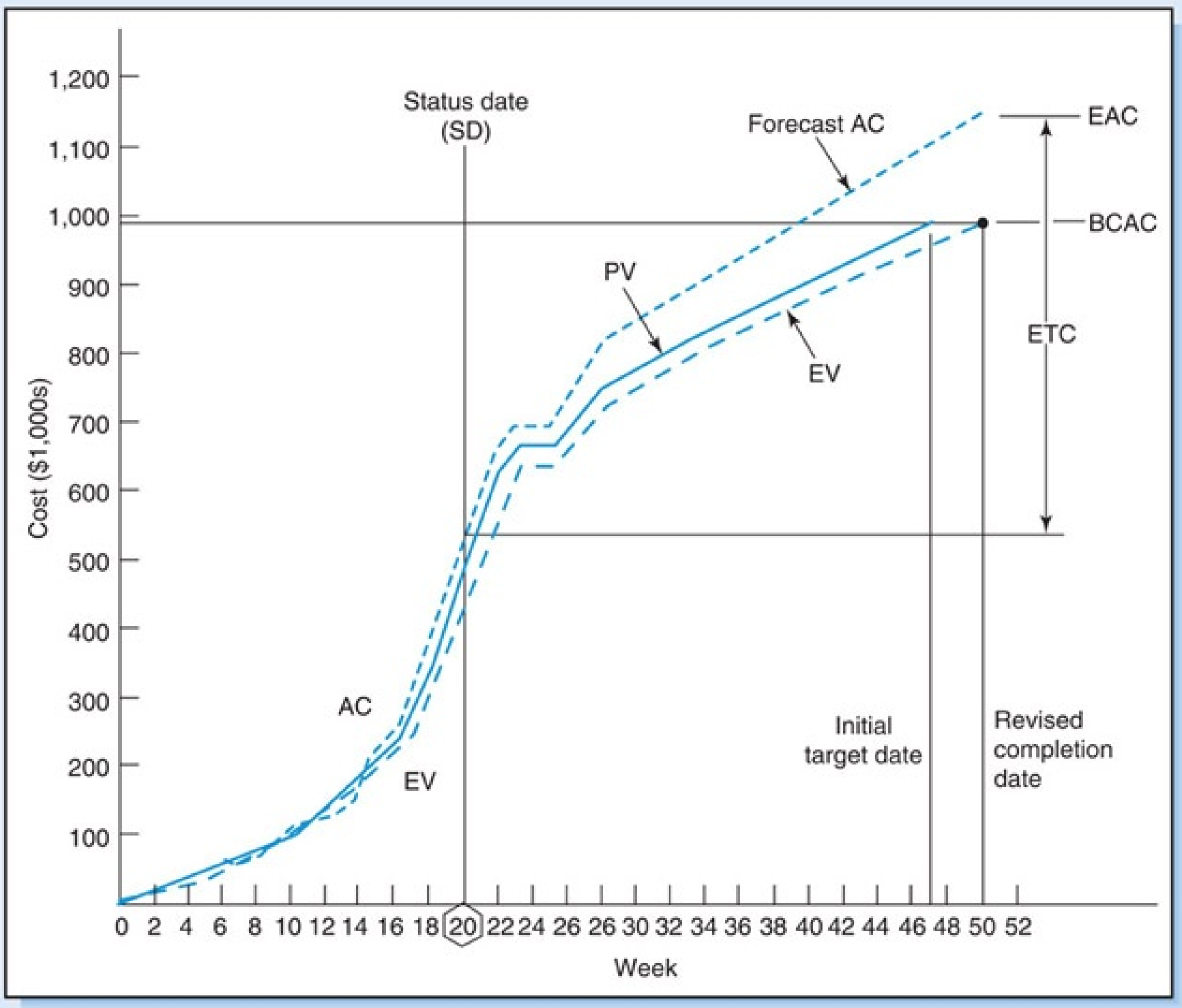


Figure 11.14 LOGON project status chart and forecast as of Week 20.

Shown in Figure 11.14 is another line, “Forecast AC,” which is an extension of the current AC line up to the EAC level ($1,159,630) and at the revised completion date of Week 50. This gives a running estimate of what "actual" costs might be until project completion.

It should be noted that the at-completion estimates assume that conditions and resources will neither improve nor worsen, which the LOGON 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 that 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 weeks (target date) – 19 weeks = 28 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 is 20 + 33.3 = Week 53.3.

Effect of Uncertainty

The EAC is based upon a single-value assessment of EV as of the SD. This assessment is usually based upon opinions about the percent complete, which means it is subject to uncertainty. Since EV is subject to uncertainty, so is EAC. A way to account for this uncertainty is to consider optimistic, pessimistic, and most likely values of EAC, as illustrated next.[9]

Example 11.10: Uncertainty in Forecasted EAC and Completion Date

The EV for the LOGON project as of Week 20 is $429,000. With the project budgeted at $990,000, this means the project is 43.3 percent completed.

Suppose an expert looks at the project and concludes that it is somewhere between 35 percent and 48 percent completed. These represent pessimistic and optimistic scenarios, respectively. The corresponding EVs are:

Pessimistic	0.35($990,000)	= $346,500
Most likely	(given)	$429,000
Optimistic	0.48($990,000)	= $475,200

Since AC = $530,000 and PV = $512,000, the range of possible CPIs, SPIs, EACs, and forecast completion dates are:

	CPI	EAC ($)	SPI	Week, PV Scheduled [a]	Revised Week of Completion [b]
Pessimistic	0.65	1,322,308	0.68	18	6.2.6
Most likely	0.81	1,222,593	0.84	19	53.3
Optimistic	0.90	1,102,222	0.93	19.5	49.6

(a) Approximate week where EV = current PV (see PV curve, Figure 11.14). For example, pessimistic EV = 346,500 since on Figure 11.14, PV = 346,500 at about Week 18.
(b) 20 weeks + {47 weeks – (PV, scheduled)}/SPI.

Figure 11.15 shows three points representing the forecasted costs (EAC) and completion times, and probability distributions for EAC and completion time.

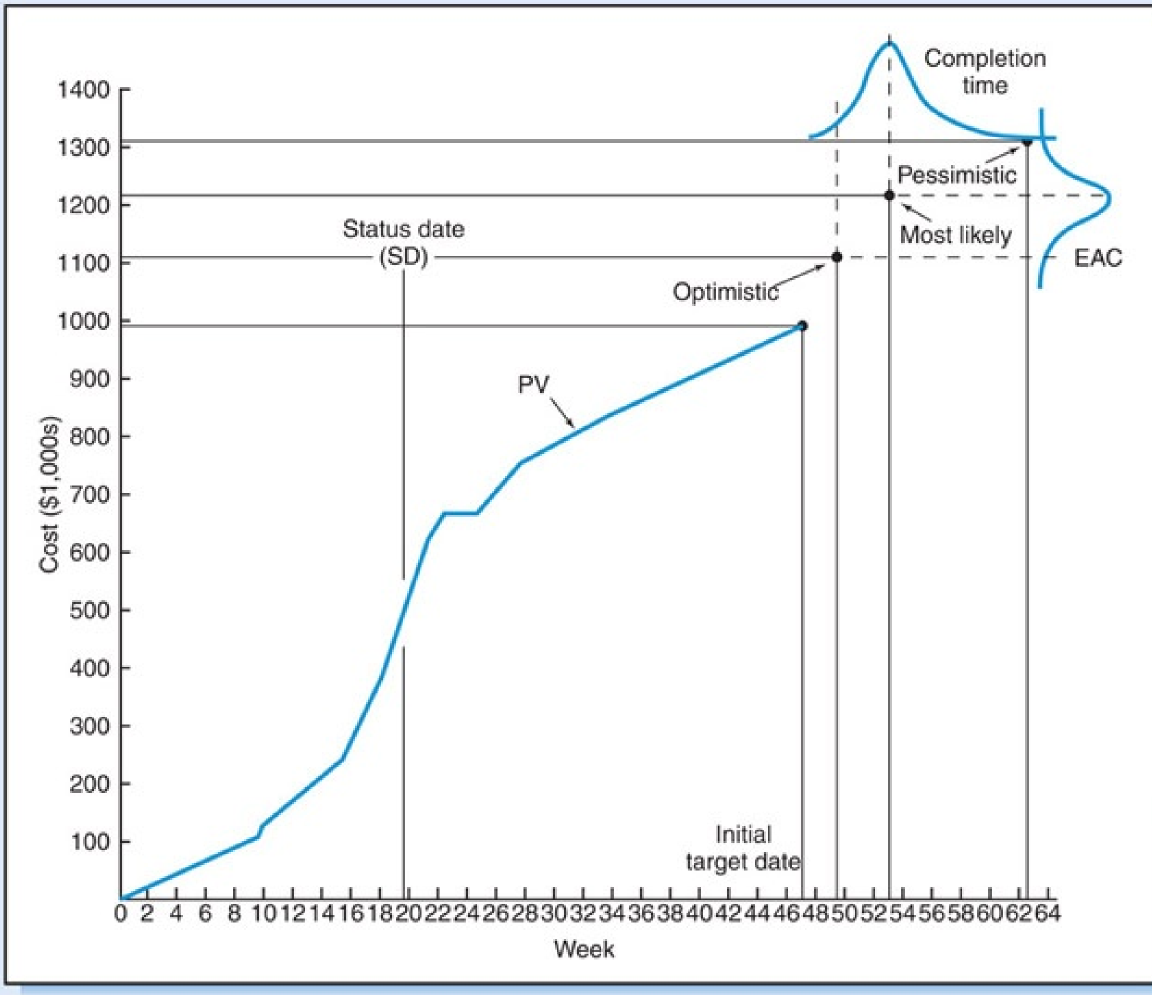


Figure 11.15 Estimated cost at completion (EAC) and completion times.

These estimated completion times do not reflect which, if any, currently behind-schedule activities are on the critical path; they reflect only the rate at which completed work was done and assume that that rate is uniform everywhere in the project, on critical and noncritical activities alike.

The optimistic, most likely, and pessimistic estimates for cost and time at-completion can be forecast periodically throughout the project by repeating the above forecasting procedure at, say, Weeks 30, 40, and 50. Figure 11.16 shows plots of the three EAC forecasts at these intervals starting in Week 20. The convergence of the pessimistic and optimistic forecasts in the figure suggests that the project is heading toward a most likely EAC of $1,150,000.

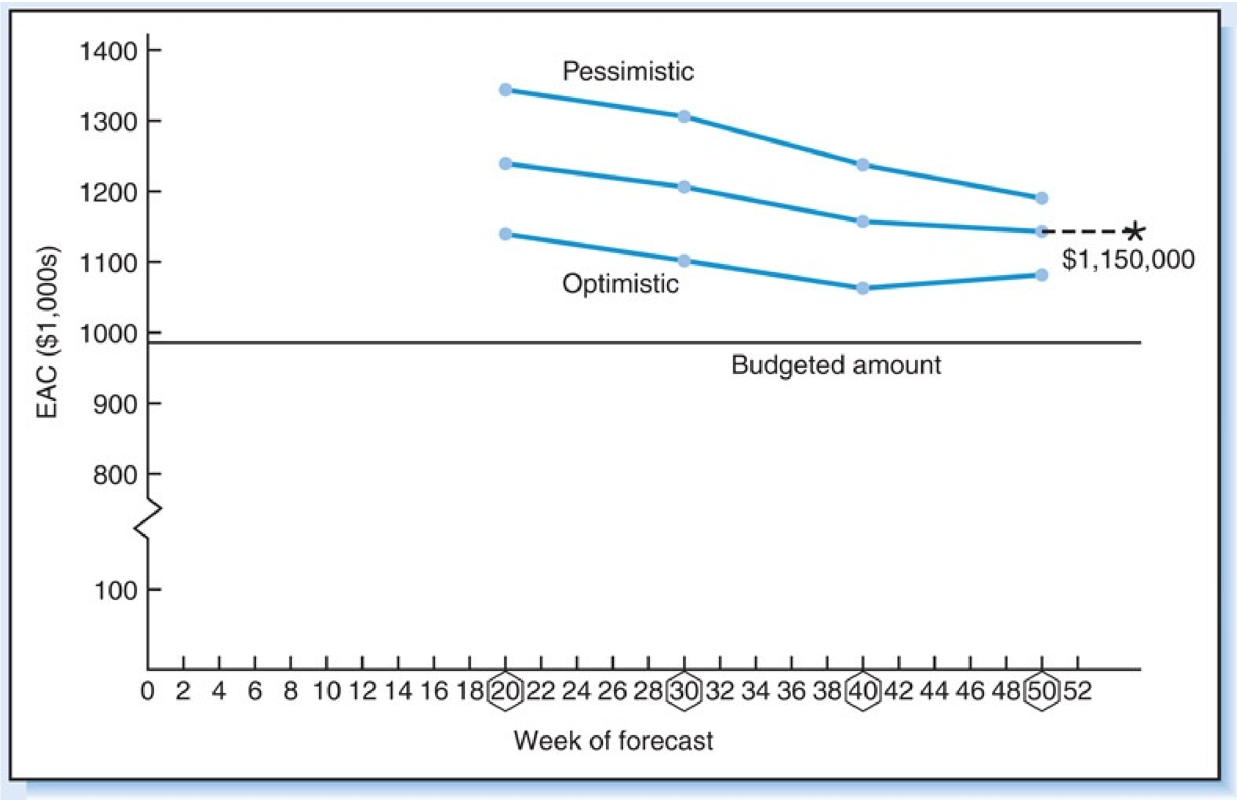


Figure 11.16 Critical chain = 24 weeks, project buffer = 12 weeks.

EVM Shortcomings

Earned value metrics can be inaccurate and misleading and so 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 simply because bills have yet to be paid. Whenever payments are made in periods other than when expenses are incurred or budgeted, the CV is skewed. This leads some companies to apply EV methods 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 should always be scrutinized to identify reasons for variances.

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 laid, miles of asphalt laid, number of fixtures installed, etc.), but are difficult or impossible when work output (e.g. drawings produced or lines of code written) cannot be measured in uniform units (not all drawings or code lines require the same amount of time to produce).

EVM and critical chain project management can be used in combination—providing the purpose of each is clearly differentiated. EVM is a cost monitoring and reporting method; it does not distinguish “critical” activities on the schedule. CCPM is a scheduling tool; it does not address costs. The two methods occasionally give conflicting signs about project performance (SPI or TV in EVM uses vs. buffer consumption in CCPM). One rule of thumb is to use EVM for cost tracking and reporting (perhaps as required by a customer), but to base resourcing and scheduling decisions on buffer consumption and CCPM.[10]

Technical Performance Measurement

Besides costs and schedules, performance depends on how well the project is meeting technical requirements. *Technical performance measurement* (TPM) is a method for tracking the history of technical objectives or requirements over time. TPM’s 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.[11]

To perform TPM, it is necessary to first specify the technical performance measures that are *key indicators* for the end-item system. These measures should be tied to customer requirements and represent major performance drivers. A large-scale system might have a dozen or so high-level measures, in which case it is necessary to define the design parameters upon which each measure depends and to set required values for these parameters. Examples of performance measures 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 measures are based upon estimates from computation, modeling, and simulations; later measures are derived from test and demonstration results on actual hardware and software. Estimates and actual measures of the technical objective are plotted on a TPM chart that shows 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 11.11: TPM for Design Tradeoff Decisions

Based on Example 10.5 in Chapter 10, design target weights for components of a spacecraft navigation system were set at 44 pounds for Subsystem A and 26.4 pounds for Subsystem B. Design margins were also established for the two subsystems to cover the risk of not meeting these targets; the margins represent amounts by which the target values could be exceeded and still achieve system requirements.

See Chapter 10

The TPM chart in Figure 11.17 shows the design progress (actual versus target values) for the two subsystems; charts like this and are used to make design tradeoff decisions. This chart shows current performance and design targets at three project milestones:

1. At the time of the *preliminary model demonstration,* the actual measured weights for both subsystems were too high, although Subsystem A was relatively much closer to its design target than was Subsystem B.
2. By the time of the *pre-critical review,* Subsystem A had been improved and was close to its target weight; however Subsystem B was still far away from its target. It was clear that Subsystem A would be able meet its target of 44 pounds but Subsystem B would *not* be able to meet its target of 26.4 pounds. The decision was made to reduce Subsystem A's unused design margin by 3.6 pounds and to increase Subsystem B's design target by that amount to 30 pounds.

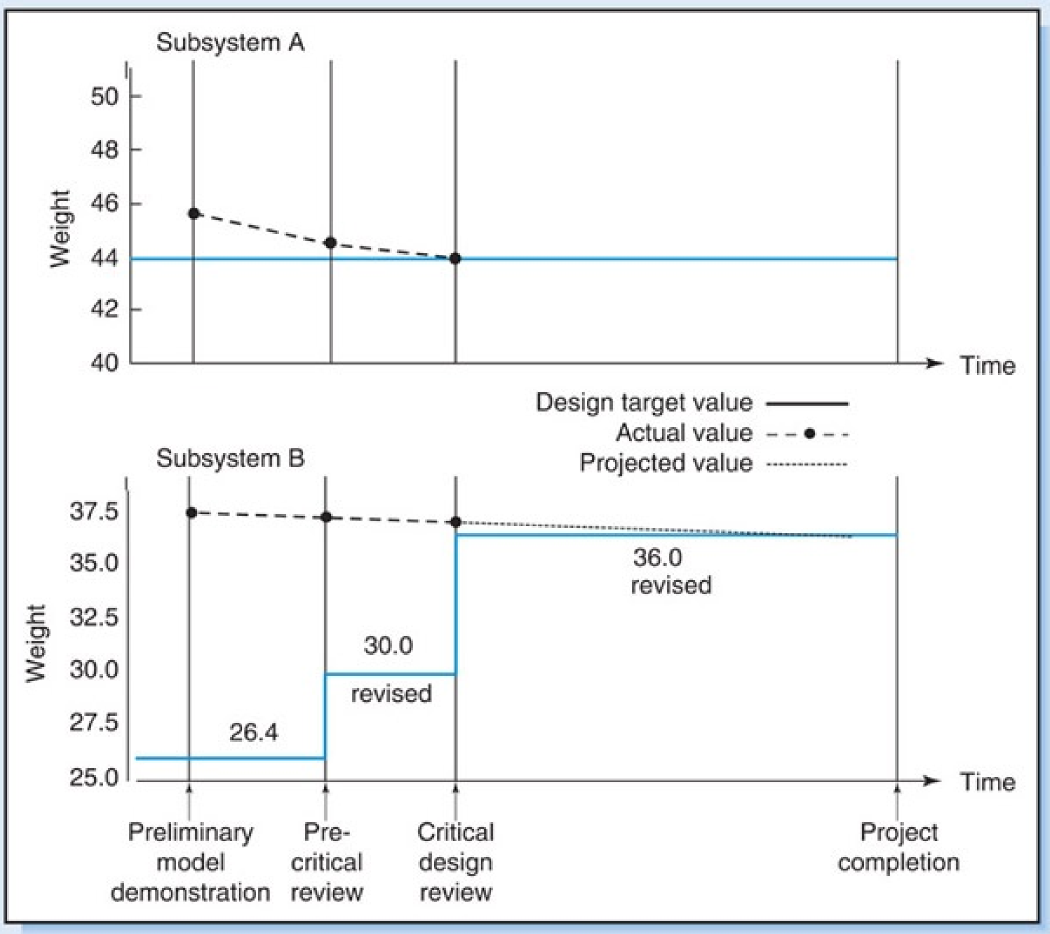


Figure 11.17 Time-phased TPM charts for Subsystems A and B.

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 6 pounds more from Subsystem A's unused design margin to Subsystem B's target, increasing it to 36 pounds. The dotted line for Subsystem B beyond the critical design review is the improvement still necessary to achieve Subsystem B's revised target value.

11.8 Issue Management

An *issue* is an emergent problem, question, or matter in dispute. It can stem from anywhere, but it 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 as they emerge. Not every detail can be addressed in the project plan, and not every circumstance can be anticipated. Issues certainty *will arise* and follow-up actions will be necessary.

Issues, risks, and changes are related: a materialized risk (a risk that has arisen or is certain to arise) or a change request can result in an issue; or, an issue can result in a change request or new risk. Each involves an "action item"—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 should be documented, prioritized, tracked, resolved, and closed out. Similar to controlling changes (discussed next), issue management involves the steps of identification, documentation, analysis/evaluation, communication, action, monitoring, and closure. These steps can be described in relation to the issue log in Figure 11.18, which is a record of all issues encountered. The log is retained throughout the project and updated whenever a new issue arises or changes status.

Shown in Figure 11.18, associated with each issue is 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. For example, some companies specify issue type as "technical," "organizational," "stakeholder," or "procurement"; others use "Risk," "Change request," or "Emergent (problem)." The *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. The result of the assessment is an assigned *priority* (e.g. 1–5); this can be based on the *severity* (1–5) of the issue and other pending issues as determined by its impact. The result of the analysis 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 issues log and an update of each issue's *current status*; this might be indicated by a single letter representing, for example: **N**ot started, Analysis, **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 10.1) and change control (substitute "issue" for "change" in Figure 11.20).

Project: LOGON

Project Manager: Frank Wesley

Last updtate:

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 11.18 Issues og.

One way to gauge project progress is by monitoring issue status. All projects encounter issues, but the "healthy" ones resolve them expediently; when issues remain unresolved, the issue backlog grows and becomes increasingly difficult to handle. Figure 11.19 shows the number of unresolved (not C or X) issues at a particular time in the project and how long they have been on the log. The shallow decline in the top curve represents a troubled project: numerous issues are unresolved, and most are many weeks old. The bottom curve represents a healthy project where issues are resolved rather quickly.

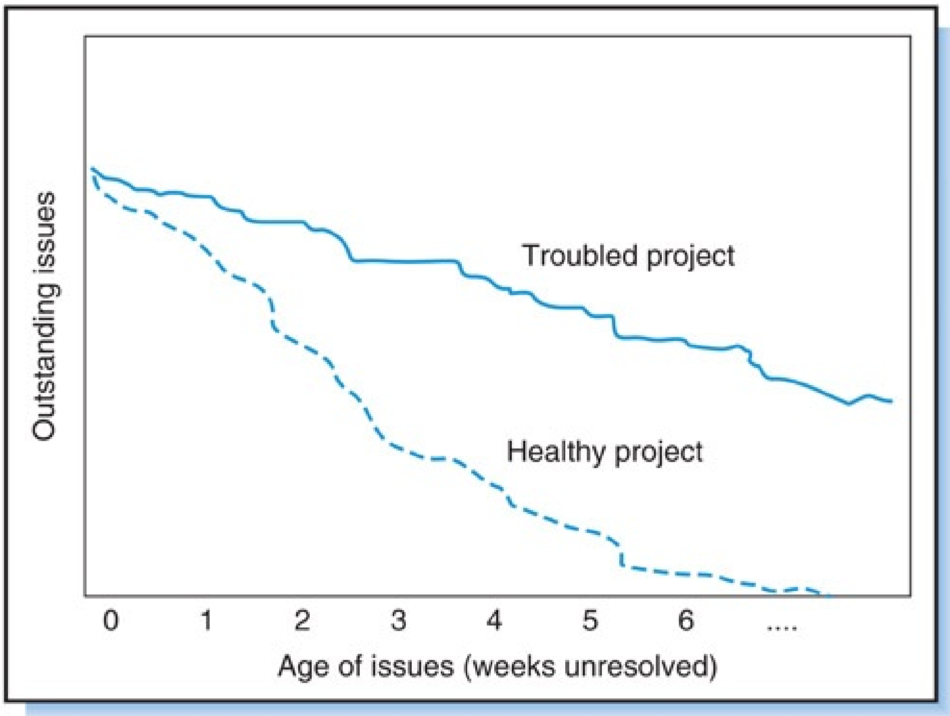


Figure 11.19 Number of unresolved (outstanding) issues versus age of issues.

11.9 Change Control

No project goes entirely according to plan. Changes to the end-item system and project plan are inevitable because of planning oversights, new opportunities, or unforeseen events and problems. Such changes require modifying the work, reorganizing or adding personnel, and trading off among time, cost, and performance. Alterations to specifications and sacrifices to technical performance are sometimes necessary to meet time and cost constraints.

The Impact of Changes

Generally, the larger and more complex the project, the greater the number of changes and the more that actual costs and schedules deviate from objectives. Changes are a chief 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, changes are then required to other elements of the end-item and project, and these impact still others.

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 or redone, and materials scrapped. Morale is affected too: people see their work dismantled, discarded, and redone; everyone is under pressure to get the project back on budget and schedule.

Reasons for Changes

A typical project experiences the following kinds of changes:[12]

1. Changes in project scope and specifications. As a rule of thumb, the more

uncertain the project, the more likely the scope and specifications will have to be altered later in the project.

2. Changes in design because of errors, omissions, unknowns, afterthoughts, or revised needs. Mistakes or omissions must of necessity be corrected or accommodated, but customers often try to squeeze in unnecessary changes that are beyond the original scope (but, they hope, for the original price).
3. Changes mandated by government codes (health, safety, labor, environment), labor contracts, suppliers, the community, or other stakeholders in the environment.
4. Changes that are believed to improve the rate of return of the project.
5. Changes perceived to improve upon original requirements. Many people want to improve upon their work; although apparently desirable, these improvements can expand the project beyond its original scope and requirements.

Examples of the above changes include: (1) After the design has been completed, increasing the payload requirement on a space probe to allow for additional necessary hardware; (2) encountering a buried cable during excavation, which must be rerouted; (3) interrupting work because of labor problems or municipal code violations; (4) altering the design capacity of a refinery under construction to increase the refinery's output rate; (5) adding more features to an already acceptable software design (bloatware) to enhance the perceived marketability. Note that some changes 1, 4, and 5 are *discretionary*, i.e., they can be rejected, while 2 and 3 are *de facto*, i.e., they already did "happen" and must be accepted.

Change Control System and Configuration Management

One way to manage changes and reduce their negative impacts is to employ a formal *change control system*. Because making changes is similar to other aspects of project work—i.e., they must be defined, scheduled, and budgeted—the process of drafting and implementing changes is similar to the project planning process. The purpose of the change control system is to review proposed design and work changes, weed out all but the necessary ones, and make sure that all related work is also reviewed, revised, and authorized. According to Harrison, the system should:[13]

1. Continually identify changes as they occur.
2. Reveal the impact of changes on project costs, project duration, and other tasks.
3. Accept or reject proposed changes based upon analysis of impacts.
4. Communicate changes to all parties concerned.
5. Specify a policy for minimizing conflicts and resolving disputes.
6. Ensure that changes are implemented.
7. Report monthly a summary of all changes to date and their impact on the project.

The change control system is established early in the project and thereafter used to appraise the impact of proposed changes on estimates and plans, and trace any variance in current performance and original estimates to specific approved changes.

See Chapter 9

The change control system focuses on the project but it is part of a broader process for controlling and integrating changes into the design, development, building, and operation of the end-item system, its subsystems, and components—the *configuration management process* mentioned in Chapter 9. When an element of the design (e.g. a performance criterion) or the project plan (e.g. scope statement, schedule, or budget) is first approved, it is referred to as the *baseline*; any time the baseline is subsequently altered to reflect approved changes, it is referred to as second baseline, third baseline, and so on.

To control and minimize design and work changes, the change control system includes the following procedures:[14]

- 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 a formal request and approval process 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.

The process, summarized in Figure 11.20, ensures that all discretionary and de facto changes to design and work tasks are documented as to their effect on the project and the end-item, formally assessed, and accepted or rejected.

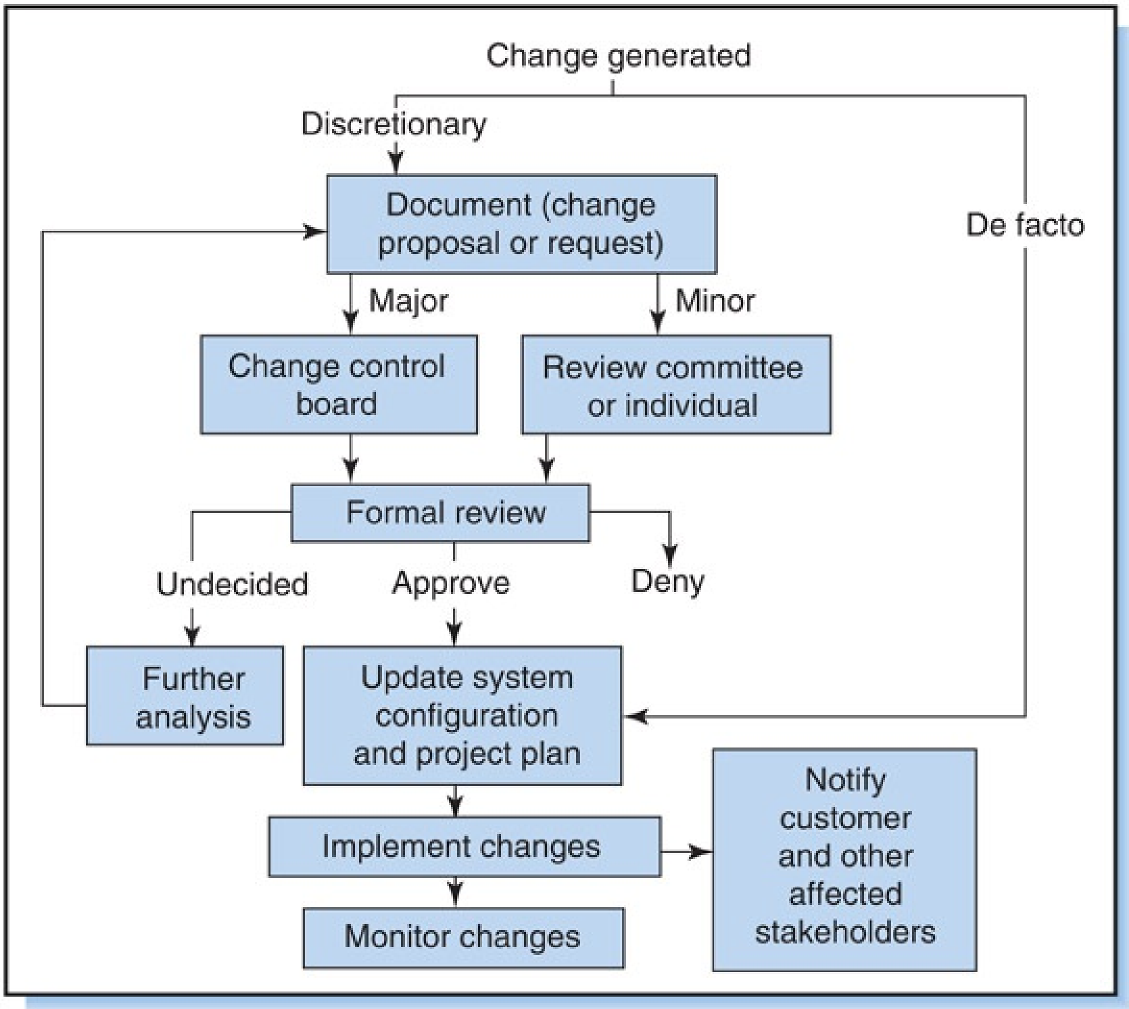
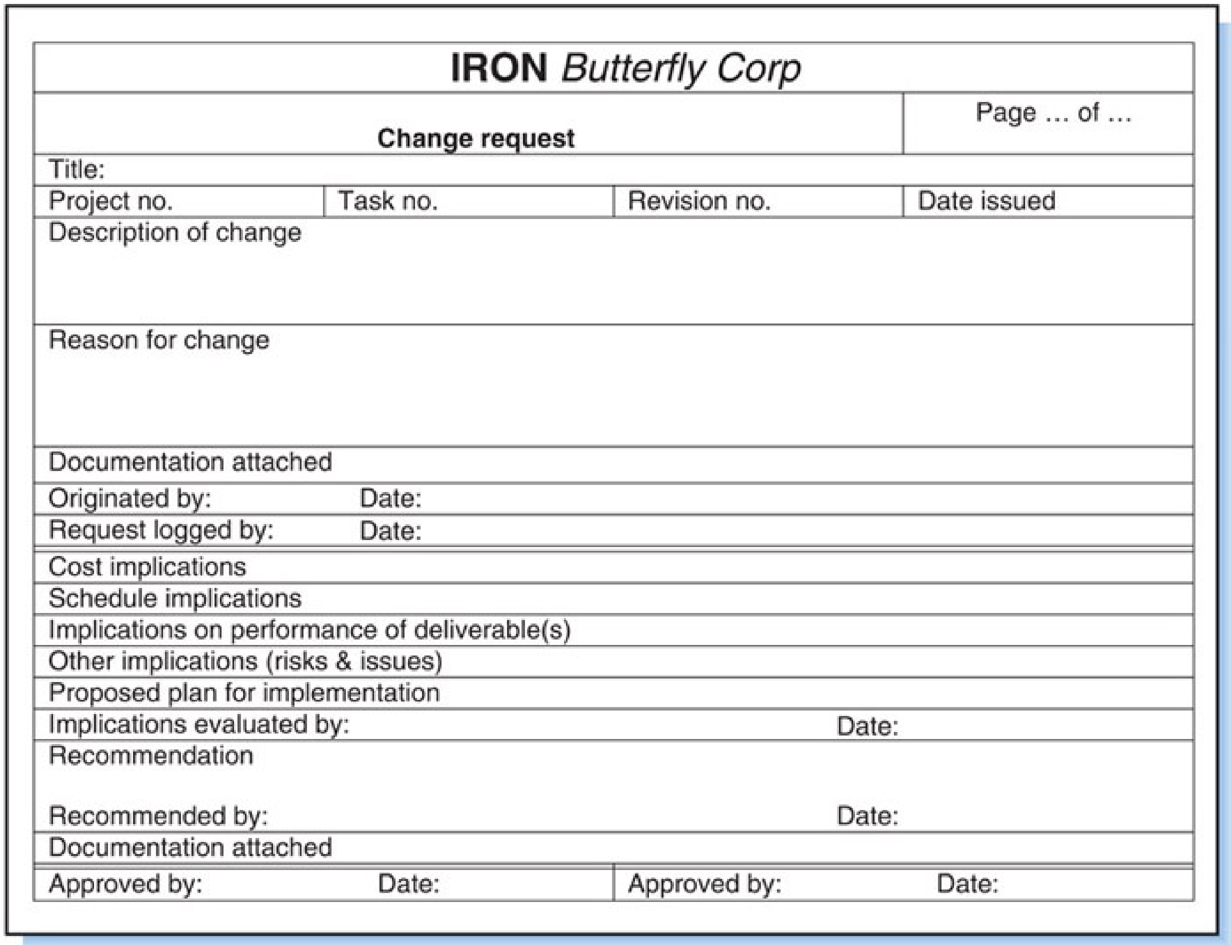


Figure 11.20 Change control process.

A key part of the process is the change request document (Figure 11.21), which provides information about and the rationale for a proposed change. 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.



IRON *Butterfly Corp*

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 11.21 Example of change request document.

In large projects a *change control board* meets 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 nature of work of a single task and related tasks must be documented. Everyone involved in the project has the potential to recognize or originate changes, and everyone must be expected to bring them to the attention of the project manager.

11.10 Contract Administration

Contract administration is responsible for ensuring that all commitments by the developer/contractor and the customer as specified in the contract are met.[15] Procurement management, discussed in Chapter 5, is an aspect of contract administration that deals specifically with managing relations with suppliers and subcontractors who provide contracted goods, work, and services.

See Chapter 5

Contract administration is an aspect of project control that pertains exclusively to contracted work; it includes authorizing work to begin; monitoring work with respect to budgets, schedules, and technical performance; ensuring quality; controlling changes; and sending and receiving payments for work completed. Under contract administration, change requests for contracted work are assessed against conditions as stated in the project contract; in case of differences, changes are implemented only after approvals are secured and the contract modified. Contract administration is managed using procedures similar to those described for task authorization, performance tracking and reporting, and change control. When the customer requires additional specific procedures for project monitoring and reporting, these must be incorporated in the contractor's project tracking and control system.

Contract administration also ensures that customers are invoiced for deliverables as specified in the contract, and that subcontractors and suppliers are paid. 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.

11.11 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 problematical for the following reasons:[16]

1. The monitoring and control process focuses on only one factor, 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 involved. Forcing compliance to one factor results in excesses or slips in other areas; 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 when they ignore those who don't comply with control procedures.
3. Project team members do not report problems they are aware of. They may not understand the situation; if they do, they might be hesitant to reveal it. The 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; this is a major obstacle to effective project control.
5. Managers act indifferently about controversial issues, believing that with time problems resolve themselves. This leads workers to believe that management doesn't care—an attitude likely to spread to others throughout the project.
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 because often it results in mischarging customers.

Management must be aware of these problems and work to eliminate them from the monitoring and control process. Above all, it must strive for a process that is impersonal, objective, and uniformly applied to all aspects of the project—people, parties, and tasks.

11.12 Summary

The execution phase includes the stages of *design, production/build*, and *implementation.* During the design stage the system concept is subdivided into tiers of subsystems, components, and parts, and for each of these designs, schematics and models are created. The result is a *functional* design and a *physical* design. In the production/build stage the main activities are fabrication and testing. Components are assembled and the end-item system is produced and tested to ensure that requirements for the system and its components are met. Project management is responsible for coordinating all activities and controlling any changes.

Throughout the execution phase, the project control process guides the project to keep it moving toward scope, budget, schedule, and quality objectives. The focal point of control is 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 measures are monitored to gauge progress toward meeting technical objectives. Performance to date is reviewed using the earned value concept, and estimates of project cost and completion date are revised.

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 system and configuration management ensures that changes are documented, assessed and authorized, and communicated.

The next chapter concludes the subject of project execution and covers the topics of project evaluation, reporting, and communication. It also covers the remainder of the execution phase—system implementation and project close out, and the last phase of the systems development cycle, Operation.

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
BCSP = budgeted cost, scheduled performance = date where PV equals EV at status date
TV = time variance

Review Questions and Problems

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. What is contract administration?
8. What are the three phases of the project monitoring and control process?
9. Explain the differences between internal and external project controls.
10. How are overhead expenses allocated in work packages?
11. If a cost or schedule variance is noticed at the project level, how is it traced to the source of the variance?
12. Describe the process of work authorization. What does a work order usually include?
13. Describe the process of collecting data about the cost, schedule, and work accomplished.
14. Discuss different ways of measuring ongoing work progress.
15. Why is scope change control an important part of the project control process?
16. Discuss quality control as applied to projects.
17. 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.
18. Refer to Example 11.3.

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 in the fever chart is the project?
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?

19. 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.
20. What does it signify if cost or schedule index figures are less than 1.00?
21. Explain TPM, its purpose, and how it is conducted.
22. What is an "issue"? Explain "issue management" and how it is implemented through the issues log.
23. Explain ETC and how it is related to EAC.
24. Discuss reasons why the project manager tries to resist project changes.
25. What should a change control system do? Describe procedures that minimize unnecessary changes.
26. What aspects of project control fall under contract administration?
27. What are some difficulties encountered when attempting to control a project?
28. Use the networks in Figure 11.22 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.
29. In the LOGON project suppose the status of the project as of Week 22 is as follows (note usage of the longer acronyms; some project management software packages use these and not the shorter acronyms PV, AC, and EV).

BCWS = $628,000

ACWP = \$640,000
BCWP = \$590,000

Answer the following questions:

a. What is the earned value of the project as of Week 22?
b. Compute SV and CV.
c. Draw a status graph similar to Figure 11.14 and plot BCWS, ACWP, and BCWP. Show SV and CV. Determine TV from the graph.
d. Compute SPI and CPI. Has the project performance improved or worsened since Week 20?

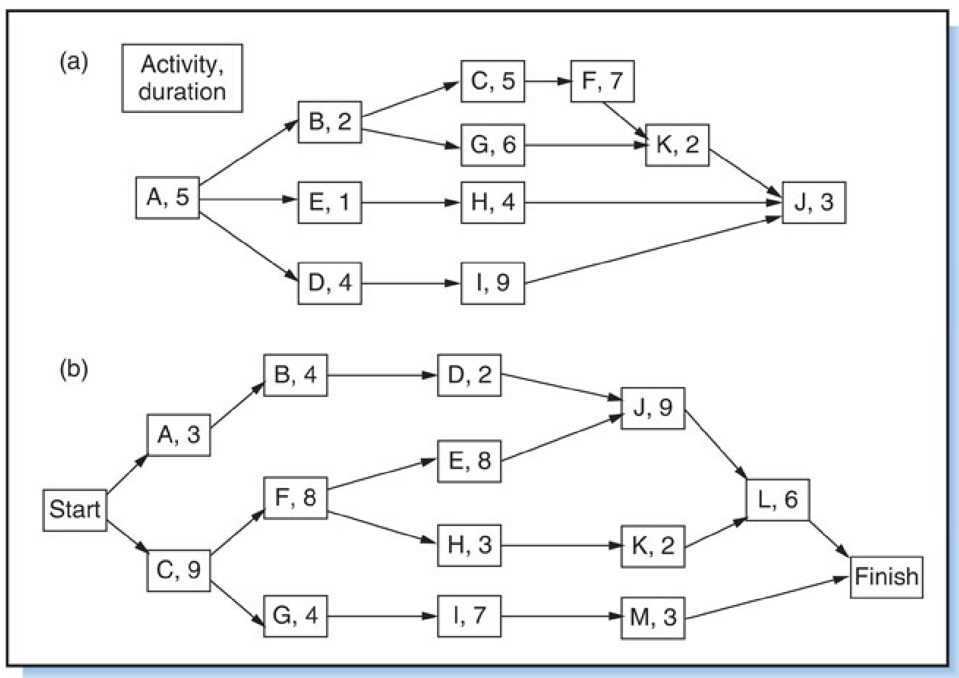


Figure 11.22 Two project networks.

e. Using 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?

30. 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.
31. Using the status chart in Figure 11.23:

    a. Estimate SV, CV, and TV, and 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.

32. 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?

33. Refer to Problem 29. Assume for Week 22 the $590,000 indicated is the "most likely" EV. Given a BAC of $990,000, this represents 59.6 percent of the project completed. Suppose an expert assesses the LOGON project at that time and concludes that LOGON is between 50 percent and 65 percent completed—these are the pessimistic and optimistic scenarios.

Compute the corresponding pessimistic, most likely and optimistic CPIs, SPIs, EACs, and forecast completion dates for the project.

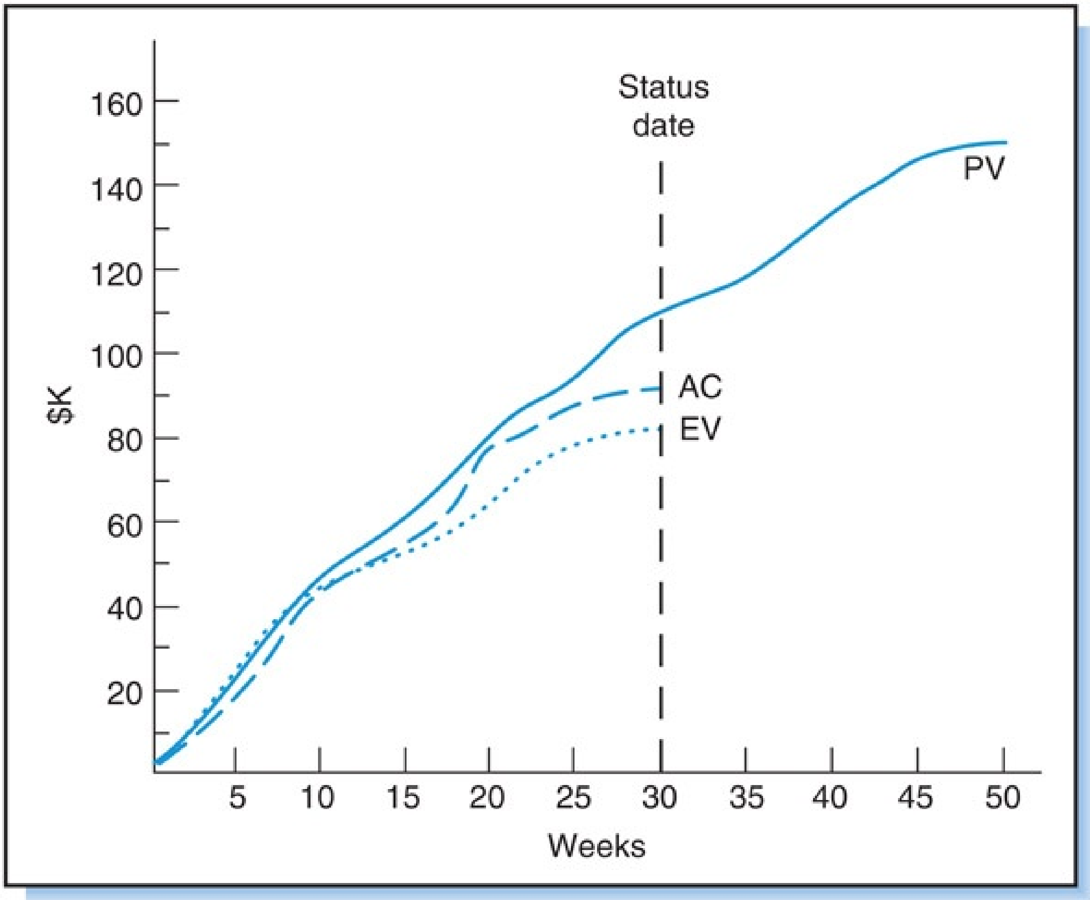


Figure 11.23 Project status as of Week 30.

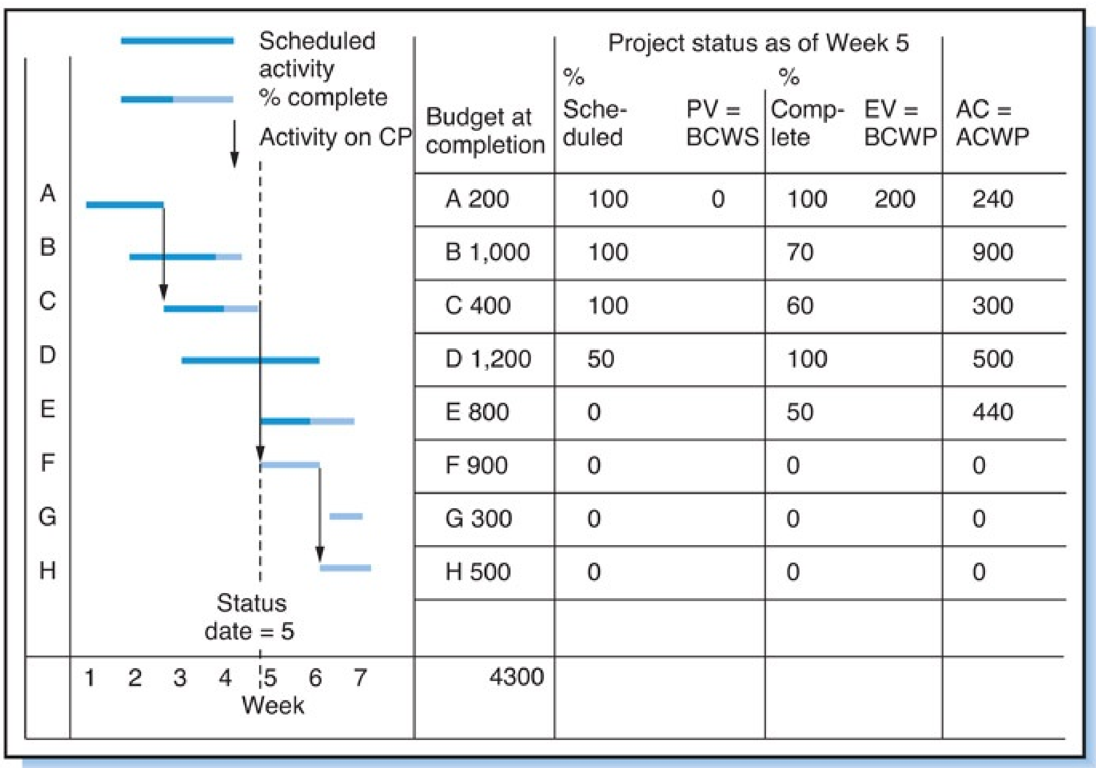
34. Refer back to Problem 28 and Figure 11.22.

    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?

35. For the following questions refer to Figure 11.24.

As of Week 5, for the project:

a. What is the planned value (PV)?
b. What is the earned value of the work completed (EV)?
c. What is actual cost of the project (AC)?
d. What is the value of work remaining?
e. What is the CPI?
f. What is estimated cost to complete the project (ETC)?
g. What is the forecasted cost at completion (EAC)?
h. What is the estimated cost variance at completion, and the percent overrun or underrun?
i. According to EV, is the project ahead or behind schedule?
j. According to the critical path (A-C-F-H), is the project ahead of or behind schedule?



	Budget at completion	% Sche-duled	PV = BCWS	% Comp-lete	EV = BCWP	AC = ACWP
A	A 200	100	0	100	200	240
B	B 1,000	100		70		900
C	C 400	100		60		300
D	D 1,200	50		100		500
E	E 800	0		50		440
F	F 900	0		0		0
G	G 300	0		0		0
H	H 500	0		0		0
	4300					

Figure 11.24 Project status as of Week 5.

Questions About the Study Project

1. What kinds of external controls, if any, were imposed by the client on the project?
2. What kinds of internal controls were used? (For instance, work package control, cost account control, etc.) Describe.
3. 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—time, invoices, etc.
    c. How was the data tallied and summarized?
    d. How was the data validated?

4. Was the concept of earned value (budgeted cost of work performed) used?
5. How was project performance monitored? What performance and variance measures were used? Was the buffer management concept used? Explain.
6. How were problems/issues pinpointed, tracked, and acted upon?
7. Were the concepts of forecasting ETC and EAC used? If so, by whom? How often?
8. Were variance limits established for project cost and performance? What were they? How were they applied?
9. When cost, schedule, or performance problems occurred, what action did the project manager take? Give examples of problems and what the project manager did.
10. What changes to the product 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 11.1 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; 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 an hour late at the next meeting, which, after 3 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 and make changes 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 11.2 SA Gold Mine: Earned Value After a Scope Change[17]

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 $10,000) per meter of shaft depth. For the station at the bottom, 3,0000 m^3 of rock would have to be excavated within 3 months at a cost of R700 per m^3. 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 11.25) with a volume of 2,0000 m^3. 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 m^3 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.

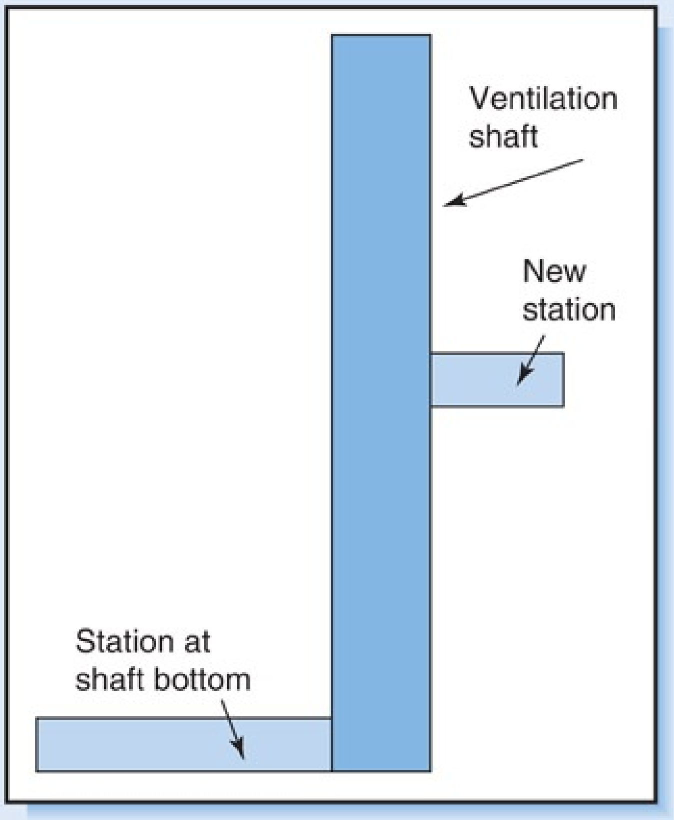


Figure 11.25 Mine shaft.

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 11.3 Change Control Process at Dynacom Company[18]

At Dynacom Company, any change that potentially affects project scope is subject to a rigorous review and approval process. Anybody requesting a 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. The TCM has no association with the project and consists of upper managers and other project managers. If the 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 well in helping it 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?

Endnotes

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. The terms "variance" and "deviation" are used here interchangeably, although in some contracts variance refers to small changes in the project plan for which compensation or correction is expected, whereas deviation refers to large changes that require a formal contractual response.

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. 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.

7. Cusumano M. and Selby R. *Microsoft Secrets.* New York, NY: Free Press; 1995, pp. 204, 221, 256–257, 417.

8. 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.

9. Sigurdsen A. Method for verifying project cost performance. *Project Management Journal* 25(4); 1994: 26–31.

10. 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.

11. For examples of analytical models used for TPM, see Eisner H. *Computer-Aided Systems Engineering.* Upper Saddle River, NJ: Prentice Hall; 1988, pp. 297–326.

12. Harrison, F. *Advanced Project Management.* Hants, England: Gower; 1981, pp. 242–244.

13. Ibid., pp. 245–246.

14. Ibid., p. 244; Archibald, *Managing High-Technology Programs and Projects*, pp. 187–190.

15. Hirsch W. *The Contracts Management Deskbook*. New York, NY: American Management Association; 1986, Chapter 6.

16. Roman, *Science, Technology, and Innovation*, pp. 327–328, 391–395.

17. Source: Mr. P. Joubert of Anglo Platinum.

18. 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 12
Project Evaluation, Communication, Implementation, and Closeout

We look at it and we do not see it.

—Lao-Tzu, *Sixth century* BC

The previous chapter described how a project is tracked and guided so as to meet schedules, costs, and performance targets. As the project advances, the project manager tracks and evaluates its progress and communicates its status to workers, upper management, and the customer. The first part of this chapter discusses methods for assessing and reporting project status and the broader topics of project communication and information systems. Clearly, project communication happens throughout the project, so these topics overlap with related topics discussed earlier in the book.

The end of the execution phase—and of the project life cycle—is implementation and closeout. As defined here, implementation refers to the point in the project when the end-item is installed in the customer's environment and turned over to the customer. In some projects, implementation occurs concurrent with system fabrication/building, in others it is a separate stage. A duty of the project manager is to assist the customer in taking ownership of the end-item.

All projects come to an end, and the project manager is responsible for the orderly closeout of the project. IPMA considers closeout a necessary technical competency; in PRINCE2 it is one of seven key project management processes. The second part of this chapter covers implementation and closeout, as well as what happens in the final phase of the system development cycle, Operation.

12.1 Project Evaluation

The purpose of project evaluation is to assess performance, reveal areas where the project is not achieving goals, and uncover extant or potential problems so they can be corrected. Although it is certain that problems and deviations will occur, it is not known a priori where or when. Evaluation for the purpose of guiding the project is called *formative evaluation;* it addresses the questions "What is happening?" and "How is the project proceeding?" Evaluation for the purposes of appraising the project after it is completed and assessing the end results is called *summary evaluation*; it addresses the questions "What happened?" and "What were the results?" It is the central theme of the "post-project" and "confirm benefits" stage of the PRINCE2 project methodology.

Project Formative Evaluation

Methods and Measures

A variety of methods, measures, and sources should be used to evaluate progress and these should be specified in the project plan. Using a variety of measures and sources increases the validity of the evaluation, particularly when they all lead to the same conclusion. The primary ways project information is obtained and conveyed are written reports, oral reports, observation, and review meetings.

Written reports are the most common and expeditious way to communicate cost, schedule, and work performance information. They can be very informative —unless the writer chooses to hide or obscure information. Oral reports provide a quick way to convey information, although their accuracy depends on the interpretative and verbal skills and honesty of the presenter. Report accuracy, both oral and written, also depends on the number of channels through which the information must pass to get to the writer or presenter; in general, the more channels, the lower the accuracy. Project managers know this, so besides reports they also gather information by walking around the project, talking to people, and making their own firsthand observations.[1]

Site Visits

Most project managers would never rely solely on second- or third-hand reports or remote sources like email to track project progress. If they cannot always be at the project site, they make it a point to visit it often—unannounced and uninvited. At the site they try to speak to team members informally at lunch or on break. In this way they show their active involvement in the project, learn what is happening, and build relations with the team.

Instead of inquiring about project "status," it is sometimes better to ask people about how life is going, what is going well or not so well, and whatever resources or support they need. Just because no one reports problems or complains does not mean everything is okay. Signs that problems might be brewing include team members being silent or not participating in meetings, avoiding discussions about the project, or giving conflicting reports about the project. The project manager watches people's facial expressions and body language. Rather than trying to talk to everyone, she concentrates on people whose tasks have traditionally been the most problematic. She tries to validate reported problems by getting at least two points of view.

Technology

The manager of a geographically dispersed project cannot be at every site and has to rely on technology — video- and audio-conferencing, websites, email, and cell phone. Video-conferencing can be effective but requires the appropriate technical facilities; audio-conferencing can be good too, but involves careful scheduling so as not to waste people's time. The Internet is effective for broadcasting plans, reports, documents and memos, however it is passive and does not *require* that people see or respond to posted documents. Most project managers, especially in construction, rely somewhat or heavily on cell phones and tablets for on-site communication.

In long-distance, international projects the best form of communication is frequent one-on-one *telephone* conversations, which allow the project manager to hear tone of voice, probe details, and obtain real-time feedback. But site managers and contractors are not always completely truthful, so the project manager also needs a trusted source at the site to *observe* work and report back progress. This is discussed in Chapter 19.

See Chapter 19

A good rule of thumb for communication is: the more sensitive the issue, the lower the technology to communicate it. For highly sensitive issues it is worth traveling to the site and meeting face-to-face. For relatively sensitive issues, telephone is okay; for non-sensitive issues, email is okay. Always follow up important discussions or commitments in writing.

12.2 Project Communication Management

Communication Plan

For larger projects the execution plan should include a communication plan that addresses the various forms of project communication—formal and informal, verbal and written. It includes a tentative schedule for all formal reviews and milestone meetings, and describes the meetings' formats, expected itineraries, advance preparations, presentation time limits, attendance policy, and who will lead. It also specifies important points of contact (who's-who) among the customer, contractor, subcontractors, supporters, and other interest groups.

The table in Figure 12.1 shows part of a communication plan that specifies the expected meetings and reports, and participants for each; not shown, the plan could also include the kickoff meeting, project charter, phase and project closeout meetings, health and safety meetings, etc. The table would be supplemented with details about the what, where, when, and how for each kind of meeting and report.

The communication plan should be distributed to everyone on the project team and discussed before the project begins. So that everyone understands the required documentation and the content and format of each, the plan should include examples of good and bad documentation from previous projects. Much of this can be posted online.

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
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
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 12.1 Sample communication plan.

Documentation Management

Projects require and generate numerous documents, the volume of which can become overwhelming. Thus, many projects need a *documentation management system* (DMS) to ensure the required documents are created, conform to standards, and are organized and stored for easy accessibility by authorized persons. A computerized DMS might be needed to track, store, access, and update versions of digital documents.

Status Review Meetings

Status review meetings are among the most important ways to assess and communicate project progress. The main function of these meetings is to identify deviations from the project plan and quickly correct them. Meeting participants discuss project progress, issue status, current and extant problems, and opportunities. They can be informal and convened as needed, or formal and scheduled at key project milestones. Large projects require both.

Informal Reviews

Informal reviews are held frequently and regularly. They are also called "peer reviews" because they are attended by a group of peers. Actual participation depends on the phase of the project and issues at hand: only those team members, customer representatives, functional and project managers who need to be involved participate. Before the meetings, issue status and estimated completion dates and costs are updated. Attendees with assignments are expected to give presentations.

A major 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 of conflict should be avoided; such behaviors only waste time and discourage attendance.

See Chapter 11

Any problem surfaced in a review is noted on the issue log, described in Chapter 11, and one of the first orders of business at each review is to assess items on the log and the status of each. Always, the project manager, not a secretary or functionary, should lead the review, take notes and, where needed, write up and distribute the notes. This reinforces the perception that the leader is committed, involved, and in charge.

Standup Meetings

The daily “standup meeting” is a form of informal review. Intended primarily to update status, identify problems, and expedite solutions, the meeting is short (15 minutes) and to-the-point. Usually held at the start of the day, the team gives a quick run-though of yesterday’s progress and today’s next steps. The occasional surprise attendance of a prominent person—senior manager from the contractor or customer—adds zip and keeps everyone on their toes. Problems that require more than a minute’s reflection are deferred for a scheduled meeting.

Formal Reviews

Formal reviews are scheduled at milestones or critical project stages. Two such reviews are the *preliminary design review* and the *critical design review* as applied to design and development projects described in Chapter 9. The preliminary design review assesses the functional design specifications’ fit to the basic operational requirements; the critical design review assesses details of the design versus the preliminary design specifications. The reviews sometimes serve as a gate for continuing the project, and the decision to continue or terminate the project depends on the results of the review.

See Chapter 9

In every project, regardless of contractual obligations, the customer should assume some responsibility as project watchdog. The *project audit* is a special formal review initiated by the customer to independently assess project progress. It can be conducted at any stage of the project or upon any significant change to project costs, timeline, or goals. Audits are discussed in Chapter 9.

See Chapter 9

Project Meeting Room

Project meetings and conferences are often convened in a central meeting place or project office. The meeting room provides physical space for preparing, storing, and displaying project information. Gantt charts, networks, and cost charts showing planned and actual performance are displayed on the walls for easy reference. The room has a conference table, chairs, filing cabinets, computers, a projector, and, sometimes, teleconferencing equipment.

Formal Reports and Documents

Company management must be kept apprised of the status, progress, and performance of ongoing and upcoming projects. Problems affecting profits, schedules, or budgets, as well as recommended actions need to be reported promptly. Stakeholders (the customer; professional, citizen, and activist groups; public agencies; stockholders; and others who have a genuine interest in the project) should also be kept up-to-date. Frequent, honest communication with stakeholders builds trust and avoids surprises.

Reports to Top Management and the PMO

The project manager and staff send reports to top management and the project management office (PMO) using information generated by the PCAS or PMIS. The reports, available in written and online formats, include:[2]

1. Summary of project status.
2. Red flag items where corrective action has been or should be taken.
3. Accomplishments to date, schedule changes, and estimates for schedule and cost at completion.
4. Current cost situation and cost performance.
5. Manpower plan and limitations.

When several projects are simultaneously underway, the PMO compiles and provides to management monthly summaries showing their relative status. Each summary includes names of the customer and the project manager; monetary and labor investment; scheduled start and finish dates; possible risks, losses, and gains; and other information. The summaries enable management to assess the relative performance of the projects and their combined influence on the company, and the PMO to coordinate plans, authorizations, and resource allocations among the projects. When projects are managed as a portfolio (Chapter 18), the summaries help management decide which projects to continue, increase or decrease resources, or terminate.

See Chapter 18

Reports to Project, Program, and Functional Managers

On a large project, work package leaders and sub-project managers send monthly reports to the project manager about work completed, current and forecasted costs, and updated completion schedules, and the project manager sends related reports to the program manager (similar to Table 11.1 and Figure 11.14 in Chapter 11). Each month the project manager also sends reports showing costs incurred to the company financial manager or controller, and reports showing labor hours and costs expended for work packages to the functional managers. The reports in Figures 8.8 through 8.12 in Chapter 8, modified to include actual expenditures, are representative.

See Chapters 8 and 11

Reports to Customers/Users

Each month the project manager should send the customer a report about work progress and the impacts of any changes on work scope, schedule, or cost. Although the contractor's marketing or customer relations director might be formally charged with communicating contract-related information to the customer, it is up to the project manager to make sure the customer remains well-informed, and she must be available to answer questions and requests for project information. The customer should never be "surprised."

12.3 Project Management Information Systems

The formal planning and control methods described in this book do not require any more input data or information than is, or should be, available in any project. What they *do* require is, in a word, a *system* for collecting, organizing, storing, processing, and disseminating that information—a *project management information system (PMIS)*.

PMIS Software

Methods such as EVM, change control, and configuration management for a large project require processing and integrating a hefty amount of information. As computers are good at this, PMIS software has become an essential tool 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.

There are numerous kinds of PMIS project software packages 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 simple software has been mastered, it is easy to upgrade to more sophisticated software.

Features of PMISs

Following is a rundown of the kinds of analytical capabilities, outputs, functions, and features offered by PMIS software. Important to note is that among the many available software packages, most do not have all of these capabilities; some perform only the most basic functions.

Scheduling and Network Planning

Virtually all project software packages perform project scheduling using network-based algorithms to compute early and late times, slack times, and the critical path or critical chain. Among the capabilities to look for are the type of procedure (CPM, PERT, PDM, CCPM), the maximum number of allowable activities, the format for activities and events (some use a WBS scheme), and the quality and clarity of outputs (e.g., network, Gantt chart, tabular reports, or multiple types).

Resource Management

Most software systems perform resource loading, leveling, and allocation, but vary in analytical sophistication and quality of reports. Major considerations are the maximum number of resources permitted per activity or project; the kind of loading/scheduling techniques used (resource-limited, time-limited, or both); split scheduling (stopping and restarting activities); interchangeable usage of different resources; and rate of resource usage.

Budgeting

Software systems vary greatly in the way they handle costs and generate budget and cost summary reports. In some, cost and expense information are not treated explicitly; in others, cost accounting is a major feature. The PMIS software for large projects should have a cost and budgeting module (like the PCAS described in Chapter 8) that is integrated with modules for planning, scheduling, procurement, and tracking.

See Chapter 8

Managing Multiple Projects and Project Portfolios

Many software systems allow data to be pooled from different projects for *multiproject* analysis, planning, and control. This feature combines information from several concurrent projects to form a picture of the overall state of the organization. Some software systems provide a “dashboard” or overview of all projects. Managers can readily distinguish which projects are performing well from those experiencing problems or overruns—an essential capability for project portfolio management (Chapter 18). By “clicking” on a particular project, they can zoom-in to view more detailed information about the project.

See Chapter 18

Cost Control, Performance Analysis, and Change Control

Here is where project software capabilities differ considerably. To perform the control function, a system must be able to compare actual costs and work completed to budgeted and planned performance. Among the features to consider are the software’s ability to compute and report cost and schedule variances and EVM metrics (performance indices, and forecasts to-complete). The most sophisticated software packages “roll up” results and allow aggregation, analysis, and reporting at all levels of the WBS. They also permit modification and updating of existing plans through input of actual start and finish dates and costs. Plus, they help manage and reveal the impacts of change requests. Software with simulation capabilities integrate network, budget, and resource information and allow the project manager to ask “what if” questions under various scenarios.

Interface and Flexibility

Some PMIS software packages are compatible with and tie into existing databases for payroll, purchasing, inventory, ERP, cost-accounting, or other PMISs; some can be used with popular DBMS, modeling, and risk analysis systems.

Systems vary widely in their flexibility. Many perform a narrow set of functions; others allow the user to develop new applications or alter existing ones, depending on need. Among the applications available are change control, configuration management, responsibility matrixes, expenditure reports, cost and technical performance reports, and technical performance summaries. Many systems allow easy access through a browser to a variety of business applications and databases utilizing Internet technology.

Web-Enabled Project Management[3]

Many project management software products take advantage of web-enabled technology that offers plans and reports on interactive websites. This technology is well suited for situations where the project team and stakeholders are geographically dispersed. Putting information on a project website or other Internet or intranet network affords the benefits of immediate information availability, rapid and easy communication between workers, and information that is current because it is communicated in real time.

With web-browser integrated project management software, team members can report progress and retrieve assignments through their own individual web pages. The manager can aggregate information received from scattered worksites to get an overview of the entire project.

In most cases, the necessary tools are already at hand. Web-enabled project software requires just one thing: access to a web browser. Most everyone uses the Internet, so team members readily adapt to web-based methods for sending and accessing project information. The costs for overhead, update, and maintenance of web-based communication are very low.

Intranets and Group Productivity[4]

An *intranet* is a private computer network that uses Internet standards and protocols to allow communication among people within an organization. It provides access to a common pool of information from computers within an organization. The intranet is accessible only by organizational members and other authorized parties, though access can be extended to trusted external organizations, partners, or clients through an extended network called an *extranet*. With an intranet it is easy for users to access *group productivity software* and to store reports, profiles, calendars, and schedules. It is also easy to locate information in these documents using special *document-sharing tools* such as file hosting services, newsgroups, chat rooms, and electronic white boards. These tools are especially useful for sharing pictorial information about product design requirements and descriptions.

One of the most common ways that project managers use intranets is for collecting information about time spent on projects. The information is retained in a project database and then processed by project management software to report and tally time spent and time still needed to complete the project.

In the past, video- or audio-conferencing were the only ways for geographically dispersed teams to hold meetings, but today video, voice, and data can be shared over the intranet or Internet at desktop locations. The information shared can be in the form of a spreadsheet, text document, presentation, graphic, photo, engineering schematic, video file, or live streaming. Other ways for participants to collectively share project information and add comments and view others' contributions are online *discussion forums* and *chat rooms*.

At Boeing, for example, all designs—which are stored electronically and kept current to reflect the most recent changes—are available immediately to anyone who needs them. Notification of any change is sent via email to everyone who needs to know, as specified on a responsibility matrix (persons with "N" responsibility). As long as team members have access to a computer and browser, they can participate in meetings. Engineers in Kansas having trouble assembling a mockup can send video images to designers in Seattle who can *see* the mockup, assess the problem, and offer suggestions; absent that technology, and designers would have to *go* to Kansas. Managing technology-enabled meetings and virtual teams is discussed in Chapter 16.

See Chapter 16

A few words about email. Any experienced project manager will say it is an important communication tool but will advise you that it is no substitute for face-to-face or phone meetings, especially when it comes to making decisions.

PMIS in the Project Life Cycle

As shown in Figure 12.2, a PMIS can assist the project manager throughout all phases of the project life cycle. The following example illustrates this use.

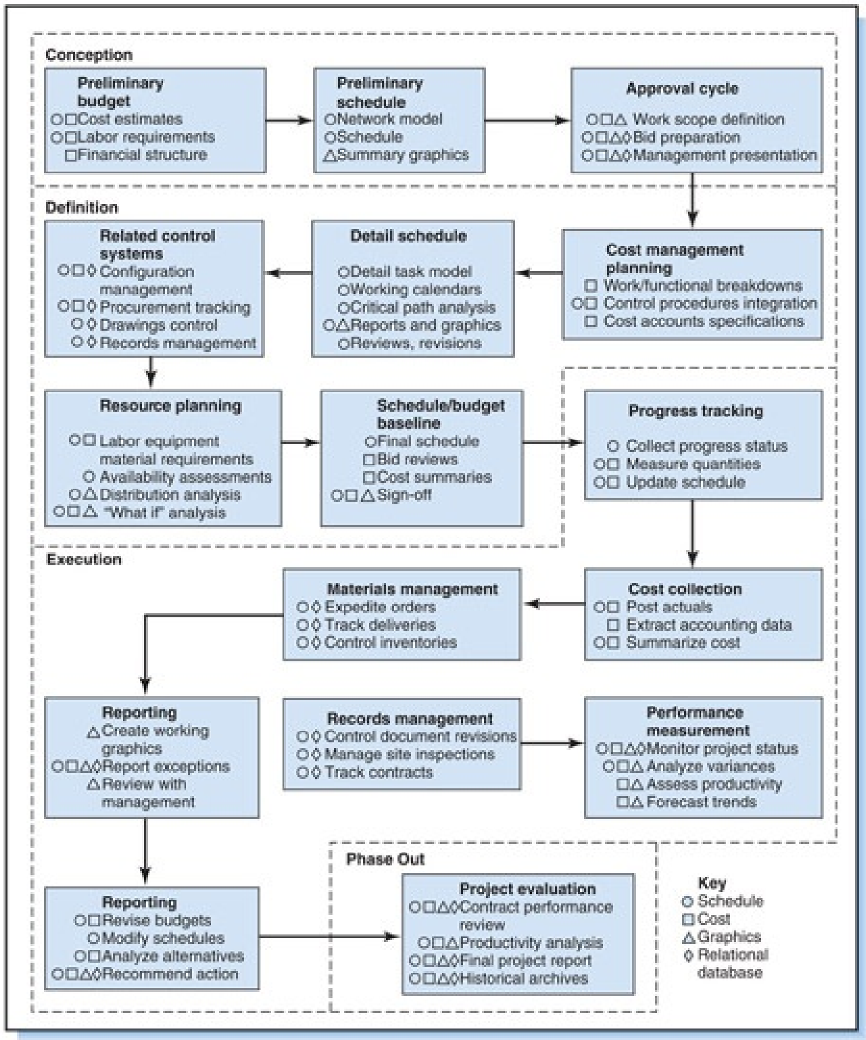


Figure 12.2 PMIS functions in the project life cycle.

Example 12.1: Sigma Associates' PMIS for Project Planning and Control

Sigma Associates, the architectural/engineering firm mentioned in Chapter 8, Example 8.7, uses a PMIS for project planning and control functions. So ubiquitous is Sigma's PMIS in its operations that employees think of it as a member of the team; they call it "Sally."

See Chapter 8

Once a project is approved, Sally's function is to routinely compare the original or current baseline project plan with actual performance, raise warnings about discrepancies, and forecast schedule and cost outcomes.

Each week, Sally receives information about current costs and accumulates estimates of weekly time spent on each activity from all project participants. Non-labor expenses and client reimbursements are input through the company's general ledger system.

Biweekly, the project manager estimates the hours needed to complete each activity, which Sally converts into a percentage completed. The system multiplies budgeted labor hours by these percentages to determine the estimated labor hours needed to bring the activity to its current level of completion (a form of earned value). By comparing this estimate with actual labor expenditures from time cards, the project manager can determine whether the activity is moving at its budgeted pace. Sally makes actual-to-plan comparisons and reports discrepancies, which managers use to spot problems. Whenever a project manager fails to provide the biweekly estimated hours, he receives a prompt about it from Sally.

Sally uses the anticipated hours-to-complete to prepare estimates of labor requirement loads for the remainder of the project. These estimates are used to adjust the remaining labor loadings and to make necessary revisions to schedules.

The comptroller uses Sally to forecast the timing and amounts of client billing, and the timing of expected payments according to each client's payment history. Based on the percentage of work completed, the system computes an estimate of earned client fees and compares them to actual project expenses in a monthly profit/loss analysis. Sally also generates monthly reports of net profit summarized by office, department, and project manager. It also combines net profit for all projects to give a picture of the company's financial health.

Sally likewise checks the correctness of the hours charged on time cards by comparing hours with dates on the schedule, and withholds any card with discrepancies and sends a memo to the employee. Each week it sends a summary report of rejected cards to the comptroller.

Fitting the PMIS to the Project

Most PM information software is no match for the capabilities of Sally, but that's okay since such capabilities are not always required. The purpose of a PMIS, in the words of Palla, is to "get the right information to the right person at the right time so the right decision can be made for the project."[5] Any PMIS able to do this is the right one. Firms often use more than one kind of PM information software—say, Microsoft Project for smaller projects and Primavera for large ones.

While project management software is essential for efficiently handling the computational aspects of project management, its role should be seen in context since computer systems are of limited value regarding numerous aspects of project management—identifying and negotiating with key stakeholders, choosing key subcontractors, motivating the team, and resolving interpersonal conflicts, to name a few. Yet many a novice project manager attends a 1-day software seminar and gains the impression that project management consists of little more than creating Gantt charts on a computer(!)

12.4 Informal Communication

Much of the communication in projects happens informally through the *grapevine*. Such communication is neither thorough nor dependable, garbles the message, and does not guarantee that people who need information will ever get it. Despite drawbacks, however, such informal communication is largely beneficial and essential. It fulfills social and work needs and conveys information more quickly and directly than most formal systems. Some theorists posit that a vast network of informal communication is essential for any organization to perform well.

While managers cannot control informal communication, they can influence it. One way is to *insist* on informality by removing status barriers and inspiring casual conversations between managers and workers. Some companies insist that everyone—from the president on down—wears a name-tag, goes by a first name basis, and that managers maintain an "open door" policy. The office's physical layout is instrumental too: removing walls and partitions, putting chairs and desks in "family groupings" for teams, and spot placement of lounges are ways to encourage informal communication.

Project management attempts to do what the informal organization sometimes does: to enable people involved in a problem or decision to directly communicate and make decisions.

12.5 Implementation Stage

The final stage of the project life cycle is implementation—the stage where 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 the clock is simple, you just plug it in and set it. If it is a digital alarm clock with a radio, you might need to first read the instructions. If it is a nuclear clock such as the one used by the US Bureau of Standards, you might need to attend a week-long training program. If the clock is to replace an existing clock connected to a timing device that controls lighting and heating in a large skyscraper, you will need to develop a *strategy* for substituting the clock so as to minimize disrupting and inconveniencing people in the building. There can be many issues associated with implementation, starting with user training and acceptance testing.

User Training

The purpose of user training is to inform 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 kind of materials needed (manuals, videos, simulators); personnel to be trained (existing or newly hired personnel); techniques to be used (classroom, independent study, role plays); 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 input afterward to improve the training. Often the user takes over training after the contractor's trainers have trained the user's trainers.

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 the user to understand the system within his environment and integrate it with existing systems. New systems create fear, stress, and anxiety; one aim of training should also be to relieve or eliminate these.

User Acceptance Testing

Among tests of the end-item performed before or during installation are the user acceptance tests. The results of these tests determine if the system can be adopted or installed as is, needs modifications or adjustments, or should be rejected.

User acceptance tests differ from tests conducted by the contractor during design and production, though the contractor tests should anticipate and be rigorous enough to exceed the user's test requirements. Nonetheless, the contractor should be prepared to make modifications pending the results of the user's tests.

Ideally users perform acceptance tests with minimal assistance from the project team. In cases where they cannot perform the tests, the project team must act as surrogate users and make every effort to test the system just as the user would, which means assuming the role of someone devoid of system-related technical expertise. Lack of user participation in these tests can to lead to later problems; therefore, even in the role of surrogate user-tester, the contractor must insist that the user be on hand to witness the tests.

System Installation and Conversion

The system installation and conversion stage is conducted according to the implementation plan. During this stage, 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 other, existing systems, so of major importance is the strategy to be used for replacing the old system with the new—the process of *conversion*. Three possible strategies, illustrated in Figure 12.3, 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 phased in as the old system is phased out.
- *Cold turkey (Big Bang)*: 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. For example, the first strategy 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 operated simultaneously and fully staffed. With the second strategy, the costs and risks are low and staff can be trained in stages. Drawback is, pilot operation is not necessarily representative of full system operation, and sometimes 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 most risky and raises issues about when the staff will be trained and what will happen if the new system fails.

Prior to installation, the project manager updates all plans and schedules, gains approvals for revisions, and renews commitments from the contractor and customer teams. Implementation is a high-stress stage for everyone, and the project manager and team must be patient with users and sensitive to their questions, concerns, and fears.

After 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.

12.6 Project Termination and Closeout

By the time the end-item has been delivered and installed, many members of the project team will be eager to move on to something new. Managers eagerly shift emphasis to upcoming projects and, as a result, might give the termination little attention. Yet, as common sense indicates, terminating a project is no less important than any other project activity. In fact, the method of termination can ultimately determine the project's success or failure.

At closeout the product or deliverable is handed over to the customer. Sometimes, contracts provide for a first handover at completion and a second handover after a *defects liability period* (aka retention period, guarantee period, or maintenance period). At first handover the customer should ensure that all *patent defects* (defects that can readily be detected by a qualified person) are identified and reported. After that the contractor is only liable for rectifying *latent defects,* which are those that could not be detected through a reasonable inspection at first handover. If, for instance, it wasn't raining at the time of the first handover, a roof that leaks later *when it does rain* would be considered a latent defect. The purpose of the second handover is to afford the customer more time to identify deviations from specifications or substandard workmanship. After second handover the contractor is no longer liable for defects; any *retention fees* withheld by the customer to ensure compliance are paid to the contractor.

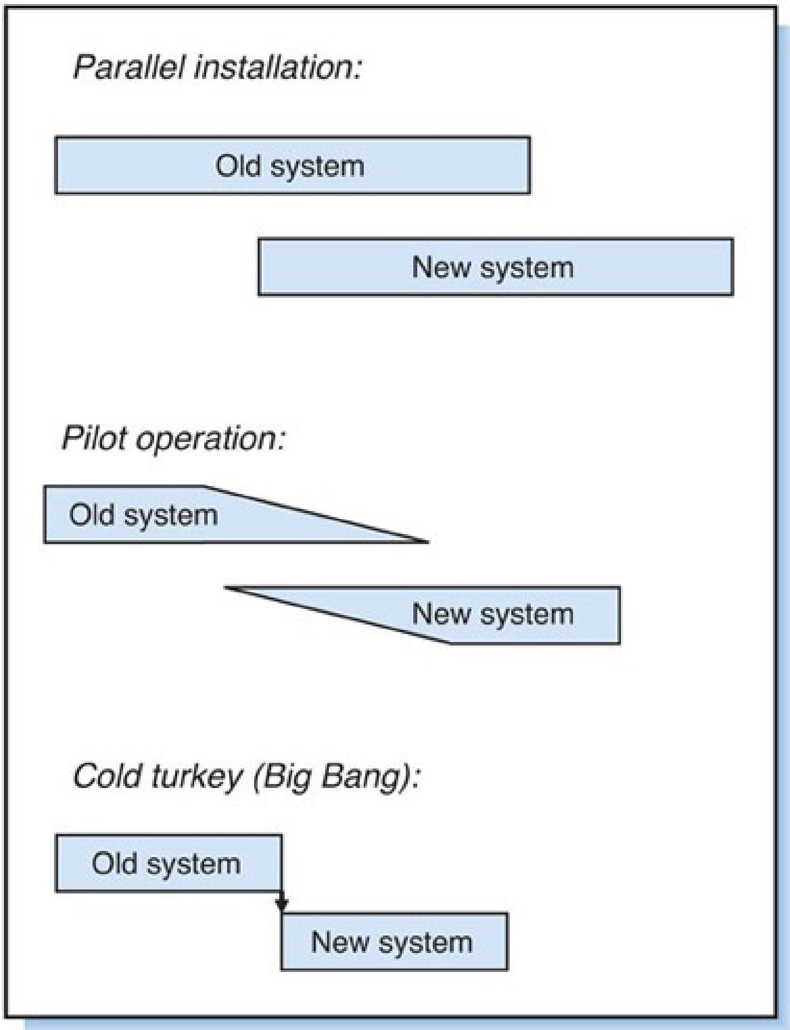


Figure 12.3 Three strategies for system conversion.

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 of resources by higher priority projects. A project can go sour simply by being allowed to "limp along" until it fizzles out. 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 after their work has been completed. Unless the project is officially terminated, work orders remain open and labor charges continue to accrue.

Kinds of Termination

The seeds of project success are sown early: since project 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 after that are approved by the contractor and customer.

There are many reasons why projects do not reach successful completion. A project may be aborted when the financial or other losses from early termination are considered less than the losses expected from completing the project. It might be a "white elephant" project with perceived low payoff or likelihood of success. The customer of the project end-item might simply change his mind and no longer want it.

Projects are also halted because of changing market conditions or technology, unsatisfactory technical performance, poor quality of materials or workmanship, violation of contract, or customer dissatisfaction with the contractor. Many of these reasons are the contractor's fault and could have been avoided had project management exercised better planning and control, respected the customer more, or acted in a more ethical manner. Such terminations leave the user requirements unmet and cast a pall over the contractor's technical competency and managerial ability.

Termination and Closeout Responsibilities

As with all other project work, the project manager is responsible for planning, scheduling, monitoring, and controlling termination and closeout activities. Archibald lists the following responsibilities:[6]

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, obligation, 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 that 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 contract administrator or others responsible for company–client negotiations and legal contracts. The final activity, obtaining the formal customer acknowledgment, may involve claims if the customer has failed to provide agreed-to data or support, or has requested items beyond contract specifications. In these cases the contractor is entitled 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 then checks off the items on the list as they are rectified.

Example 12.2: Punch List for the Chunnel[7]

Five months before the scheduled completion date of the Chunnel, the punch list still contained a lot of items—over 22,000. Incredibly, by the day before scheduled handover of the Chunnel to its owner/operator, that number had been whittled down to only 100. Problem was, the contract allowed for *no* (zero) items on the punch list; any open items at the handover 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 Chunnel 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 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 sign-off 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 Chunnel's ten prime contracting companies plus 1,000 other guests—was a minor project in itself.

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. The termination responsibilities listed before should be systematically delegated and checked off as completed. Termination requires the same degree of attention as do other project management responsibilities.

Closing the Contract

Delivery, installation, and user acceptance of the main contract end-item (hardware, software, or service) does not necessarily mean that the project is closed. Project completion can be delayed pending delivery of necessary, ancillary articles—*side-items*—or payment of compensation for failure to meet contractual agreements. This applies to not only the contract between the customer and the contractor but also to contracts between the contractor and subcontractors.

Side-Items

The installation, operation, maintenance, and monitoring of the contract end-item is often contingent upon availability of numerous contract side-items such as special tools, instruments, spare parts, reports, drawings, courses of instruction, and user operating and maintenance manuals. Side-items are usually provided by subcontractors and can range from the simple and mundane to the complex and innovative. The former is exemplified be an operating manual for a network server, the latter by a high-fidelity computer simulator for training operators of a large chemical processing facility. Simple or complex, successful completion of side-items is important to 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, however, perhaps because they are deemed “side” items, the time and effort required to develop and produce them is often underestimated. The result is a delay in implementing the end-item and closing the project.

Side-items should be included in all aspects of project planning and control. The project manager must make certain that the scope of side-item work is well understood, and qualified personnel are assigned with time to fulfill their requirements.[8] Side-items are part of the contracted work, not afterthoughts or project extensions. They must be given full consideration in the WBS, project schedule, and budget.

Handover and Negotiated Adjustments

In many projects, the contractor receives payment for only a portion of the total project cost, say 80 to 90 percent, and the remainder is contingent upon the performance of the end-item, the contractor's compliance with contractual agreements, or the quality of the working relationship with the contractor.[9]

These final payment contingencies are considered *post-acceptance* issues because they occur after the customer has accepted the major end-item. If the delivered end-item is satisfactory yet does not perform up to the contracted specifications, if it is found defective after a trial period due to design or production inadequacies, or if it is delivered late, the contractor might have to pay a negotiated compensation to the customer.

Contract sign-off might also be contingent upon how well the product functions after installation or delivery, and the contractor might be obligated to provide on-site user support, at no additional fee, to remove any operating 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 the above discussion applies to contracts between the contractor and subcon-tractors as well as the contract between the customer and the contractor.

12.7 Project Summary Evaluation

Among the final activities of the project team after project closeout is to perform a formal evaluation. This final *summary evaluation* gives project and company management the opportunity to learn from its successes and mistakes in the project. 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?") The project summary evaluation reviews and assesses the performance of the project team and the end-item system. Its purpose is to identify and assess what was done and what remains to be done—not to find fault or pass blame. 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* (perversely also called a *postmortem*) is a summary review and assessment of the *project* conducted by the contractor immediately after project closeout—early enough so project team members are still around, available to participate, and remember what happened.[10] It is an important task for which funds and time should be included in the project's budget and schedule. Post-completion reviews are one way companies try to continuously improve future projects through lessons learned from past projects—an opportunity that many companies forego.

The post-completion project review should review:

1. Initial project objectives in terms of technical performance, schedule, and cost; and the soundness of objectives given the needs and problem the system should have resolved.
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 life cycle, including the effectiveness of project management; relationships

among top management, the project team, the functional organization, and the customer; and customer reactions and satisfaction.
4. The involvement and performance of all stakeholders, including subcontractors and vendors, the client, and outside support groups.
5. Expenditures, sources of costs, and profitability.
6. Areas of the project where performance was particularly good, noting reasons for success and identifying processes that worked 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-day 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 note things that went right and wrong with the project; they then share their notes and create lists of lessons learned and recommendations for future projects. The completed lists are then formally presented to stakeholders, others on the project team, and to project, functional, and senior managers.

The review seeks to determine lessons that may be applied to future projects, not to criticize or place blame. Its results are documented in a *project summary report,* which becomes the authoritative document on the project. The summary report describes the project, its evolution, and the outcome. It describes the project plan, where it worked and where it failed. Because projects affect different parties in different ways, any opinions of the customer, the 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. Post-completion reviews and summaries are ways to capture and reapply knowledge to future projects—tools for project *knowledge management,* discussed in Chapter 17.

See Chapter 17

Example 12.3: Microsoft Postmortems[11]

Product development projects at Microsoft often conclude with a written postmortem report that is circulated to team members, senior executives, and 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 require as much as 6 months to prepare and can range from under 10 pages to over 100 pages in length. 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.* Some months after its delivery, the *end-item* or *system* should also be evaluated to assess its performance in the user environment and under ongoing operational conditions. This *post-installation system review* focuses on the end-item system and serves a variety of purposes, such as providing operation and maintenance information for the system's designers, and revealing possible, needed enhancements for the system's users. 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? Is the user getting the expected benefits from it? 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 system modifications and improvements after installation; although there is nothing wrong with this per se, the system is physically or functionally changed from the one delivered or installed, a fact that must be considered when evaluating its performance.

During the course of 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 only after the system has been in routine operation.

Results of the review are summarized in a report that describes the system's performance compared to its objectives, 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.

12.8 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 for which there is nothing physical to operate (e.g. a rock concert, company relocation, corporate merger, or audit). In the former case (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) by agreeing to maintain/repair the system, or (2) by initiating a new project to enhance or replace the system.

System Evaluation and Maintenance

The contractor may perform evaluation of the system either as part of the original contract agreement or by an additional agreement. The evaluation may occur as the last scheduled activity of the contractor in the form of a *post-installation review*, described earlier, or as an extended agreement to provide periodic review and/or service to the system on a continuing basis. The agreement can be a warranty arrangement whereby the contractor reviews and maintains the system for a pre-specified time period as part of the original contract, or it can be an "extended" arrangement that continues the contractor's involvement for a longer time period. The contractor may assign *system representatives* and technicians to the user site to perform preventive maintenance and system upgrades on a scheduled basis or as requested by the user.

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.

There are two kinds of extensions: discretionary and essential. *Discretionary extensions* are requested by the customer or proposed by the contractor for the purpose of improving the operation, performance, or convenience of the original project end-item. The environment remains the same, but new and better ways now have appeared that can improve the system. The other kind, *essential extensions,* are compulsory; without them the system will cease to operate or become obsolete. An end-item that is no longer adequate because of changes in the environment or design deficiencies *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 is initiated with a request from the user (e.g. an RFP) or proposal from the contractor. The extension itself becomes a new project. Humankind engages in few dead-end projects; each spurs others, and the systems development cycle keeps rolling along—hence the term "cycle."

12.9 Summary

Project evaluation relies on a variety of sources and measures for collecting and communicating evaluation information, including written and oral reports, observations, and review meetings. Site visits and one-to-one conversations are the best sources, as are informal reviews and formal review meetings. Informal reviews are held regularly and conducted by peer members of the project team. Formal reviews are special reviews or audits held at key stages or milestones in the project and conducted by experienced outsiders. They provide independent assessments of overall project performance, suggestions or instructions for improving the project, and sometimes a decision about whether or not to continue the project. The kinds of reports and reviews, and details about contents, formats, schedules, participants, and points of contact are specified in the project communication plan.

In many projects, PMIS software packages perform 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.

Implementation, the final stage of the project life cycle, is when the end-item system or other deliverable is completed and turned over to the user. Among important tasks during implementation are user training, user tests of acceptance, and system installation and conversion. The contractor trains the user to operate, maintain, and service the system, and develops a strategy for installing the system in the user's environment; three possible strategies are parallel, pilot, and cold turkey. The user performs his own set of tests to determine whether or not the installed end-item system is acceptable.

The project is terminated through a series of formal procedures overseen by the project manager. Following project completion, a post-completion project review (postmortem) is conducted to assess the effectiveness of the project organization. Additionally, after the end-item system has been in operation, a post-installation system review is conducted to assess its performance and determine possible maintenance or enhancement needs. The documented results are combined in a summary report to provide a reference document for future project teams.

Review Questions

1. Describe the difference between formative evaluation and summary evaluation in project management.
2. Why is it better to rely on a variety of information sources for evaluation rather than just a few? Give some examples of how several sources are used in project evaluation.
3. What are the advantages and disadvantages of the following sources of information: (a) charts and tables, (b) oral and written reports, (c) firsthand evaluation?
4. What is the purpose of internal peer reviews? When are they held? Who participates?
5. What is a formal critical review? When is a formal review held and what does it look at? Why do outsiders conduct it? Why would a customer or project supporter want a formal review?
6. What should be included in status reports to top management?
7. What reports should the project manager receive? How does the project manager use these reports?
8. What reports are sent to functional managers?
9. When and what kind of reports are sent to the customer? Why is reporting to customers so important?
10. What is the role of the PMIS in project management?
11. Discuss the applications and benefits of web-based project management.
12. Discuss the uses of the PMIS throughout the phases of the project life cycle.
13. How is the system implemented? Describe the important considerations for turning the system over to the user.
14. Discuss user training and why it is sometimes included in the implementation stage.
15. How is the project end-item tested and checked out for approval?
16. Describe the different strategies for installing or converting over to the new system.

17. What are the reasons for project termination? How can termination for reasons other than achievement of project goals be avoided?
18. What is involved in planning and scheduling the project termination?
19. What is the role of the project manager and contract administrator in receiving customer acceptance of the work and final payment?
20. What are side-items? Give examples not used in this book. How can they delay project completion?
21. 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?
22. What is a punch list?
23. What are project extensions and how do they originate? How is a project extension managed?
24. What are the differences between the two kinds of project summary reviews: the post-completion project review (or postmortem) and the post-installation system review? Describe each.
25. Describe what happens during the operation phase. What role does the systems development organization (contractor) play in this phase?

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 top management and the customer. Who issued these reports? What kinds of reports were sent to project and functional managers? Who issued them?
6. Describe the PMIS used in the project. Was it the same one used for cost-accounting (PCAS) and project scheduling? Did it combine scheduling, budgeting, authorization, and control, or were several different systems used? If several systems were used, how were they integrated?
7. What were the strong and weak points of the PMIS system? Did the system adequately meet the information requirements to plan and control the project? Were any inadequacies the fault of the computer PMIS or the manual support system that provided inputs and utilized the outputs? Was a web-based system used? What improvements would you suggest to the system?
8. Did the project manager encourage open, informal communication? If so, in what way?
9. 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.
10. 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.
11. How was the contract closed out? Were there any side-items or negotiated adjustments to the contract?
12. Did any follow-up projects grow out of the project being investigated?

13. Describe the post-project summary review (report prepared at the *end* of the project). Who prepared it? Who was it sent to? How was it used? Where is it now? Show an example (or portion of one).
14. 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?
15. What happened to the project team after the project was completed?
16. Did the contractor remain involved with the customer and end-item through an extended agreement?

Case 12.1 Status Report for the LOGON Project

Assum LOGON project began as scheduled in May 2020. 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; 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 for MPD's top management and should address issues of most importance to them: deliverables and other requirements, schedule, and budget, as noted in Appendix C. The report should also note any problems encountered to date, anticipated challenges, and recommended suggestions or changes to the plan.

See Appendices A, B and C

Case 12.2 SLU Information Central Building

Construction of the new Information Central building at South Land University (SLU) is completed on time and on budget. Administrators at SLU and managers at Finley Construction Company, the building's prime contractor, are very 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 of the technical requirements. Much of the technology is leading edge, and some of it is being applied for the first time ever 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 of project termination and payment. The meeting goes well and ends with discussion of future projects at SLU and possible involvement of Finley. 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 just 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 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 way the 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 taken after the project was completed?

Case 12.3 Formal and Informal Communication

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 it has 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 will in the future supply HeavyEng with electronic copies of the drawings instead of the “hard” copies they have 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?
5. What is a good general rule regarding formal and informal communication for any project?

Endnotes

1. 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.

2. Archibald R. *Managing High-Technology Programs and Projects.* New York: John Wiley & Sons, 1976. p. 191.

3. Portions of this section were prepared with the assistance of Elisa Denney.

4. Palla R. Introduction to micro-computer software tools for project management, *Project Management Journal.* August 1987: 61–68.

5. Ibid.

6. See Archibald R. *Managing High-Technology Programs and Projects*, pp. 235–236 and 264–270, for a complete checklist of closeout activities.

7. Fetherston D. *Chunnel.* New York: Time Books; 1997, pp. 372–375.

8. 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.

9. Ibid, pp. 241–242.

10. 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.

11. Cusumano M. and Selby R. *Microsoft Secrets.* New York: Free Press; 1995, pp. 331–334.

Chapter 13
Agile Project Management and Lean

Some methods of project management as practiced in construction, infrastructure, and large-scale product development are ill-suited for projects where the solutions or end-item requirements are uncertain or subject to change. Such is the case for software development projects, and in 2001 a group of 17 developers published *The Agile Manifesto* report wherein they proposed the principles of a form of management better-suited to software development, namely:

> 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]

For each principle on the list, the implication is that things on the left (e.g. individuals and interactions) should be emphasized *over* things on the right (processes and tools).

Innovative practices in project management arise constantly, and the *Manifesto* endorsed some that, until then, had been applied only on a limited basis. Accompanying the Manifesto was the rise of "agile software development," which was the precursor of *agile project management* (APM) and particular practices such as Scrum, which that are described in this chapter. APM is a form of project management *methodology*, which is covered more broadly in Chapter 17.

See Chapter 17

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* had been inspired by Toyota's lean principles—continuous improvement, elimination of waste, and respect for people[2]—so it is no coincidence that APM and lean somewhat overlap. But, as this chapter discusses, lean applications extend beyond software projects and have taken hold elsewhere, especially in product development projects.[3]

13.1 Traditional Project Management

All projects can be conceptualized in terms of the life cycle phases of Conception, Definition, and Execution, although exactly what happens in each phase depends on the kind of project and, in particular, on how clear, complete, or well-understood are the problem/need, goals, and solution/end-item of the project. When these are well-understood, the project moves readily through the phases: goals and the solution/end-item are established in Conception; requirements and project plan are drawn up in Definition; and work toward the goals/requirements is undertaken in Execution. But in many projects the solution/end-item is not certain, and neither are the requirements. Such projects cannot "readily" move through the phases, and to manage them requires deviating from the methods of traditional project management as described in earlier chapters.

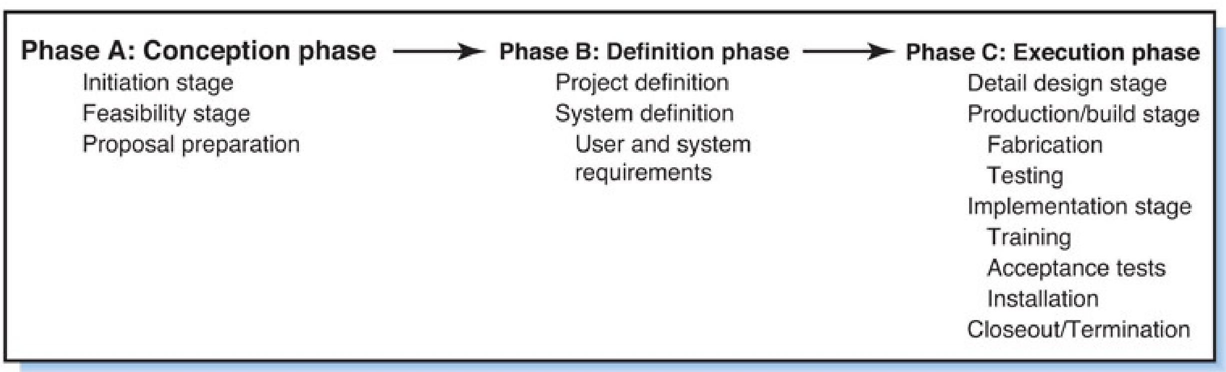


Figure 13.1 TPM methodology: applies when the problem/goals and solution/end-item can be well-defined.

The traditional project management methodology, or TPM—the so-called *waterfall methodology*—applies to projects where the problem/goal and the solution/end-item are well-understood and can be defined early in the project. The phases of definition and execution each happen just once: the solution/end-item requirements and project plan are clearly laid out in definition, and the work is executed in compliance with the plan in execution (Figure 13.1). TPM projects are plan-driven projects; execution means, simply, "execute the plan."

Incremental TPM

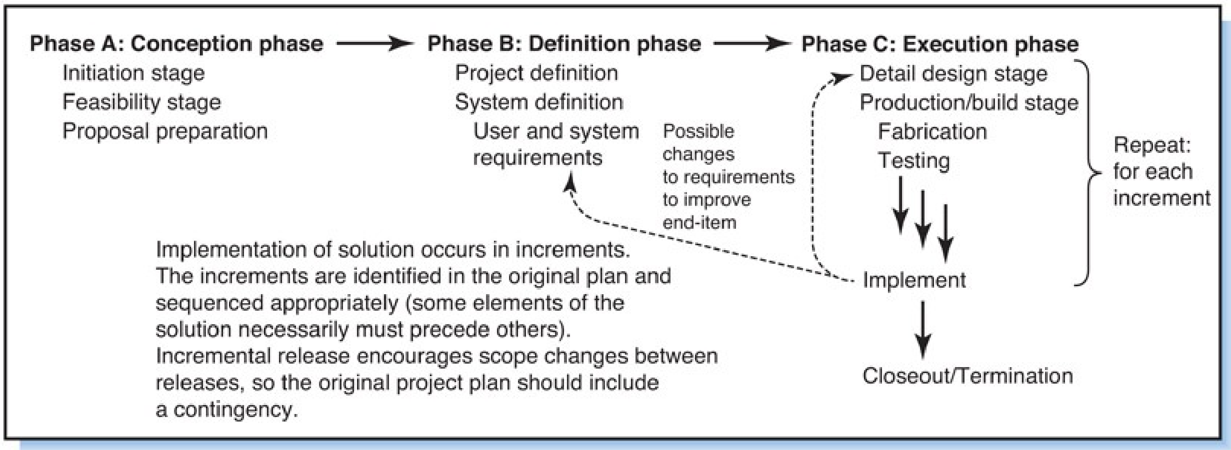


Figure 13.2 Incremental TPM: applies when problem/goals and solution/end-item are well-defined and the end-item is to be built/launched in increments.

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 13.2). The lower floors of a building, for example, are finished 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 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 needed changes or improvements. But incremental TPM is still TPM: it assumes that the solution 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 in a series of steps, say, 6 weeks or 6 months apart.

Inappropriate Use of TPM

When the solution cannot be clearly identified or is uncertain (e.g. development of a revolutionary new technology or product), then the end-item requirements cannot be fully defined and the project cannot be fully planned. Instead, both requirements and plan will have to be developed over time, as things become better understood. The consequence of this is that the requirements and plan will expand and change, perhaps often. The change control process described in Chapter 11 is not well-suited for this kind of continuous change and tends to extend project duration, increase costs, and, often, yield mediocre or poor quality results.

See Chapter 11

Inability to define a problem or solution should not be confused with ignorance. Maybe little is known, but usually *something* is known, and that can be enough to start a project. Unlike TPM wherein much of the plan is completed early in the project (or, for long projects, planned at a high level with details being added later), an alternative is to *acknowledge* that relatively little is known and to use the project as the vehicle *to learn more.* Rather than implementing a known solution, the focus of the project is on *learning what the solution should be*; rather than moving through the definition and execution phases once, the project cycles through them iteratively in a learn-as-you-go fashion. In short, all projects start with initiation and end with closeout, but whether the phases/stages in between happen once or repeatedly depends on the certainty of what the solution/end-item should be.

13.2 Agile Project Management, APM

Project management methodologies that accommodate uncertainty in an iterative, learn-as-you-go manner are referred to as Agile Project Management, APM. In APM projects, the customer/developer does not know or is unclear about the desired solution and cannot define the end-item requirements upfront. Thus, the main purpose of the definition phase in APM is to identify, as best as possible, the customer's wants and needs, and create a *high-level* plan to address those needs. The actual requirements and features of the end-item will be developed later in a series of iterations. The high-level plan specifies the anticipated number, length, and objectives of the iterations.

Learning from Iterations

The iterations occur in the execution phase (Figure 13.3). Each iteration leads to discoveries and better-defined requirements; each concludes with a release or end-item "increment"—a partial solution that addresses the customer's needs or problem, but in a limited fashion. Planning happens on a just-in-time basis: at each iteration, the customer and team assess progress made so far and plan for upcoming iterations. Thus, at each iteration includes not only the typical execution stages of design–build–test, but also aspects of the definition stage—requirements definition and project planning. The emphasis in planning and execution is on creativity and delivering a solution of value. Budgets and schedules are of less or little importance.

To facilitate learning and discovery, the iterations tend to be short, in some cases just 2–4 weeks each, in other cases several months. The result of each increment is assessed: Does it provide the expected benefits? Is it useful and usable? What does it lack or what is wrong with it? This enables the developer to improve subsequent releases so that by project closeout the cumulative increments will have fully or substantially satisfied the customer's needs and wants.

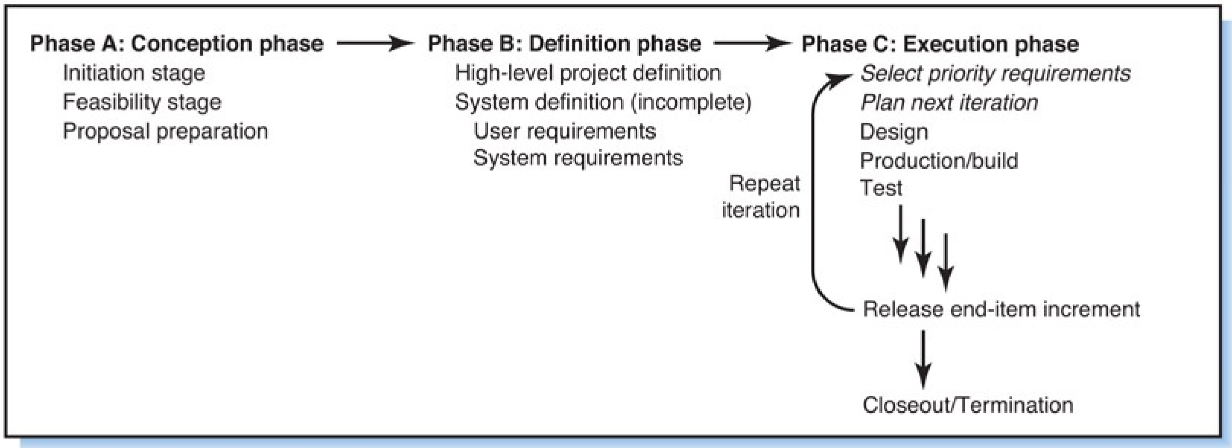


Figure 13.3 APM: applies when problem/goals are well-defined, but solution is poorly defined.

APM Variants[4]

APM comes in several forms and the one to use depends on what is known about the end-item/solution. For example, do we know the range of different, possible solutions available 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—the solution is known 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 features and functions into the solution/end-item. The project might have no deadline and continue until either the customer is satisfied or the budget runs out. The customer is involved throughout the project, assessing changes/additions to the solution, and giving feedback to the development team. The spiral model is an example.

Example 13.1: The Spiral Model

Using the spiral model, the end-item or product is released in a series of multiple 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 initial cycle delivers, optimistically, a product prototype, which the customer uses and assesses. The process repeats, with each cycle delivering an improved prototype version based upon customer feedback from earlier versions (Figure 13.4). After several cycles—either a predetermined number or whenever the customer is fully satisfactory—the "production" or operational version of the end-item is released.

The spiral model is most commonly applied to successively improve a prototype leading to a final product, but it can also be applied to successively improve a final product. An example is a software product or software portion of hardware where every few months or annually an upgraded or enhanced version of the product is released. This "outward spiral" of product development (Figure 13.4) continues for as long as the product remains "improvable" and viable in the market.

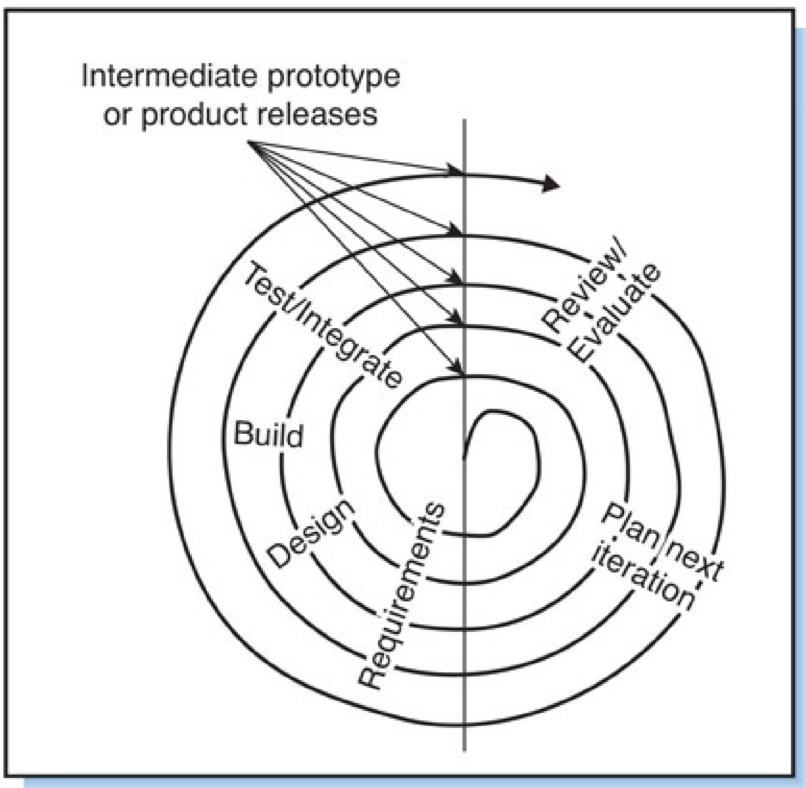


Figure 13.4 Spiral development methodology.

There is a big difference between APM (including the spiral model) and incremental TPM. With incremental TPM, which applies to all kinds of end-items, including one-of-a kind (space station, aircraft carrier, skyscraper, highway), the end-item is not expected to "evolve." Although portions or elements of the end-item might evolve, the overall end-item does not; its fundamental structure/composition is fixed at the start of the project. Furthermore, the overall end-item/solution and the elements to be added incrementally are largely known and defined early in the project.

With APM, details of the solution and end-item system *evolve and expand* with each iteration—which is of necessity since much of 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 aspect of the solution and builds on knowledge from the releases of previous iterations. The process iteratively *converges* on a complete solution.

Modern products are a combination of hardware and software. Usually the hardware must be developed using the TPM approach, although the software might more-appropriately use an APM approach. A challenge in managing hardware–software development projects is to coordinate the iterations so that the needed software is available to integrate with hardware that is being produced in traditional waterfall fashion.[6]

Typically the duration of each iteration in APM corresponds to how much is known about the solution: the less known, the shorter the duration. When the solution is somewhat well-known, iterations can each last several months; this is common in spiral-model projects. When it is less well-known, they must be shorter; in. In the popular Scrum method of APM, each iteration lasts just weeks.

13.3 Scrum[7]

The Scrum process is shown in Figure 13.5. The 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. But initially the 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 stories that describe how a user might use or benefit from the product.

Just before each iteration—called a "sprint"—the development team and product owner review the product backlog and select from it a few items on which to focus the next sprint; these items constitute the "sprint backlog." The team agrees on the functionality it must build to address those items and that will fit within the sprint deadline or "time-box" of, 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); it also reviews its work performance to learn lessons for subsequent sprints.

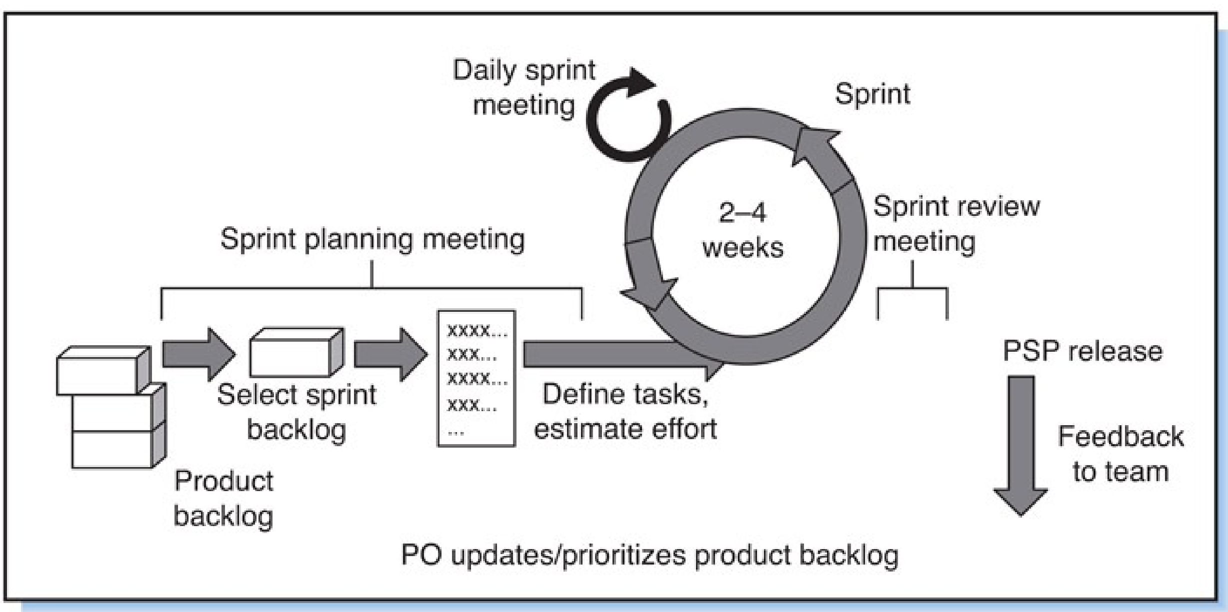


Figure 13.5 Scrum process.

Source: Adapted from Schwaber K. and Beedle M. *Agile Software Development with Scrum*. Upper Saddle River, NJ: Prentice Hall; 2002.

Scrum Roles[8]

In Scrum, management responsibilities are shared among other roles—scrum master (SM), product owner (PO), and project team. There is no project manager, so a "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 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 and know how to get needed resources and remove barriers.

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 the backlog with the team.

The PO represents the "customer" and therefore must thoroughly understand the customers and users, the business, or the market. She must be good at communicating with and listening to customers, users, and the team. In Scrum, the PO is *the* decision maker regarding product matters and product-related issues facing the team, and thus must have decision-making authority in the organization.

The SM and the PO should not be the same person. 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 of the team.

Project Team

The team comprises whatever the project requires—analysts, programmers, testers, etc. Members work primarily in their specialties but contribute to all other specialties. Work assignments are tentative because members are expected to provide assistance wherever it is needed. If a member finishes her task early, she is expected to start a new one or help others too, regardless of specialty. Says Cohen, in Scrum there is no "my work" and "your work"; there is only "our work."[9]

Through rotating and sharing of roles (which similarly occurs in concurrent engineering, discussed in Chapters 4 and 14), members develop an understanding of every task in the project. Everyone is able to do and does a little of everything. Since everyone participates in all tasks, they know what has been and still needs to be done. This reduces the need for formal documentation and work "hand-offs" (down-time from waiting on others).

See Chapter 4 and Chapter 14

Team Structure

Scrum teams are small, typically five to ten people; Amazon calls them "two-pizza" teams, which affords advantages: they are better able to 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.

A Scrum team is self-organizing and self-managing. It decides itself how it 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.

Sprints are fast-paced and emphasize rapid learning, so real-time communication among team members is essential. Consequently, team members usually work full-time on a project and in the same location. But 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 point 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, 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 on satisfying 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 eventually be broken down into simpler user stories that can each be handled in a sprint.

The stories and features 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, and reprioritizing items.

In Scrum, the emphasis in system definition shifts from documentation to discussion. Prior to each sprint the team discusses stories in the backlog with the PO and chooses the ones it best understands and wants to attack in the next sprint; these form the *sprint backlog*.

For each story selected, the team and PO define "conditions of satisfaction" (COS), which state what the user expects and doesn't expect. The COS and user stories constitute the sole "requirements documentation" for a sprint. When the product must 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 "minimum viable product" or a "potentially shippable product"—PSP. 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) and is a stand-alone working product. With each PSP the customer gets a usable product and the team gets feedback about what 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 benefits from the PSPs delivered so far.

Within each sprint all the stages to produce a PSP are performed (analysis-code-test-checkout), however they are *overlapped* and done in "chunks"—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 development—analysis, coding, testing, etc.; this is why people on the team must be capable of doing anything whenever it is needed.

As mentioned, the sprint duration is fixed or timeboxed, usually 2–4 weeks, based on whatever time the team wants. Timeboxing simplifies planning and estimating since the team learns how much it can do each sprint (work rhythm) and can monitor its output rate. Knowing its work rhythm, the team selects user stories for each sprint backlog that it thinks it is capable of achieving during the timebox. The timebox duration is held constant throughout the project; it is strictly reinforced and rarely is overtime allowed. A typical project involves many sprints, and teams are more effective when they work at a uniform, sustainable work rhythm. If the team cannot finish everything on the sprint backlog, the PO decides which items to drop—and to be picked up in later sprints.

Once a sprint is underway, the PO steps out of the way but is always available whenever the team requests.

Sprint Increments versus Production Versions

Each sprint produces a new end-item increment or PSP, but rarely are customers able to absorb such frequent changes into their operational or production environments. Most likely the customer will be able to implement a revised "production" or user-operated version only a few times a year—although that version will incorporate the cumulative discoveries and improvements from all 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 customer and key users reviews and critiques the released increments or PSPs for each sprint. This provides the 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.

Planning and Control

Scrum plans are developed progressively. A plan is prepared early in the project, but it contains only fundamentals of the solution/end-item and has few details. Any commitments specified allow broad latitude for change. The planning process is analogous to the rolling-wave, phased approach described in Chapter 4, the difference being that the "goal" and its associated requirements in agile projects are somewhat in motion and the path to reach them is less clear.

See Chapter 4

As the team's knowledge expands during the project, so do details about the plan. This progressive refinement of the plan has many benefits: it avoids planning for things that are unknown or uncertain; it defers decisions until there is adequate information; and it allows wiggle room to change direction. Planning, along with work review and control, happens in a series of meetings before, during, and after each sprint.

Sprint Planning Meeting

Prior to each sprint the team devotes 4–8 hours to preparing for the next few sprints. It selects from the product backlog user stories for the next sprint backlog. For each item on the sprint backlog, the team determines the specific tasks to be done and estimates the time; this enables the team to know that the work it has chosen is "doable" within the sprint timebox and, later, to track its progress. The team also selects stories for the following two sprints (e.g. at end of sprint 2, it plans for sprint 3 and creates backlogs for sprints 4 and 5).

Daily Sprint (aka Daily Scrum) Meeting

The team meets each morning during the sprint for 15 minutes to review progress and update the white board and burn down chart (Example 13.2 and 13.3 below). The meetings are no-nonsense and are held at the same time and location. 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

Scrum uses simple, visual tools for tracking and controlling work. The following are two examples.

Example 13.2: White Board[11]

Each job is written on a sticky note and placed on the "To do" section of a white board; *To-do* is the first of a hypothetical four-step process followed by *In-process* (being done), *Verify* (done correctly), and *Done* (Figure 13.6). Each job is one task that must be done on a user story on the sprint backlog, for example analysis, code, test, etc. Different kinds of tasks can be represented by different color notes. As jobs progress, the notes are moved from one section on the white board to the next. Suppose the development team had decided that no more than three jobs at a time can be in a section; i.e., when a section has three notes, no more are allowed. Restricting the jobs in each section like this, called Kanban, prevents the team from taking on additional tasks before it has completed existing tasks, and keeps work moving at a uniform pace—called *velocity* or *cycle time* (discussed later). Simply by scanning the white board, the team can see at any given time which jobs are holding up progress and need additional resources.

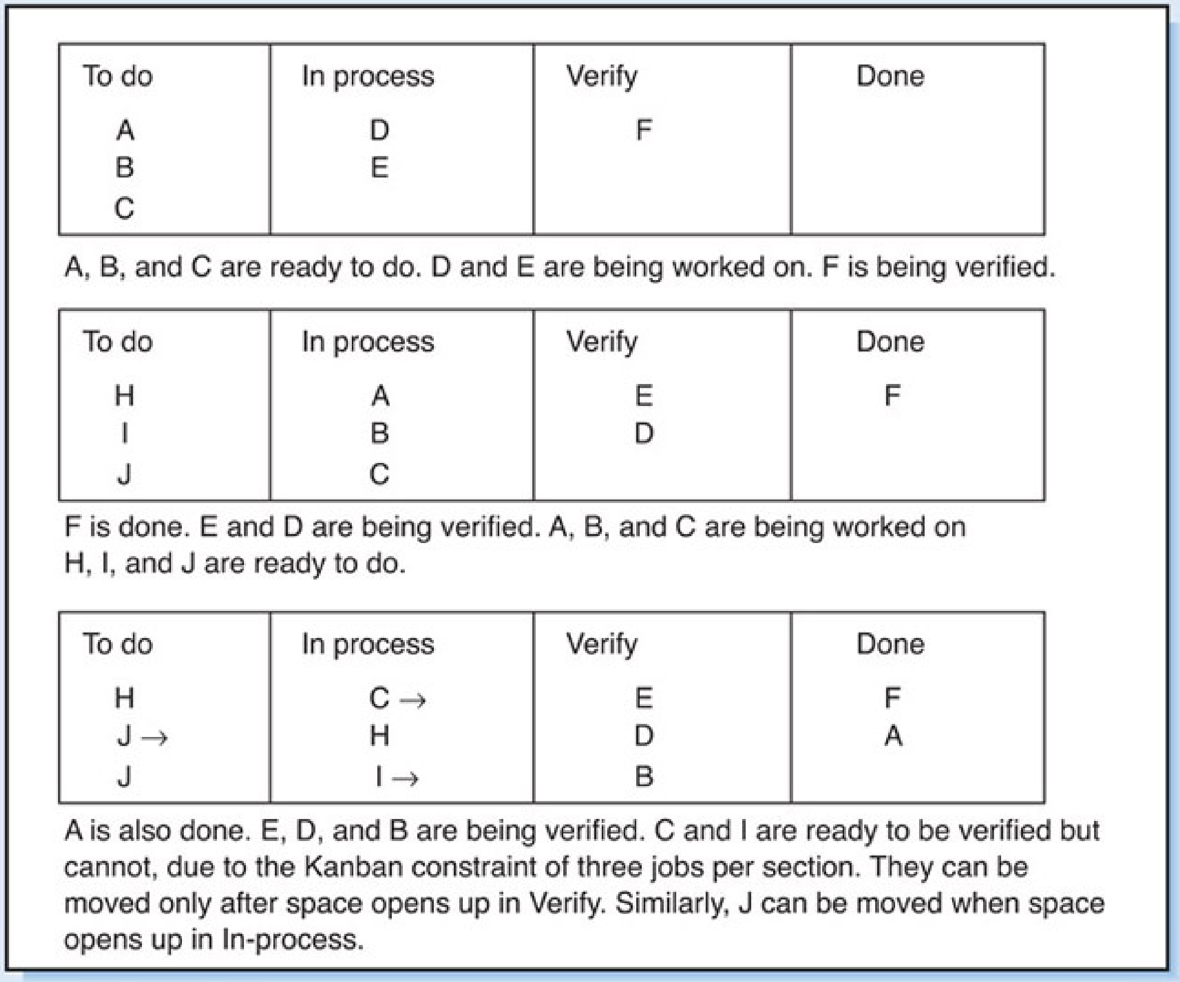


Figure 13.6 White board for tracking and controlling work tasks.

Example 13.3: Burn Down Chart[12]

Progress in each sprint is also 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 13.7). 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 required, the more points assigned); if hours of effort, 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 timebox—the hours available for 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 the user stories selected for the sprint backlog must be doable within that constraint. In fact, in selecting stories from the product backlog, the team keeps a running tally of the hours of effort involved, selecting no more stories than can be "fit" within the available labor hours of effort. Although the team will certainly work at least 8 hours a day, by using only 6 hours a day in the estimate it is being conservative and somewhat guaranteeing that it will be able to complete the sprint backlog.

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 each day the hours of effort remaining on the burn down chart should decrease by 36 hours (Figure 13.7).

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 13.8 this happened on Days 2, 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 the reverse of) the estimated line. In other words, at the end of the day, even though the team may have actually 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 13.8 this occurs on Day 5 and 7.

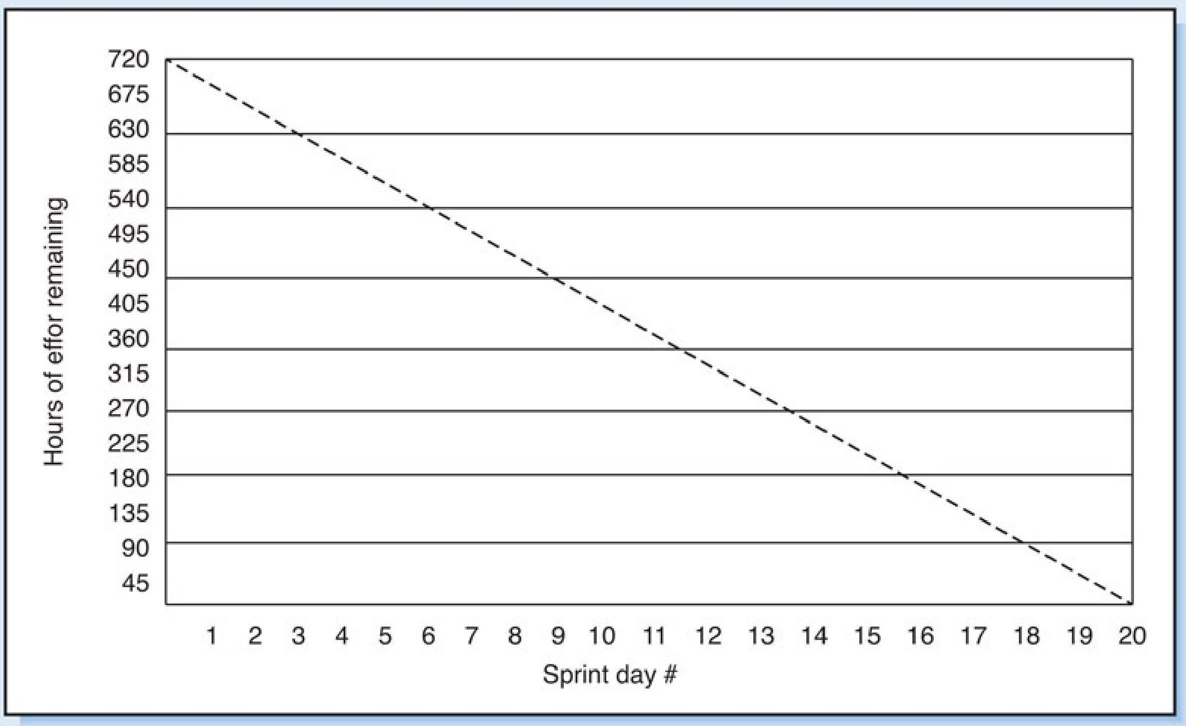


Figure 13.7 Burn down chart, estimated, expected performance.

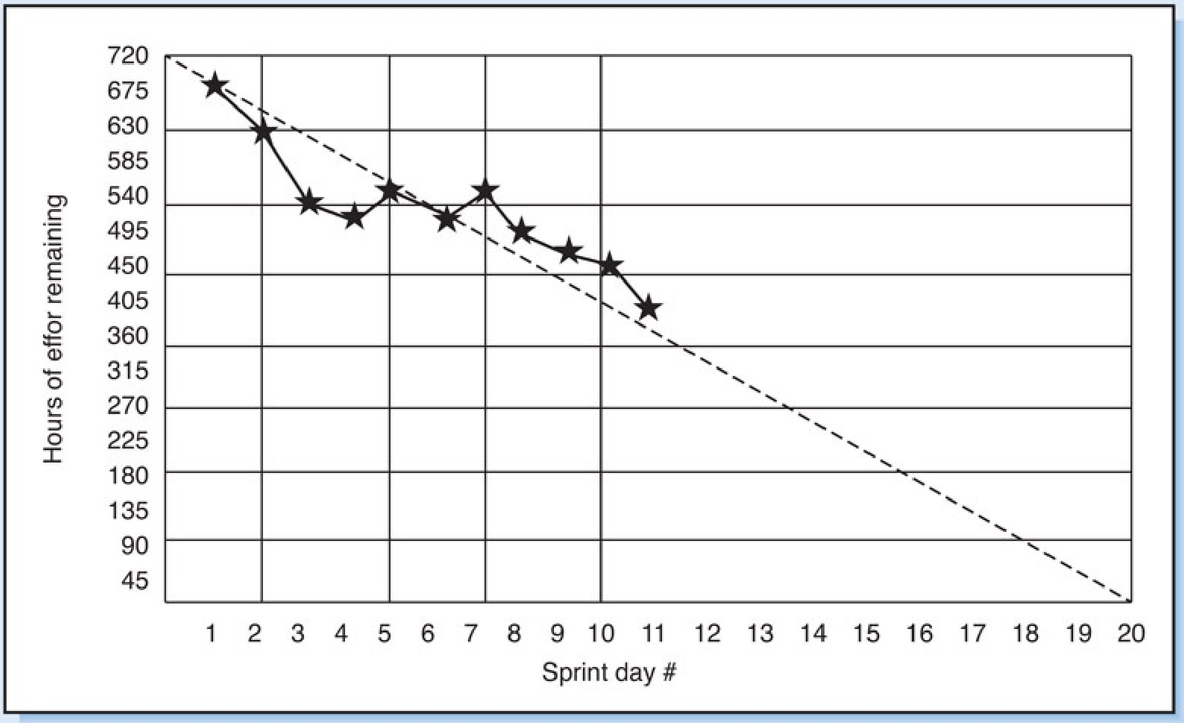


Figure 13.8 Burn down chart, actual performance for 11 days.

Monitoring the chart, the team can readily identify problems with the sprint scope (number and size of tasks) or work pace or 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 timebox.

The work pace in Scrum is called "velocity." Figure 13.8 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:

Effort Completed/Elapsed Days = 315 hours/11 days = 28.6 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 timebox by 6 days.

When a sprint or entire project falls behind schedule, i.e., the planned work cannot be completed within the timebox, 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. If the PO had put 50 features in the prioritized product backlog, and if 40 of those were delivered in the timeboxed sprints, then the PO will have gotten a fully functional and operational product with the 40 highest priority features—all within the time and money allotted.

Sprint 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 planned, completed, and not completed, and 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, and what it needs to improve in the next sprint.

13.4 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 book. Applied to software development, however, the same methods and tools led to widespread failure. Simply, software development is different than construction and hardware development—hence the origin of the *Agile Manifesto* and APM.

Bits versus Atoms Projects

But it works the other way too: APM approaches developed for software development (so-called "bits") projects are of limited applicability in hardware ("atoms") projects. Paraphrasing Paul Lohnes:

> 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.[13]

For example, once an aircraft (a hardware product) has been designed with an 18 meter diameter fuse-lage, a change in requirements to 17.5 meters would require significant redesign to many of the fuse-lage's components, redesign of related systems and components, and involve significant wasted time and cost. You cannot design and build an airplane (or an artificial heart, or a bridge) through iterative sprints. Simply, APM won't work there. (Neither will it work in projects subject to regulatory laws that *mandate* thorough requirements definition and project documentation; e.g. pharmaceuticals development.)

You can, however, design and build some of the airplane's avionics software through sprints,[14] which raises an important point. More and more, products once thought of as hardware are actually hardware-software products, and the software part overshadows or dictates the hardware part. Cell phones are an example, but so are airplanes, some of which are able to fly only by virtue of the software. Thus, although APM methods may be limited to software end-items, given the iniquitousness of software-within-hardware, APM's applicability is vast and growing.[15]

APM: A Project Management Methodology?[16]

Some have argued that APM is not a project management methodology, per se, but a *product development methodology* aimed primarily at creating software, not managing projects. While it is true that APM originated and has been applied mostly in software development projects, we think practices for "development" (of hardware or software, and systems in general) sufficiently intertwine with management and leadership to merit calling them project management methods, and we have done so throughout this book (see Chapters 2, 3, 4, 11, 14).

Project Management Principles

With the spread of "agile practices," some proponents have prescribed that they replace practices originating in TPM, no matter what 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.

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 and all at once, it *should* be done that way, as in TPM; when it cannot, then it will have to be done iteratively and evolutionarily, as in APM. Either way, learning the requirements and striving to meet them is a big part of project management.

Project control is always important too: TPM uses Gantt charts, networks, earned value, and formal change control methods; APM uses controlled iterations —timeboxed and tracked with burn down charts. Methods in both serve to keep projects on schedule and moving toward project goals.

APM has been called "lightweight" project management because it emphasizes informal communication and minimal documentation. But APM applies to projects for which the solutions/end-items are vague or unknown, so there isn't much to document, and creating documentation for its own sake would be a waste of time. APM replaces documentation's information storage and sharing functions with something that works 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: to enable operators and future developers to understand the end-item, its functioning, capability, and limitations.

In summary, tools and methods of APM are appropriate and desirable for software deliverables, but are less so or just won't work for some physical deliverables. APM has a place among project management practices, but many projects still require a more formal, structured approach to definition and execution. So it would be incorrect to try to replace TPM with APM; more appropriate would be to consider APM as an addition to the project manager's toolbox.

13.5 Lean Production and Project Management

Lean production refers to 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 cost but not to the value of a product or end result. Toyota, the originator of lean production, identified particular kinds of wastes. 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 or transfer). An additional waste is non-utilized human talent, i.e., not taking full advantage of workers' ideas, skills, and abilities.[17]

Lean production emphasizes "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 to be 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. Despite originating in manufacturing, many lean production concepts and practices apply to projects; these include *small batch flow, minimal waiting, cycle timing, standardization, minimum handoffs, visual management*, and *Kanban*. Not coincidentally, these concepts overlap with APM practices, hence APM is sometimes referred to as "lean project management." Yet lean concepts and practices can be found in all kinds of projects, not only in bits-oriented (software) projects but also in atoms-oriented (construction) and bits- and atoms-oriented (product development) ones.[18]

Small Batch Flow[19]

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 are done in sequence. Upon completion of one stage, the results and work are transferred to the next.

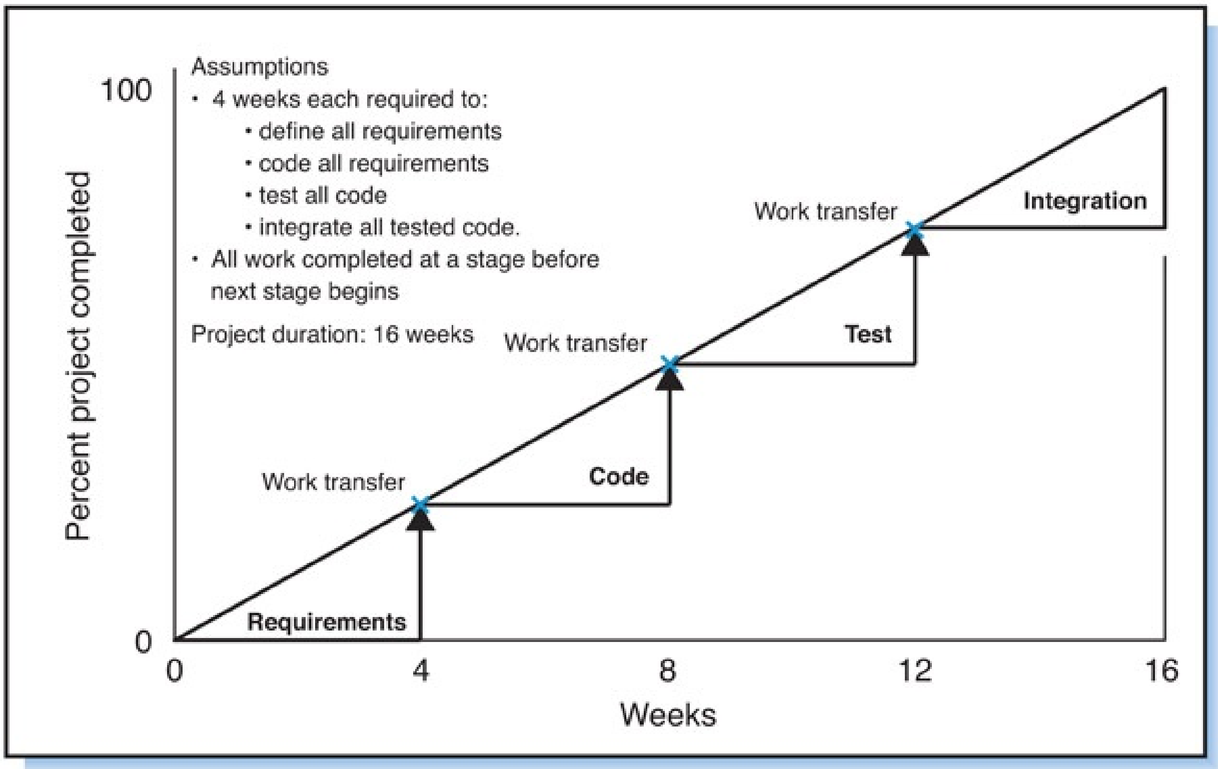


Figure 13.9 Four-stage process, work transferred monthly at completion of each stage.

In manufacturing, work going from stage to stage like this is called “batch production” and the implication is that work moves through the process in batches. If the work batch size is 40 items, 40 items are processed at each stage and nothing moves to the next stage until 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 coded modules, 40 tests, 40 drawings, etc.

Figure 13.9 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 results are released and the next stage can begin. In a product-development company you can sometimes see this happening as stacks of paper (requirements, drawings, work orders, and other documents) moving from department to department. Digital information also moves like this but is less obvious since it is stored and transferred unseen.

The example in Figure 13.9 requires 4 weeks at each stage and, thus, a release and transfer every 4 weeks. But there is an alternative way to do this, which is to transfer *some* of the work at each stage *each week*. For example, instead of waiting until all requirements are completed, transfer whatever requirements were completed each week so the next stage can get started.[20] Figure 13.10 illustrates this, showing work flowing from stage to stage on a weekly basis. 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.—between stages one by one is called "one-piece flow," meaning that everything moves through the project in batches as small as size one.

Reducing the batch size like this has many benefits.

Project Duration

With smaller batches, project stages overlap and the project finishes sooner. In Figure 13.9 where results are transferred between stages every 4 weeks, the project takes 16 weeks to complete. In Figure 13.10, where results are transferred weekly, the project takes 7 weeks.

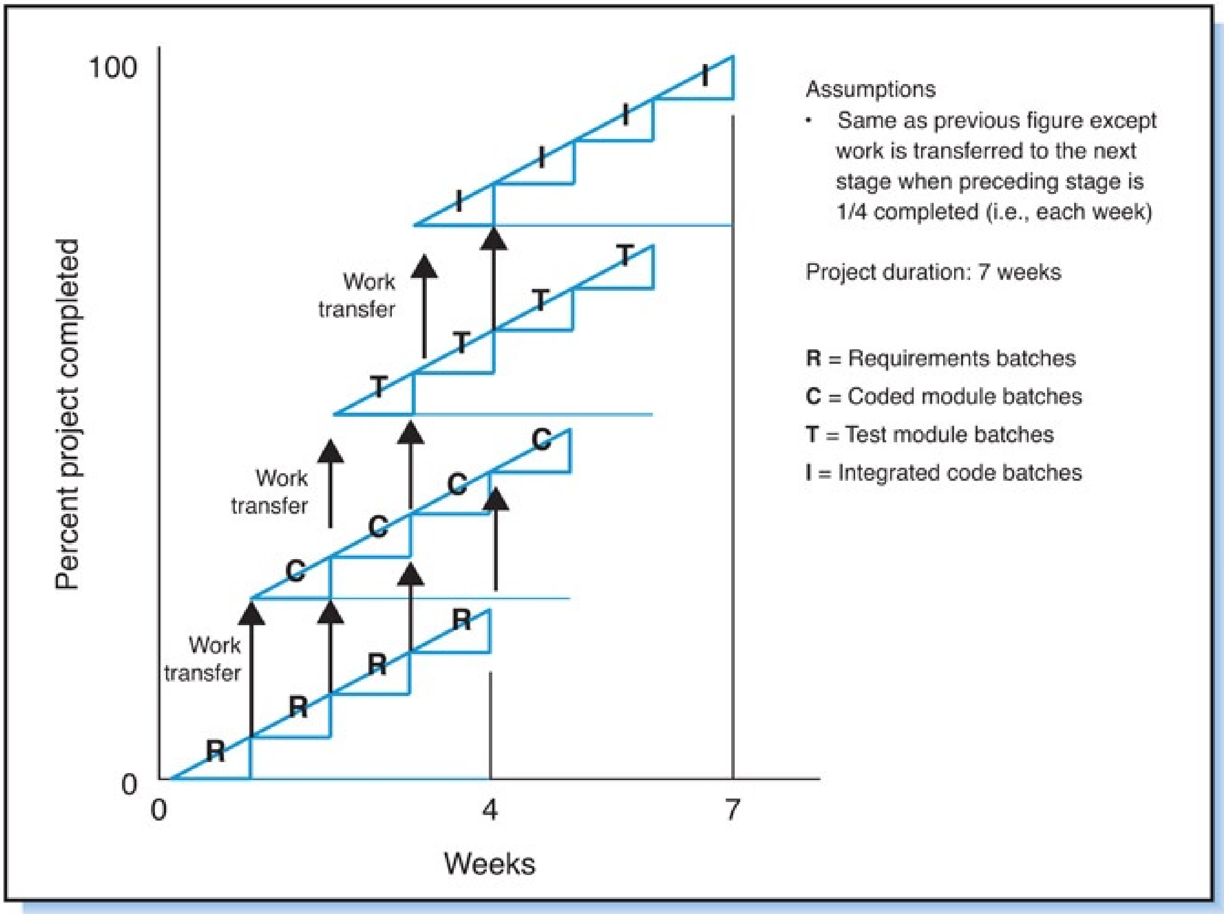


Figure 13.10 Four-stage process, work transferred weekly.

Quality

Suppose a mistake made in Requirements is not detected until Test, which per Figure 13.9 happens 4 or more 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 13.10, 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 yet have influenced them. Also at Week 3 the Requirements and Code stages are still underway, and 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 for it.

Feedback

The quality benefit of smaller batches stems from accelerated feedback. In Figure 13.9 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 tasks/projects and it will be difficult to bring them back. Plus, people forget, and it might be difficult for them to determine the nature of the 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 based on *discoveries made later*—in the Test, Integration, and Launch stages. Small batches permit this: if someone in the Test stage determines a requirement must be changed, she can communicate that back to the people working in Requirements, which is still underway (Figure 13.11). If she determines 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. This is what happens in APM.

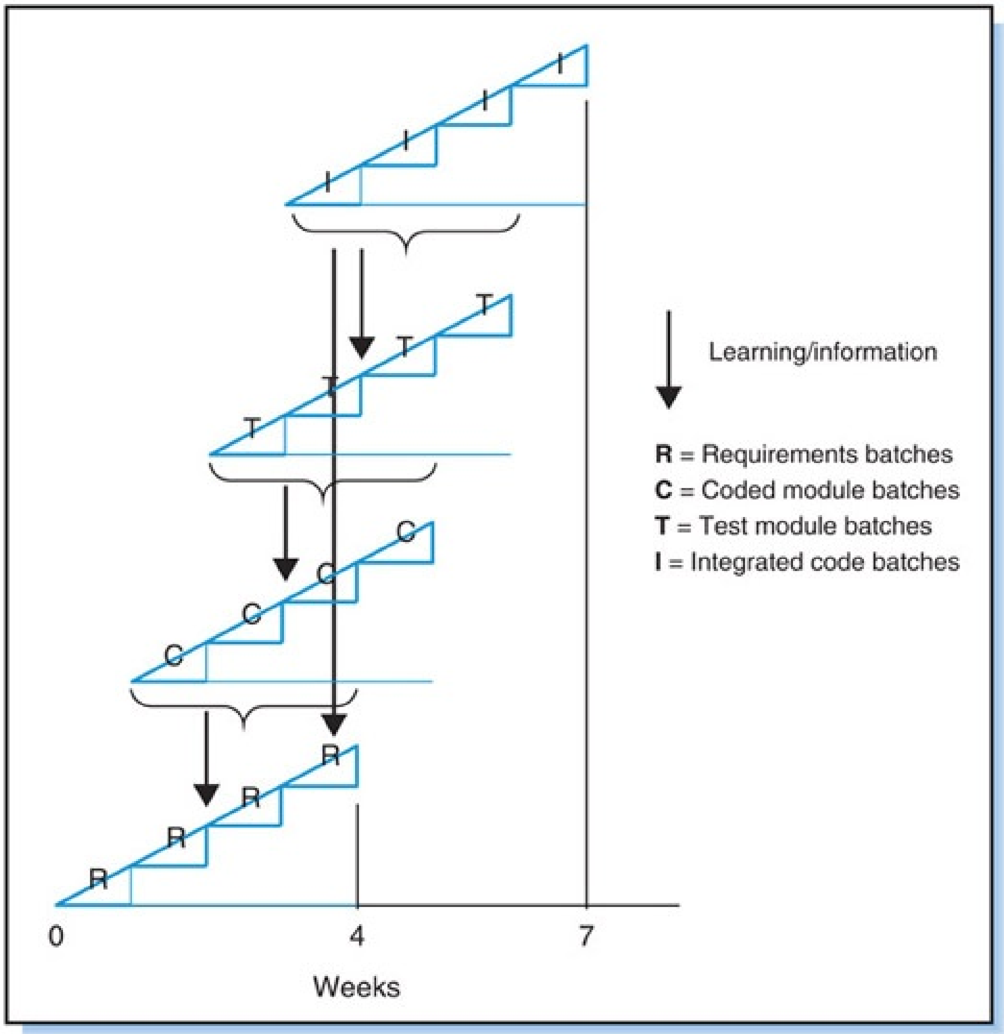


Figure 13.11 Feedback: sharing learning and information with preceding stages.

Efficiency

Suppose the Code stage created programs from a batch of 40 requirements, but then discovered that all 40 requirements were based on an initial, incorrect assumption. This will require that all 40 requirements be reviewed and corrected, and that code based on those requirements also be reviewed and 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, when people get quick feedback they are able to fix things quickly; when they get late feedback, they waste time figuring 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 batches cause smaller workload variability and result in 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 other programmers' code.

Overhead Cost

The larger the batch the more items that have to 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. And, since 10 bugs will be resolved much sooner than 100, they will not have to be reported on for nearly as long.

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 enable quicker feedback about all of these, and they enable the project to finish quicker and, thus, be vulnerable to the risks for a shorter time.

There is of course also an argument for larger batches: decreased cost of batch setup and transport/transfer. Usually work on a batch must be preceded by preparation, and afterward the batch must be transferred; thus, concomitant with smaller sized batches are higher setup and transfer costs. In projects, however, such costs are often low or negligible (what is the cost to transfer batches of information?), and the comparative advantages of small batches win out.

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-iteration: repeat stages iteratively; 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; with the last sprint, combine the results to create a completed product.

In general, batch size and iteration frequency should be varied depending on 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: give the design group requirements as soon as they are created; give the modeling group drawings as soon as they are 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 Wait[23]

Queues—backlogs of work waiting to be done—are the project equivalent of inventory. In general, the longer the queue, the longer the waiting time (think of people waiting in line to get through a check-out counter; 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 13.12, which is similar to Figure 13.9, 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 have to wait in line at the next stage to be processed. Figure 13.13 show what happens with smaller batches. The smaller triangles represent smaller queues and shorter waiting times.

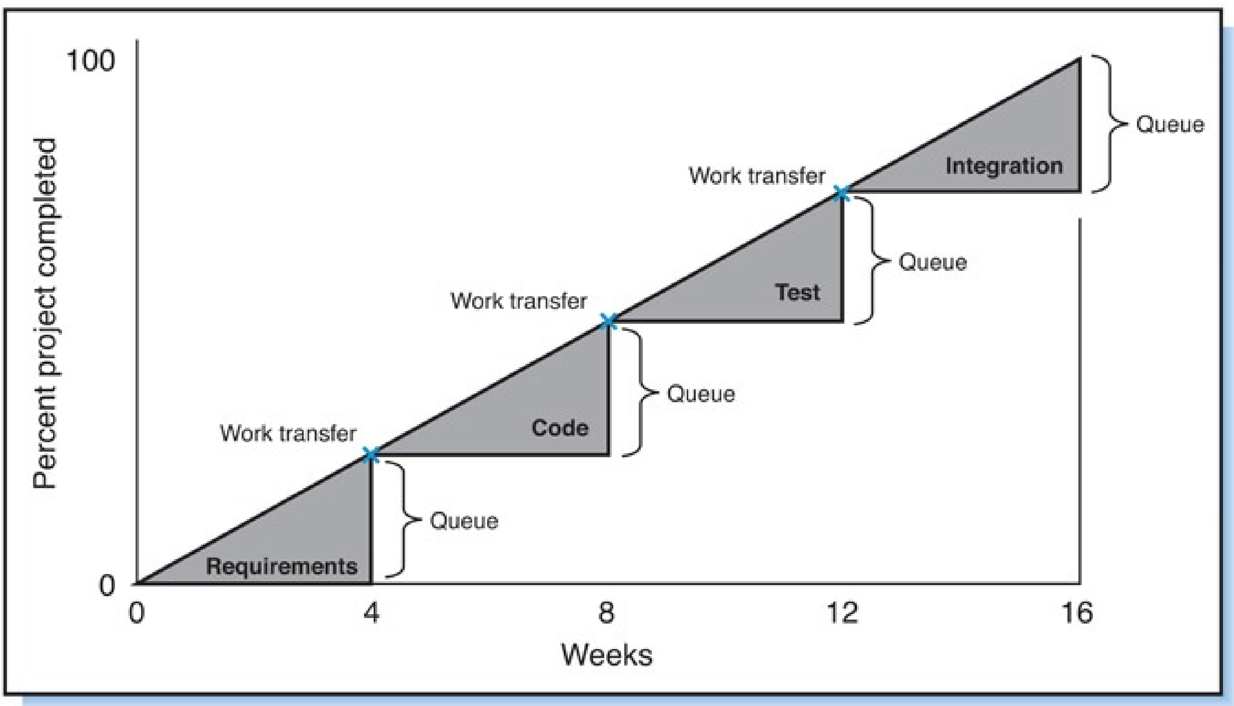


Figure 13.12 Queue length, work transferred monthly.

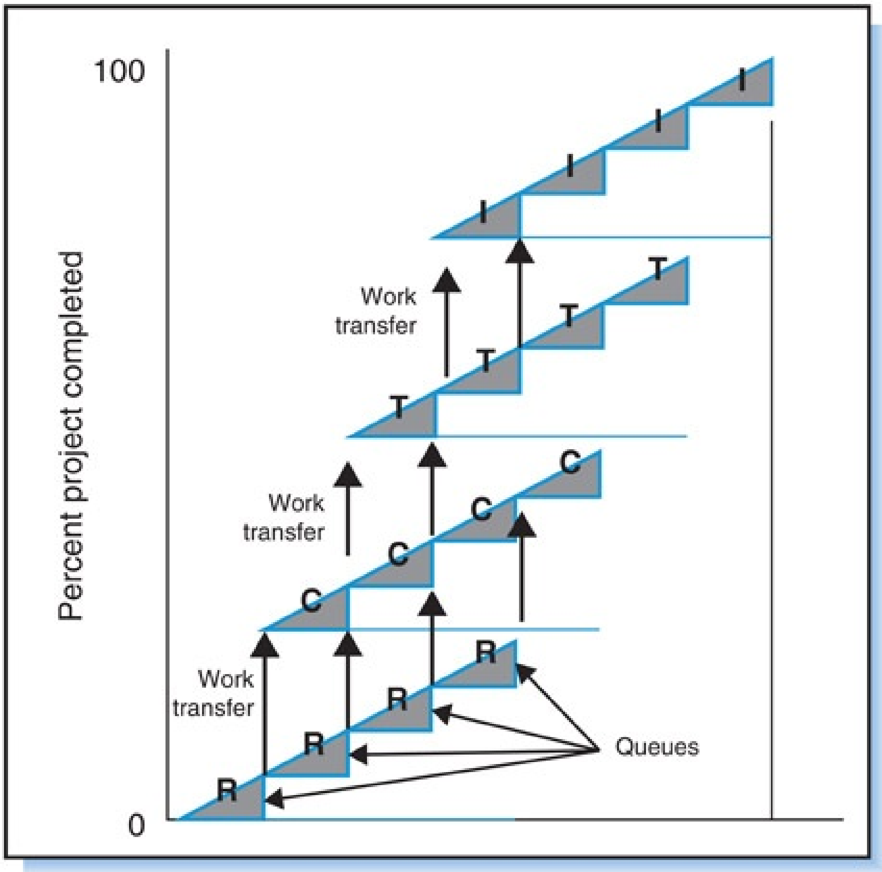


Figure 13.13 Queue length, work transferred weekly.

In manufacturing systems, queues are easy to see—they are physical items—and when they get too large managers try to shorten them. But in projects, queues are often invisible—they consist of information stored on computers; unaware of them, managers don't feel compelled to do anything to shorten them. 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 items 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," the sizes of batches arriving at the stage often can be reduced.

The "gating process" described in Chapters 4 and 18, whereby the project is halted at various stages for review and authorization, illustrates the large-batch, long queue case in Figure 13.12. Despite its purported advantages, the gating process holds up work and information at each stage-gate and does not release it until everything is approved. If speed-to-market is a goal of the project, gating is not the way to achieve it.

See Chapter 4 and Chapter 18

Cycle Timing and Standardization

Cycle time is the time to complete a job or unit. In a project, cycle time can be the time to build a model, code a module, complete a test, launch a new product, and so on. *Cycle timing* is the concept of regularity or rhythm, and it means 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, day after day” or “we will complete one major test every week, week after week” or “we will release a new version of the product every month, month after month.”

Cycle timing is related to the concept of *smooth flow*, or work that is performed uniformly to a beat, rhythm, or pace, without interruption. Such uniformity reduces uncertainty about workload variability and enables managers and workers to know and meet expectations; for example, each day a programmer knows she is expected to complete one coded module. Smooth flow is achieved in part by moving work stage-to-stage on a periodic basis and in small batches. The work is sized small enough so it *can* be completed in the assigned cycle time.

In APM projects the work flows in this manner—the aforementioned concept of velocity. Consider for example a project that involves daily cycles of code-test-integrate. 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 lean principle of *standardization*, which refers to setting standards on processes, tasks, schedules, etc., so that everyone tends to do work in the same way, for the same amount of time, and use the same steps and resources. Standardization reduces uncertainty and variability regarding what must be done and the resources needed to do it. In lean organizations, the work standards are set by people most familiar with the work—those actually doing it; this is part of the self-management principle described later.

Meetings and Reviews[24]

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 total time of meetings stays the same since frequent short meetings require no more time than infrequent long meetings.

Meeting schedules are standardized and cycle-timed, i.e., meetings are convened at the same time and place, every day or same day every week. 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 same standardized-cycle-timed concept applies to project status reviews: convene them at regular intervals, at the same time and same day every month, regardless of project progress or issues. Everyone knows the meeting dates in advance and can fit them into their schedules.

The benefits of short, regular, frequent meetings are seen everywhere. Reinertsen tells the story of HP where each 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. After the coffee cart was terminated, engineers had to go to the cafeteria for coffee. Everyone went at different times and the cross-pollinating dropped off significantly.

Similar standardized-cycle-timed rationale applies to resources that support multiple projects: make them available to each project at a scheduled time every day or week. For example, the manufacturing representative is available to a project team 9 am to10 am each morning. That way anyone on the project who needs to work with the representative will have an hour during which the representative is fully available. If an important issue arises, anyone can request the representative's assistance at other times; otherwise they wait until the next day. The time available can be adjusted for the stage of the project and demand for the resource.

Self-Management, Minimal Hand-Offs, Visual Management, and Kanban

Lean production philosophy recognizes that problems and opportunities in a process are often first identified by workers in the process and, given appropriate skills, they are the people best-suited to make decisions and take corrective action. This is the lean principle of *self-management*, which refers to empowering worker teams and providing them with the information necessary to take action without direction from supervisors or managers.

In a self-managed project team everyone is able to assist everyone else, and each person is able to do a little of everything, regardless of her specialty. Reinertsen calls these "T-shaped resources," people deep in one area but broad in many—Jack-of-all trades, master of one![26] T-shaped resources are developed by hiring "I-shaped resources" (deep 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 indicated on a white board), 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.

Commonly in projects, team members 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—frequently and in small batches. They learn about each other's families, backgrounds, and interests, which builds team cohesion.

Collocation also eliminates a main source of inefficiency in projects—*handoffs*, which refers to the transfer of tasks or items between project stages, workers, departments, or contractors. Handoffs are wasteful: jobs are waiting to be processed, and information transfer is inadequate or incorrect. Collocation minimizes these wastes because everyone already knows about everything.

Self-managed teams are aided with *visual management*—visual cues all around them that provide information to enable them to decide what, when, and how much to do and when and where problems are occurring. On a production line, for example, each worker observes other workers at stages before and after her. If she sees those stages are being underutilized or overloaded, she accelerates or decelerates her own work or walks over to assist those stages.

A self-managed project team acts in similar fashion. It tracks and controls workflow through daily standup meetings, burn down charts, and white boards. Sticky notes on white boards show jobs or user stories at each stage of the process: as jobs move from stage to stage, notes are moved from section to section, which enables everyone to see the jobs in each stage of the project and which stages are over- or under-loaded.

Similarly the team manages the number of jobs in stages of the project, and using a white board it can *restrict* the number of jobs (queue size) in a stage and prevent downstream (later) stages from being overloaded by work coming from upstream (prior) stages. The lean production concept of regulating and smoothing the work flow by restricting the work volume at each stage is called *Kanban*. With Kanban, also called "pull production," work is "pulled" through a process: no stage is permitted to transfer work to the next until it receives a signal that the next stage is ready to receive it. For example, on the white board in Figure 13.6 each stage can hold a maximum of three jobs. The signal that a stage is ready to receive another job is simply when the number of jobs (sticky notes) at the stage drops below three. The team is able to use this simple, visual method to monitor progress and control work to keep jobs flowing at a uniform pace.

13.6 Summary

Traditional project management (TPM) methodology applies to projects where the problem/goal 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 flow—that anything going through a multi-stage process should move unhindered. APM is sometimes called "lean project management," but lean practices apply to TPM projects as well.

One lean concept is 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. Another benefit is that project teams learn and can do things quicker, which reduces risk.

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. Batch size and queue length are also related. The smaller the batch arriving at a stage, the shorter the work queue and the time to move work through each stage.

Another lean concept, cycle timing (CT), means devoting the same time to do each kind of task; everyone receives, produces, and delivers known entities every day. CT is facilitated by standardizing work procedures.

Lean philosophy recognizes that the problems and opportunities in a process are often first identified by workers in the process; given the right skills, they are often the best-suited to make decisions and take corrective actions. This is the concept of self-management—empowering worker teams to decide and take action aided by visual management—providing teams visual information to enable them to decide what, when, and how much to do. The white board is an example of a visual management tool: it enables the team to monitor jobs in a project and restrict the volume of work at stages. That practice, called Kanban, smooths the work flow and facilitates CT. For project managers, such concepts suggest ways to improves efficiency and remove wastes—ways limited only by the imagination.

The preceding sections of the book described how project managers, organizations, and teams plan, organize, and guide projects from start to finish, though, little was said about the project managers, organizations, or teams *themselves*. The following section focuses on managers and teams; it addresses project *organizational behavior* and the topics of organization structure, leadership, teamwork, and conflict and stress.

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, timebox, conditions of satisfaction, potentially shippable product.
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 white board 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 timeboxed to 15 days. What is the velocity? Will the project finish within the timebox?
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? How does it apply to project management?

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 white boards, burn down charts, or other methods used?
9. Is the project manager aware of "lean methods"? Does she apply them to project management?
10. Would you say work in 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 13.1 Grand Entry for Accent, Inc.

The goal of the Grand Entry project was to provide a new web portal for employees of Accent, Inc. that would replace the existing portal, which employees felt was 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 would be reviewed and approved by "stakeholders."

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. Being 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 approaches that the maintenance group had already proven 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 the GE team was at the mercy of Metasoft. Metasoft had assigned no special priority to Grand Entry and in each case took several days to fix problems that halted work on the project.

The result of each sprint was a beta version that the GE team demonstrated to the focus group. The focus group's response typically 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 of 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.

As the requirements list grew Theodora imposed more work on the team, which resulted in overtime every day. She had tried to avoid overtime, but never being completely aware of the tasks her team was working on, she would request 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 she realize she was asking too much. Theodora didn't know exactly what team members were doing because her tracking method was entirely verbal. She had told team members they would be responsible for informing 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 estimated dollar range. The most significant requirements as identified by the focus group and senior managers were incorporated into the portal, but opinions about Grand Entry's effectiveness from the broader employee population are still pending. Operation and future upgrade and repair of Grand Entry will be handled by the maintenance group, which, said the group, might be difficult since the GE team's "agile process" produced little documentation—so little that the functioning of portions of the site is hard to comprehend, and determining where and how to make fixes could prove challenging. 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 13.2 Technology to Track Stolen Vehicles[27]

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 for insurance purposes. 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 DevOps software development methodology. The restructure split the IT department into two teams, one for development (50 employees) and one for operations (30 employees). The 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 more quickly and accurately. SoftTech was included in the project 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, SoftTec would create software for the GPS-based tracking application, the TFI team would create hardware to add functionality to the vehicle receiver unit, and SoftTec would integrate the hardware and software. SoftTec was capable of also developing hardware, but in the interest of the DevOp methodology TFI management gave the task to its own development team to encourage a close working relationship with the operations team.

SoftTec 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 SoftTec, made sure that virtually 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 TFIApp 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 farther and farther 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 seems to have been missed 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 some 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?

Endnotes

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*. Boca Raton, FL: CRC/Productivity Press; 2011.

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*, ACM, 11(4): 14–24, August 1986.

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. Ibid., pp. 117–140.

9. Ibid., p. 202.

10. Ibid., 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. Adapted from Ries E. *The Lean Startup*. New York: Crown Business; 2011. pp. 138–140.

12. See Schwaber K. and Beedle M. *Agile Software Development with Scrum*. Upper Saddle River, NJ: Pearson; 2001.

13. 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.

14. 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.

15. 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.

16. See Weaver P. Agile is not a project management methodology! Blog posted April 2, 2012. http://network.projectmanagers.net/profiles/blogs/agile-is-not-a-project-management-methodology/Accessed Sept. 10, 2014.

17. Nicholas, *Lean Production for Competitive Advantage.*

18. Lean production applied to construction projects; see Ballard G. The Last Planner. *Northern California Construction Institute Spring Conference.* Monterey, CS: 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.

19. Lean project management concepts in product development: see Reinertsen D. *Managing the Design Factory*, op cit., and *The Principles of Product Development Flow: Second Generation Lean Product Development.* Redondo Beach, CA: Celebras, 2009. The following sections draw from the latter, especially Chapters 5 and 7.

20. 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.*

21. Reinertsen, *The Principles of Product Development Flow*, p. 112.

22. Ibid., p. 115.

23. Ibid., pp. 111–120.

24. Ibid., pp. 180–185.

25. Ibid., p. 185.

26. Ibid., pp. 155–156.

27. Bond-Barnard T., University of Pretoria.

Part IV
Organization Behavior

Projects are organizations of individuals and groups created for the purpose of delivering results or end-items, and project success depends in part 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.

Chapter 14
Project Organization 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 grouping of elements. The other is the *informal structure*, the unpublished one that describes 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 organization structure of projects and 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 occasionally projects are conducted without any formal project organization, per se. In other words, a manager and people are tasked with working on a project, but no project group or organization is explicitly recognized. Lack of an identified project organization makes everything more difficult to manage because of uncertainty over reporting relationships and who, exactly, is on the project. Also noteworthy, and as will be discussed in the chapter, is that most project organizations are "superimposed" on the existing organizational structure; although this makes them better suited to accomplish project goals, people in the formal organization sometimes view the project organization as disruptive to business-as-usual

14.1 Formal Organization 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 publicized in a chart such as the one for NASA in Figure 14.1. A quick glance at it reveals both organizational hierarchy and groupings for specialized tasks. The chart in Figure 14.1 shows, for example:

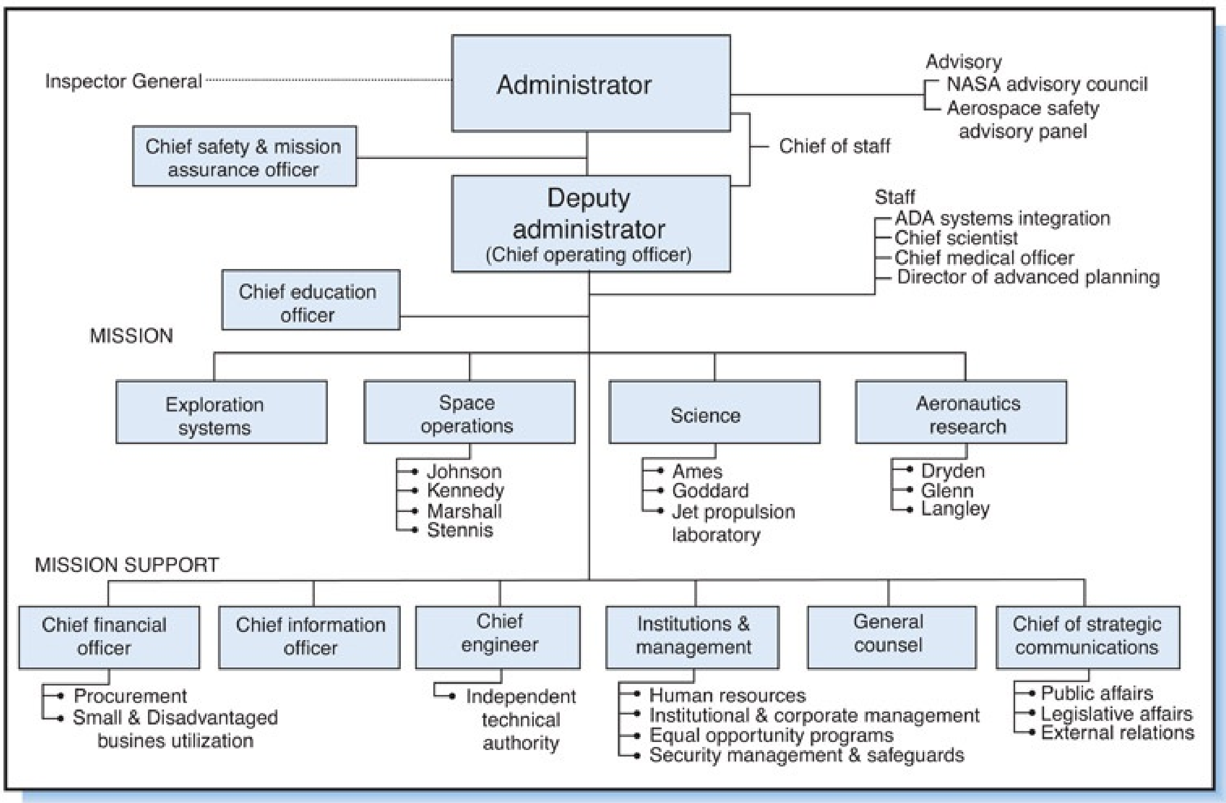


Figure 14.1 NASA organization and program chart.

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 each subdivision (e.g. projects at 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 along the lines from one box to the next).

There are things the chart does not show, for instance, personal contacts whereby, for example, workers at Jet Propulsion Lab communicate directly with workers at Dryden via email and telephone, not (as the chart implies) via the directors of these centers. Nonetheless, the chart is useful for giving an overview of the organizational departments and roles and formal relationships among them.

14.2 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, they add new subdivisions and groupings to handle new situations and emerging problems. For example, when a company expands its product line, it may subdivide its manufacturing area into product-oriented divisions to better address problems specific to each line. As a company expands its sales territory, it may subdivide its marketing force geographically to better handle problems of regional origin. This subdivision into specialized areas is called *differentiation*.

Ordinarily the subunits of an organization do not act as independent entities 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 traditional organizations differentiate into subunits: functional, geographic, product, customer, process, and project. We will start by looking at the first five forms of differentiation and then delve more deeply into the project form.

Functional Differentiation

Functional differentiation is so called because it divides the organization into functional subunits such as marketing, finance, production, human resources, and research; the structure of the Iron Butterfly Co. in Figure 14.2 is an example. Most of the integration between subunits is handled by rules, procedures, coordinated plans, and budgets. When discrepancies occur that cannot be handled by these measures, the managers of the affected subunits must 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

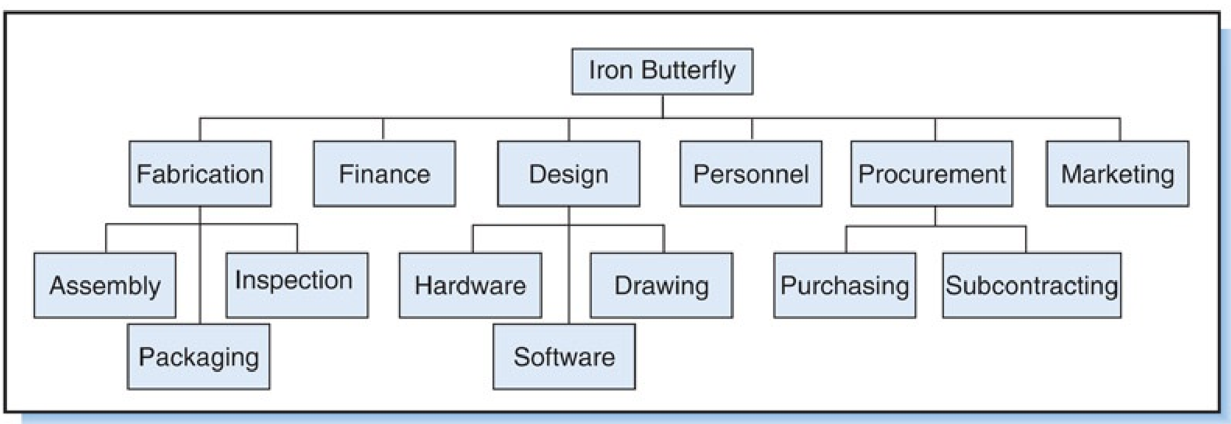


Figure 14.2 Functional differentiation in organization structure for Iron Butterfly Company.

Most organizations have more than one basis for differentiation. The Roman army was also geographically differentiated; that is, it was structured according to region or location. Organizations subdivide according to region (e.g. Atlantic branch; European division; Far East 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. Regional subunits may operaterelatively autonomously and any integration between them might be achieved 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 General Electric are split into subdivisions wherein each designs, 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 subdivisions 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 large military and commercial sales often establish a separate division because federal requirements for proposals, contracting, and product specifications differ substantially from those for commercial customers. The level of integration between customer divisions depends on the interdependency of the product lines; typically, however, there is little integration.

Process Differentiation

Organizations also differentiate according to process or sequence of steps. This is illustrated in Figure 14.2 for the Fabrication division of Iron Butterfly Co., which includes departments for assembly, inspection, and packaging—three steps in the fabrication process. The subunits in this form of differentiation 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 and 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, the price of which is less flexibility and greater difficulty integrating the subunits.

The traditional organization forms work on the assumption that all problems or tasks can be neatly classified and resolved within specialized units that can work independently. Problem is, when a problem arises that doesn't fit into any of the subunits, there may be no place for it. Such problems fall through the cracks.

One way to handle unanticipated, unclassifiable problems is to redesign the organization each time one arises. However, the process of redesigning an organization to suit unique problems is slow and expensive. The alternative to redesign 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 the chain of command can get quickly overwhelmed, and the response is to add more managers (i.e., further increase the size of the bureaucracy). In short, traditional organization forms are not well-suited for situations with frequent change and high uncertainty. Nonetheless, most projects are conducted within or use resources provided by organizations with a traditional forms of structure.

14.3 Requirements of Project Organizations

Project environments are characterized by frequent change and uncertainty and they typically require the resources and coordinated work effort of multiple subunits and organizations. Each project is a new undertaking, somewhat unique, aimed at a new goal; because of that, uncertainty and risk are inherent. Changes, mistakes, or delays in one subunit have consequences to all others. Because of that, resources from 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 demands, and resources; consequently, their organizations must be adaptable to the unexpected; in a word, they must be *organic*, which means both highly differentiated *and* highly integrated to accommodate a variety of problems and situations. To achieve this, all project organizations have two properties:

- Subunits differentiated to suit the unique requirements of the project and the environment.
- Subunits integrated using horizontal relations.

These properties are discussed next.

14.4 Integration of Subunits in Projects[1]

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 their *horizontalness* or use of direct communication between the parties involved in a problem. Horizontalness means cutting across formal lines of authority and moving decisions to the level of the parties affected.

All organizations have elements of horizontalness, mostly in the form of personal contacts, informal relationships, and friendships. Horizontalness helps expedite communication and resolve problems between subunits. For example, whenever the assembly department in Figure 14.2 experiences a minor parts shortage, George, the assembly foreman, phones Helen in purchasing for a "rush order" favor. The informal call bypasses the formal structure (George 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. For example, Helen must charge all purchases to an account; if George is not privy to the dollar amount in the account his informal requests might deplete the account before additional funds can be credited, which involves someone in the accounting area who is not aware of George's requests. Further, if George does not tell anyone about the parts shortages, then the reason for the problem—pilferage, defective parts, or under ordering—will never get resolved. In short, informal processes in many regards are inadequate.

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 all 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, need, 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.

14.5 Liaison Roles, Task Forces, and Teams

The *liaison role* is a specialized person or group that links two or more departments. In Figure 14.3 the dotted line represents the liaison role of "inventory controller." This person performs duties in the assembly department, but also notifies purchasing of impending shortages and keeps track of orders placed. The role relieves the assembly foreman of the responsibility and, by legitimizing the process, ensures that orders get placed and are funded and documented.

But the liaison role is not always effective. Though the inventory controller in the example expedites parts ordering, the reason for part shortages goes unresolved. To unravel the problem it might be necessary to involve people from elsewhere in the company. 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 meet to solve a problem. When such a group is formed and begins addressing the problem, they are, in fact, conducting a project. For example, when a shortage occurs, the assembly foreman might call together representatives from the areas of inspection, finance, and purchasing. The task force meets as often as necessary to solve the problem, and then it disbands. The most effective task forces have ten members or less and a team leader or coordinator, and are short-lived.[3]

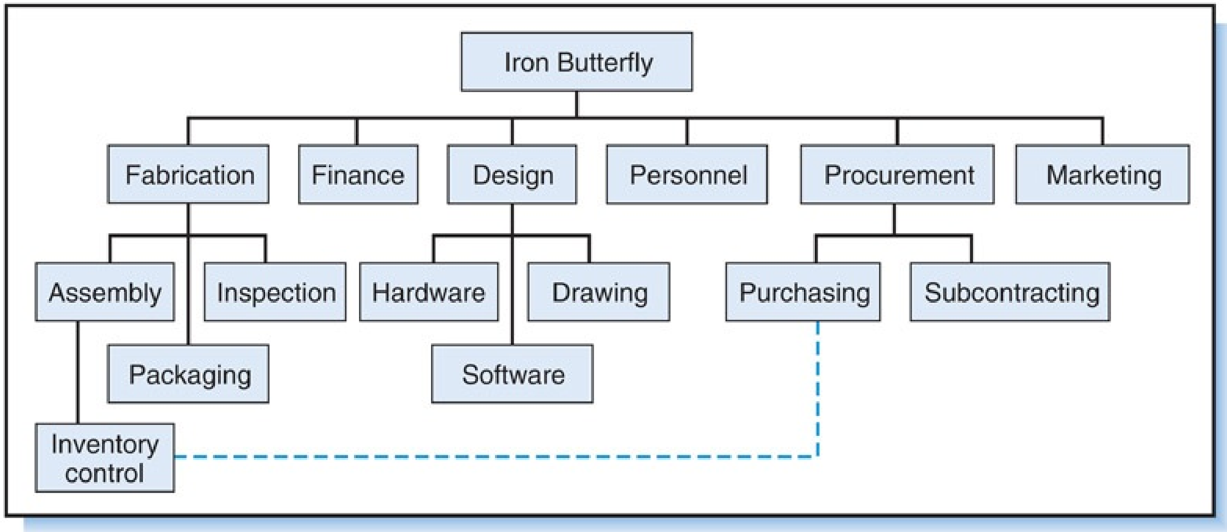


Figure 14.3 Liaison role linking Assembly and Purchasing departments.

Both the team leader and members are selected by (and the leader reports directly to) whoever initiated or sponsors the project—a functional manager, vice president, or CEO. Team leaders are responsible for expediting and coordinating the efforts of the team, and they may have authority to direct tasks and to contract out work. Usually, though, they have little formal authority over team members who, often,divide time between the task force and their "usual" work. Task forces undertake an unlimited variety of projects and special assignments, including the following:

- 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 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 each requires design changes throughout the year, then representatives from design, fabrication, procurement, and other areas need to meet face-to-face on a regular basis to make decisions regarding these changes in response to changing markets and competition. Members work on the team either part-time or full-time.

Most projects involve several kinds of teams; some convene during a single phase of the project life cycle, others for the entire project. An example of the latter is the *change board* discussed in Chapter 11, a multifunctional team that meets periodically to discuss and approve design changes.

See Chapter 11

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 area of work, it delimits their ability to work with people from other areas. For multifunctional projects, the team-building methods described in Chapter 16 help break down barriers and forge bonds between project team members.

See Chapter 16

14.6 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 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 14.4 because all the necessary resources and expertise are there. The team does everything—it prepares the survey instrument, obtains mailing lists, distributes the survey, and processes 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, keeps the project moving, and apprises the manager. 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. A large organization might have over 100 such projects being conducted in its functional departments at any given time.

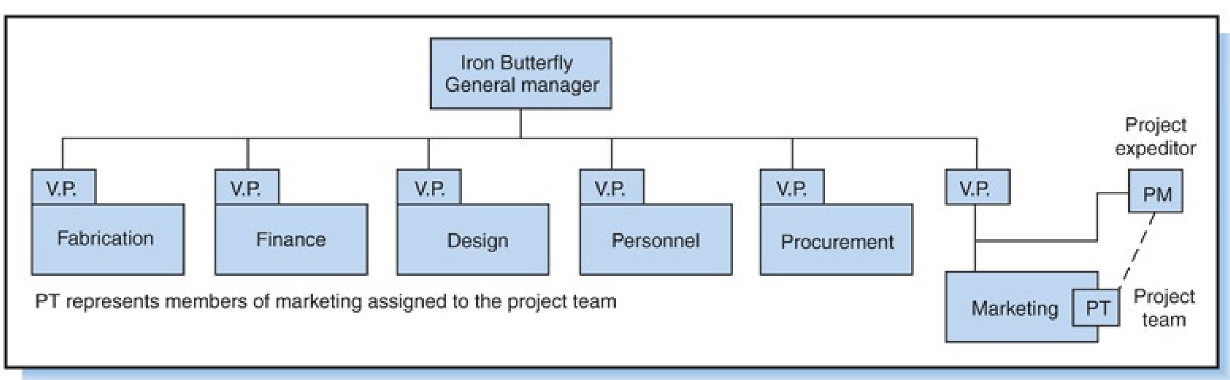


Figure 14.4 Project expeditor within a single functional area.

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, which might be called the “ERP Task Force,” 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 teams are commonplace in product development. By forming closely knit teams of engineers, designers, manufacturers, assemblers, marketers, suppliers, dealers, and customers, 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 in Chapter 4 and later in this chapter.

See Chapter 4

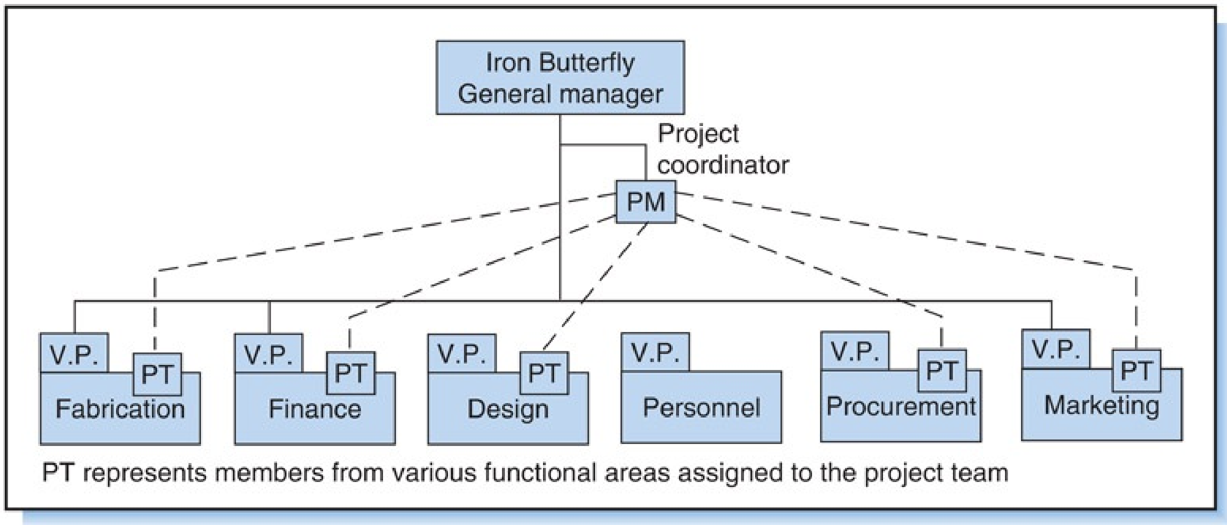


Figure 14.5 Multifunctional project team.

Multifunctional project teams are commonly not associated with any particular functional area (they are *multi*-functional), and they report to a higher-level manager as shown in Figure 14.5, 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 is his reporting to a high-level general manager and, like that of the expeditor, his knowledge of and central position in the project.

14.7 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 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*. Unlike a coordinator or expeditor, the pure project manager has formal authority over all people and physical resources assigned to the project and, thus, maximum control. 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 is involved 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 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 subcontractors. She oversees and coordinates their work with other areas of the project. The project managers in the Delamir Roofing, disaster recovery, and NASA examples in Chapter 1 are pure project managers.

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 14.6 for the Iron Butterfly Company, which shows two pure-project arms, LOGON and SPECTOR. (Of course, unlike people, organizations can have any number of arms!) 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 vehicle size 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 was used to oversee the relocation of its corporate offices. The project 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 areas in the parent company for support tasks. In Figure 14.6, for example, the LOGON project manager might fully control design and fabrication but rely on the areas of finance, human resources, and marketing for functional support.

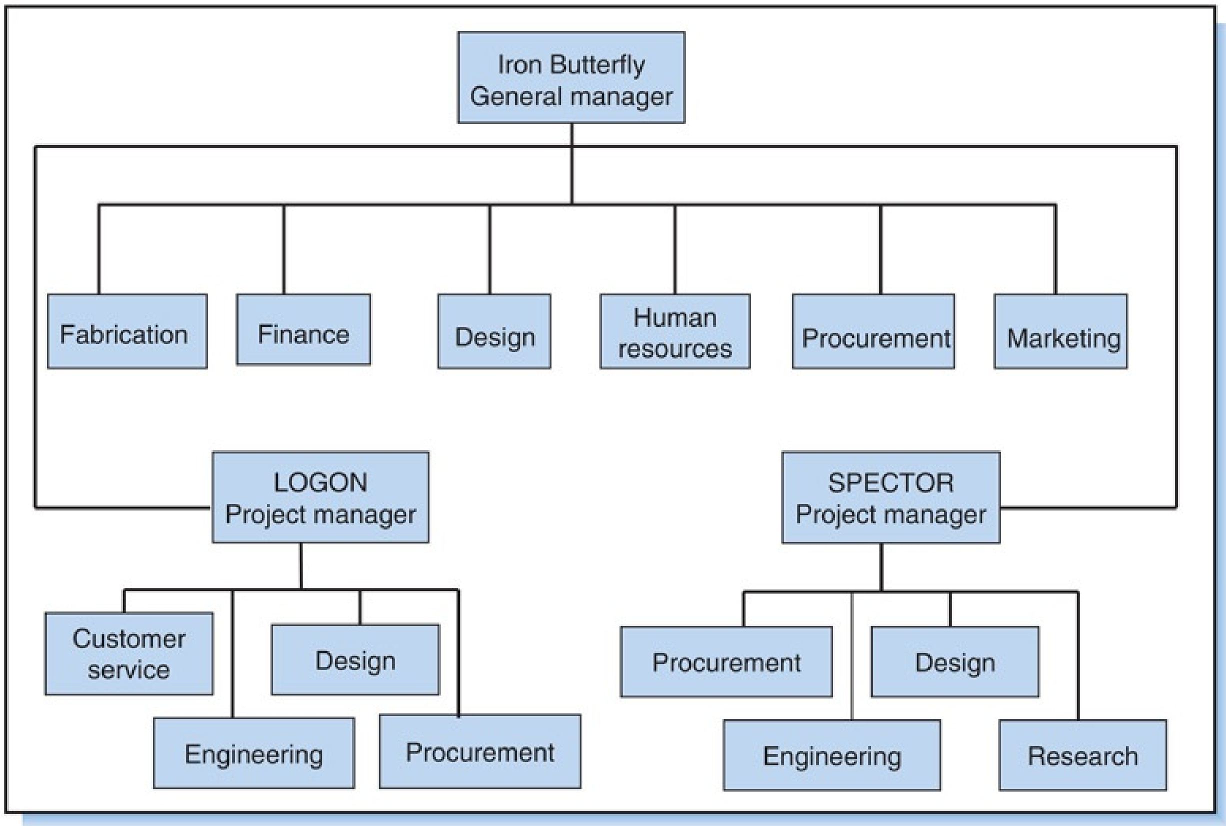


Figure 14.6 Pure projects as "arms" to the functional organization.

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. 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 are examples. When these projects are completed, a function remains to operate the system but the rest of the organization dissolves. Stand-alone projects are discussed later in this chapter in the sections on integration in large-scale projects and concurrent engineering.

Disadvantages

The chief disadvantage of the pure project organization is its *cost.* Because each pure project is a completely or partially independent organization, it must be fully 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 overheads.

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 awarded, and everyone had to be transferred elsewhere or laid off. The payroll loss alone amounted to hundreds of man-months.

In most organizations, the functional manager is the driving force behind workers further developing their technical competencies. Most functional managers encourage their professional workers to expand their capabilities, and they back it up with raises and promotions. But in a pure project organization there might be no functional managers and hence no one to emphasize competency development. The usual tact of the project manager is, lacking suitable in-house technical competency, to contract out the work. While this might suit the project's needs, it represents a missed opportunity for the organization to develop its own in-house expertise. Further, those workers that do have considerable competency often resign after completing what they consider the interesting part of the project because they cannot foresee what they'll be doing next in the project—or what the next project will be.

This suggests 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 after the project ends. Personnel who have worked on long-term projects often become so specialized that they cannot be placed in projects requiring more generalized or up-to-date skills.

Pure project organizations are strictly temporary; 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 jobs for himself and his team that he becomes neglectful of his closeout responsibilities for the current project.

14.8 Matrix Organizations

Although the pure project form often provides the only way to do a large-scale, one-time project, 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 concept, design, manufacture, and promotion is a project; IT, where every hardware and software installation is a project; law and accounting, where every case and audit is a project; and 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 are involved in many projects at a time—they are *multi-project* organizations—and they need the capability to create large project groups quickly without the personnel and cost disadvantages associated with pure project organizations.

To achieve this capability the *matrix* form of organization evolved. First adopted in the aerospace industry by such firms as Boeing and Lockheed-Martin, the *matrix,* illustrated in Figure 14.7, 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. Each project manager creates a project team by negotiating with functional managers to "borrow" the expertise and physical resources needed for her project. Each project is composed of workers who are on loan to work on the team during the course of the project. This sharing of the same workforce across several projects reduces duplication of effort.

Second, while in their "functional homes" workers associate with colleagues in their fields of specialization; this not only keeps them current in their profession or trade, but 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. This makes all of them more effective in their respective projects.

Third, when their assignments are fulfilled or the project is completed, workers return to their functional homes 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.

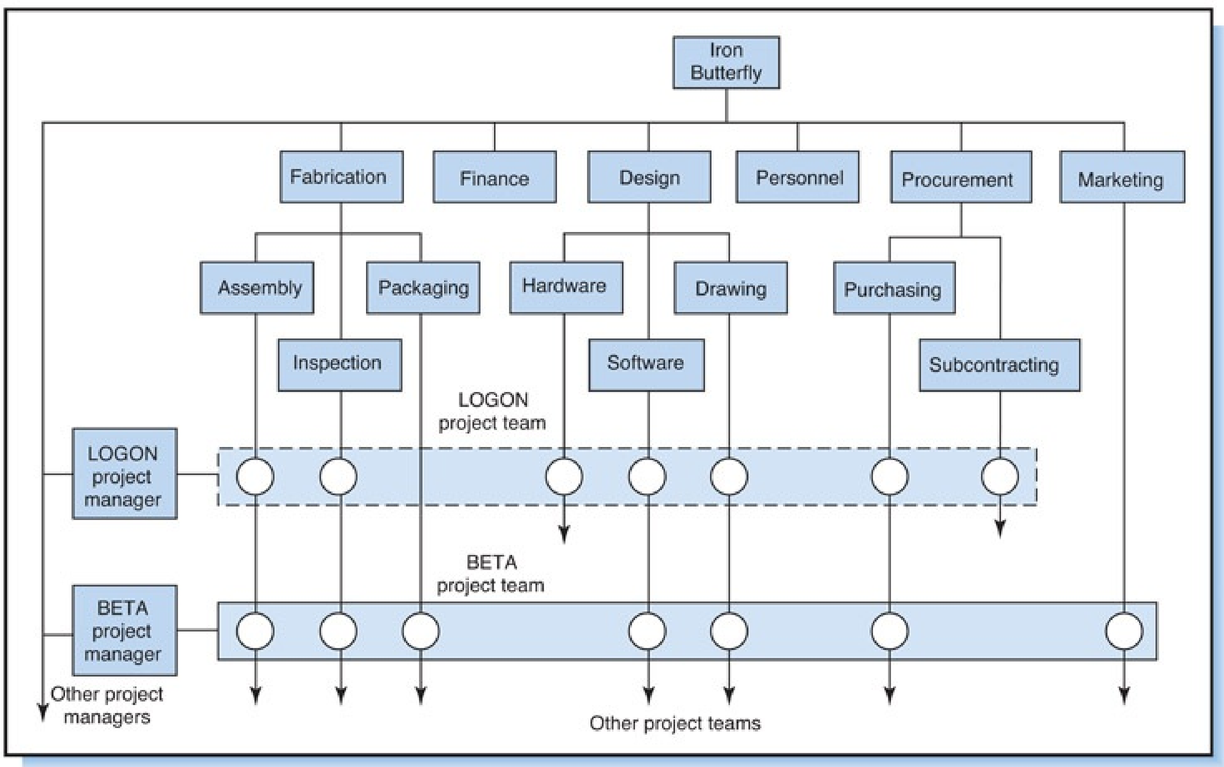


Figure 14.7 Matrix form of project organization.

Although the matrix structure shares the virtue with pure-project organizations of having dedicated resources and a project manager to give the project visibility, the range of authority of the project manager 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 and control over 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 quite fully-fledged project managers and must fit the role into 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 organization, prioritizing and balancing resources shared by the different projects is the responsibility of the *manager of project* or the *PMO director* (the "vice president of projects" in Figure 14.8). The manager of projects attends to 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. With its emphasis on horizontal relations, lateral information flow, and decentralized decision making, the matrix is clearly contrary. 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. It can be done, but it tends to be somewhat difficult and expensive.

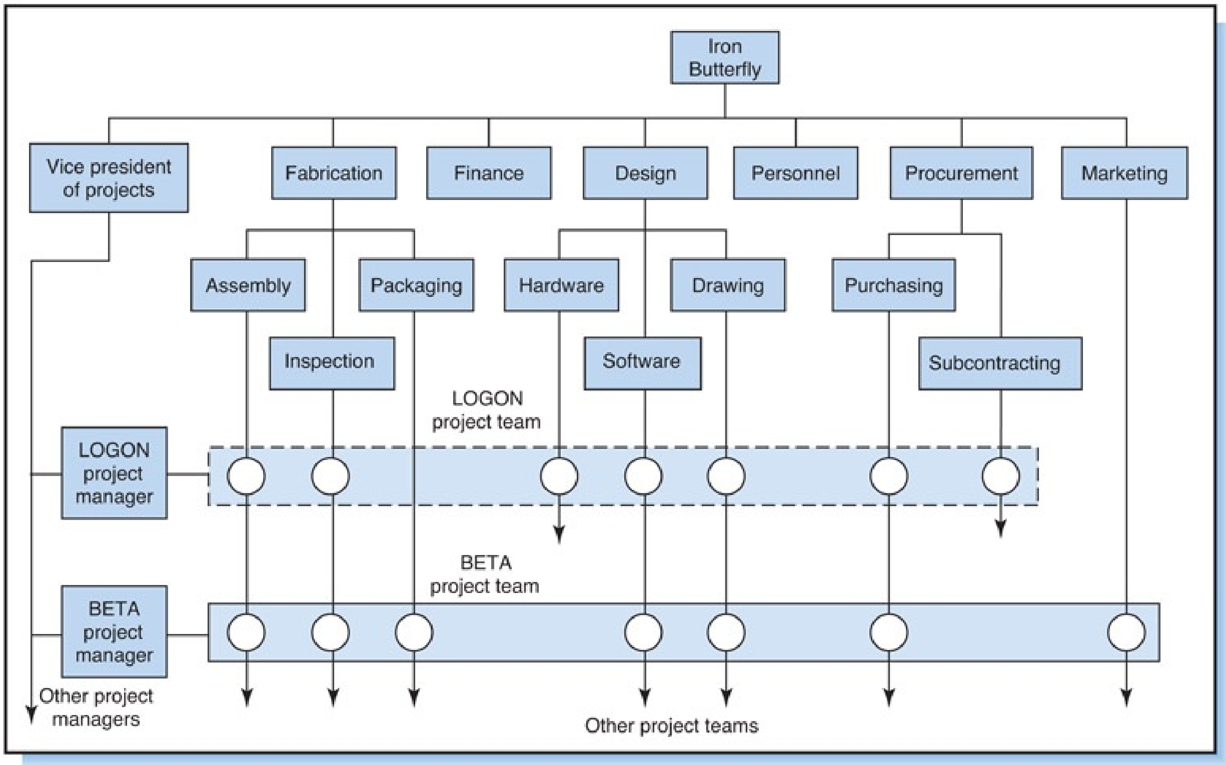


Figure 14.8 Location of the vice president of projects in a strong matrix organization.

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 decisions to be made for the benefit of the project. It assumes, however, that a balance of power exists between functional and project managers. Often, however, authority in the matrix is unclear, and functional and project managers jockey to control one another. In multi-project organizations functional managers control project resources, so conflict arises over which project gets priority and which project managers get the best resources.

The matrix structure attempts to be the best of both worlds—functional and projectized. The main problem with the matrix is rooted in something few people like to admit—fear and power: functional managers fear that project managers (who are sometimes perceived as having the more interesting and challenging work) might take control of "their" resources and reduce their role to a mere "support/staff" function. They become even more worried when the Vice President of Projects controls project funding and threatens to outsource work normally provided by the functional areas. Project managers get frustrated too, because, unlike functional managers, they have little or no control 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 a large extent, *who* within the function should do it).

Here is another problem: each project worker in the matrix has 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 and confusion in the matrix, everyone—managers and workers—must have a common reference, and the organization must establish clear priorities. Boeing, for example, which has used the matrix successfully for years, sets priorities day-to-day: people operate *either* in a project team *or* in a functional area, and they put priority on whichever area they are in.[7] The matrix can lead to still other dilemmas, explained below.

Example 14.1 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 he hasn't grabbed the best resources for the projects that *he* is managing. Also, to people from his department who are working on his projects, the "project hat" is invisible. All they see is that "functional hat," ever mindful of the fact that he controls their wages and promotions; as a result, they *always* give his projects priority 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.

14.9 Selecting an Organization Form for Projects

Project managers are seldom involved in designing the organization 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 14.9 shows the approximate applicability of different project organization forms based upon four criteria:

- Frequency of new projects (how often, or to what degree the parent company is involved in project-related activity).
- Duration of projects (how long a typical project lasts).
- Size of projects (level of human, capital, or other resources in relation to other activities of the company).
- Complexity of relationships (number of functional areas involved in the project and degree of interdependency).

Matrix and pure project forms are applicable to projects of medium and higher complexity and of medium or larger size and in companies that are always undertaking projects. These kinds of projects require large amounts of resources and information and need project managers and integrators with strong authority. 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."

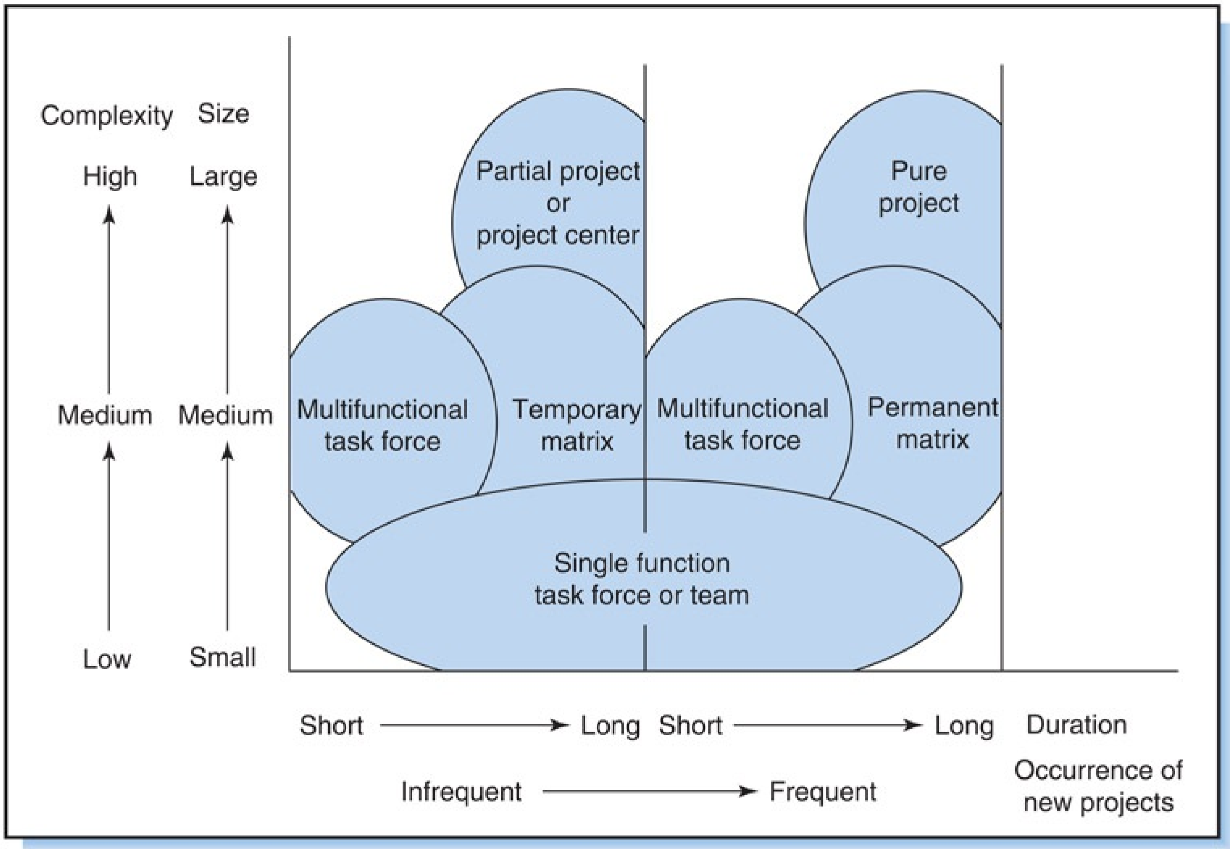


Figure 14.9 Some criteria for selecting the appropriate project organizational form.

For smaller projects that involve several functional areas, task forces and cross-functional teams are more appropriate. Part-time task forces managed by expeditors can effectively handle short-duration projects involving one or a few functional areas. When several areas are involved, a multifunctional task force lead by a coordinator who reports to the general manager is more suitable. Projects of longer duration, but small in scope, are best handled by full-time project teams with coordinators. When the team size needed to accomplish the task becomes large and the interrelationships complex, then a temporary matrix or partial project should be set up. Teams, task forces, and project centers are appropriate when the existing structure and work flow of the organization cannot be disrupted.

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 involves high certainty and little risk, and when time and cost are not major factors. When the risk and uncertainty is great, when time and cost goals are critical, or when there is much at stake, matrix and pure project forms better afford the obligatory high level of 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.

These considerations all relate to the project itself which, in fact, is sometimes less important than attributes and experiences of the parent company. For example, matrix and pure project forms are seldom found in small organizations, which usually don't have sufficient resources and managers to commit. 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 are involved in a variety of projects and use a variety of 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 a 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 14.2 Product Development Organization at Microsoft[9]

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 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 working 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.

14.10 Project office and PMO

The term *project office* has dual meaning: it can refer to a support staff group that reports to the project manager, and 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 (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 projects are 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 various services to project managers, as described in Chapter 17.

See Chapter 17

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 14.10 is for a large-scale engineering development effort. 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.

See Appendix, Chapter 2

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 to Chapter 2), and furnishes inputs for planning and control. Configuration management (discussed in Chapters 9 and 11) defines the initial product configuration and controls changes in product and system requirements and project plans. As shown, the project office also handles contracting and financial control.

See Chapter 9 and Chapter 11

Integrating the functions within a project is achieved by structuring the project office to mirror the functional areas it must integrate.[10] This happens by including in the office a representative from each functional area working in the project (in Figure 14.10, 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 and meet face-to-face. In smaller projects the size of the project office can be reduced by allowing some or most of the functional representatives to remain in their functional areas.

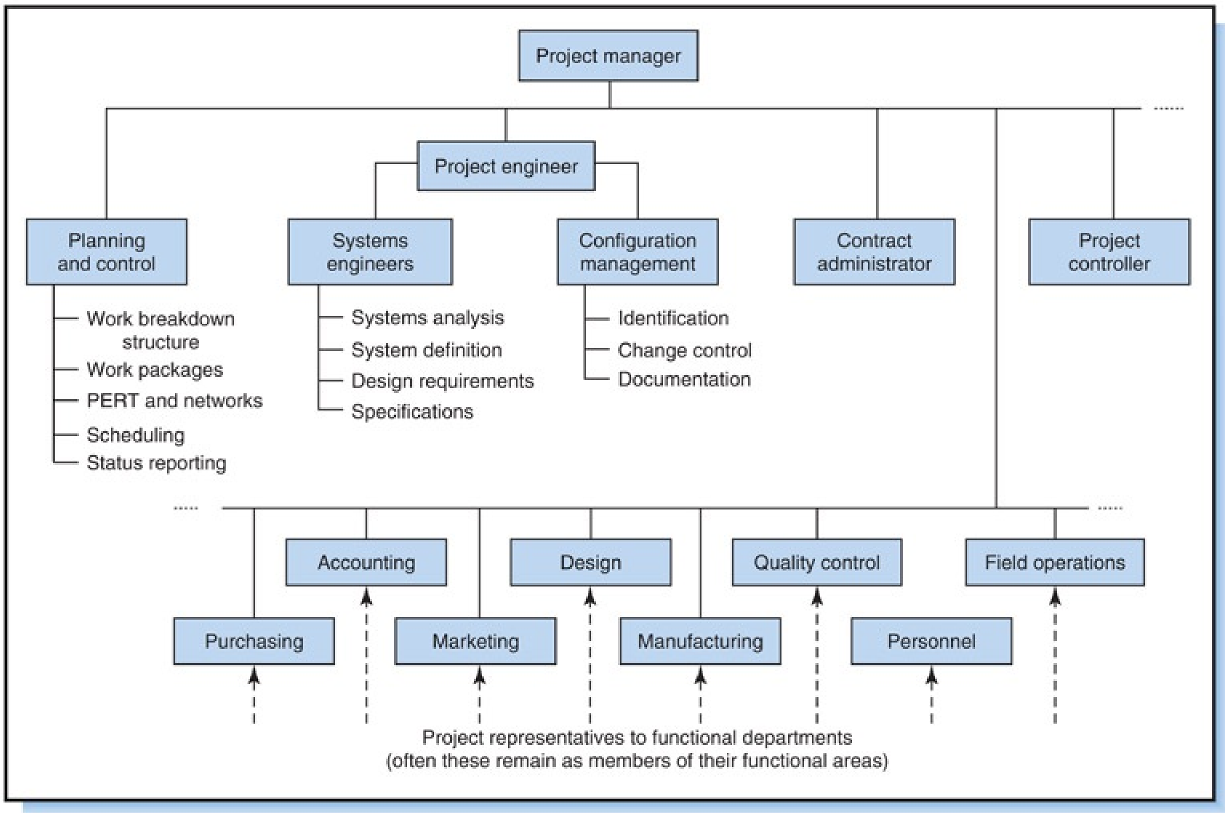


Figure 14.10 Project office for a large development project.

Office of Projects, PMO, and the Program Office

Multiproject organizations also have an *office of projects* (not to be confused with the project office), program office, or *PMO*. This was shown in Figure 14.8 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 14.6, it would be located below the general manager and on the line connected to the LOGON and SPECTOR projects). When projects are small, the office of projects substitutes for individual project offices and handles 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 17) 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.

See Chapters 1 and 17

14.11 Integration in Large-Scale Projects

In a large-scale project (LSP) or *mega project*, numerous parties—sponsors, prime contractors, subcon-tractors, consultants, and suppliers—contribute to one effort. Figure 14.11 shows the principal contributors and relationships in an LSP. Relationships are complex and lines of authority connecting the parties are often weak, sometimes based entirely on contracts and purchase orders. If Figure 14.11 appears somewhat 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 (Canada's La Grande hydroelectric venture, Holland's Delta flood control project, the Channel Tunnel, China's Three Gorges Dam), company relocations (involving the client, movers, construction companies, recruiters, consultants, and suppliers), and corporate mergers (dual sets of clients, consultants, and attorneys).

Notice in Figure 14.11 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 helps 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 a number of contributors, each responsible for a specific subsystem or component that must be integrated with others to form the overall system. Figure 14.12, 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's orbit, support systems, and the numerous organizations that work to develop, produce, and integrate 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.

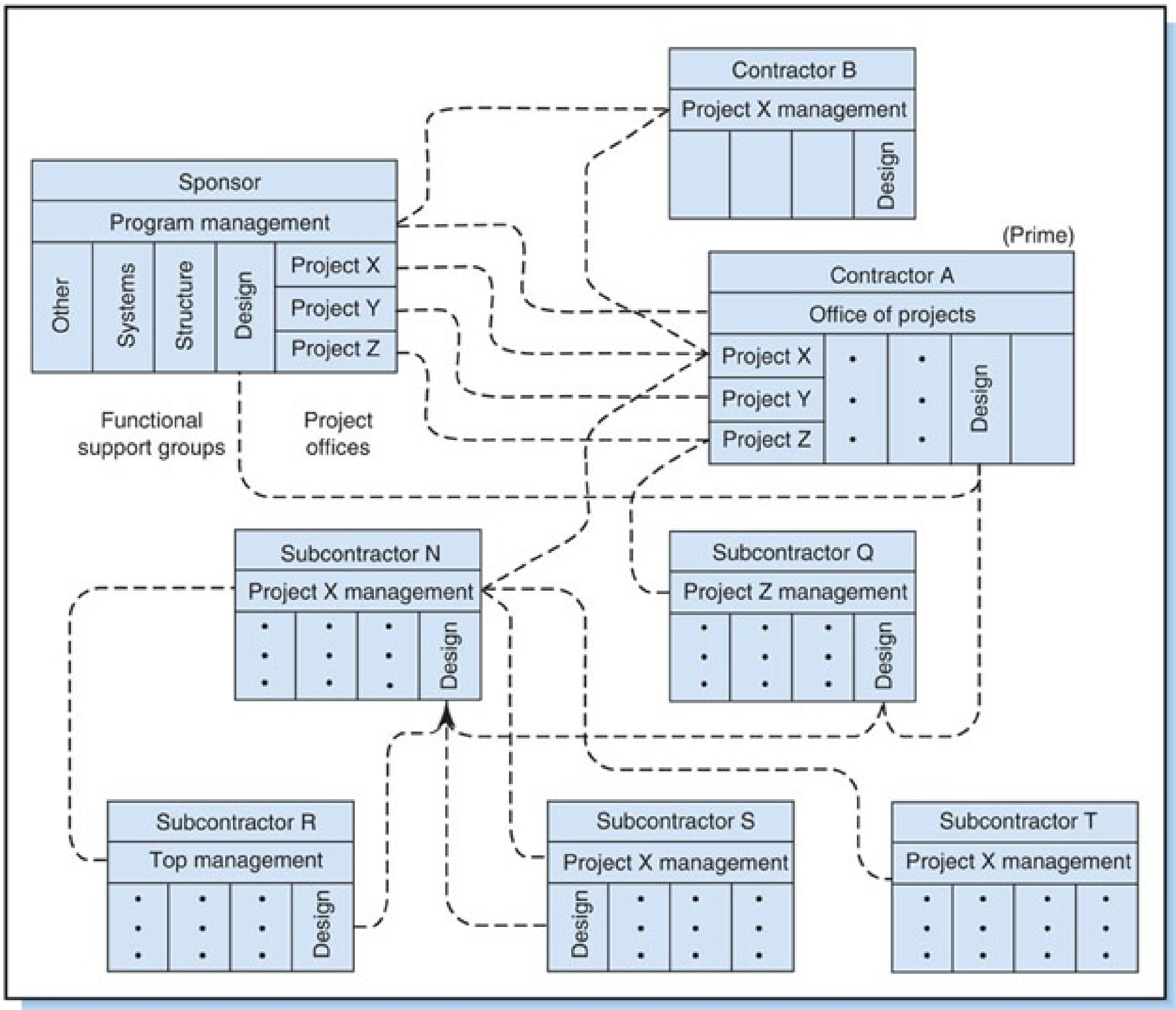


Figure 14.11 Integration relationships in a large-scale project.

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 were different, so plugs and fasteners would not fit, 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 and only worsened matters, as explained by Livingston:[13]

> A contractor wanted to change the clock in an airplane cockpit from a one-day to an eight-day mechanism. It wrote a request and gave it to the military representative, who forwarded it to the military technical group. The tech group reviewed the request and asked the contractor for a more-detailed request. The contractor revised the request and resubmitted it. The tech group approved the request and sent it to the change committee. The change committee reviewed and accepted the change, then sent written authorization back to the representative, who forwarded it to the contractor. In all, this simple change request took *three months* to approve.

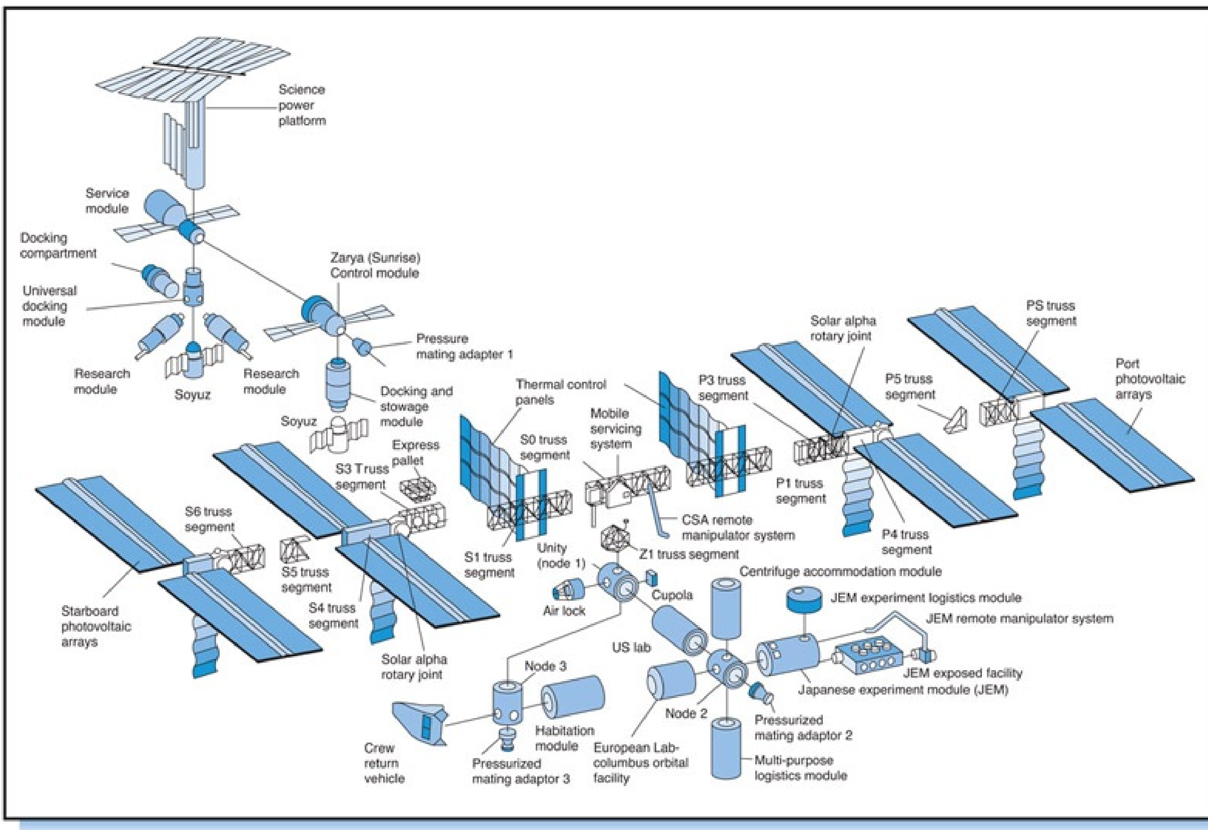


Figure 14.12 Major components in the hardware and assembly of the International Space Station.

Source: NASA

Today the process is expedited by giving integration responsibility to a single "oversight" body, usually the *lead* or *prime* contractor (the "prime"), a role similar to that of a wedding consultant or general contractor, but on a larger scale. The project sponsor is still responsible for contracting with *associate* contractors (subsystem manufacturers), making major decisions, and resolving conflicts between the prime and associates. The associates become subcontractors ("subs") to the prime contractor, take orders from the prime, and are subject to its surveillance and approval. Figure 14.13 shows the relationships among the sponsor, prime, and subs for a large urban transit project.[14]

Sometimes the prime 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 subs are competitors since, understandably, subs 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 aren't big enough. At such times they form teams ("joint ventures") and submit joint proposals where one company serves as leader and takes on responsibility for systems engineering and management of the others. This appeals to small- and mediumsized firms that ordinarily would not have the resources to contract independently. With this approach, however, unless the lead company is strong there could be serious interface problems. But no team is likely to have all the best ideas, and the sponsor may require the lead firm to open up development of subsystems to competitors and, if necessary, change the members of the team.

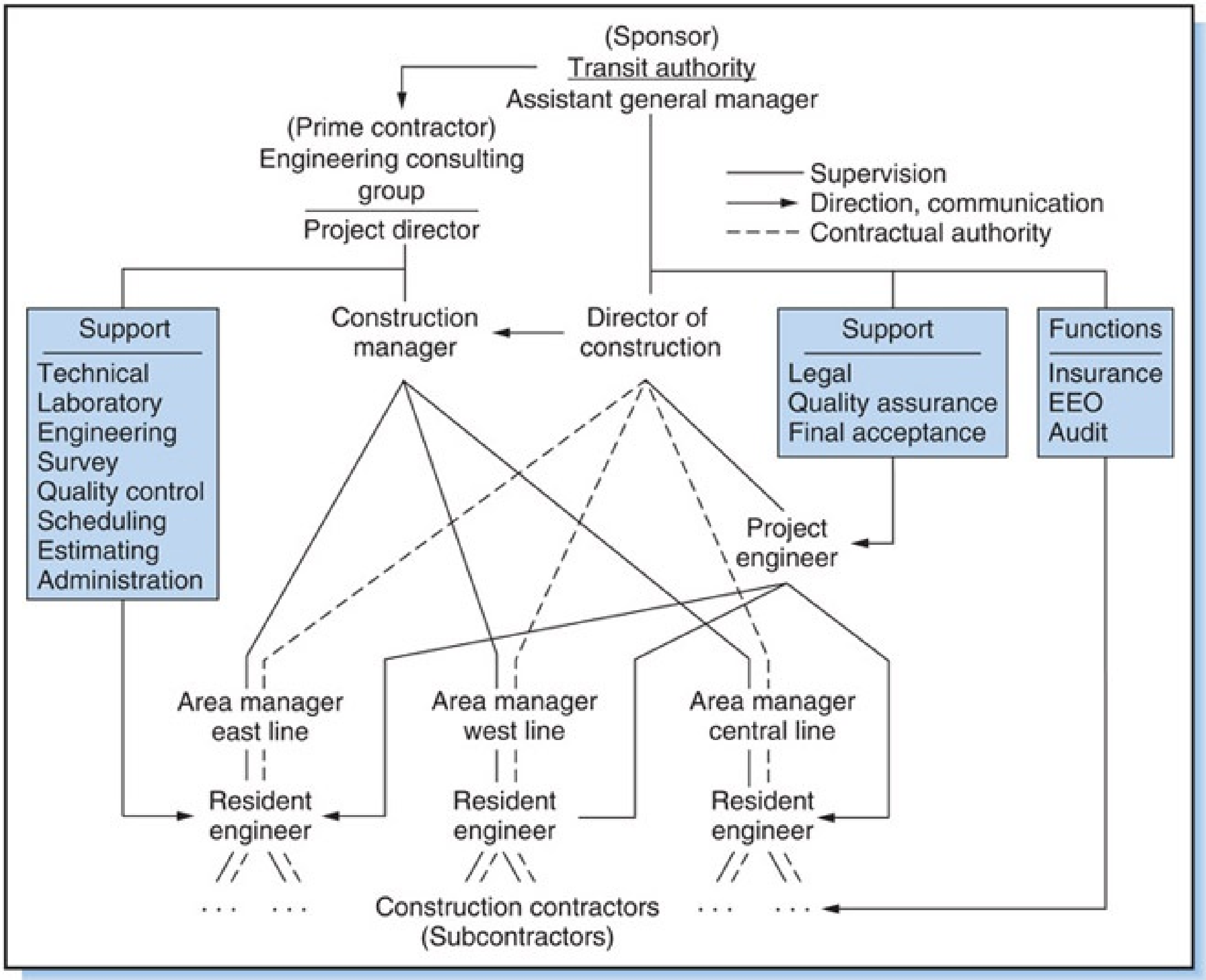


Figure 14.13 Management and authority relationships in a large construction project.

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.

Example 14.3 Corporate Merger—large Scale Non-Technical Project[16]

Special integration management is necessary not only for technical projects but any *large, complex* project. When one of the US's largest pharmaceutical companies 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 purpose of the AIM team was to plan, manage, and execute the integration across 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 corporations. The consulting firm composed the teams, structured the work of the teams, participated in their major decisions, watched over critical path activities, and consulted with the European company's managers and functional teams. By hiring 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.

14.12 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 14.14). The two ways are interrelated because integration of the project phases also usually requires integration of the functional areas working within the phases.

Large-scale projects in product and software development 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 does not always happen.

Nonintegrated (Serial) Systems Development

In a traditional development project, a different functional group is responsible for each project stage. For example, the marketing group specifies the initial concept and user requirements, then the design group defines the technical specifications and creates the system design, then the manufacturing group determines how they will make the product, and so on. Even when a project manager oversees the process, work in each stage remains largely centered in one functional area and minimally involves the other areas. The project is a series of "hand-offs" 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. As a result, the design group must create a product design that conforms to the requirements it inherited from marketing; in turn, the manufacturing group must develop a production process that conforms to the design it inherited from the design group. This serial development process, illustrated in Figure 14.15, involves little interaction and knowledge sharing between the participating groups

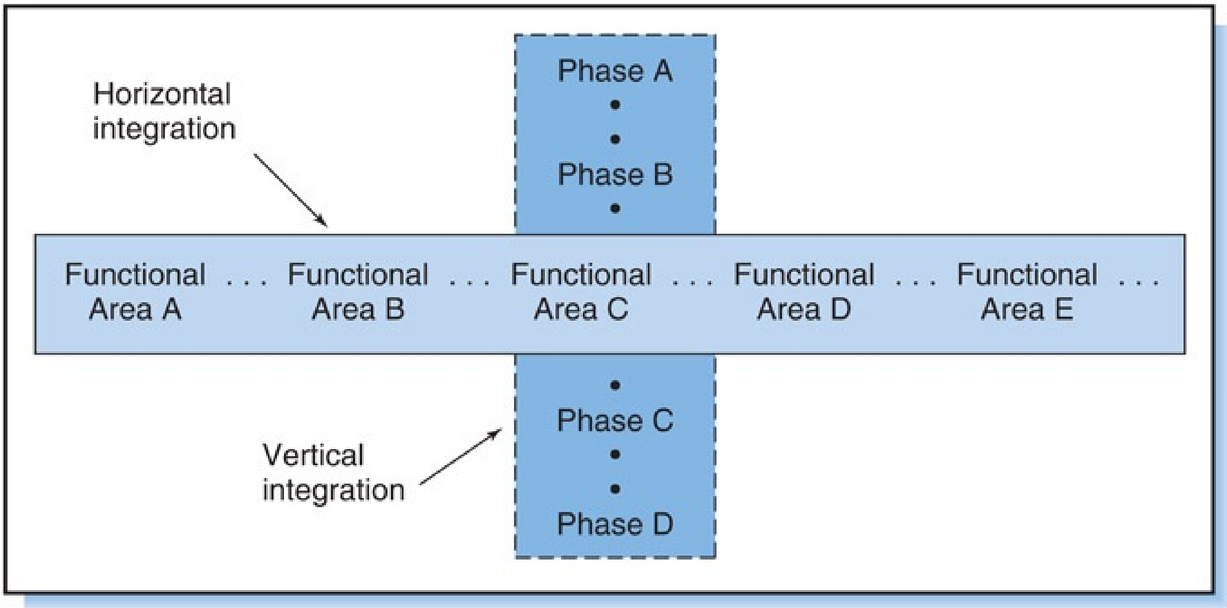


Figure 14.14 Horizontal and vertical integration in systems development projects.

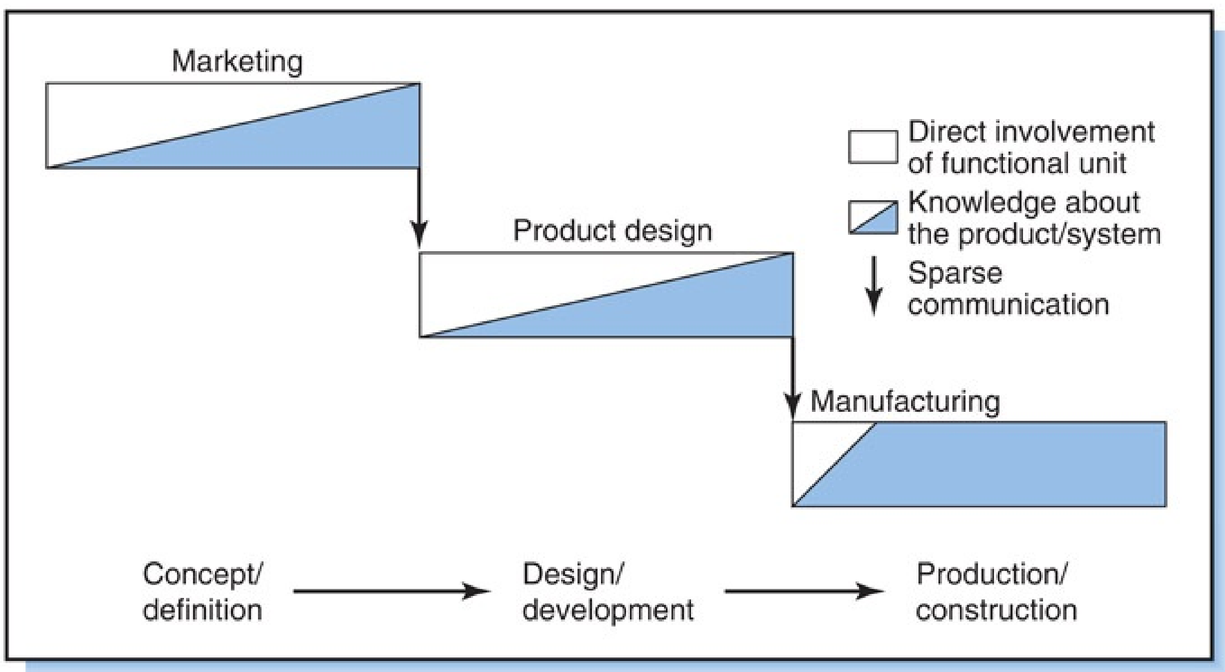


Figure 14.15 Traditional interaction between functional areas during phases of a systems development project.

Source: Adapted from Wheelwright S. and Clark K. *Revolutionizing Product Development*. New York, NY: Free Press; 1992, p.178.

Since 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. As a result, for example, the marketing group specifies a requirement that is not really necessary, but the design group inherits and must incorporate it into the product design; the manufacturing group then inherits the design, which it discovers will be difficult or costly to produce. 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 theother 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.

An integrated approach to systems development is *concurrent engineering.*

14.13 Concurrent Engineering

Concurrent engineering 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. They contribute to decisions long before they actually begin to design, produce, test, or operate the system (Figure 14.16). Unlike serial systems development, hand-offs are nonexistent because all parties have a hand in everything. Horizontal integration and vertical integration are achieved in one fell swoop.

In concurrent engineering decisions about design, development, production, and operation overlap, which greatly reduces the project duration. 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. With concurrent engineering, however, the dies are designed and produced soon after the body panels have been 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.

Concurrent engineering also improves tradeoffs between product features and production capabilities. Working together, product and process designers can discover subtle changes to product design features that would be transparent to the customer but that take advantage of existing production capabilities. The result: lower production costs from fewer production bugs and rework, and fewer customer usage problems and warranty claims.

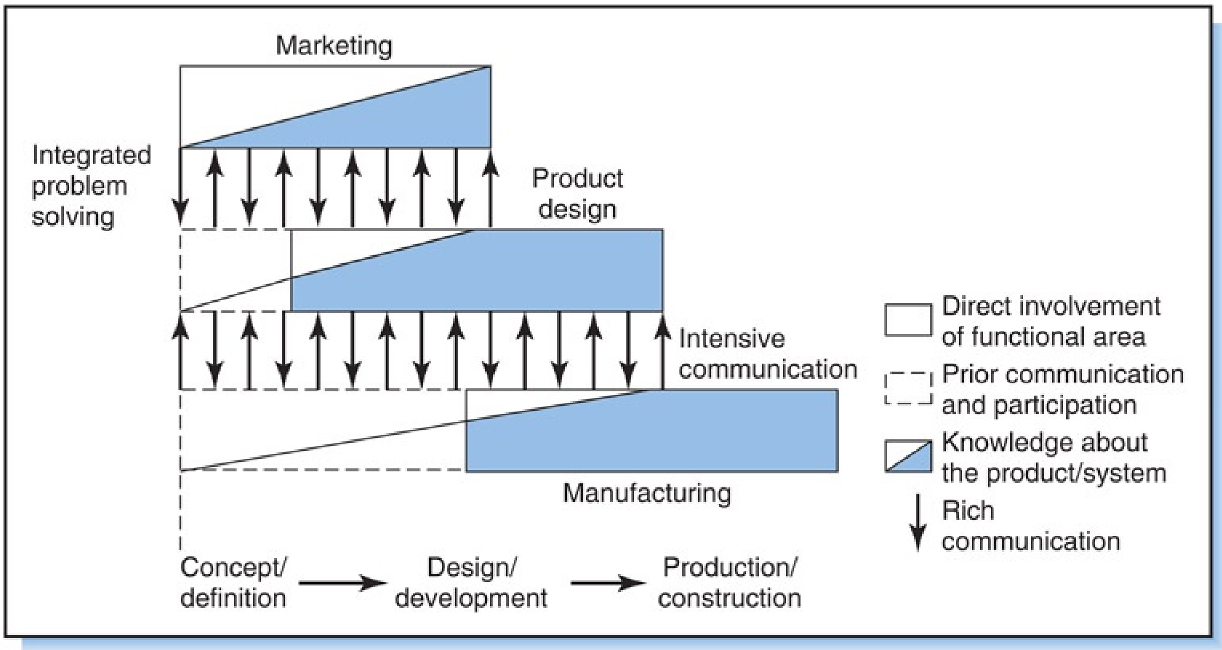


Figure 14.16 Concurrent interaction between functional areas in systems development.

Source: Adapted from Wheelwright S. and Clark K. *Revolutionizing Product Development*. New York, NY: Free Press; 1992, p.178.

Team organization

A concurrent engineering team is organized to maximize team member communication and control over design decisions. This is achieved as follows:[17]

- Autonomy. Members of a concurrent engineering team are relieved of unrelated obligations and commit fully to the development effort.
- Best case is full-time, full-duration. Ideally, members are involved in *all* decisions throughout the *entire* systems development process. That's what "concurrent" means.
- Colocated. 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, and do modeling, assembly work, and whatever else that needs to be done.

These features mirror those of agile teams as described in the previous chapter.

But concurrent engineering is more than just organizing people into a team. 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 auto manufacturers require their designers to spend a full day every few months assembling the portion of the car they helped design.

There are other ways in which organizations organize teams to achieve vertical and horizontal integration. The following example describes two.

Example 14.4 Systems Development at Motorola and Lockheed-Martin[18]

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 pager system, the team consisted of a project manager and eight members from industrial engineering, robotics, process engineering, 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" on the team and offer suggestions, they worked in a glass-enclosed office located in the middle of a manufacturing facility. The team created the work plan for the project, and after gaining senior management approval, took responsibility for performing the bulk of the project. The project was completed 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 the Skunk Works handpicks the project manager and cross-functional team. Unlike the core teams at Motorola, which rely on functional teams for resources and support, 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. Emphasis in Skunk Works teams is on technical excellence and speed. 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 team-work 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, the Skunk Works has become a model for creating highly innovative, leading-edge aircraft and space vehicles quickly and on budget.

See Chapter 10

An example is the F117 Stealth fighter mentioned in Chapter 10.[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 pre-existing aircraft. The project was completed in record time—31 months. The cost for research, development, and production of 59 airplanes was $6.6 billion, considered low at the time when other aircraft programs were running $1 billion over budget. 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 Motorola teams described in Example 14.4 are what Wheelwright and Clark[21] 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, giving them organizational clout 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.

Both cross-functional core teams and autonomous teams provide for strong 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 can understand. 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 design reviews, mentioned in Chapters 9 and 12.

See Chapter 9 and Chapter 12

14.14 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 most 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 startup and shut down.

The matrix organization form creates project teams by sharing members and resources from across functional subunits. It is effective for a creating a continuous stream of project teams, however it can be difficult to implement and induces organizational conflict.

Many companies use a *composite* of 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 project office handles contracting, planning, scheduling, and control, but its major role is integrating functional units. In a large-scale project that involves multiple organizations, integration of all the organizations' 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 must coordinate the efforts of multiple projects, oversight and integration of the projects is handled 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. 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 the traditional forms of differentiation? List some companies that presently use each.
3. List the various forms of integration. Give examples of each. Which of these are “lateral” forms of integration?
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:

    - Project expeditor
    - Project coordinator
    - Project leader in a pure project
    - Project leader in a matrix.

7. Describe the applications, advantages, and disadvantages for each of the following:

    - Project task force
    - Project team
    - Pure project and project center
    - 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 its beneficial aspects be influenced by the project manager?
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:

    - List the various groups, organizations, and individual parties that are involved and must be integrated.
    - 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 of the major difficulties in changing from a traditional nonintegrated development approach to a concurrent engineering approach?

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 design. 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 or a program office? 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? If so, discuss how it was applied.

Case 14.1 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 products and assembles them to satisfy 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 warehousing systems despite its rather limited experience and expertise in that field.

IBC has been awarded a large contract for a robotic 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 at $14.5 million and includes design, fabrication, and installation at the center. IBC was awarded the contract because it was the lowest bid 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 has set up ad hoc project management teams comprised of a project coordinator and members from the functional areas. These teams are then responsible for planning, scheduling, and budgeting the actual work to be done by the functional departments. Team members serve 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 remote terminals, and in its 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 CreativeRobotics to provide the computer and robotics 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.

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 14.2 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 14-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 won't 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.

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 14.3 Implementing a Matrix Structure in an R&D Laboratory[23]

The R&D laboratory of a large Dutch multinational corporation served two roles: product-process development and supporting functional departments; it split these two roles roughly 50/50. The lab's employees were grouped into

13 departments—seven (with 85 employees) mainly devoted to product/process development, eight (with 84 employees) to service support activities.

The decision was made to restructure the laboratory from a functional to a matrix form, and a policy committee was appointed to draft a proposal for the restructure. After a year of discussion, a “balanced matrix” was introduced and five project managers appointed. The functional managers expressed unease over the balanced matrix and suggested instead a “weak matrix,” but they were overruled. Functional managers charged with product/process development complained about loss of operational authority. Managers responsible for support activities had a different grievance: in the past, their work supporting stakeholders outside the R&D lab had always took precedence over development activities, which they performed in whatever time remained. The matrix now changed that, with priority going to development activities with enforced due dates.

The functional managers, who “didn’t feel called upon to cooperate much,” rebelled and ceased making constructive contributions to the projects their departments were involved in. This forced the project managers to attempt to manage the projects single-handedly, which resulted in serious work overloads. Trying to speed up project work, they stealthily bypassed functional managers whenever they visited the functional departments. Further contributing to the rift was the fact that project managers received higher salaries and nicer company cars than the functional managers.

Twice the functional managers requested that some projects in the project portfolio be delegated to their departments. The first time they created a list of 22 big and 26 small projects (“small” defined as requiring less than 1,000 labor hours per half-year, many that involved only one or two departments) and they proposed that the small ones be delegated to their departments. The policy committee 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 coordinating work among them was difficult. They proposed that big projects be divided into sub-projects with responsibility delegated to sub-project managers within the functional departments. This proposal was opposed by the project managers and rejected.

The project managers, frustrated that personnel were often shifted between projects, proposed that personnel assigned to bigger projects should temporarily be *transferred* (rather than shifted) to other departments. They also proposed that people on larger projects be assigned to stay there "semi-permanently," for a period of, say, 6 months, and within that period they would not be reassigned. After a year of deliberation both these ideas were implemented despite the opposition of the support-orientated managers because it compromised their ability to give service requests the highest priority. The production and marketing managers, who wanted quick response to their requests, supported these objections.

Initially, service support tasks that required more than 300 hours were handled by project managers and those requiring less 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 or functional managers, but the project managers suspected that functional managers manipulated the rule by creating service projects such that they usually required less than 100 hours. They therefore proposed that service requests be sent directly to them, but this was rejected.

Three years after initiating the matrix structure, the destructive behavior slowly decreased; disagreements still existed, but the atmosphere improved. Managers of the support-oriented departments admitted that the matrix structure improved objectives-setting and project control, but they still favored a weaker form of matrix over the balanced matrix. Managers of the development-oriented departments came to accept the balanced matrix, although the general manager remained unsatisfied and suggested that all departments should split their staff into two groups, one for product-process development, and the other for support work.

Questions

1. What would the roles and responsibilities of functional managers have been prior to implementation of a matrix structure? How did the the change in roles contribute to conflict after implementation of the matrix?
2. Comment on the complaint of functional managers of support-orientated departments about due dates being enforced for product development work. What possible solutions do you suggest?
3. Many of the projects required the involvement of only one or two departments. Comment on how this fact should have been taken into account 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 for 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 this have been prevented in this case? Comment on the idea that project managers should display "integrative leadership."
7. In this case it took a year of discussions prior to implementing the matrix structure and another 3 years before it started to work reasonably well. Comment on the time you think it should take to implement a matrix structure. What factors must be considered?

Endnotes

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* 1962; 9(3): 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* 1983; 109(4): 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. Ibid., 85.

14. Adapted from Lammie J. and Shah D. Construction management: MARTA in retrospect. *Journal of Construction Engineering and Management* 1984; 110(4): 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* 1994; 35(3): 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, PB. Matrix management of projects and power struggles: A case study of an R&D laboratory. *IEEE Engineering Management Review,* Winter, 1995.

Chapter 15
Project Roles and Stakeholders

All the world's a stage,
And all the men and women merely players.

—William Shakespeare *As You Like It*

When an organization undertakes a project, it forms a pure project team, matrix team, or task force, but unless it is a pure project most people on the team are "borrowed"—they come from functional departments or contractors and work on the project for only as long they are needed. Project management "gets work done through outsiders"[1]—people from various functional and professional groups scattered throughout the parent company and outside subcontractors. As Sayles and Chandler describe, project management

> is dealing laterally, but not in the informal-group, informal-organization sense. It 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]

Part of being a project manager is the ability to influence people without giving orders or 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 and leadership style than traditional managers.

Of course, the project manager is just one of numerous individuals and groups who are involved in, influence, or are influenced by a project—collectively referred to as *project stakeholders*. This chapter discusses the project manager's role, project stakeholders, and the project manager's role in managing stakeholder expectations.

15.1 The Project Manager

Project Manager's Role

The project manager is the centerpiece of project management; she is the glue that holds the project together and the mover and shaker that spurs it on. Being a project manager requires wearing different hats, some at the same time; these hats are those 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.[3]

The project manager is the *communication hub*, the funnel through which most reports, requests, memoranda, and complaints flow. She accepts inputs 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 say that project managers spend 90 percent of their time communicating; not all project managers will agree with that figure, but virtually all will say they spend *most* of their time communicating.

Being the communication hub, the project manager is also 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 is often well-situated to influence others who do make the decisions.

The prime *motivational factor* in any diverse group is strong commitment to a central goal. The successful project manager is able to foster enthusiasm, team spirit, confidence, and drive the team forward, even when the work becomes 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 conceptual phase, she is often the only person who sees the big picture, and whether or not the project gets funded often depends on her ability to gain the endorsement of influential stakeholders.

The project manager is like an *entrepreneur* too, driven to procure the funds, facilities, and people needed to get the project off the ground and keep it flying. She must win over reluctant stakeholders who question supporting or assigning resources to the project. After the work is underway, she must continue to champion the project and sometimes fight for its existence. In the end, whether the project succeeds or fails, the project manager is ultimately held accountable.

Example 15.1: Gutzon Borglum: Project Manager and Sculptor[4]

If you are familiar with the carvings pictured in Figure 15.1 then you know the handiwork of Gutzon Borglum. More than two million people a year visit Mount Rushmore National Memorial. Most 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. 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 have to 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 rested or slept. Of course, he was by no means perfect; he did not always have project problems under control, and his efforts in the early years were criticized for being unorganized. When the project began in 1927, Borglum wasn't completely sure what the monument was going to look like. People familiar with Borglum 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."[5] 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.

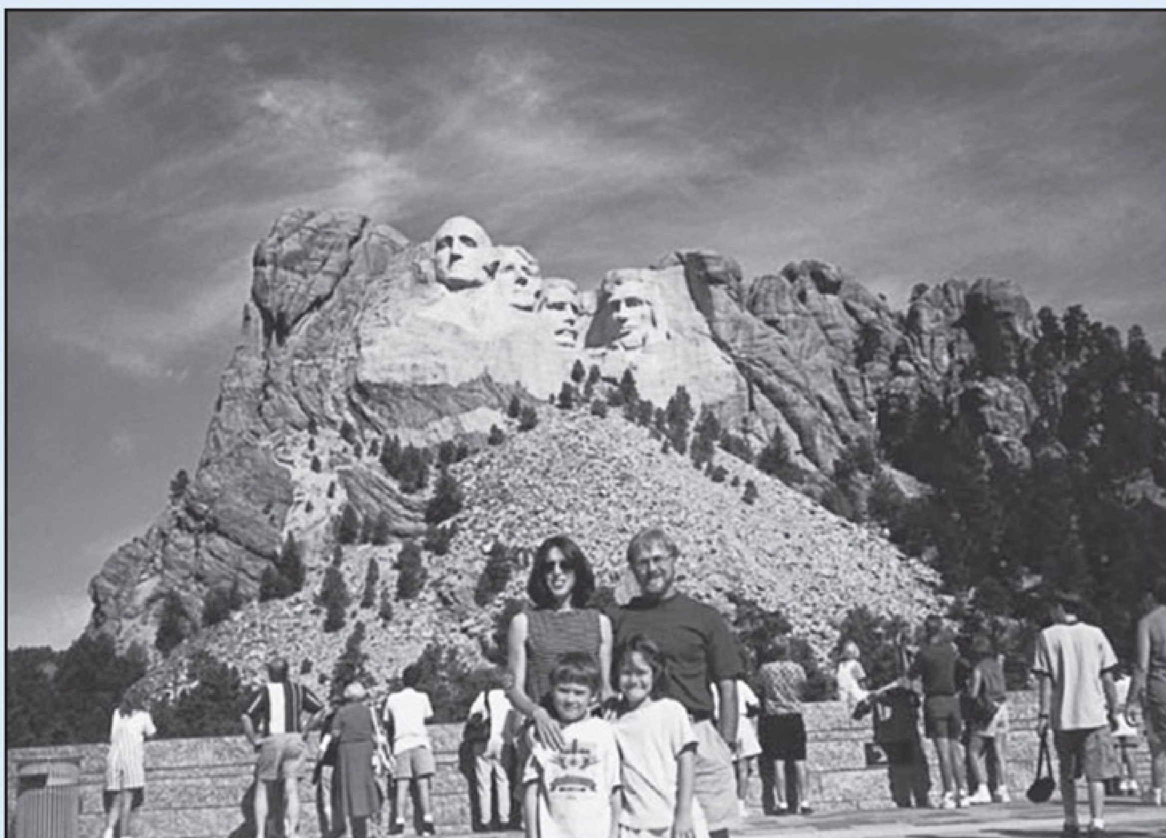


Figure 15.1 Gutzon Borglum's most famous work attracts millions of visitors a year.

Photo courtesy of John Nicholas.

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."[6]

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. In fact, 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. Just months before the project was canceled, Borglum died. He had been the project's prominent driving force, 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 the project end-item in accordance with requirements and contract terms and, when specified, in fulfillment of profit objectives. Other, specific responsibilities vary depending on the size and nature of the project, the stage of the project, and the 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.

Though responsibilities vary, they usually include:[7]

- Planning project tasks and end results, creating the WBS, schedule, and budget, coordinating tasks, and allocating resources.
- Selecting and organizing the project team.
- Working and negotiating 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.
- Solving problems directly or knowing where to find help.
- Dealing with crises and resolving conflicts.

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 errors, problems, or overruns.

Domain Competency and Orientation

Project managers work at the *interfaces* between top management, the customer, and technologists or contractors, so they must have managerial ability, technical competency, and other broad qualifications. They must feel as much at home in the office talking with executives and customers about policies, schedules, and budgets as in the plant, shop, or on-site talking to subject matter experts and supervisors about technical matters.

Broad background is also essential. The more differentiated the functional areas, the more prone they are to conflict. To effectively integrate multiple, diverse functional areas, the project manager needs to understand each of the areas—the jargon, techniques, and procedures used—and their contributions to the project. Referred to as *domain competency*, the project manager must have a good understanding of *all areas* of the project. Another way of saying this is that 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 work breakdown structure. Although most project managers cannot be experts in all areas of the project, they must be sufficiently familiar with the areas to be "credible"—to intelligently discuss issues, ponder ideas offered by specialists, and evaluate and make appropriate, balanced decisions. Along the same lines, to deal effectively with top management and the customer, they must be familiar with the workings and businesses of the parent and customer organizations.

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. As a rule of thumb, a project manager tries never to bluff since doing so risks losing all credibility with stakeholders.

Studies indicate that the most effective project managers adopt goals, time, and interpersonal orientations *intermediate* to the functional units they integrate. In other words, they take a balanced outlook.[8] For instance, to integrate the efforts of a production department and a research department, the project manager adopts a time perspective intermediate between production's short-term, weekly outlook and research's long-term, futuristic outlook.

As far as the relative importance of technical ability versus managerial competency, that depends on the project. Project managers in most research and engineering projects need greater technical competency because the problems and the orientation of the project team are technical. In non-technical projects, however, project managers need greater managerial ability because the projects involve multiple, diverse functional areas. In general, project managers must be sufficiently technical to be able to understand project issues, but not so technical as to neglect their managerial role. There is no substitute for strong managerial competency in the role of the project manager.

15.2 Project Management Authority

Authority refers to a manager's power to command others to act or not to act. There are different kinds of authority, the most familiar is that conferred by the organization and written in the manager's job description, called *legal authority*. Given legal authority, people in higher organizational positions are viewed as having the "right" to control the actions of people below 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, personality, and appearance. People both inside and outside the formal organization can increase their authority by being charismatic.

Traditional Authority

Management theory says that authority is always greater at higher levels in the organization and is delegated downward. This is presumed the way it ought to be because managers at higher levels are assumed to "know more" and, therefore, be able to make decisions and "command" workers at lower levels. This point has been challenged on the grounds that managers, particularly in technology-based organizations, cannot possibly know everything needed to make complex decisions. They often lack technical expertise and so, increasingly, must rely upon subordinate specialists for advice. Even managers who are technically skilled cannot always manage alone; they rely upon staff groups for personnel and budgetary assistance. Especially in projects, this aspect of "participatory management" (described in Chapter 16) is commonplace.

See Chapter 16

Influence

It is important to distinguish between legal authority and the *ability to influence.* Managers with legal authority influence subordinates by giving orders and controlling salaries and promotions. Generally, however, the most effective managers are able to exert influence *without* "ordering" others or making an issue of their superior–subordinate relationship. This is especially true when subordinates are well-educated or highly experienced. In fact, managers who rely solely on legal authority are often relatively ineffective. Effective managers tend to rely instead on two other sources of influence: *knowledge* and *personality*.[9] The first source, called *expert power*, refers to a special level of knowledge or competency; subordinates believe that the person possesses knowledge and information that is important and that they themselves do not have. Simply, the expert power holder is viewed as being right because he knows more, and others readily defer to his requests.

The other, called *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 his requests.

Given expert power and referent power, a person can influence others irrespective of the formal hierarchy. These forms of power can even subtly reverse the authority relationship. A subordinate may exert considerable influence over her superior 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

Functional managers tend to rely on different forms of influence—knowledge, expertise, persuasion, and personal relationships; when these fail, however, they are able to fall back on their legal authority. But project managers are not able to 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. The project manager exists *outside* the traditional hierarchy. His role is temporary, superimposed on the existing structure, and is not afforded the leverage inherent to a hierarchical position. Project managers work across functional and organizational lines and, except 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 must share authority with functional managers.

Thus, despite the heavy responsibility they carry, most project managers lack a comparable level of formal authority. Instead they have *project authority*, meaning they can make decisions about project objectives, policies, schedules and budgets, but lack authority to give orders backing up those decisions.

The disparity between high formal responsibility and low formal authority has been referred to as the *authority gap*.[10] 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.

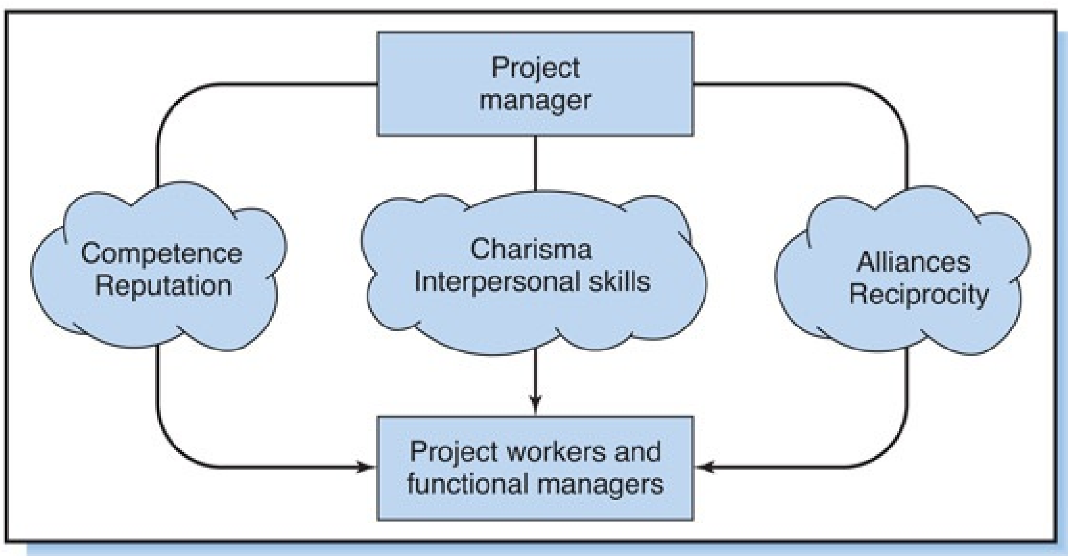


Figure 15.2 Project managers' sources of influence.

Project Manager's Authority

Most project managers handle the authority gap in similar ways: having no legal authority, their recourse is to rely on influence derived from expert power and referent power. They have to do this also because no matter the project, they depend on others to get the job done. Numerous decisions must be made, many for which they have neither the time nor expertise. Even in the rare case of a project manager having legal authority, she seldom uses it because unilateral decisions and orders are inconsistent with the need for reciprocity and tradeoffs in projects. Project managers know that not all information need be channeled through them, and they encourage direct informal contact between the individuals involved.

Project managers also gain influence through networks of alliances and informal connections they build with managers and other stakeholders. The strength and breadth of these networks increase with the project manager's perceived competency, reputation from prior project accomplishments, and charisma. The final feature, charisma, 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 associate with, 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 alliances and friends to call upon for help.

In summary, project managers tend to rely upon knowledge, experience, personal relationships, and personality to influence others (Figure 15.2). 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 ways that different project managers employ these sources of influence is illustrated in Example 15.2.

Example 15.2: Effective Project Managers, Contrast in Styles[11]

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 also in the whole 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 people with whom he had dealings sweat—bureaucrats and engineers—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 the bad temper and somewhat unkempt appearance 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 re-compute 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, therefore, would rely on his teams for most decisions. His first move was to loosen the reigns and allow the teams latitude 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 stuck to schmoozing and cheerleading through an endless supply of one-liners. As one employee said, "Whereas Kelly ruled by his bad temper, Ben ruled by his bad jokes." Ben used a non-threatening approach. 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.

Johnson and Rich led using different styles, yet both have been acknowledged by the industry 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, no technical slouch, acknowledged that he had some smarter people working for him. Unlike Kelly, however, he had charisma and many personal friends, and with that he too was able to accomplish great things at the Skunk Works.

The Balance of Power

Typically project managers and functional managers share authority, but in high-performing projects that authority is clearly differentiated.[12] Project managers have power to decide *what* must be done, procure resources, coordinate work efforts, and mediate conflicts; in contrast, functional managers have power to decide *how* things must be done—the technology to be used—and to resolve technical problems.

Although a project manager rarely has any form of reward power, workers tend to listen more if they *perceive* the project manager has some influence over their salaries and promotions.

15.3 Project Manager Qualifications

Qualifications of successful project managers fall into four categories: personal characteristics, interpersonal skills, general business skills, and technical skills.

Personal Characteristics

Archibald lists the following as essential personal characteristics:[13]

- 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 close outs, 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 he connects with people, stays informed, and builds morale.

Interpersonal Skills

A project manager needs strong behavioral and interpersonal skills and ability to "actively listen to" and "read people."[14] Active listening means asking questions 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 project stakeholders' attitudes. Specialists on the project team often disdain anything non-technical and resent schedule and budgetary constraints. The project manager must be able to convince them why these matter.

The project manager must also know how to build trust, promote team spirit, and reward cooperation; often she does this through the only forms of reward she is able to give: praise and credit. A good project manager understands the personalities, attitudes, and strengths of her team members and knows how to utilize their talents, even when they do not measure up to her standards. She is sensitive to human frailties, needs, and greed, and able to resolve conflict, manage stress, and coach and counsel team members. It seems like a tall order, but a good project manager can do all of that.

General 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 cost 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 the phasing and scheduling of work to meet delivery dates, so they must be familiar with the work tasks, processes, and resources in executing the project. And they are responsible for enforcing schedules, hence must be knowledgeable about tools and techniques for planning, tracking, and control.

Technical Skills

To be able to make informed decisions, project managers must have a strong grasp of the technical aspects of the project. As mentioned, their "domain competency" must span the full scope of the project. In non- or low-technology projects, 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 science or engineering.[15]

Although project managers seldom do technical analysis, they must be qualified to integrate concepts from different disciplines and make technical judgments. Often technically-qualified people are not good at integrating concepts from different specialties because undergraduate training in engineering and technology typically emphasizes analysis and ignores integration.[16] The project manager must be able to understand and speak the language of all the specialists on the team, regardless of their specialty; this is minimally necessary for communicating with and integrating the work of specialists.

Selection and Recruiting

The manager to head up a given project is selected from among the ranks of product and functional managers, subject-matter specialists, and experienced project managers. The last source is the best though not always feasible. It might be difficult to find an experienced project manager who has the right mix of qualifications and is available for the new project. As a result, when an experienced project manager is needed often he is recruited from the outside; this is readily observable in job listings online and in major newspapers (Figure 15.3 shows a sampling). The downside with hiring an outsider is that it will take time for him to make friends, build alliances, and learn organizational policies. On the plus side, he is likely better suited to objectively take on the task (without political influence), and will not have any enemies—at least initially.

The project manager can also be selected from among functional managers, although functional managers sometimes have difficulty shifting to a project perspective, which requires switching from managing just one area to overseeing and integrating the work of many areas. Unless he has abundant well-rounded experience, everyone will likely perceive him as just another functional manager.

Project managers can also be “created” by promoting subject-matter specialists (engineers, scientists, system analysts, product designers, etc.), although this has the same drawback as with placing any non-manager into a managerial role: he has to first learn how to manage. Being a good engineer or auditor is no assurance the person will be a good project manager. In addition, the specialist must learn how to remove himself from his 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 and give him authority over his former boss.

Training

Project management skills cannot be learned quickly, so organizations devote substantial time and expense in 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. In recent years there has been a proliferation of both kinds of programs, accompanied by a rise in training oriented 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.

Figure 15.3 Advertisements for project management positions.

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.[17] As part of their career paths, they rotate assignments throughout all areas of the organization to develop sufficient domain competency to enable them to manage projects that involve those areas. Technical specialists work full- or part-time as assistants to experienced project managers, and 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 15.3: On-the-Job Training of Project Managers[18]

Microsoft Corporation's approach to preparing project managers (which they term "program managers") is typical. There is neither an official training program for 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 videotaped luncheons where managers present their experiences, and then circulates the videotapes.

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 the role is eased when the project manager:[19]

- Understands what has to 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 is able to do well, and where he falls short.
- Is aware of what can and should be done to correct a bad situation.
- Believes that his superiors have an interest and believe in him, and are eager for him to succeed.

In ideal cases senior management provides all of these to the project manager; sometimes, however, it is up the project manager to seek, request, or demand these.

15.4 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. The most effective project management role occurs when one person becomes involved during proposal preparation and remains until project completion. When no one is available or competent enough to manage the project, the role is filled in other ways. For example, it may be filled by the general manager or plant manager, though these managers usually have neither the necessary 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. This “two-hat” role has other problems as mentioned in Chapter 14.

See Chapter 14

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 managerial or technical continuity from one stage to the next (to integrate the stages). The 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. This happens 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, his 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; only occasionally are they the same person. Because the shooting of a motion picture is an artistic pursuit, directors need flexibility in budgets and shooting schedules, but costs matter too, and the producer faces the question of “at what price creativity?” It is no surprise that the two do not always have a happy relationship.[20] 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 manages 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.

Although ideally the person filling the project management role devotes 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 his responsibilities to 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.

15.5 Roles in the Project Team

Early in a project the project manager and functional managers divide the overall project into work packages. This division determines skill requirements and serves as the basis for personnel selection and subcontracting. Those in functional support areas, contractors, and the project office who will contribute to the project become part of the *project team*. This section describes roles in the team.

Members in the Project Office

Chapter 14 described the purpose of the project office. An example project office (not to be confused with a project *management* office or PMO) for a large engineering-development project is shown in Figure 15.4. This section describes the members of the project office.

See Chapter 14

The *project engineer* (aka systems engineer or systems designer) shoulders responsibility for coordinating technological areas and assures integrated design of the end-item. When several functional areas or subcontractors are involved, the project engineer:[21]

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.

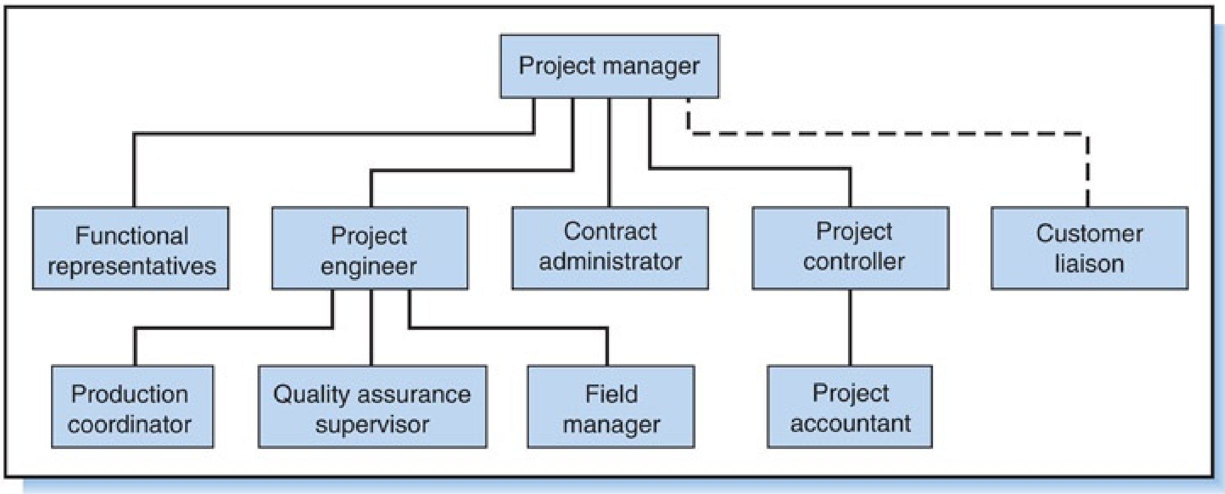


Figure 15.4 Members of the project office in a large engineering-development project.

Sometimes the title “project engineer” denotes a person having full project manager responsibilities, although more commonly it refers to the more limited role described here.

The *contract administrator*[22] is responsible for project legal aspects such as authorization to begin work and subcontracting with outside firms. The administrator assists in preparing proposals, defining and negotiating contracts, integrating contract requirements into project plans, ensuring fulfillment of contractual obligations, and monitoring and communicating to the customer changes to project scope. During closeout, he notifies the customer of fulfilled obligations, documents customer acceptance of the end-item, and initiates formal requests for payment. He is also responsible for collecting and storing RFPs, correspondence, legal documents, contract changes, bills, payment vouchers, and related documents.

The *project controller*[23] 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 completion.

The *project accountant* provides accounting support to the project. He establishes procedures for using the PMIS, 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 ongoing reviews (within the bounds of 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 all 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 handing 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 encompass raising quality awareness and instituting 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. They work in and charge their time to the project office whenever they must meet with the project manager and the representatives from the other areas, and return to their functional departments as soon as their work has ended.

The number of staff in the project office should be as small as is practical. This keeps the office flexible, minimizes personnel costs and assignment problems, and is simpler for the project manager to manage. Members of the office staff contribute full- or part-time as needed and might be physically located in different places.

Functional Managers

Often the glamour of work sits on the project side, and functional managers may perceive their roles as 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. Functional and project managers depend on one another to achieve project goals. Functional managers are responsible 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, a functional manager is more likely to look out for the interests of the people being solicited.

In most project organizations, functional managers retain much the same authority and responsibility as in non-project environments. Nevertheless, some functional managers believe that the project manager role undercuts their authority and that they could handle the project better if it were exclusively within their domain. Project managers who try to undermine the authority of functional managers will have difficulty obtaining support when they need it (see, for example, Case 14.3 in Chapter 14).

Before a project begins, the responsibilities and contributions to technical content for each functional manager should be clearly delineated.[24] This will ensure a continued strong technical base for all projects and alleviate potential animosity between functional and project managers.

See Chapter 14

Project Functional Leaders and Work Package Supervisors

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.

15.6 Roles Outside the Project Team

This section discusses some of the roles of individuals and groups outside the project team who influence or control the management, resources, and outcomes of a project (Figure 15.5).

Manager of Projects or PMO Director

The *manager of projects* (also called the PMO director, vice president of projects, or director of projects) is at the same level in the hierarchy as functional managers (see Figure 14.8 in Chapter 14). This manager oversees multiple projects and:[25]

See Chapter 14

- Directs and evaluates all the project managers.
- Ensures that projects are consistent 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.

Chapter 17 describes the role of the PMO more fully.

See Chapter 17

Top Management

Top 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 allocation of resources, this responsibility is headed by the project portfolio manager, discussed in Chapter 18.

See Chapter 18

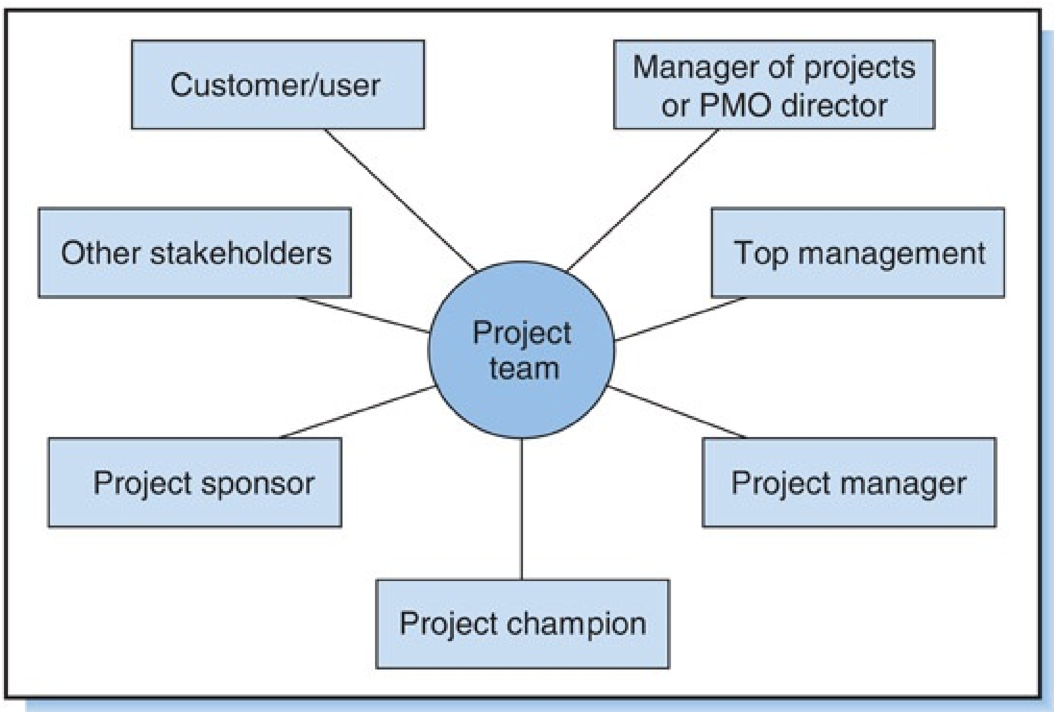


Figure 15.5 Roles outside the project team.

Top management establishes the rules and policies that govern the organization's project management. Directly or through the PMO, it:[26]

- Defines the project manager's responsibility and authority relative to other managers.
- Defines the scope 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 only as granted 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.

Program Manager

When the project is part of a larger effort called a *program*, the project manager works with a program manager who is responsible for coordinating the project with other projects and work efforts to achieve program goals. The role of program manager is covered in Chapter 17.

See Chapter 17

Project Supporters: Champion and Sponsor

Every project needs the support of two key outsiders, the *project champion* and the *project sponsor*. The champion is someone who firmly believes in the project and argues in its favor, both at its inception and thereafter. The champion must have "clout" and be the sort of person able to convince stakeholders of the project's value or benefits. The champion is often the project's most visible spokesperson. When the project does not have a champion, the project manager might have to put on his "evangelist" hat and go scout for one.

The project sponsor is someone who works to ensure that the project gets the necessary priority, funding, and resources. Like the champion, this person is influential, although more in terms of formal authority to clear roadblocks and influence top management's decisions; as a consequence, for projects not performed for external customers, she often holds a position in top management. The sponsor ordinarily does not actually devote much time to the project but is nonetheless accessible to the project manager and available to help whenever the project hits a snag and needs high-level assistance. When the project is part of a program, sometimes the program manager serves as the project's sponsor. Occasionally the champion and sponsor are the same person.

15.7 Project Stakeholder Engagement[27]

The term "stakeholders" appears frequently throughout this chapter and everywhere in the book. That's because stakeholders are key players in projects and perform a critical role in project management. By definition, a *project stakeholder* is any group or individual affected by, interested in, or having potentially influence on the project. Among the most important stakeholders are customers and users as discussed elsewhere in the book, and the project-related roles discussed in sections above 15.5 and 15.6. Other stakeholders include prospective customers/users, partners, lenders, governments, the press, and trade groups. With growing recognition that projects have widespread—even global—environmental impacts, project stakeholders potentially also comprise everyone concerned about or affected by the project's environmental impacts, including everyone in the larger community, society as whole, and, for that matter, all of Earth's living organisms.

Some stakeholders support the project and want it to succeed; others resist it and want it killed. The latter include managers of areas that compete with the project for resources, environmental or political interest groups or lobbies, and anyone who perceives the project as detrimental to their own or society's interests. Most stakeholders are not aware of and don't care about other stakeholders. Before a project starts the project manager needs to learn who the stakeholders are. In essence, he 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. This is part of *stakeholder engagement*, which begins with learning who the key stakeholders are, understanding their interests, needs, and attitudes regarding the project, and preparing strategies to accommodate them. To do that usually requires talking directly to stakeholders to learn their views and opinions, what they hope for from the project (and project manager), what they need (explicit requirements) and expect (unstated or undefined requirements), and how they might be influenced. Given limited resources, technical capability, or other stakeholders' demands, not every need and expectation can be met, in which case the project manager does the best she can. Sometimes she does what she does simply because it is the "right" or ethical thing to do.

Example 15.4: 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; as it rises, it blocks stunning views the loft residents once had of the river and skyline—a common problem in cities wherever one building goes up next to another. To acknowledge their annoyance, Chris arranged every morning for coffee, roles 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. 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.

Every morning construction on the new building resumes at 5 am and, with it, bright lights and cacophony. To remind him of her irritation, every morning Hilda leaves a message on Chris's cell, something like, "Here it is 5 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 there isn't more he can do to make things better for her.

The list of stakeholders may be long, and the amount of communication with, information provided to, and interaction with each should depend on their interests in the project and power or ability to influence the project's outcomes. The table below shows appropriate strategies for dealing with stakeholders depending on their level of interest and potential level of influence. For example, stake-holders with high influence but low interest need to be "kept satisfied," lest they become opposed to the project if they become unsatisfied. The chosen strategy should address whether the stakeholder's "interest" is supportive, opposing, or neutral, how the project manager will engage the stakeholder throughout the project, as well as the stakeholder's communication preferences (face-to-face, phone, email, etc.). For stakeholders opposing the project, the strategy should specify how to gain their support or, failing that, how to mitigate or accommodate their opposition. (As Example 15.4 shows, the strategy might be to try to satisfy *all* stakeholders with high interest, even those with low influence.) All of this can be included in a *stakeholder engagement plan* that lists the stakeholders and for each their interests and level of influence, and strategies for communicating with, engaging, or otherwise dealing with them. The plan should be reflected in the project communication plan and the project execution plan.

	Stakeholders with low influence on the project	*Stakeholders with high influence on the project*
Stakeholders with high interest in project	Keep informed	Focus on these: keep informed and satisfied
Stakeholders with low interest in project	Possibly ignore; monitor for changes in interest/influence	Keep satisfied

Example 15.5: The Big Dig[28]

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 numerous interest groups (Figure 15.6).[29]

The central artery portion of the project replaced the 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—a total of 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; the project eventually cost over four times that amount.

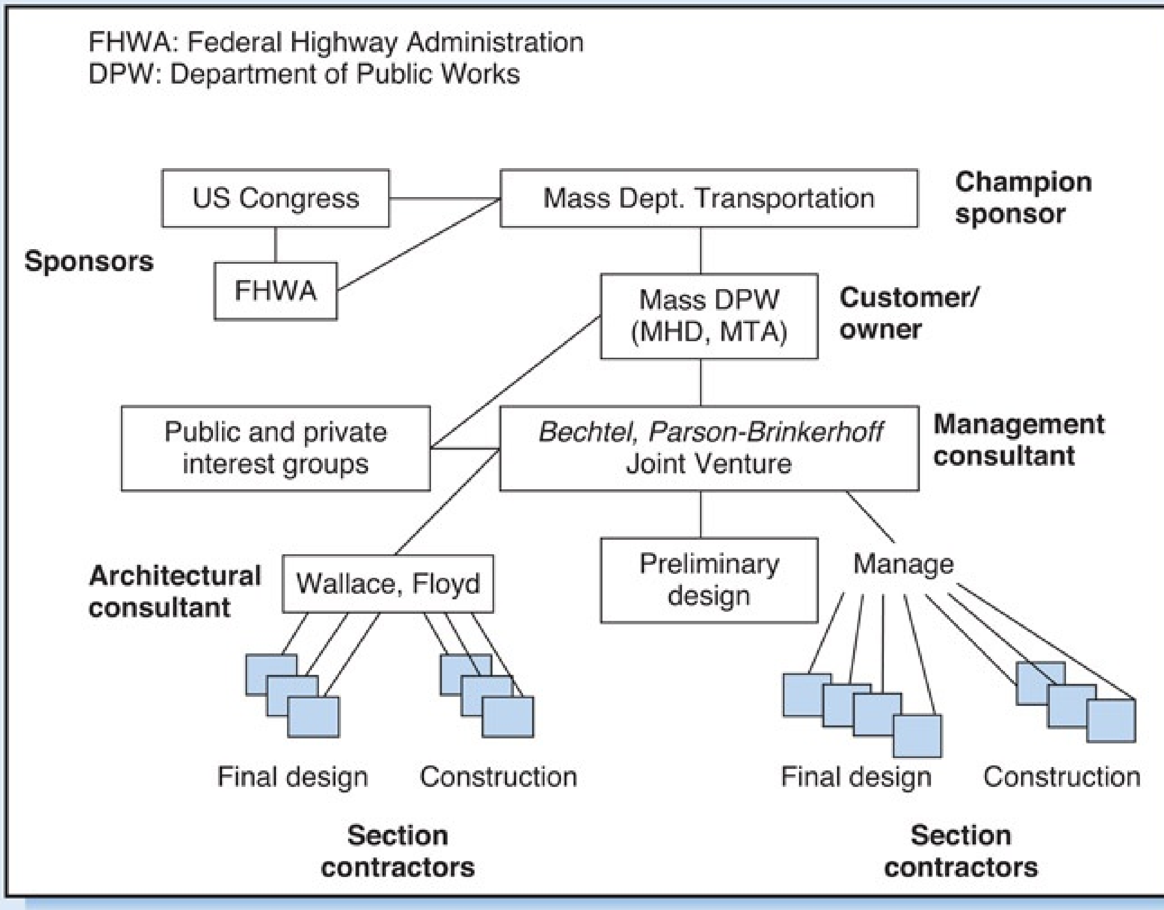


Figure 15.6 Key stakeholders in the Big Dig project prior to 1992.

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 taking into account the interests of—and making promises to—a large host of ad hoc congressional allies. With 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, DPW was given the job on the rational that MBTA is a transit, not a highway, agency. To manage the project, DPW hired the experienced contractor team of Bechtel/Parson Brink-erhoff, 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 MGM 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 neighborhood, church, business, 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.

Getting a project off the ground involves negotiating hoops and hurdles posed by stakeholders. The project manager is always mindful of those stakeholders and works to gain and retain their support in ways big and small.

Example 15.6: McCormick Place West[30]

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 became 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 needy family. These and other charitable programs benefited the local community and helped the contractor to gain the community's support.

15.8 Summary

Project managers work at the project–functional–customer interface, integrating project elements to achieve time, cost, and performance objectives. They have ultimate responsibility for the success of projects, 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. They also have strong behavioral and communication skills and are able to function effectively in uncertain, changing conditions.

The role of project manager is best filled by one person who is involved in the project from 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 subcontractors. The project office provides administrative assistance. 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 play key roles in the project. Top management establishes 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 organization 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 gets the project the needed priority and resources.

Numerous other stakeholders support or resist the project and can 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.

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 dealing with groups and individuals. These are covered in 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, 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? What should the project manager do to “manage” stakeholders and their expectations?

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 15.1 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 13.1, prepare a list of qualifications—general background and experience, personality 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 expediters, 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 13

Case 15.2 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 likely require no less than a *culture 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 be a good opportunity for her to learn.

Assume that you must act on the information available in the case: if it was your choice, who would you select to manage the project: Noware, Roberta, or someone else? Explain.

Case 15.3 Stakeholders in Boston's Big Dig[31]

(Refer to Example 15.5) 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 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 "economic 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 persons 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 15.4, create a list of the project's stakeholders. Revise Figure 15.6 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?

Endnotes

1. Sayles L. and Chandler M. *Managing Large Systems: Organizations for the Future*. New York: Harper & Row; 1971, p. 204.

2. Ibid., p. 212.

3. Ibid., p. 204.

4. Portions adapted from Smith R. *The Carving of Mount Rushmore*. New York: Abbeville Press; 1985.

5. Ibid., pp. 17–18.

6. Ibid., pp. 164.

7. 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, p. 34; and Roman D. *Managing Projects: A Systems Approach*. New York: Elsevier; 1986, p. 419.

8. Lawrence P. and Lorsch J. *Organization and Environment: Managing Differentiation and Integration*. Boston: Graduate School of Business, Harvard University; 1967, Chapter 3.

9. 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.

10. Hodgetts R. Leadership techniques in the project organization. *Academy of Management Journal* 1968; 11: 211–219.

11. Rich B. and Janos L. *Skunk Works*. Boston: Little, Brown & Co; 1994.

12. Katz R. and Allen T. Project performance and the locus of influence in the R&D matrix. *Academy of Management Journal* 1985; 28(1): 67–87.

13. Archibald. *Managing High-Technology Programs*, p. 55.

14. Adams J., Barndt S. and Martin M. *Managing by Project Management*. Dayton: Universal Technology; 1979. p. 137.

15. Gaddis P. The project manager. *Harvard Business Review*; May–June 1959, 89–97.

16. Ibid., 95.

17. Roman, *Managing Projects*, pp. 439–440.

18. Cusumano M. and Selby R. *Microsoft Secrets*. New York: Free Press; 1995, pp. 105–106.

19. Kerzner H. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. New York: Van Nostrand Reinhold; 1979, p. 99.

20. 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.

21. Responsibilities for project engineers are described in Chase W. *Management of Systems Engineering*. New York: Wiley-Interscience; 1974, pp. 25–29.

22. According to Archibald, *Managing High-Technology Programs*, pp.124–128, 199.

23. Ibid., pp. 128–131.

24. Katz and Allen, Project performance and the locus of influence, pp. 83–84.

25. Cleland D. and King W. *Systems Analysis and Project Management*, 3rd edn. New York: McGraw-Hill; 1983, p. 358.

26. Ibid., pp. 362–363.

27. See Schibi O. *Managing Stakeholder Expectations for Project Success*. Plantation, FL: J Ross Publishing; 2013; Roeder T. *Managing Project Stakeholders*. New York: Wiley; 2013.

28. 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_tri

29. Figure 15.6 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.

30. 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.

31. Source: Hughes T. *Rescuing Prometheus*.

Chapter 16
Managing Participation, Teamwork, and Conflict

Eh! je suis leur chef, il fallait bien les suivre.
Ah well! I am their leader, I really ought to follow them!

—Alexandre Auguste Ledru-Rollin

Teambuilding. We don't need that!
I'll skip this chapter.

—Anonymous Project Manager

During the manned landings on the moon, researcher Richard Chapman conducted a study of NASA project management.[1] This was during NASA's heyday—a period marked by extraordinary achievements and a time when NASA was upheld as exemplar of a large 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 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. Kloman 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 Lunar Orbiter's customer/contractor organizations 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] Kloman 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.
>
> (p. 359)

These remarks are the crux of this chapter: behavioral issues such as decentralized decision making, interpersonal skills and teamwork are important in project management. Unfortunately they are often overlooked in project management practice and education, possibly because inexperienced managers and specialists in the "hard" disciplines (technicians, engineers, and business people) 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 the two studies cited: participative decision making, team-work, conflict resolution, and the related matter of emotional stress in work.

16.1 Leadership in Project Management

Leadership Style

Chapter 14 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-oriented* and *relations-oriented*. Task-oriented managers show higher concern for the goal and the work and tend to behave in a more autocratic fashion. Relations-oriented managers show greater concern for people and tend to behave more democratically.

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, and the nature and environment of the task. This perspective, called the *contingency* or *situational approach* to leadership, suggests that the leader should apply the style that best fits the situation and 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.

See Chapter 14

Contingency and Situational Approaches

According to Fiedler,[3] 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. A project manager might encounter any of these situations, although commonly:

- The project manager gets along with team members and is respected for his ability and expertise.
- The task is relatively complex and requires a good deal of judgment or creativity.
- The project manager has relatively low formal authority.

Fiedler's research indicates that under these conditions 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 Blanchard[4] developed a model called *situational leadership* that weighs the interplay of three variables: (a) the amount of direction and guidance a leader gives (task behavior), (b) the amount of socio-emotional support he gives (relations behavior), and (c) 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 highly trained 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 facilitating, supporting, and communicating with followers; the leader shares decision making with followers. For Group (2), the model recommends a *delegating* style; i.e., the leader identifies the problem or goal, then delegates to followers responsibility for solving the problem and determining how and where 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 necessary capabilities 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 will not take initiative. In cases where participation or delegation fails, a project manager with legal authority may need to cajole, give orders, and even terminate workers.

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; i.e., 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.

16.2 Participative Management

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—supportive, facilitative, and encouraging. Sometimes project managers must give orders or tell people what to do, but in *most* project situations delegation works best, even when combined with task-oriented behavior.

This conclusion is further supported by research in large aerospace projects showing that the most effective leadership style is *participative management*. 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" reaction, and partly because the "subordinates" are, after all, the experts. Project managers use participative management because, to an extent, they *must*. Although they have a good purview of the total system, usually they are farther removed from technical problems and less qualified to resolve them than the people on their teams.[5]

Motivation

Project work can be stimulating, satisfying, and provide a great sense of achievement; combined with constant pressure to meet project goals, these are natural motivators. Elements inherent to project management—contracts, WBSs, Gantt charts, responsibility matrixes, etc.—are also motivators. They provide clear goals that, combined with financial and career rewards, motivate people in the same way as the management-by-objective approach.

But project work includes de-motivators as well. Too much pressure leads to stress, tension, and conflict. On large jobs individuals can lose sight of the end-item, become alienated, and feel threatened. One advantage of participative management is that it helps garner worker commitment to project decisions and diminish potential de-motivators.

Participative project managers do not relinquish responsibility; they delegate it. Even when they have legal authority, they involve others by acquainting them with problems, consulting them for their opinions, and giving frequent feedback. Knowledgeable workers are allowed to help prepare project plans and budgets. Through such participation they gain an appreciation for how their work fits in. They feel more closely associated with the project and dedicated to its success. But, as stated earlier, people and situations vary, and the project manager must determine for each worker how much responsibility she can handle and how much she needs to be monitored and directed. Usually project managers are supportive, involve others in decision making, and avoid dogmatic or impatient behavior. Especially in projects with high potential for conflict, they invest considerable emotional energy in developing trust and interpersonal relationships.

But simply telling project managers they need to develop a participative, relations- and task-oriented style is not enough. Unless a project manager receives support in adjusting styles, she might not be able to do it; old patterns of behavior remain; new ones are hard to develop. Often, companies provide training in interpersonal skills and team building to help managers to 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 steer his behavior.

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.[6] The ample evidence, anecdotal and empirical, is that effective project managers are strong leaders who utilize participative management.

16.3 Teams in Project Management

Project organizations are comprised of groups. As Figure 16.1 illustrates, in a large project some of these groups are comprised of people from within one organization (on the figure, the project office, mid-level management team, and functional and cross-functional work package teams), while others come from multiple organizations (the project management (PM) team and cross-organization functional team). Membership in many of these groups overlaps and people serve multiple roles that link the groups together.

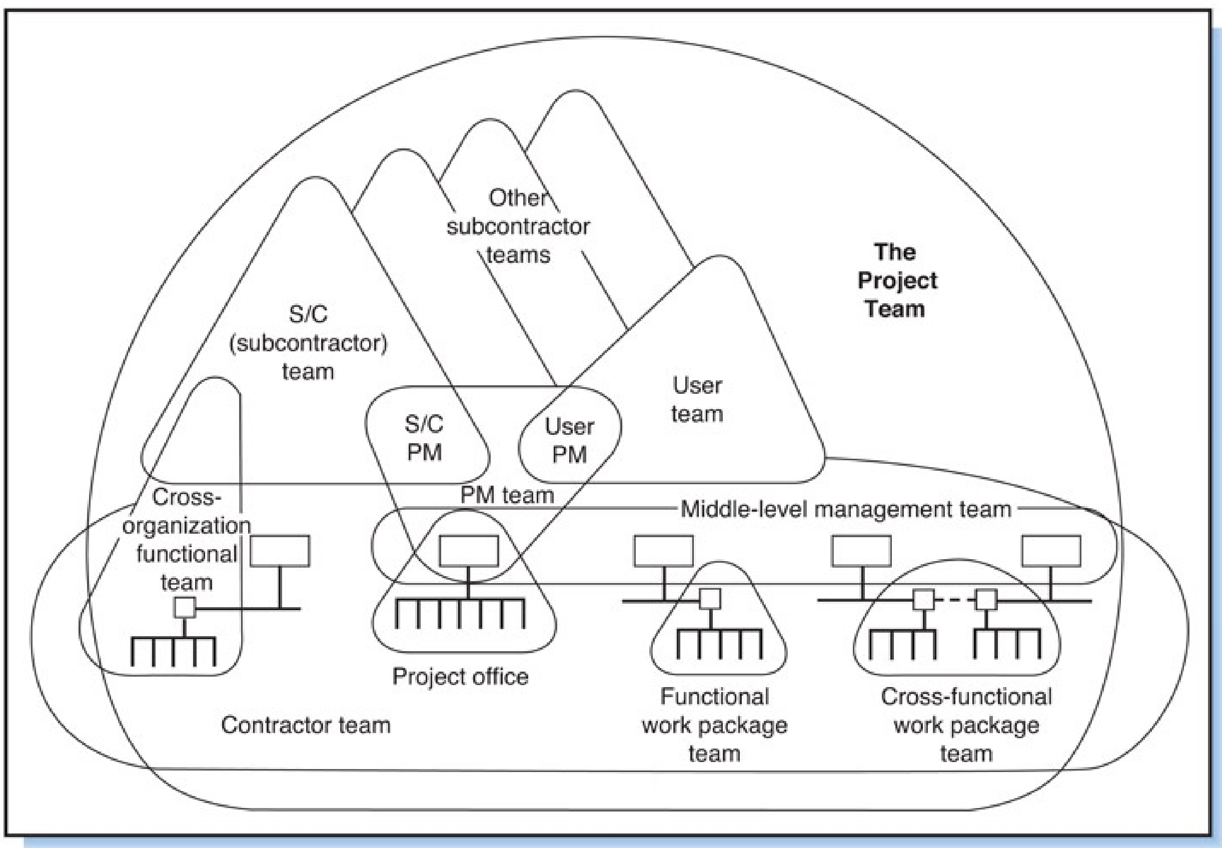


Figure 16.1 Groups comprising the project team.

The term *project team* as used here refers to any group that works in the project or to all the groups in combination. 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*. Virtually all work done in a project, mental and physical, is the product of teams. To be successful, a project needs *teamwork*.

The Trouble with Teams

Failures in projects 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 decisions made haphazardly. Teams often are more concerned with getting the task *done* than with doing it *right*. Many teams never know what their *purpose* is, so they never know when or if they have achieved it.

In projects with multiple teams, each might be oriented to different goals. They might be in separate offices and physically isolated, which creates and reinforces perceived boundaries and an attitude among the teams of "us versus them." These make for a portentous project environment that bodes ill for project success.

High-Performing Teams

In contrast, successful projects are the result of 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."[7]

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:

- High motivation 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 taxicabs—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 in the system (focus). Do it in terms of your energy (time). Put your whole psyche into it (feeling).”[8]

For project managers, Vaill’s 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 16.1 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 large-scale project.[9] Over 25 firms submitted proposals for the task of leading the team of 500 engineers, architects, artisans, and craftsmen who would do the renovation. Selected for the job was the small construction management firm of Lehrer/McGovern, Inc.

Hofer describes the firm’s partners: Lehrer is soft-spoken and generally conservative in appearance; McGovern clean-shaves his head, has a handlebar mustache, and wears cowboy boots. Despite differences in appearance, the two share 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 making arrangements with subcontractors and ensuring that 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."[10] 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."[11] 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.

Effective Project Teams

Project work requires close collaboration. People in project teams must rely on and accept one another's judgments and support each other. Managers 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.

One way to increase collaboration and commitment is by locating everyone in the project team in the same office quarters. Frequent daily contact makes it more likely individuals will identify with the team and project goals.

But even if co-locating team members were possible, close 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 as a team. In many projects however, especially where people have not previously worked together, team members don't know the team's goals or their own responsibilities, and they never learn to work together. A purpose of team building is to ensure that doesn't happen.

16.4 The Team-Building Approach

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.[12] In another study involving 32 research and product development projects, *group cohesiveness* was identified as the *single most* important factor to achieving project goals.[13]

Group cohesiveness and effectiveness do not just happen. Like any other 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, a group explores such issues and then *plans* how it will address the issues 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 the stress and to disclose and 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.

16.5 Improving Ongoing Work Teams

Consider how team building is 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.

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 individual 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. 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:[14]

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:

> "Our meetings are always dominated by the same two or three people."
> "Our way of getting things done is slow and unorganized."
> "I have no voice in decisions that affect my functional group."
> "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 leads 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 worked with project teams in workshops to effectively resolve problems ranging from technical issues to interpersonal conflict.

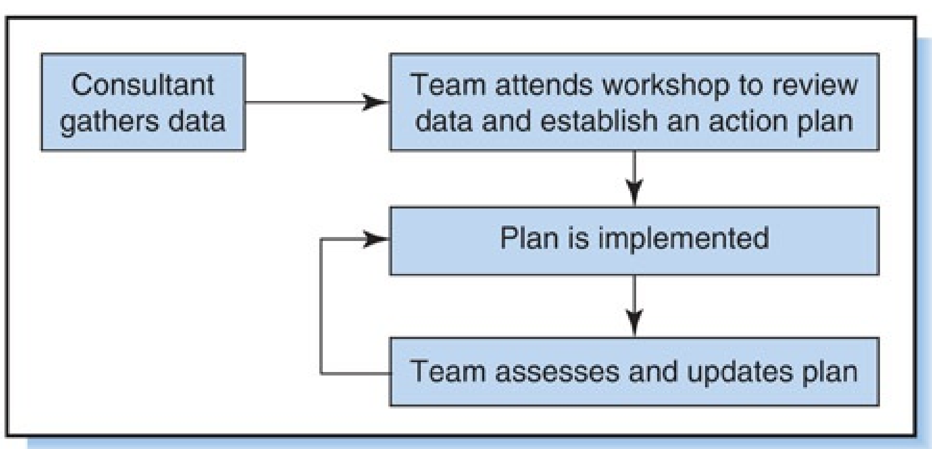


Figure 16.2 The team-building process.

To ensure that action steps are implemented, follow-up work is scheduled formally in sessions 2 to 3 months later or less formally during regular meetings. The team takes stock of its functioning, any improvements it has made, and what is still needed. The group itself takes over the consultant's role; should new problems emerge, it repeats the process, as summarized in Figure 16.2.

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.

16.6 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 in terms of team cohesion and performance is that they develop bonds around shared skills, competencies, and tasks. Thus early team-building efforts should provide team members an opportunity to work together on tasks related to project goals, and to develop competency-or task-oriented relationships.[15]

The purpose of team building for a newly formed team is for the team to reach agreement on its purpose, how it will achieve its 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 and one another.

A team-building workshop is convened by a facilitator. During the workshop members will 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:[16]

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 she can devote to the project each day or week. The information is tallied and posted on a chart similar to Figure 16.3. The team members discuss the differences in their commitments to the project and individuals are invited to explain their positions on the chart. The discussion helps reduce the potential resentment of some members committing to more work or less work than others.

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. Sometimes this is straightforward, such as when the objectives have already been set; other times the group will have to define its objectives from scratch. Either way, the purpose and objectives should be clearly defined and accepted by everyone. The group then develops sub-objectives so that members may be given specific assignments. The objectives should complement any user and system objectives and requirements as defined in the SOW or Charter (described in Chapters 3–5). In fact, a session like this can be used to create the SOW or charter.

see chapters 3–5

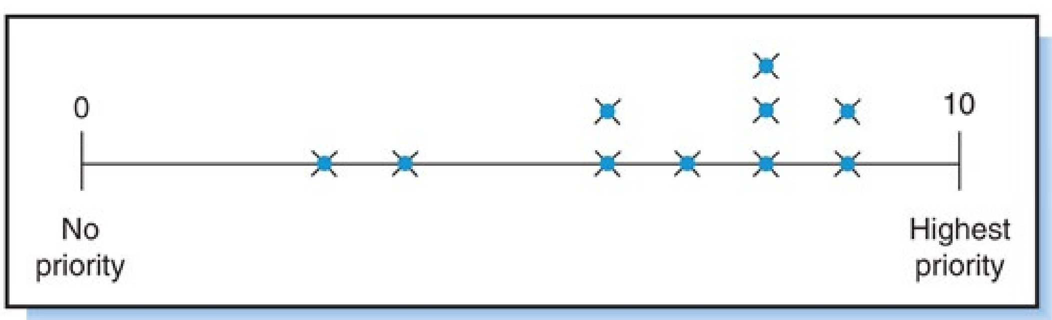


Figure 16.3 Priority ranking of project for ten team members.

Step 4: Formulate Operating Guidelines

Group dysfunction often arises over mixed expectations about work roles, job assignments, and how the group ought to work. This can be avoided by the team establishing operating guidelines that address, for example:

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? For example, in some cases maybe only two or three members should be involved; in others, only the best-informed people; in still others, the entire team.
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 current roles and responsibilities of its members and identify any ambiguity, overlap, or conflict.

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

Opposite team building is team disbanding. 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.

16.7 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.[17]

The two groups meet together for a day. At that time:

1. Each group compiles four lists: (1) what they believe the *other* group are 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 discuss what they learned from each other's lists and to list and prioritize the issues that need to be resolved.
4. Finally, the groups meet together again to discuss 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*, and overall goals of the project or program suffer. 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.

16.8 Virtual Teams

Sometimes everyone on the project team—the manager and all its members—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 other members 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 described earlier 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 to communicate. This makes it more difficult to make decisions, follow up on commitments, monitor results, and build relationships and team cohesiveness. And everything gets worse when the team is spread across different time zones, languages, and cultures.

Communication Technology[18]

Virtual teams exist by virtue of technology. When travel budgets are meager, most communication happens electronically. There are many available technology options, and a project will usually employ several. 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
- Audio-conferencing
- 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 largely depends on the task. Ambiguous or challenging tasks and decisions require technologies that are "media-rich" and mimic normal conversation—i.e., synchronous technologies. Asynchronous technologies such as email are not media rich; their use should be confined to sharing information and documentation. Virtual teams, however, share lots of documentation because, in general, in the absence of face-to-face meetings, writing replaces conversation; simply, virtual teams *need to write more.* Even audio conferences and old-fashioned phone calls need to be followed up in writing to ensure clarity.

The technology to be used must be compatible with the hardware/software at different team members' sites. Also, members must be comfortable with using the technology and have access to training.

Team Cohesion[19]

In general, in a cohesive team the members share a vision and trust each other. 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 results-oriented performance measures for each member; the measures must be specific enough so the project manager is able to gauge

each member's performance.

- Related to the previous point, 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 16.6. The leader must expect and reinforce compliance to the guidelines.

Trust[20]

Team cohesion also depends on the level of 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 on a regular basis, although it can occasionally by allowing/encouraging members to visit *each other's sites*. An alternative is for representatives from subteams at different sites to "float" among the other project sites. Another is for the entire team to meet 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, though ideally more often, if possible. 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![21]

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.

Definitive clues about performance in virtual teams are lacking, so even a little negative information can destroy an individual's or team's reputation. The project manager and team members must support the team and the project in good times and bad; this is true for face-to-face teams, but more so for virtual teams. Information indicating poor performance should never be accepted without investigating it first.

Virtual Meetings[22]

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 not invited the reason why, 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 much), and in what ways. Try to get peoples' 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* is talking. Cohen suggests at audio-conferences appointing one person at each location with a good ear for voices to hold up a picture of whoever is speaking on the other end. People get accustomed to this and actually look at the photo as if it were the photo doing the talking![23]
- **Be inclusive**. 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 a diverse point of view, such as asking one member to play devil's advocate. Practice communicating in ways that lead to trust: show respect, use names people prefer to be called, create dialogue—not monologue, and listen attentively. Be forgiving when someone makes a mistake.

- **Pace the meeting**. Guide discussion toward a resolution or postponement; remind people of remaining time. Set time allotment for each item; ask if the team wants to extend the meeting.
- **Enforce participation**. Check frequently that everyone is staying with the agenda. Notice if members have not spoken and ask for their input. In meetings not conducted in the team's native language, members might have trouble keeping up with the discussion. Provide breaks for them to organize and collect their thoughts.
- **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 of the meeting. Be careful to make sure the minutes (and interpretations of what was said) are correct.

16.9 Conflict

In all organizations, differences in objectives, 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 departments. It 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 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.[24]

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. Such scrutiny is a constant irritant to the contractor. In a fixed price contract, the contractor might request periodic upward revisions, also a source of conflict. Any contract with poorly-specified terms for cost, schedule, or performance is likely to 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, at the same time, the chances of conflict. The 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 compromise the high 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 release certain personnel to projects or try to retain authority over personnel they do release. Workers with dual reporting relationships often feel conflicted over priorities and loyalties.

Moreover, people are ordinarily reluctant to accept change, yet with projects change is the norm. Administrative procedures, group interfaces, project scope, and resource allocations are in constant flux. Changes in the labor force make it difficult to establish lasting reporting relationships.

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 well before a project begins.

The Project life Cycle

Thamhain and Wilemon[25] 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 costs 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 16.4.

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.

Start	Project life cycle		Finish
Project conception	*Project definition*	*Project execution*	*Project closeout*
Priorities	Priorities	Schedules	Schedules
Procedures	Schedules	Technical	Personality
Schedules	Procedures	Manpower	Manpower
Manpower	Technical	Priorities	Priorities

Figure 16.4 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 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:[26]

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*, 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 is 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 16.2 Conflict in Product Development Teams[27]

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 the 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. Designers are from the arts; developers from math and science. Designers tend to be female, vegetarians, talkative, and live in lofts; developers tend to be male, eat fast food, and talk little except to say "Not true." 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."

16.10 Managing group Conflict

How do people deal with conflicts? In general there are five ways:

1. Withdraw or retreat from the disagreement.
2. Smooth over or de-emphasize the importance of the disagreement (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. Confront the conflict directly; work through the disagreement 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 five 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.[28]

At the individual level, a project manager confronts conflicts by raising questions and challenges such as:[29]

> 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.

Role Clarification Technique[30]

Conflict in projects often arises because people have mixed expectations about work plans, roles, and responsibilities. In particular, disagreements arise because:

- 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.
- Everyone thinks someone else is handling a situation that, really, no one is.

- People do not understand what their group or other groups are doing.

The *role clarification technique* (RCT) is a systematic procedure to help resolve these sources of conflict. As the title "role clarification" suggests, the goal is that everyone understands their own and other's major responsibilities and duties, and that everyone knows what others expect of them.

RCT is similar to 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:[31]

1. What does the organization expect of you in your job?
2. What do you actually do in your job?
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 group 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.

16.11 Managing Emotional Stress[32]

Working in projects can be stressful. Long hours, tight schedules, high risks, and high stakes take a toll on social relationships and individual mental and physical health. Projects achieve great things, but they also instigate ulcers, divorce, mental breakdowns, and heart attacks. Emotional stress affects the performance and physical health of project workers 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.[33]

We will restrict discussion to three main causes of stress in projects: work overload, role conflict, 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 is distressful. A related problem, *work underload*, occurs with too *little* workload or work beneath a person's ability. Underload can occur during a long hiatus between projects.

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 in particular might feel frustrated because they have limited authority to satisfy the requirements of numerous stakeholders.

Stress also develops from the demands and pressures of *social 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 environmental stressors—stress is inevitable.

Stress Management

Most people accept stress as the price of success; however, although stress is 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.

In general, ways to reduce negative stress at work are aimed either at changing the organizational conditions that cause stress or at helping people better cope with stress. Because stress results from the interaction of people with their environment, both are necessary. Organizational means 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 the stress in projects.[34]

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; so does the opposite, the do-nothing, under-demanding. 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 can help reduce stress in projects. 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 one gets 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.”

16.12 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 slight 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, properly managed, beneficial. The primary conflict sources in projects include 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.

Stress in projects is also 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-Blanchard. What do these models say about leadership in the situations faced by project managers?
4. How is participative management 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?

Questions about the Study Project

1. How would you characterize the leadership style of the project manager in your project? Is it 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 16.1 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—at least. 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; 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 16.2 Mars Climate Orbiter Spacecraft[35]

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?

Endnotes

1. Chapman R. *Project Management in NASA: The System and the Men*. Washington, D.C.: 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, D.C.: NASA SP-4102; 1972, p. 23.

3. Fiedler F. *A Theory of Leadership Effectiveness*. New York: McGraw-Hill; 1967.

4. Hersey P. and Blanchard K. *Management of Organization Behavior: Utilizing Human Resources*, 4th edn. Upper Saddle River, NJ: Prentice Hall; 1982, pp. 150–173.

5. Sayles L. and Chandler M. *Managing Large Systems: Organizations for the Future*. New York: Harper & Row; 1971, p. 219.

6. Bennis W. and Nanus B. *Leadership: Strategies for Taking Charge*. New York: Harper & Row; 1985, pp. 224–225.

7. Vaill P. The purposing of high-performing systems. *Organizational Dynamics*, Autumn 1982: 23–39.

8. Ibid., 38.

9. 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.

10. Ibid., 28.

11. Ibid., 21.

12. 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.

13. Keller R. Predictors of the performance of project groups in R&D organizations. *Academy of Management Journal* 29 (4) December 1986: 715–726.

14. See, for example, Dyer W. *Team Building: Current Issues and New Alternatives*, 3rd edn. Reading MA: Addison-Wesley; 1995; Reilly A. and Jones J. Team building, in Pfeiffer J. and Jones J. (eds.). *Annual Handbook of Group Facilitators*. LaJolla, CA: University Associates; 1974.

15. Based on research in Gratton L., Voight A. and Erickson, T. Bridging faultiness in diverse teams. *MIT*

Sloan Management Review 48(4); 2007: 3

16. This discussion largely based on Dyer, *Team Building*.

17. Blake R., Shepard H. and Mouton J. *Managing Intergroup Conflicts in Industry*. Houston: Gulf Publishing; 1965.

18. This section is largely based upon Duarte D. and Snyder N. *Mastering Virtual Teams*. San Francisco, CA: Jossey-Bass; 2006, pp. 31–80.

19. Portions of this section are from Duarte and Snyder, *Mastering Virtual Teams*, p. 77.

20. 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.

21. Cohen, *Succeeding with Agile,* p. 370.

22. Much of this section adapted from Duarte and Snyder, *Mastering Virtual Teams,* pp. 170–183.

23. Cohen M. *Succeeding with Agile*, pp. 376–377

24. Sayles and Chandler, Ibid., p. 278.

25. 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.

26. Schmidt W. Conflict: A powerful process for (good or bad) change. *Management Review* 63, December 1974: 5.

27. Moody F. *I Sing the Body Electronic*. New York: Viking; 1995, pp. 110–115.

28. Ibid., pp. 46–47.

29. Sayles and Chandler, *Managing Large Systems*, p. 216.

30. Method based upon Dyer, *Team Building*, pp. 109–116.

31. Ibid., p. 112.

32. 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.

33. 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.

34. 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.

35. NASA website, December 1999.

Part V
Project Management in the Corporate Context

Beyond leadership skills and project management tools and methods, what else is needed to enhance project success? One part of the answer is the organization must support its managers and encourage and enable them to apply project management best practices; another is that it undertake projects that are viable and beneficial to the organization; i.e., it must select projects that meet sound criteria based upon the organization's objectives and available resources. These are the topics of Chapter 17 and Chapter 18, respectively.

In today's growing business and technology globalization, a topic of increasing importance is international project management; this is the subject of Chapter 19. The topic spans most everything else covered in this book (although from an international perspective), so it serves as a fitting summary and review.

Chapter 17
Meta-Management of Projects and Program Management

Meta-management of projects refers to important aspects of project management over which project managers usually have little or no influence. For example, the general approach to be taken and preparedness of an organization to perform projects may depend on others—senior managers and directors. In other words, project managers are constrained in what they can accomplish, and the likelihood of project success partly depends 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.

17.1 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 maturity levels are shown in Figure 17.1.

The process of increasing maturity in project management begins when a few people start to understand 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, and for that to happen requires executive-level awareness about the importance of project management and a willingness to support the spread of those principles throughout the company. These steps include documenting lessons learned from every project for the benefit of other projects and developing a common language of project management terms to be used everywhere in the company. A company with projects across the globe might create a glossary of terms in multiple languages. Naturally, moving to higher-level maturity also requires that the organization develop a project management methodology. Ultimately the organization is in the position to benchmark its project management capabilities against organizations that are industry leaders.[1]

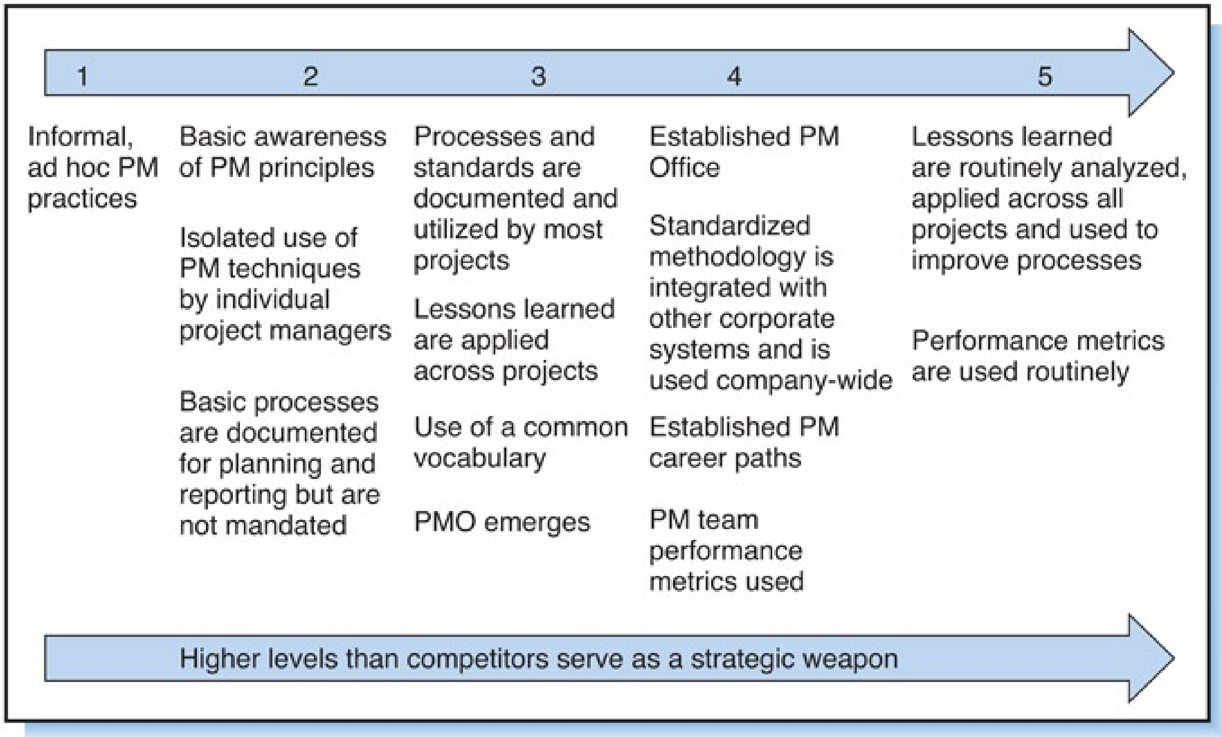


Figure 17.1 Levels of project management maturity/competency.

Maturity Models

An organization's project management maturity is gauged according to so-called "maturity models"; there are many recognized models although none has achieved acceptance worldwide.[2] Maturity models fall into three categories:[3]

- Technical Delivery Process Models
- Project Management Process Models
- Total Organization Models.

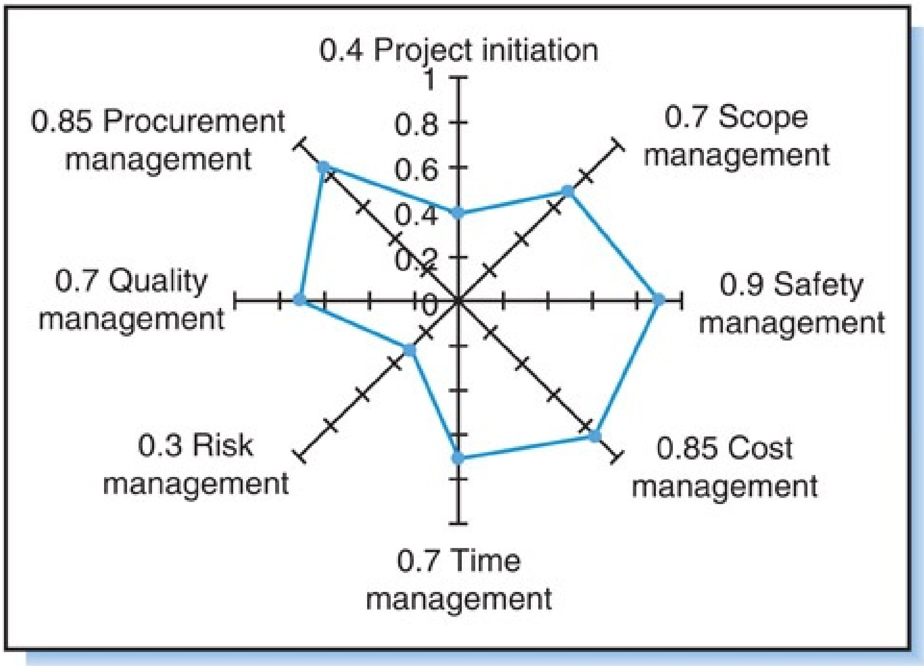
Technical Delivery Process Models originated in the Total Quality Management movement of the 1980s when companies started to measure their quality management capabilities. An example is the Capability Maturity Model (CMM) developed by the Software Engineering Institute of Carnegie-Mellon University during the 1980s and 1990s to help identify competent software contractors. The model, which emphasizes process documentation, similar to ISO quality standards, has five levels of maturity.

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 17.2 shows the audit results for the assessed maturity levels for "Initiation" is (according to the PBMOK) areas.

Process models commonly specify five levels of maturity, comparable to the levels in Figure 17.1:[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 the development of 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 accounts for the entire organization and how it manages projects, programs (discussed later in this chapter), and project portfolios (discussed in Chapter 18).



[Figure 17.2](#) Results of a maturity assessment regarding project management aspects.

See [Chapter 18](#)

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, beyond which they reduce success; i.e., conformity to industry standards can only take you so far.[8] No single maturity model enables project success across all industries and types of 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 Models[9]

Having achieved a high rating according to a standard maturity model gives a company bragging rights.

In a proposal, a company can point out it has achieved high-level maturity for a recognized 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]

Project Maturity and Project Success

Studies indicate that about two-thirds of organizations rate at levels 1 or 2 on the five-level maturity scale. Companies in petrochemical and defense industries are relatively more mature; those in insurance, financial and health services, pharmaceutical R&D, and telecommunications are less mature.[11]

Does achieving 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, 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 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 17.1, processes and standards for managing projects, a *project management methodology* should be created and utilized. Also needed are documenting and utilizing Lessons learned, which relates to *knowledge management*, and creating a *PMO* (Project Management Office). These topics are described next.

17.2 Project Management Methodology[15]

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, though 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 that projects will be successful. An organization creates or adopts the methodology so as to uniquely fit 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 emphasizing 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 will 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. Of course, every manager should practice it. Managers accustomed to a structured, documented approach to project management will more readily adopt the methodology; those not accustomed might not.

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 stepwise fashion. The project management methodology defines the steps—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 stage might be a "gate," so-called because at that point the current status of the project and plans for the remainder of the project are assessed, and a decision made whether to continue, hold, or cancel the project. The number of stages and gates depends on the methodology, with the minimum usually four or five; Motorola in Case 17.2 had 16 sub-phases (stages) 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 ever 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 17.3 illustrates a project management methodology for a seven-stage project life cycle (Initiation through Maintenance). In general, the methodology should conform to the technical and business practices of the organization; for example, stages 4 and 5 in the methodology in Figure 17.3 must be compatible with whatever development methodology the organization employs—waterfall, spiral, iterative, or Scrum.

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.

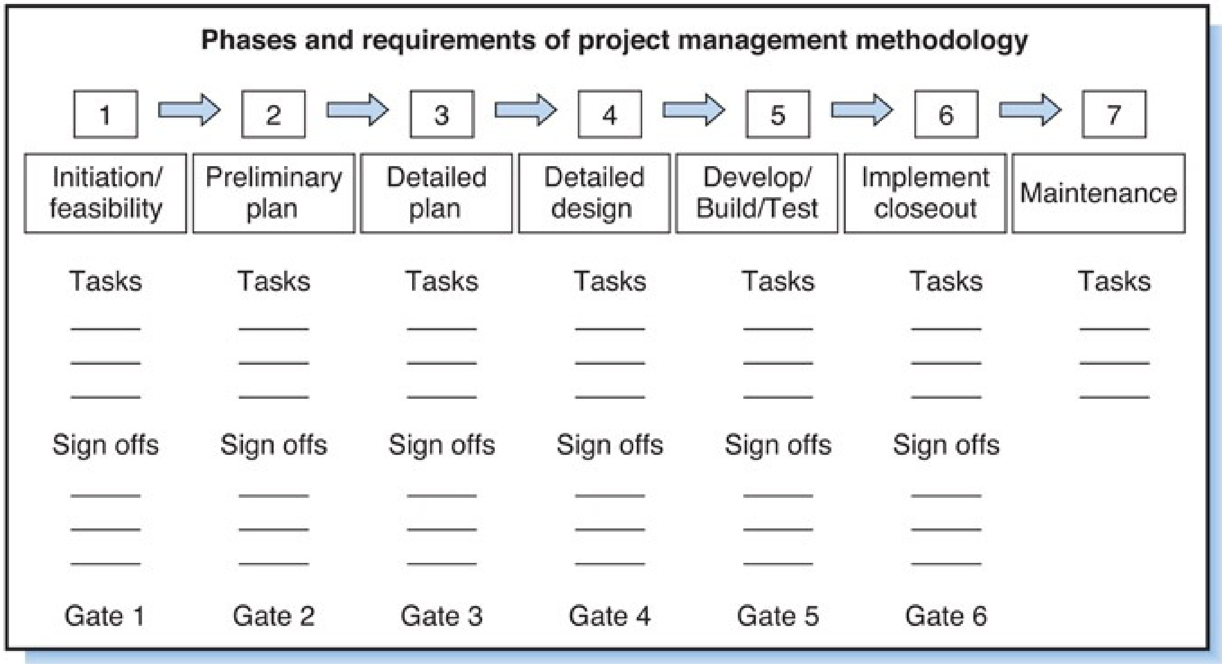


figure 17.3 Project life cycle phases versus project management methodology.

A typical methodology defines the phases or stages of the project life cycle and for each phase the tasks and deliverables, and stakeholders and their responsibilities.

Project Life Cycle

This book has used the phases of Conception, Definition, and Execution, each with a series of stages; in general, though, a particular project can be defined in terms of 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* and *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 (e.g. the Figure 17.3 methodology includes the post-project stage of maintenance).

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 17.3 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 include tasks and deliverables that cover virtually all of the topics covered in this book, such as those in Table 17.1.

Table 17.1 Project Management Tasks and Deliverables

Project initiation/proposal	Procurement/Contract management
Stakeholder identification	Recruiting, taining, layoffs
Project selection	Project tracking/review
Proposal development	Data entry

Project planning	Reporting to management
Requirements/specification	Project auditing
Work definition	Qualit control/assurance
Resource needs	Process control
Time and cost estimating	Change control
Scheduling	Project closeout
Budgeting/accounting	Post-project review
Risk analysis	Post-implementation review
	Knowledge management

Who is Responsible—Sign-offs and Approvals

As mentioned, the methodology might include gates at which the project must be approved. The methodology would specify 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 corporation is shown in Figure 17.4. Interesting in its details, it exemplifies the scope of the tasks, deliverables, and responsibilities covered in a comprehensive project management methodology.

The PRINCE2 Methodology

An example of a standard methodology is PRINCE2, which the UK Government developed as a guide for organizations to develop their own unique methodologies. It defines the roles of higher-level managers (corporate or program), the Project Board, the project manager, and team managers. It prescribes the project stages as *Pre-project, Initiation, Subsequent-delivery stage(s)* and *Final delivery*. The Pre-project stage is initiated by a *mandate document*, the Initiation stage by a *brief*, and the Subsequent delivery stages by a *PID* (*Project Initiation Documents*); there might be only one Subsequent-delivery stage for small projects but several for large projects. PRINCE2 also prescribes a stage-boundary management process to be followed upon completion of the Initiation stage and Subsequent delivery stages. The process defines activities of the project manager to enable the Project Board to assess each stage and approve the plan for the next stage and updated overall project plan, and procedures to control stages and manage product delivery in the Final delivery stage.

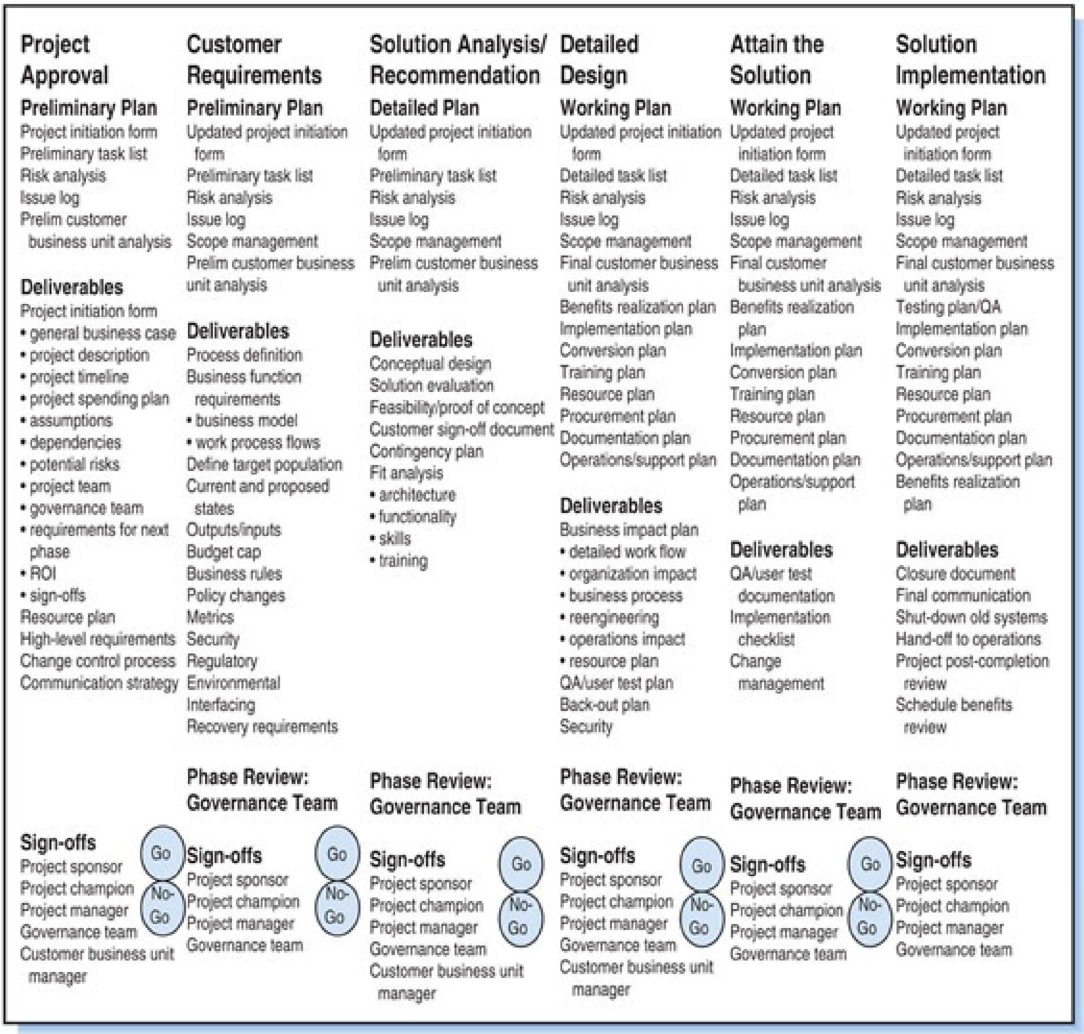


figure 17.4 Comprehensive six-stage project management methodology.

One Size Fits All?

Most methodologies are somewhat flexible. They specify project management requirements for a generic kind of project but allow for inclusion of other requirements, 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 suit 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, 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. The technical stages, tasks, and life cycles of these projects vary and, thus, require different project management methodologies.

Creating the Methodology

Two ways an organization develops a methodology are to create it 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 uses those as a baseline from which to create its own methodology and precisely tailorit to its projects and business practices. Ideally the tailoring is done by a group of the organization's best project managers (*not* by 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.

Evolving, Continually Improving Methodology

A project management methodology is not a static thing; it is subject to change and improvement based upon experience and a changing environment. A methodology should be periodically reviewed to incorporate 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 must exceed the effort in creating and maintaining it. The methodology must not become yet more red tape, forcing managers to attend more to 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 job."

17.3 Managing Project Knowledge

One potential pitfall in managing projects is treating each project as if it were completely unique 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 that it exists or how to access it. Often the waste occurs because the organization has no formal process for capturing and disseminating knowledge; i.e. it has no *knowledge management*. In a project organization, knowledge management would help 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 apply it to future projects.

Capturing Knowledge

Knowledge is information put to use. Everything experienced in projects is a source of information, but to learn from that experience managers and teams must reflect on each experience and draw conclusions; they must think about what happened, what they did, and the outcomes; otherwise they won't learn or will forget.

One opportunity to learn from a project is to conduct a *post-completion project review* or postmortem discussed under continuous improvement in Chapter 9. During the review the team carefully looks at what it did and what it learned from that. It reflects on significant events, successes, and failures, and the actions that led to them. This is discussed under project closeout in Chapter 12.

Sometimes a post-completion review is not enough; it happens at the end of the project, and by that time memories of events have faded, recollections of details dimmed, and 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.

See Chapters 9 and 12

Common Knowledge and Knowledge Transfer[19]

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 for, among other things, “analogy” estimating in project proposals.

Example 17.1 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. Jacque 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 Jacque's firm frequently holds for the express purpose of enabling people to meet each other and share project experiences. Jacque arranges a phone conference with Leslee, but before then Leslee checks the company database for background about Jacque's client. During the phone conference Leslee and Jacque work out the details of the proposal, which Jacque completes and sends to the client—barely a week after having received 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. 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 knowledge 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—and to know where to look in the database and what questions to ask.

Some companies actually *impose* potentially useful knowledge on the persons who need or could use it. A project support group (PSG) or PMO tracks information that might be of use to people, and forwards 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 reports on "best practices" 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. In Example 17.1, Jacque relied not only on databases but also on Leslee for advice. Peer advice is the form of one-on-one knowledge transfer that Jacque's firm encourages through networking meetings where people make acquaintances they might one day rely upon.

Such personal interaction 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 not be able 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.

After-Action Reviews

Teams that remain intact from project to project can learn and develop a growing knowledge storehouse through after-actions 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 event 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 performs its work, 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 the truth without fear of recrimination. AARs are most effective for projects that have specific, clear 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 Consultation and Project Resource 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; the trick is to connect the people who have the knowledge (providers) with those who need it (receivers) so the two parties can personally *interact* one on one.

Why personally interact? Because when knowledge providers and receivers do so, 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 17.2 Peer Consultation

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 them ("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 the company in this example, any project manager needing peer consultation can request it and the company will cover the consultants' expenses for travel and time off. The consultation process emphasizes questioning, analysis, and feedback; 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 17.3 Project Support group[24]

The project support group (PSG) of a large pharmaceutical corporation includes ten consultants available on request 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 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 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, could have saved $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 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 their knowledge 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: 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, a 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 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 managing project knowledge? 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.

17.4 Project Management office[27]

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 17.4 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 remaining 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 and 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.

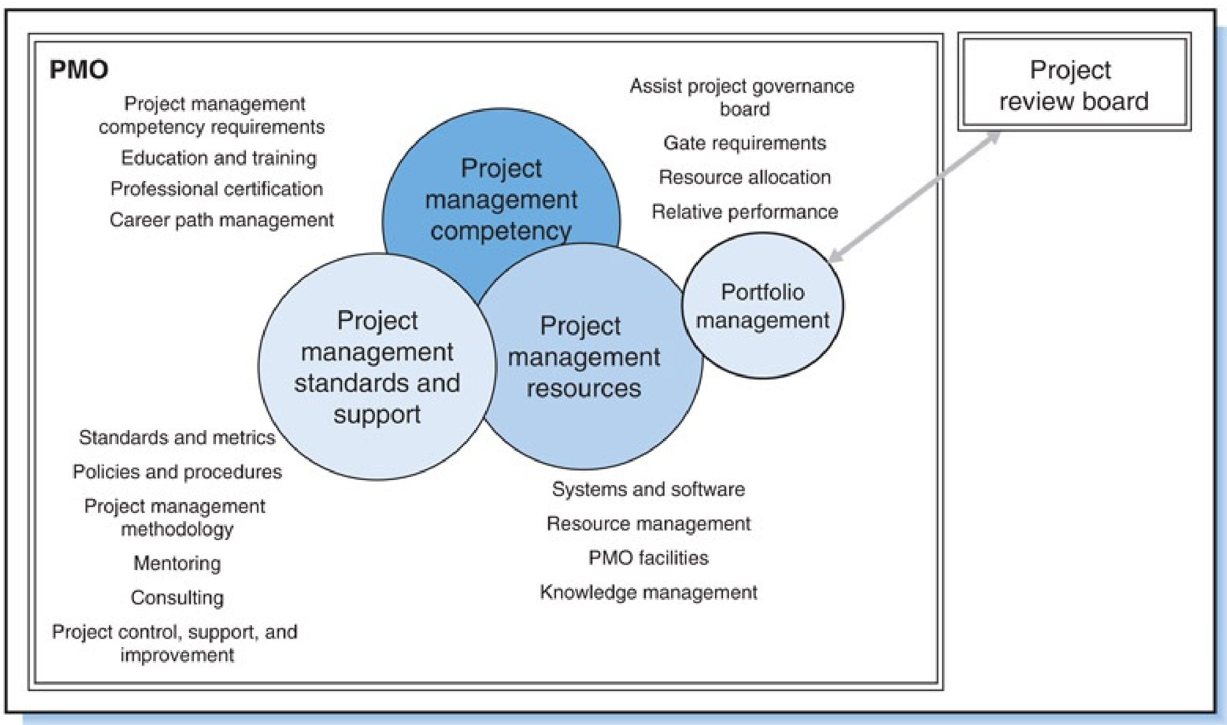


figure 17.5 Major functions and responsibilities of the PMO.

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, for example, serving managers in projects for a certain client, or for departments such as IT, research, or product development—units where the work is largely project-based.

What exactly does a PMO do? The foci and activities of a PMO, shown in Figure 17.5, 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

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.

2. Project Resources and Project Management Support

Resource Management

A common problem in project-based organizations is that projects simply do not have adequate resources. This happens when projects are initiated and approved without considering the resources needed versus the resources available. As a result, resources are shifted from one project to another, and 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 number of full-time employees in each job title or skill category. The record includes for each resource the number allocated to current projects versus 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 cancelled.

In many organizations, the selection and relative priority of projects are set by a Project Review Board (PRB). The PMO provides the information about resources so the PRB can determine the feasibility of undertaking each new project and allocate resources to projects with the highest priority. Chapter 18 covers this.

See Chapter 18

Project Management Software and Communication Technology

In most project-based companies all project managers use the same project management software. 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. Often the PMO is responsible for procurement, installation, and upgrade of the software, as well as providing training in its usage and applications. For software that requires time-consuming data entry for scheduling, budgeting, planning, and tracking, the PMO often provides clerical and data-entry support.

Project Facilities

Projects situated at stand-alone sites (construction projects) or away from the home organization (including overseas projects) or involve multiple functions or organizations need a physical office, a central place for the project staff to meet and work. The PMO arranges for the project office and related 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.

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.

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

For now suffice it to say that the PRB is charged with oversight of all significant projects, including deciding which to fund, which to defer, and which to kill (discussed Chapter 18). The PMO serves as advisor to the PRB and provides the PRB the information necessary to make these decisions. As each project moves through the gating process, the PRB assesses its performance, in part 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 posts information about the projects 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.

See Chapter 18

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; often 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 the 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 champion, create a project management methodology. They also create 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 position and 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 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 knowledge databases and competency lists (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 better management of projects and enable project managers to do their jobs better.

17.5 Program Management

A program (or programme) is *a set of projects and other activities* organized and coordinated to achieve *an over-arching purpose* or *goal.* The projects are interrelated (interdependent or linked in predecessor–successor relationships), often 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 program.

Program management prioritizes and coordinates the set of projects and other activities to meet program goals and attain benefits not achievable from any single project. Sometimes program management is considered an extension of strategic management since it implements strategic initiatives and manages change in the organization or community. The green vehicle program, for instance, could be seen as the manufacturer's way to implement a 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, which is necessary because, simply, programs are different than projects.

Programs vis-à-vis Projects

The main differences between a project and a program are:[28]

- A project provides specific *deliverables*—products, services, or other results—for a specific cost at a specific date. A program provides *benefits*—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 duration of a program might be indefinite with no set end date.
- 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 benefits that exceed those achievable from any one project.

The last point says a program's benefits go *beyond* those of its component projects. Take for example a construction company that forms programs based upon the kinds of projects it does. By grouping projects into programs for, say, roadway construction, roadway resurfacing, and retention systems, each program provides benefits—e.g. consolidation of work approvals and streamlining of processes—not attainable from the individual projects.

Kinds of Programs[29]

Among 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 the projects and other activities to achieve overarching benefits tied to business strategies and broad organizational goals. The green vehicle program is one example; the Cosmic Mercury Exploration program in Case 17.4 is another.

An *improvement program* provides regular enhancements to 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, or performance—even 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," which 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. Another example is a government-sponsored jobs training program that serves as a clearinghouse for institutions offering training/education for high school equivalency, adult basic education, career-skill training, college degrees, certifications, apprenticeships, and internships.

A *portfolio program* is a group of projects that are otherwise independent but share something such as 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 around the same types of projects is an example.

Programs can also be classified by how they are initiated.[30] Some derive from a clear strategy: the program organization is created around a strategic vision and purpose, then projects and other initiatives are created to achieve the purpose; goal-oriented programs are formed in this way. Alternately, a program "emerges" when someone recognizes that pre-existing projects could be better managed if they were organized and coordinated. If the projects are largely independent, they might be grouped into a portfolio; if they are related and contribute to a greater purpose, they might be grouped into an improvement-related program.

17.6 Program Phases[31]

Programs do not have specific deliverables or end dates, so they do not follow the same life cycle as projects. Nonetheless, goal-oriented and some improvement programs do have life cycles with phases such as *Program Initiation, Definition, Project Execution, Renewal, and Closeout*. Whereas *project* life cycles typically end upon delivery of the end-item, *program* life cycles can extend to include the organization's transitioning to and initial operating of the end-item. The program phases might repeat: projects are executed, end-items are delivered, then the program is renewed (repeated) or closed out.

Program Initiation (or Formulation) Phase

A program is initiated in response to some pressure or need on the organization, e.g.: new or changing customers, goals, challenges, or strategies; competition; or a review of current and proposed programs and projects. An executive creates a high-level business case that defines the program's objectives and alignment with organizational strategies, and justifies its feasibility. A gate review after this phase by the program governance board either approves or cancels the program. If approved, a program manager and the program team are selected.

Definition Phase

This phase follows either Program Initiation or a "renewal" decision, described below. After Initiation, this phase includes establishing a more detailed business case and program objectives, a program plan, resource requirements, and budget, and defining and sequencing the initial, as-known projects and other work to comprise the program. It also requires 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, Definition consists of updating strategies and goals, establishing new or revised projects and other initiatives, and updating the program plan and responsibilities. The gate after this phase determines if the program and its component projects are ready for the Project Execution phase.

Project Execution Phase

The projects and other work activities constituting the program are executed. The program team monitors the projects' collective performance and interdependencies; senior management evaluates program benefits and determines if changes are necessary to keep the projects or the program aligned with program objectives. The phase includes directing delivery of project end-items, preparing the organization for change, and ensuring support for the adoption 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") or closed out. With renewal, the program continues as-is or with changes to its direction or composition, and initiates a new cycle of Definition and Project Execution.

Closeout Phase

The program is terminated because the objectives were achieved, the program is no longer justifiable, or greater benefits are available elsewhere. 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.

17.7 Program Management Themes[32]

Four themes pervade program management: decision management, benefits management, program governance, and managing stakeholder expectations.

Decision Management[33]

Decision management refers to the way decisions are made and implemented. In programs, it must account for the complexity of decisions resulting from multiple stakeholders, uncertainty about the future, ambiguity over alternatives, and linkages between outcomes and strategic objectives.

Decision management in programs happens iteratively throughout all program phases. Stakeholders discuss matters, agree on a shared vision, and make a series of small decisions aligned with the vision. They assess the benefits of project deliverables and other activities to make decisions and take actions regarding the projects, program, or strategies. Organizations without program management tend not to assess the business benefits of projects.

Benefits Management

Benefits are tangible business improvements that support strategic objectives.[34] 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 are first defined during Program Initiation in the business case. In Definition a *benefits plan* is developed, which specifies how the benefits align with organizational strategies and will derive from program outcomes. The plan also stipulates 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 Closeout responsibility to sustain 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 developing a vision and program objectives based on business strategies and stakeholders' needs, and then creating and using the necessary mechanisms to monitor the program and keep it aligned with objectives and strategies. As mentioned it includes phase-gate reviews to decide whether the program should be continued 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 15, includes the steps of identifying the stake-holders, their interests and expectations, and their influence; communicating with them; and working to increase their acceptance or support of the program and decrease any resistance. It is sometimes called "stakeholder engagement" because stakeholders must be engaged in defining and executing the program; plus, generally, they don't like being "managed." Often stakeholders have considerable authority and formal power, which requires a facilitative, participative leadership style on the part of the program manager. Stakeholders frequently have differing or conflicting expectations, so reaching agreement involves much negotiation.

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 meet program expectations—e.g. cooperate in defining program requirements and expected benefits and, later, supply information and give approvals upon request. Stakeholder management must account for the fact that in programs both the stakeholders and their expectations might change.

See Chapter 15

17.8 Program Organization[35]

The roles and relationships in a program organization are illustrated in Figure 17.6.

Topping the organization are the program sponsor, governance board, director, 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 a supportive environment for the program. The board is usually led by a *program director*, an executive who "owns" the program and has overall responsibility for its meeting objectives and delivering benefits. Often, however, the key decision maker on the board is the *program sponsor*; she champions the program and gives it top-management's ratification. She is responsible for the smooth "transitioning" of project end-items and results into ongoing business operations, and informing the board about resulting benefits. (Sometimes the transitioning responsibility is handled by someone in a special role, the *business change manager*.)

Figure 17.6 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 projects or other initiatives *within the program*. For the projects within the program, the office handles myriad functions: contracting, budgeting, training, risk management; eliminating redundant tasks; 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.

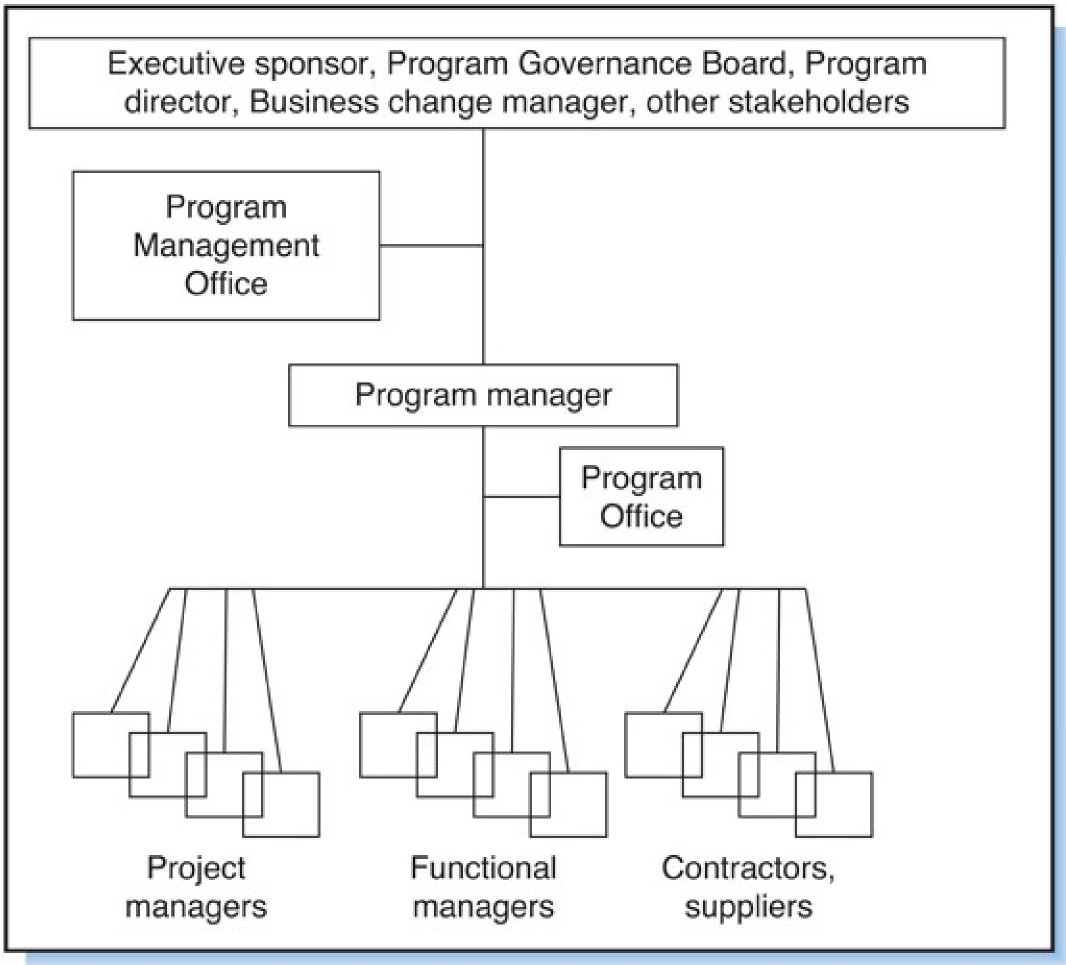


figure 17.6 Program organization, roles, and relationships.

In the center of a program organization is the *program manager* whose overall responsibility is to achieve the program outcomes as defined in the benefits plan. Specific duties include:[36]

- Develop program-level plans and schedules.
- Review and approve project plans for conformance 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 17.6

includes the Program Office, managers of constituent projects and other program initiatives, and anyone else involved in administering the program.

17.9 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 several ways, such as the following.

Transition Management

Transitioning is a central issue in program management; this is the transfer of project outcomes—products, capabilities, knowledge—to program 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 includes 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; 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.

Risk and Interface Management

Program risk management involves identifying, assessing, and managing *program-level* risks. A big program-level risk is "interface risk," i.e., risk caused by interdependencies among the component projects. Each project might be the successor or predecessor of at least one other project and, thus, can delay or be delayed by other projects for lack of products, information, services, or resources. Addressing such risks is called "interface management," which involves identifying interfaces (inputs/outputs) among projects and assuring that the necessary inputs 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 tend not to meddle with individual project activities. They view projects as black boxes with inputs and outputs, and schedule projects to account for their interdependencies, when deliverables are needed, or 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 interfacing projects about what each can expect from the others.[37]

Work Definition

How is the work in a program defined? In a project, often work is defined with a WBS. Can a similar approach be used to define program work? The answer depends on the kind of program. In a portfolio program, the program work is largely "predefined": it is simply the work defined for 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.

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, as with a portfolio program, 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 or new technology and uncertain requirements, so program planning happens "incrementally"; knowledge gained from earlier projects determines next steps and future projects.[38]

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.

See Chapter 4

Planning and Control

Planning for goal-oriented and improvement programs includes identifying the program’s constituent projects. The managers of these projects develop a plan that shows each project’s time and resource estimates 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 buffer consumption, expenditures, and other performance measures for each project, looks for potential problems, and assesses each project’s impact of the others and the overall program. Methods for doing this are covered by sources in the end-notes.[39] 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 project that do not affect other projects or the program are handled by individual project managers; those that 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 change requests, approving or denying them, and communicating follow-up actions. Required or requested changes emanating from the outside must also be assessed for their impacts on existing and future projects, and the managers of the affected projects notified to take appropriate action.

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, "mega project" management, or managing several projects as if each was a work package.

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.

Sometimes a large project—a "mega" or "super" project—is more-easily managed by breaking it down into multiple sub-projects. But unless managing the sub-projects in this way 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 project management, not program management.

Finally, program managers cannot manage the projects in a program in the same way a project manager manages work packages in a project. There are many reasons why; here are two: 1) Program managers usually do not have authority to "control" project budgets and schedules in the same way project managers control work packages; and 2) they cannot use a technique like earned value to assess program progress since such progress derives from project *interdependencies* and is more than just the sum of the component projects' progress.

17.10 Summary

The first 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 all 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 that an organization has achieved a high level of standardization in its project management practices and processes.

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 knowledge from projects include checklists, databases, and other forms of documentation (for explicit knowledge), and peer consultation, project resources support groups, or expert knowledge consultants (for tacit knowledge).

The PMO is a unit or department devoted to improving the practice of project management and supporting project managers. The PMO establishes and maintains the methodology, instigates initiatives to increase 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 a collection of projects and other activities grouped to meet goals and provide a collective capability or benefits beyond that of the individual projects. Program management and project management differ in many ways, including: life cycle phases and stages, roles and organization structure, themes of emphasis, and methods for initiation, planning, definition, and control. Consequently, managing programs requires tools and practices 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 four themes of program management?
24. Explain the program phases of Program Initiation, Definition, Project Execution, Renewal, and Closeout.
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 17.1 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 or 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 recommended that if MCA outsourced all IT infrastructure operations, it would save $30–50 million annually. 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 education 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 "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.

See Chapters 14 and 15

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 PMO director's role compare to the VP of projects illustrated in Figure 14.8 and PMO director discussed in Chapter 15?

Case 17.2 Motorola's M-Gate Methodology and the Razr Project[40]

Motorola employs the following 16-stage project methodology called M-Gate:

M15 Idea Concept	M7 Contract Book
MM 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

The methodology corresponds roughly to a five-phase product life cycle:

M15 M14 M13	M12 M11	M10 M9 M8 M7 M6	M5 M4 M3	M2 M1 M0

Business Case	Portfolio Planning	Project Definition	Implementation	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 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 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 skunkworks-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 18 (Figure 18.3b), the process starts with selected and prioritized product concepts streaming from the narrow end of the funnel. The concepts then go through five stages:

See Chapter 18

- 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?

Case 17.3 Tecknokrat Company

Tecknokrat Company 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 that too.

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 17.4 Mercury Exploration Program[41]

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.[42] 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 17.7).

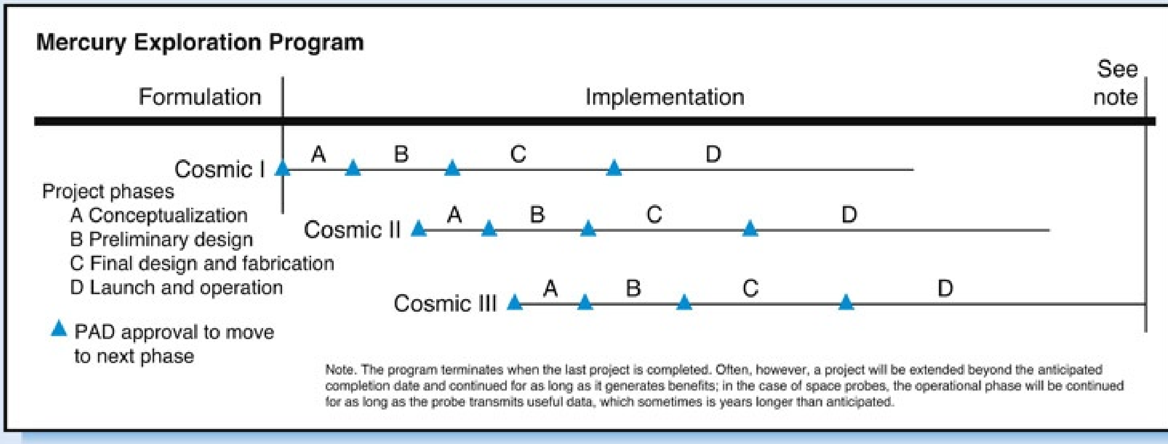


figure 17.7 Mercury Exploration Program and Cosmic projects; not shown are various other support projects and activities in the program. Program timescale is 10–15 years.

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 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.*

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—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.[43]

Upon approval of the PAD, MEP was authorized to move into the Implementation phase and Cosmic I, 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 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 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 initiated obtaining the funds 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 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.

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, Proposal for the Apollo Spacecraft) 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 Cape Kennedy 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 with the 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?

Endnotes

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 K. and Thomas J. Project management maturity models.

5. *A Guide to the Project Management Body of Knowledge (PMBOK Guides),* 5th edn. Newton Square, PA: Project Management Institute; 2013.

6. Kwak Y. and Ibbs C. *Project Management Process Maturity (PM)2 Model (Berkeley PM Model),* www.ce.berkeley.edu/~ibbs/yhkwak/pmmatrutiy.htmel; 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: 11–18.

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 Cooper; 2004.

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*. http://www.ey.com/global/content.nsf/Middle_East/Knowledge_Management_-_Tools, downloaded April 13, 2007.

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*; National Library for Health, Knowledge Management Specialists Library, *Peer Assist*; http:/www.library.nhs.uk/knowledgemanagement/ViewResource.aspx?resID=125167, downloaded April 14, 2007,

24. Other examples provided by ibid, pp. 103–104, and Welch N. *Peer Assist Overview*, http://003cce4.netsolhost.com/PeerAssit.htm, accessed April 12, 2007.

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. 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.

29. Ibid., Pellegrinelli.

30. Thiry M. *Program Management*. Dorchester, UK: Gower; 2010, p. 39.

31. Pellegrinelli, Programme management.

32. *The Standard for Program Management*.

33. Thiry, *Program Management*, pp. 59–63.

34. Ibid., 77

35. *The Standard for Program Management*; Thiry, pp. 83–88.

36. Hanford M. Program management: Different from project management. *DeveloperWorks*, http://www.ibm.com/developerworks/rational/library/4751.html Downloaded Dec. 22, 2014.

37. Kendrick T. *Identifying and Managing Project Risk*. New York: Amacom; 2003, pp. 99–100.

38. Op cit., Pellegrinelli.

39. Milosevic D. and Martinelli R. *Program Management for Improved Business Results*. New York: Wiley; 2007; Thiry, *Program Management*.

40. 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; *Gizmodo* website, Motorola Insider Blame Game: Engineers Shoved Designers Aside, B. Koerner, http://gizmodo.com/5038839/motorola-insider-blame-game-engineers-shoved-designers-aside, accessed August 24, 2010.

41. Case based upon Chapman R. *Project Management in NASA: The System and the Men*. NASA, SP–324 Washington, D.C.; 1973, pp. 13–19.

42. *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.

43. 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 18
Project Selection and Portfolio Management

Lilies that fester smell far worse than weeds.

—Shakespeare, *Sonnets*

Errors, like straws, upon the surface flow;
He who would search for pearls must dive below.

—John Dryden

This chapter is directed to anyone who wonders how companies choose projects. In many companies; project managers have no involvement in selecting or approving projects; simply, they are assigned to projects already chosen by someone else. Occasionally, however, especially in smaller companies, project managers do assist in project selection; they help to choose the projects that they or others will manage.

Projects are the means by which organizations pursue their strategic objectives, hence doing the *right* projects is critical to their business success. If an organization's objectives are to "be the low-cost leader," "expand market share in Europe," or "preserve the natural environment" then you would expect that 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. “Hobbyhorse” projects of senior executives get sacred-cow 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.

18.1 Project Portfolio Management

A project portfolio is a group of projects and programs aimed at 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; thus, most organizations would have several portfolios.

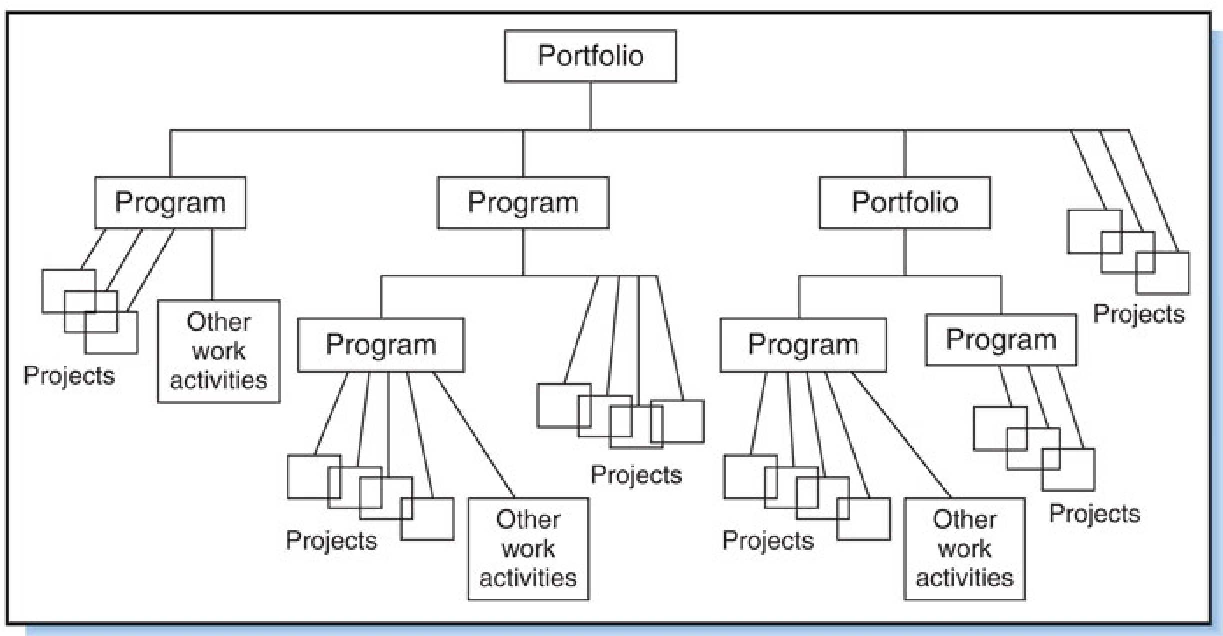


figure 18.1 Portfolios, programs, and projects.

As shown in Figure 18.1, a portfolio can consist of programs, projects, and other work activities—even other portfolios. Unlike the projects in a program, however, the projects and other elements in a portfolio are *not necessarily* interdependent. They are grouped in the portfolio for the purpose of better managing them as a whole, and prioritizing and allocating resources among them to best achieve organizational goals.

Almost 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*; briefly, this involves two steps: (1) creating portfolios—defining "themes" around which to form portfolios and criteria for including projects and programs in each portfolio; categorizing current and proposed projects and programs into particular portfolios; and (2) managing portfolios—assessing proposed projects and programs to decide whether each should be approved, put on hold, or rejected; prioritizing approved projects and allocating resources so priority projects get adequate funding; and tracking and managing projects and programs collectively for the good of the portfolio and the overall business.

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 portfolio management process is not trivial since managers routinely start projects without registering them and don't keep lists of current and proposed projects.

Academics, consultants, and software firms have proposed many approaches to project portfolio management. The breadth of the subject fills books; hence, treatment here is limited to a survey of the most common approaches.

Process for Successful Projects[3]

Successful projects and programs depend upon two things: doing the *right projects*, and doing those *projects the right way*. The two happen in a 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 the projects right***. Project managers guide projects using principles and practices of project management as described throughout this book.

Project Portfolio Manager

The project portfolio manager is the person 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 an important 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:

- Assure projects and programs align with strategic objectives and initiatives.
- Assess proposed projects and programs in terms of potential benefits, required resources, and risk.
- Approve projects that can achieve strategic objectives within acceptable risk and resource constraints. This happens in an early project phase such as FEL-1.
- 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.

Aspects of the portfolio manager's role may appear similar to a project or program manager's, but the portfolio manager has ultimate authority. Each project manager champions and may fight for his project's existence, but the portfolio manager looks at the project's benefits to the organization and may recommend scaling back or terminating the project.

Ideally the portfolio manager 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.

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 Governance Board, Steering Committee, Project Council). PRB membership typically includes the portfolio manager, chief financial officer (CFO), chief risk manager (CRO), chief human resource officer (CHRO), project management office (PMO) director, and chief technical officer (CTO)—the last being 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.[4] The portfolio manager typically chairs the PRB and has 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.[5]

18.2 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, it is used to determine to which RFPs the company should respond.

Projects differ with regard to 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 prescribed 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 18.2. In Phase I each project is independently evaluated and screened; in Phase II, all projects are compared and a subset is approved.[6] The framework largely also applies to programs—to selection of programs as well as 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.[7] 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 justification is documented in a preliminary business case 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 make a decision are rejected. Sometimes a simple checklist with a small number of criteria is employed and each proposal is rated as excellent, good, poor, etc. Proposals falling below a "threshold" composite rating are automatically rejected.

See Chapter 4

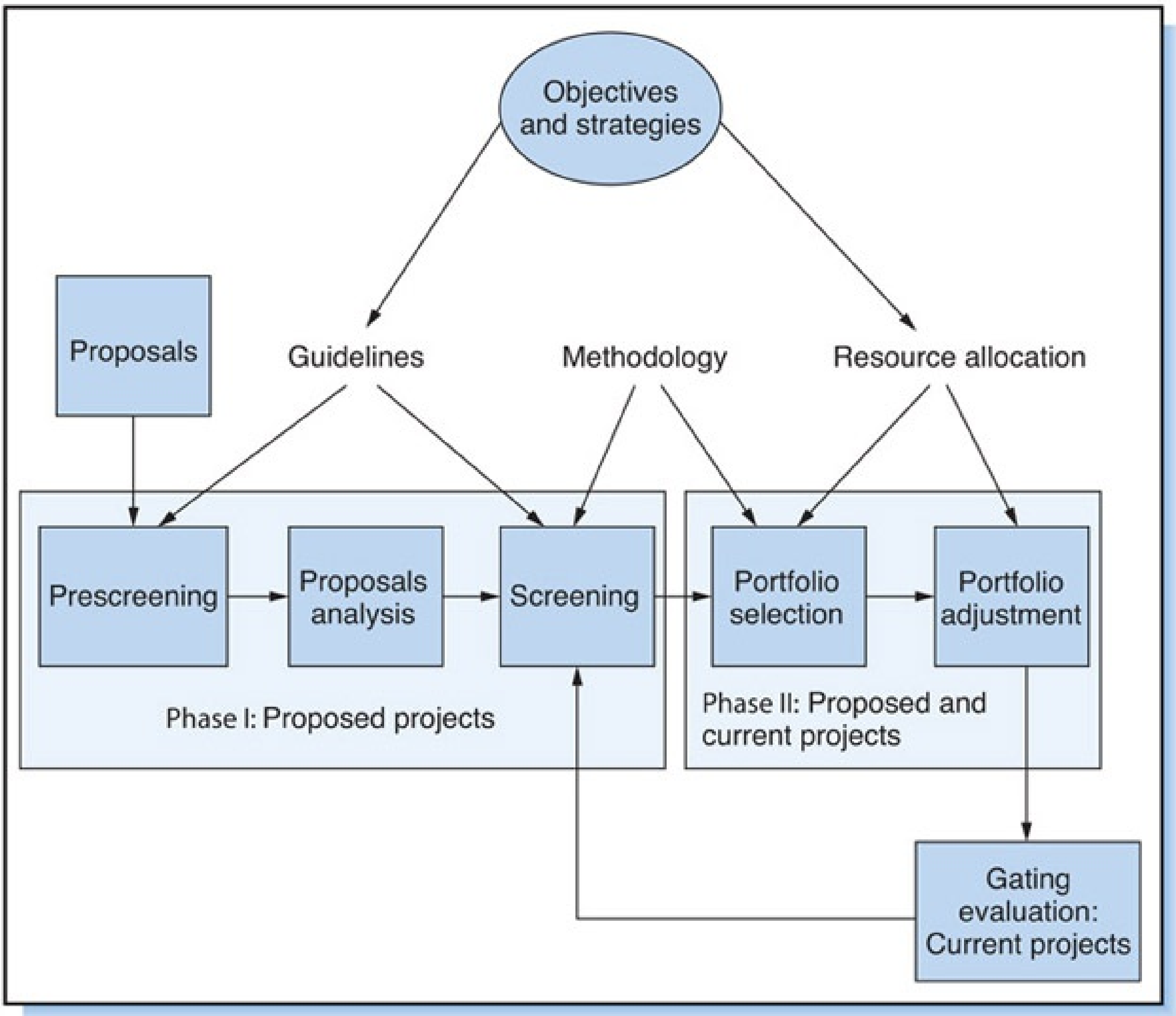


figure 18.2 Project analysis, selection, and portfolio management methodology.

Source: Adapted from Archer N. and Ghasemzadeh F. An integrated framework for project portfolio selection. *International Journal of Project Management* 17(4); 1999: 207–216.

See Chapter 4

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 lead to a more detailed, verified business case (e.g. as in FEL-2, discussed in Chapter 4).

To be considered for Phase II and possible 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, and to approve.[8]

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.

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 a continuing process. Existing projects are evaluated for expected benefits, 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 reconsideration about continuation, reduced or increased support, or cancellation. Rank ordering 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.

Funnel and Filter

The purpose of the analysis and selection process is not to discourage projects but to make sure only the "right" projects are pursued. Whether a company does work externally under contract or internally to develop new products or processes, it relies on projects for survival and growth; ideally, it has the option of *choosing* projects from among many proposals, RFPs, and initiation requests. Hence, the company's selection process can be likened to a *funnel* into which many proposals flow, and a *filter* through which only the best emerge, as illustrated in Figure 18.3. 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.[9]

18.3 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.

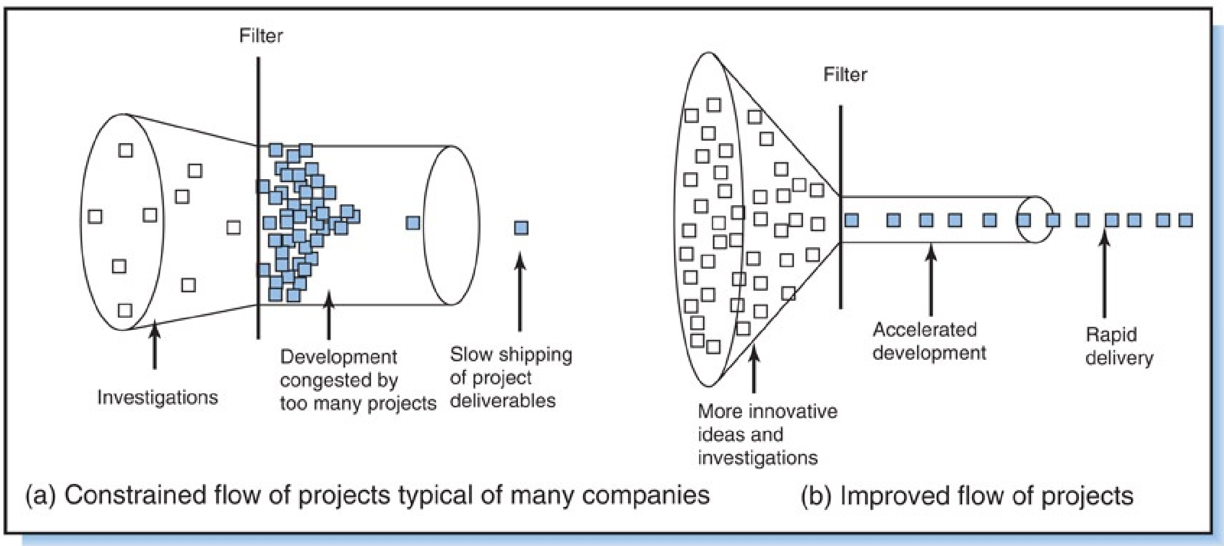


figure 18.3 Project selection process as a funnel and filter.

Source: Adapted from S. Wheelwright and K. Clark, *Revolutionizing Product Development.* New York: Free Press; 1992.

Financial Models

Financial models measure project proposals in term of economic or financial criteria such as net present value (NPV), internal rate of return (IRR), return on original investment (ROI), payback period, and life-cycle cost (discussed in Chapter 9). One common such model is expected commercial value (ECV), an application of decision-tree analysis (see Appendix to Chapter 10). This model, illustrated in Figure 18.4, considers the costs, earnings, and success likelihood of the development and launch of a new product.[10] 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,

$$\text{ECV} = [(\$50)0.6 - \$1.5\text{M}]\ 0.80 - \$10\text{M} = \$12.8\text{M}$$

Generally, the higher the ECV, the more preferred the project.

See Chapters 9 and 10

Another financial model is benefit/cost ratio (B/C), which weighs the benefits of a project against its costs. A simple example is:

$$\text{B/C} = \frac{\text{Estimated revenues} \times \text{probability of success}}{\text{Estimated cost}}$$

Values in the numerator and denominator are expressed in the same form, either as annualized or present value amounts. For example, if estimated annual revenue of the project is $100,000, estimated annual cost is $25,000, and probability of success is 50 percent, the resulting ratio is 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.[11] For instance, in the ratio

$$\text{B/C} = \frac{\text{Worth of benefits}}{\text{Capital recovery cost} + (\text{Operating cost} + \text{Maintenance cost})}$$

the "worth" 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.

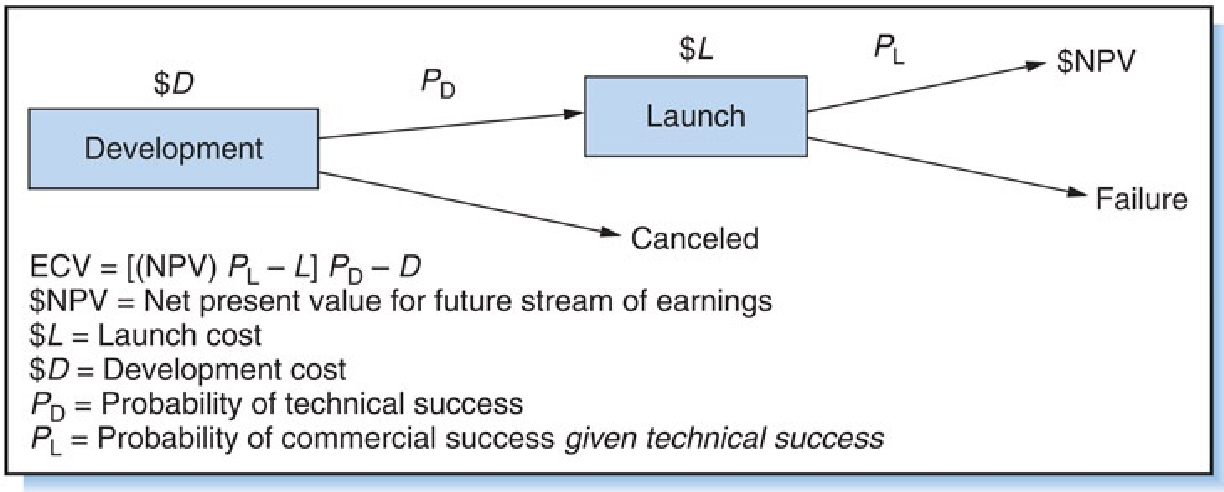


figure 18.4 Model for computing expected commercial value, ECV.

Source: Adapted from Cooper R., Edgett S., and Kleinschmidt E. Portfolio management in new product development: Lessons from leaders, phase I. In Dye L. and Pennypacker J. (ed.), *Project Portfolio Management*. West Chester, PA: Center for Business Practices; 1999, pp. 97–116.

Of course, the accuracy of the ratio 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.[12] Hidden costs and benefits can be difficult to identify and measure and often exceed by far the more obvious costs and benefits. In the renovation 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 is sole reliance on one (financial) criterion and neglect of other criteria of equal or more importance; NPV, for example, does not measure the strategic value of a project or the extent to which, say, a project would contribute to the goal of "expanding market into Europe."

Scoring Models

Scoring models rate projects in terms of *multiple* criteria that, besides quantifiable measures, include non-quantifiable ones such as market risk, customer enthusiasm, or fit with company goals—whatever criteria are thought important and discriminate between projects.

In the simplest scoring 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 the criteria are summed to yield an overall project score. Weighted ratings are used when some criteria are considered more important than others.

Table 18.1 illustrates a scoring method that includes probabilities and weights. The first column is the scoring criteria, and the next five columns ("Very Good" through "Very Poor") are *the expected probability* that the project will fit the criteria. For example, the probability that the long-range outlook for the project will be "Very Good" is 80 percent and will be "Good" is 20 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 10. 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 18.1 are calculated as the sum of the probabilities 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, the Weighted Expected Score is $3.8 \times 10 = 38$.

The bottom of Table 18.1 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.

See Chapter 10

Table 18.1 Project Weighted Scoring Model

Criteria		Very Good 4	Good 3	Fair 2	Poor 1	Very Poor 0	Expected Rating	Weight	Weighted Expected Score
Long-range	1. Product	0.8	0.2				3.8	10	38
outlook	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	18.0
	3. Image		0.6	0.4			2.6	4	10.4
Fits strategy	Phase 1	0.8	0.2				3.8	10	38
	Phase 2	1.0					1.0	5	5
	Phase 3	0.6	0.2	0.2			3.4	5	17
Goal	Goal A	0.2	0.8				3.2	10	32
contribution	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

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.

18.4 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:[13]

- 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 18.4 or score in Table 18.1), 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 18.1), 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.

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.[14]

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, 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 valued low on another. To overcome the conundrum of which criterion to use are methods that employ multiple criteria.[15] For example, Table 18.2 rates each project for three criteria: Fit with Corporate Strategy (subjective rating 0–4; 0 is poor fit, 4 is perfect fit); Reward (ECV, computed from financial model); and Risk (subjective rating 0–4; 0 is no risk, 4 is high risk). Project scores for each criterion are compared and the projects ranked. For example, in Table 18.2 Project Adrastea is ranked 5 for Strategic Fit because it scored lowest, 0 (note, some projects are ranked the same because their scores are tied); it is ranked 4 for ECV because it scored fourth in ECV value; and it is ranked 4 for Risk because it scored high in risk (tied with Projects Thebe and Europa). The Ranking Score column is computed as the average of the three rankings; for Project Adrastea it is (5 + 4 + 4) /3 = 4.33; this places the project seventh, and last.

Table 18.2 **Multiple Criteria Rank-Ordered List**

Project	Strategic Fit	Reward (ECV)	Risk	Ranking Score
Project Metis	4 (1)	2.3 (7)	3 (3)	3.67 (5)
Project Adrastea	0(5)	3.5 (4)	4 (4)	4.33 (7)
Project Thebe	2 (3)	3.1 (5)	4 (4)	4.0 (6)
Project 10	3 (2)	2.6 (6)	2 (2)	3.33 (4)
Project Europa	1 (4)	6.4 (1)	4 (4)	3.0 (3)
Project Ganymede	3 (2)	4.6 (3)	3 (3)	2.67 (2)
Project Callisto	4 (1)	5.3 (2)	2 (2)	1.67(1)

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.

Portfolio Balance[16]

Wise investors avoid taking on too much risk. Rather than put all their eggs into one basket, they diversify and try to balance investments 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 to balance projects that are gambles with projects that are safe bets.

A way to display this balance is with a “bubble chart.” In Figure 18.5 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, low);

similarly, the likelihood axis can be interval (0–100 probability) or ordinal (low, 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 success likelihood 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 in advance.

Bread and butter projects are the most common: rewards are low to moderate, but the success likelihood is high. Too many bread and butter projects, however, detract resources from the pearls and oysters and reduce future business opportunities.

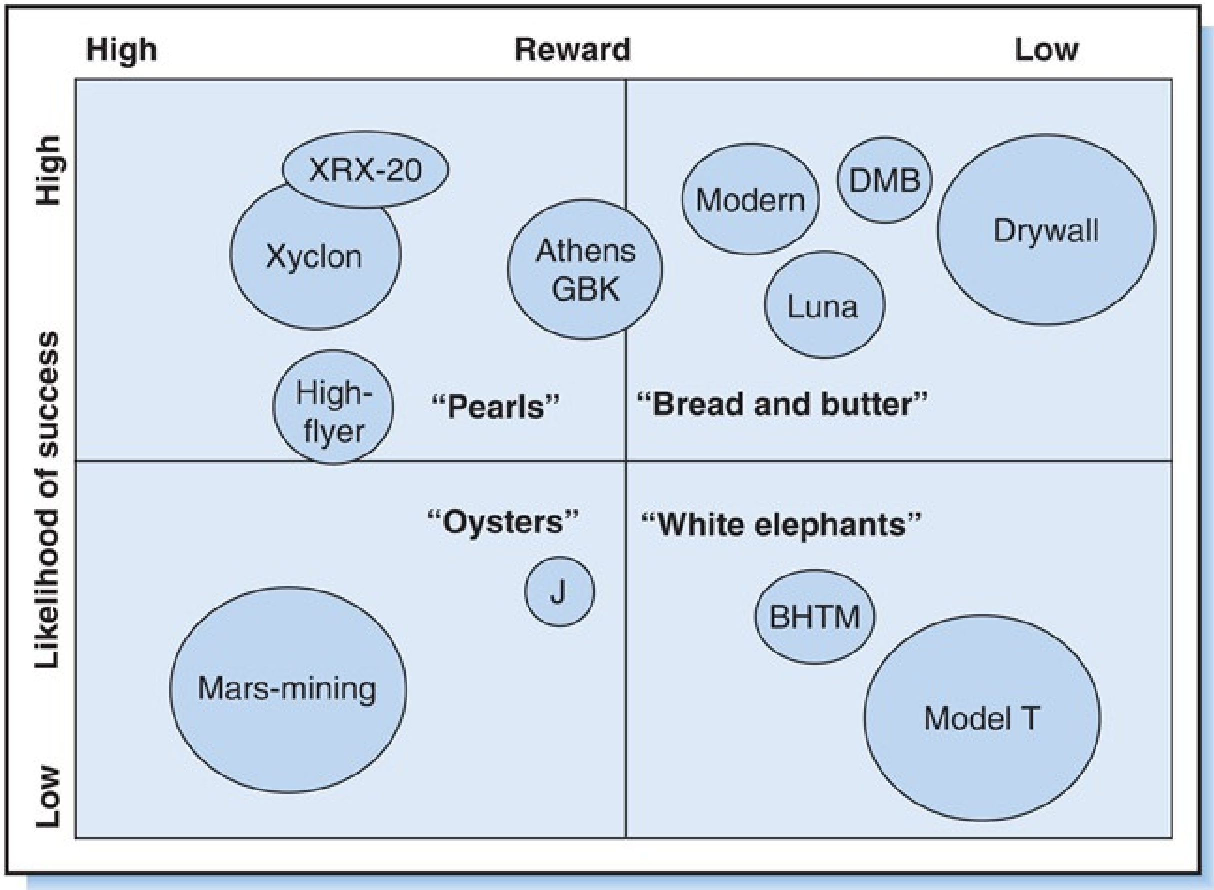


figure 18.5 Bubble chart for project likelihood of success and reward, and project size.

Source: Adapted from Cooper R., Edgett S., and Kleinschmidt E. (1999).

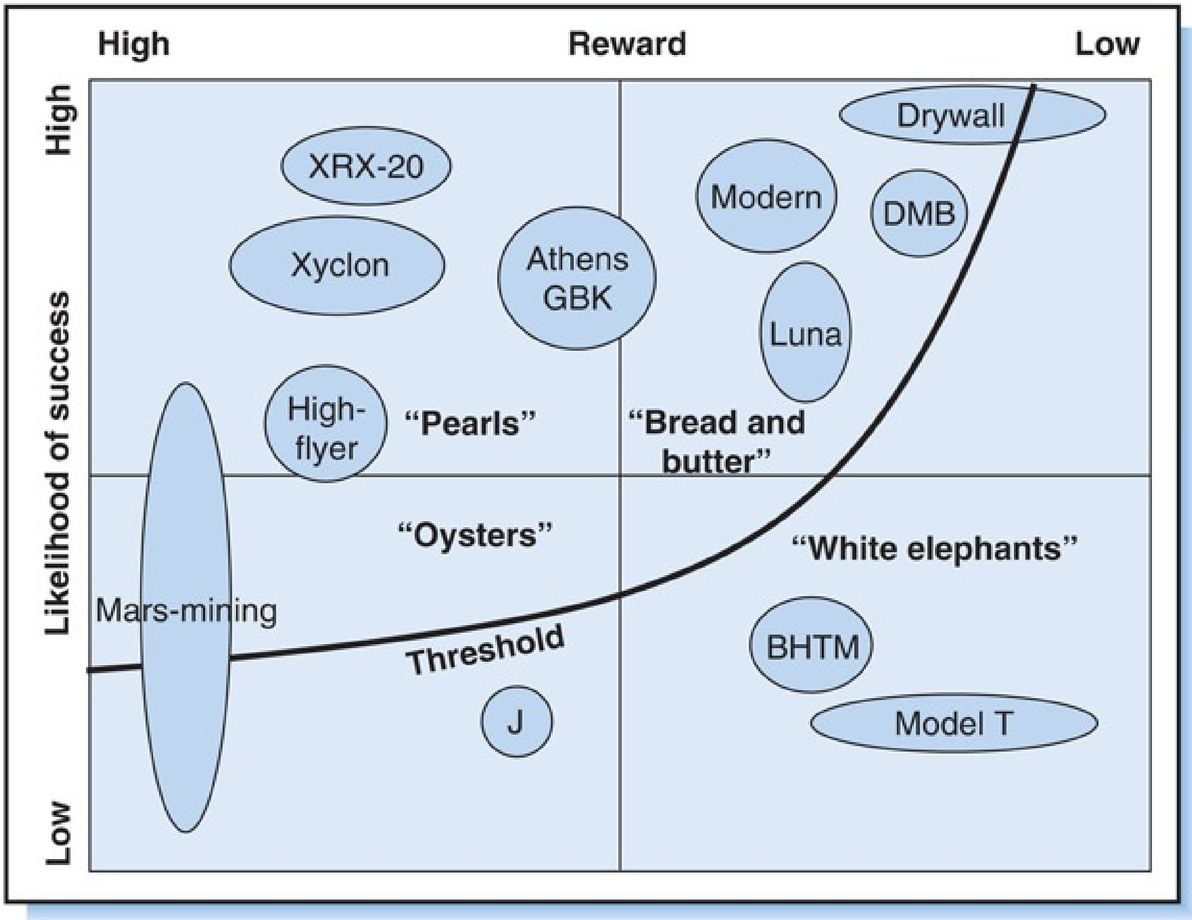


figure 18.6 Bubble chart for likelihood of success and reward, and uncertainty range.

Source: Adapted from Cooper R., Edgett S., and Kleinschmidt E. (1999).

White elephants are projects with low likelihood of success *and* low payoff. You have to wonder why a company would retain any projects in this category. Fact is, having spent money and effort on a project, companies feel they *should* continue; they stay committed to the projects because of funds already spent.[17]

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.

Figure 18.6 shows another kind of bubble chart wherein the size and shape of each bubble reflects uncertainty about the likelihood of project success and the reward; the larger or longer the bubble, the greater the uncertainty.[18]

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, "High-flyer," "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 18.6) above which projects are accepted, below which they are rejected.

Strategic Fit

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:[19]

- 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 18.3. Associated with each category is an allocated funding amount ($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 18.3 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 18.3 are rank ordered, the underlined projects represent the cutoff. In the last category, for instance, Project S is the cutoff; and Projects A1 and E1 will not be funded.

An approved project is admitted to the portfolio, but its ultimate execution depends on 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.

Table 18.3 Projects Rank Ordered by Category

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	Project R	3.1**	Project T	1.3	Project S	2.7**
Project W	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*	Project FI	1.9**	Total	11.7**	Project Z	1.2**	Total	8.8**
		Total	8.1			Project DI	2.2		
						Project HI	0.2		
						Total	9.4*		

* Required funding exceeds allocation.
** Cutoff project.

Deciding on the categories and the funding for each is top management's responsibility. Such decisions presumably are based upon consideration of organization 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.[20] 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 this way, the result looks like Grid A in Figure 18.7, 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, intangible benefits, fit with company business strategy, and so on, and plots the results on other grids (Figure 18.7).

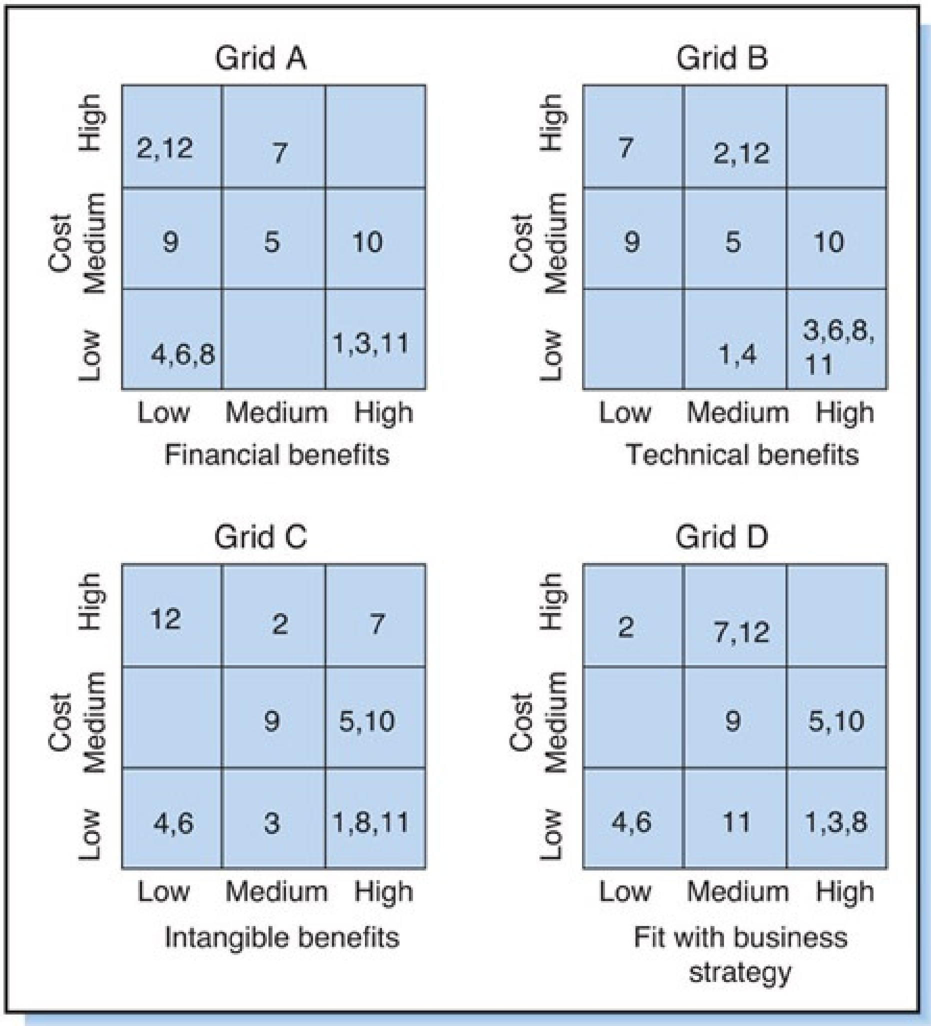


figure 18.7 Buss's cost–benefit grids, ratings for 12 projects.

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.[21]

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 18.7 projects 1, 3, 8, and 11 appear in the lower-right 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.).[22]

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 Analysis[23]

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 18.1. 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 18.4 shows three proposed projects and seven factors.[24] Each project is scored for each factor 0–100 for effectiveness (E) and for weighted effectiveness (WE = weight × 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 18.8 shows the three projects from Table 18.4 and five other projects. Projects *j, h,* and *m* (in the shaded area) fall below the minimal effectiveness threshold of 75 and would be dropped from consideration. The line connecting the uppermost points (*j, A, n, C*)—called the "efficient frontier"—represents the *maximum effectiveness* attainable for a given cost (or minimum cost for a given effectiveness level). Projects *B* and *k* are below this line, which means they are inferior to at least one other project in terms of both cost and effectiveness and should also be dropped. Only projects *A, 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.

Table 18.4 Cost-Effectiveness Data Analysis

		Project A		Project B		Project C	
Factors	W(weight %)	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			$1.9B		$2.0B		$3.0B

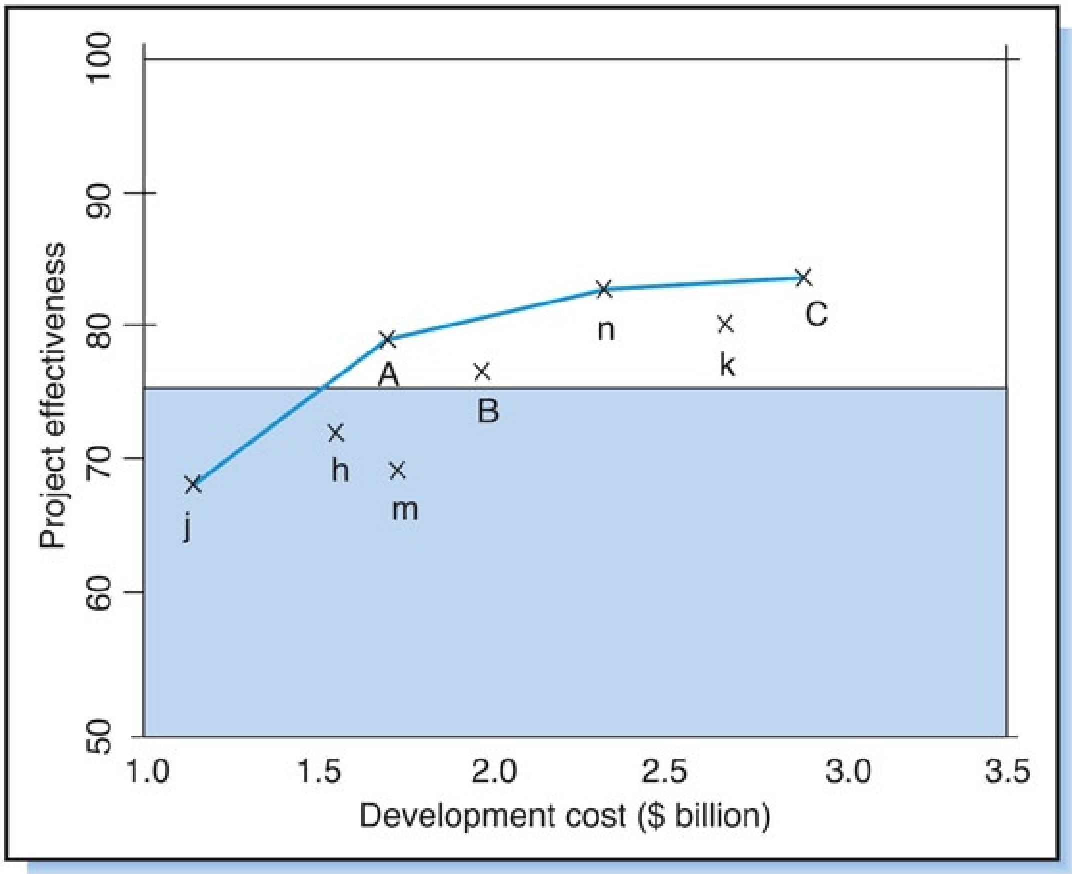


figure 18.8 Effectiveness versus development cost for eight projects.

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.

18.5 Integrating the Gating Process and Portfolio Management[25]

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. Underperforming or no-longer necessary projects are cancelled to make way for essential or more-promising projects. A sign of an effective portfolio management process is that periodically some projects do get cancelled.

In many companies, periodic project review happens through a gating or stage-gate process. Gating and portfolio management augment each other, but the two processes are very different. In gating, each project is reviewed at certain milestones or stages, and the fate of the project is based upon an assessment of its progress; 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 benefits, costs, and resources. This involves considerable effort and, consequently, might happen only three or four times a year—maybe less. Since companies are usually involved in many projects and proposals at any given time, each arriving at a decision gate at a different time, 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.

18.6 Summary and Discussion

Portfolio management is the process of choosing and managing those projects that best achieve organizational objectives subject to resource constraints. Portfolio management in combination with strategic management, the gating process, and project management help assure 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 review of approved new and existing projects. Top management establishes the 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 has approved.[26] 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 head and shoulders 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 benefit–cost 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 assure that the projects selected are the "right" ones.

Review Questions and Problems

1. What are the three or four 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 ("pick the low-hanging fruit")?
4. Compare the following; for each state the focus and how it relates to projects:

    - Strategic management
    - Portfolio management
    - 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:

    - Project W: Return, high; Risk, medium; Environment, medium; Service, low.
    - Project X: Return, medium; Risk, high; Environment, medium;

Service, low.

- Project Y: Return, medium; Risk, medium; Environment, high; Service, high.
- Project Z: Return, medium; Risk, medium; Environment, high; Service, low.

Create a scheme for screening the projects, assuming equal weight for all criteria. Which project comes out best; which 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, 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 earning were in euros instead of dollars?
17. In problem 15, suppose the likelihood of project success is $0.5 \times 0.8 \times 0.7 = 0.28$.

    - What is the B/C ratio for the project?
    - 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. Repeat the example in Table 18.2, but besides Fit, Reward, and Risk 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 ranking on list):

Project:	Public Image
Project Metis	0 (5)
Project Adrastea	1 (4)
Project Thebe	4 (1)
Project 10	3 (2)
Project Europa	4 (1)
Project Ganymede	2 (3)
Project Callisto	3 (2)

Include this ranking with the rankings for Strategic Fit, Reward, and Risk; re-compute the Ranking Scores for the projects based on all four criteria.

24. Draw a bubble chart for Risk versus Reward similar to Figure 18.5 for the projects in Table 2. Assume ECV < 3 is "Low" and ECV > 6 is "High." For risk, 1 is "Low" and 4 is "High." Use the Public Image ratings (not rankings) from problem 23 to "size" the projects on the chart: i.e., 4 is the largest project, 3 is smaller, etc., and 0 is a dot! Based on the bubble chart, which projects are pearls? Which are white elephants? Which are

bread and butter?

25. Top management has decided to reallocate funds among the five categories of projects listed in Table 18.3 as follows: $13M, $8M, $7.5M, $10M, and $7.8M, respectively. What are the cutoff projects in each of the five categories?
26. Explain how cost–benefit grids can be used to rank order projects.
27. Discuss similarities and differences between bubble charts and cost–benefit grids.
28. Suppose Project D is added to the projects in Table 18.4 and has been rated for effectiveness as shown in Table 18.5.

    Compute the total weighted effectiveness using the weights in Table 18.4. How does Project D compare to the others in Figure 18.8?

29. Once a project has been approved and admitted to the project portfolio, how is it monitored thereafter? Under what circumstances might it be cancelled?
30. 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?

Table 18.5

Project D	
Factors	E
Speed	80
Range	90
Efficiency	95
Comfort	85
Capacity	95
Loaded mass	90
Maintainability	80
Cost	$2.5B

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.

See Chapters 3 and 16

Note: Case 3.3 and Case 16.2 are related to topics in this chapter.

Case 18.1 Consolidated Energy Company

Consolidated Energy (CE) is a public utility that generates and distributes electricity throughout the US. The company is involved in many kinds of projects, including 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 five regions, information technology 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 the 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 if the IT, equipment, or construction unit for its region can handle the job; if so it assigns the job to them, otherwise it contracts the work using the RFP/proposal process. A corporate PRB makes decisions for projects that exceed $20M. When the PRB approves a project it awards the job either to the internal unit assigned to 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 giving the project automatically to the pre-assigned internal unit, it would send an RFP to *all* of the company's IT, construction, or equipment units. 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 transporting equipment and moving 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 18.2 Proposed Cement Factory for PCS Company

(Note, for this case you might wish to 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?”

See Chapter 4

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.

Endnotes

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. This section adopted from Cooper et al., Portfolio management in new product development, pp. 34–35; and Bridges D. Project portfolio management: Ideas and practices. In Dye and Pennypacker, *Project Portfolio Management*, pp. 97–116.

4. Levine H. Project portfolio management: A song without words? *PM Network.* July 1999.

5. Davidson Frame J. *The New Project Management.* San Francisco, CA: Jossey-Bass Publishers; 1994, p. 190.

6. Archer N. and Ghasemzadeh F. An integrated framework for project portfolio selection. *International Journal of Project Management* 17(4); 1999: 207–216.

7. Sommer, Portfolio management for projects: A new paradigm; and Bridges D. Project portfolio management: Ideas and practices.

8. Bridges D. Project portfolio management: Ideas and practices.

9. Project selection as a funnel is described in Wheelwright S. and Clark K. *Revolutionizing Product Development.* New York, NY: Free Press; 1992.

10. 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.

11. 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.

12. Ibid., 186–187.

13. Cooper et al., Portfolio management in new product development: Lessons from leaders, phase I.

14. Mathematical programming optimization approaches for project selection are described in the literature in operations research. For an 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.

15. Method described in Cooper et al., Portfolio management in new product development: Lessons from leaders, phase I.

16. Ibid. Bubble charts are easily created with commercial software (Google the term "bubble chart" for examples of methods and products).

17. 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.

18. Cooper et al., Portfolio management in new product development: Lessons from leaders, phase I.

19. Method described in Cooper et, Portfolio management in new product development: Lessons from leaders, phase II.

20. Frame, *The New Project Management*, pp. 181–185; Buss M. How to rank computer projects. *Harvard Business Review.* Jan–Feb; 1983.

21. Buss, ibid.

22. Frame, *The New Project Management*, p. 185.

23. Overview of method as presented in Shtub et al., *Project Management: Engineering, Technology, and Implementation*, pp. 127–130.

24. 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, Jenkinson L., Simpkin P. and Rhodes D. *Civil Aircraft Design.* London: Arnold; 1999.

25. This section adopted from Cooper et al., Portfolio management in new product development: Lessons from leaders, phase II; and Nelson et al., Building on the stage/gate: An enterprise-wide architecture for new product development.

26. On selection of dependent projects, see Dickinson et al., Technology portfolio management: Optimizing interdependent projects over multiple time periods.

Chapter 19
International Project Management

Consider three recent projects:

1. General Electric divided the development project for a new cardiac monitoring device between two teams, one in Milwaukee, one in Bangalore. The hardware development work was done by the US team, the software work by the Indian team. The manager coordinating them was based in Milwaukee but made frequent trips to Bangalore. The project required continual back-and-forth exchange of people, equipment, software, and information.
2. Bechtel, a US corporation with divisions worldwide, oversaw the construction of a complete industrial city in Saudi Arabia. As prime contractor it managed and coordinated on-site work, materials, and major systems provided by subcontractors from Europe, the US, and Saudi Arabia. The Bechtel project manager remained on-site during most of the project but traveled globally to meet with Bechtel senior managers and contractor associates.[1]
3. Boeing Commercial Airplane Division is the principle designer, systems integrator, and final assembler for the 787 commercial aircraft, but virtually all of the design and manufacture for the plane's major

> components and subsystems—wings, fuselage sections, engines, and instrumentation—is done by contract suppliers in Japan, Canada, Spain, Italy, and the US. Oversight and integration of suppliers and other Boeing divisions contributing to the program is run by Boeing's program management office in Washington State.

The obvious commonality among these projects is that they are "international" or "global" in scope. Unlike single-country, domestic projects where most or all stakeholders and project work are confined to one country, stakeholders in these projects are cross-national and cross-cultural, and project work happens in different countries.

19.1 International Projects

International projects have become ubiquitous as more companies establish divisions, seek customers, and outsource work to suppliers and contractors in different countries. Thanks in large part to lower costs and increased capacity of global air and sea transportation, enhanced communication technologies fueled by the Internet, and emerging business and technological capabilities in nations such as China, Brazil, and India, companies seek out and execute projects everywhere.

While such projects are enticing because of the benefits and opportunities that come with operating on an international scale, they are at the same time vulnerable to considerable 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 regardless of the issues and problems that the manager of a domestic, one-country project faces, the manager in an international project automatically faces an "extra layer" of issues. That extra layer touches most everything about management—leadership, interpersonal relations, stakeholders, communication, work definition, estimating, risk management, and work tracking and control. Language, communication, local customs, transportation, and infrastructure—all of little or no concern in a home-country project—become potential showstoppers in an international project.

19.2 Problems Managing International Projects[2]

Each new international project poses a new set of unknowns. To illustrate, think of an international project as analogous to a play with actors, scripts, sets, and props. Actors are the project stakeholders and social networks, scripts are the social institutions that guide and constrain peoples' behavior, the set is the project's work site, and props are the project technologies. Just as the actors, scripts, sets, and props differ in every play, so do the stakeholders, institutions, site location, and technologies differ in every international project. Such differences expose the project to potential mistakes and oversights in organizing, planning, and execution.

Table 19.1 lists aspects of an international project that tend to make planning and undertaking it 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 "unknown" the host country and its people are to the project manager and team members, 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. Ignorance about the unknowns makes it difficult to anticipate problems, set priorities, and act appropriately. It is why international projects often have trouble meeting schedule, budget, or requirements commitments.

Table 19.1 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 location 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, occupational, and legal" rules and regulations to the local level.[3]

19.3 Local Institutions and Culture

Stakeholders in international projects encompass different languages and cultures that influence communication, attitudes, behavior, work practices, decision patterns, and, ultimately, project performance. Additionally, they are 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 tend to share. Among many ways to measure culture is an oft-cited study by Hofstede of IBM employees in over 50 countries. The study identified five dimensions of culture.[3]

- Power distribution (PD). The extent to which the less powerful members of a culture accept or expect power to be unequally distributed, versus feeling 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 being part of and looked after by a group to which they are loyal.
- Achievement orientation (ACH). The extent to which roles are distributed along gender lines: "masculine," which implies assertive, tough, or achievement orientation, versus "feminine," which implies a more relationship, helping, or quality-of-life orientation. People from high ACH countries care more about earnings and signs of 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. People from low-UNA countries are more comfortable with ambiguity and feel *less* uncomfortable in the absence of detailed plans and formalized team roles and responsibilities.

- 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 19.2 gives some results. Larger values imply the tendency to accept unequal power distribution, be individualistic, seek achievement, and be masculine, more comfortable with uncertainty, and long-term oriented; smaller values imply the opposite.

For a project with team members in different countries, differences in these dimensions might merit attention. For example, a project team with members located in UK, US, and China might expect differences in terms of PD, IND, UNA, and LTO. According to the table, team members in China are more likely to accept authority differences than those in the US and UK; they are also likely more willing to "blend" into the team and not want to be singled out as individuals than members in the US or UK. US team members will possibly be slightly more comfortable with uncertainty or ambiguity than their colleagues in the UK or China; and members in China will likely be less influenced by short-term gains and incentives than those in the UK or US. Any of these might lead to different responses to management expectations by members from different countries.

One danger with findings like this is the temptation to generalize even though, of course, people are unique and don't necessarily fit the average. Some have criticized the findings for a number of reasons, including methodology and basic assumptions.[4] Nevertheless, the fact remains that people in different regions and countries *do differ*, which can be challenging when they have to work together. While the challenges might be significant even among workers from different developed nations, they are exacerbated when the workforce combines members from developed countries and developing countries (aka emerging economies).

Table 19.2 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

Like any challenge, the solution starts with airing of differences, and that might happen as part of a team-building session. As described in Chapter 16, one purpose of team building is for members to acknowledge their differences—in this case their values, belief systems, and expectations—and to develop guidelines for behavior that bridge those differences. Besides team building, project management should seek to strengthen interpersonal relationships, trust, and mutual respect, all of which tend to reduce stereotyping and build cohesion in team projects.

Of note is that national culture sometimes matters less to people than the culture of their profession or personal interests. This says, for example, that an Indian software engineer might feel more in common with an American engineer than his average fellow Indian.[5] A project manager might take advantage of such an affinity by developing "communities of practice" to overcome cultural differences and build team unity.

See Chapter 16

Language

When project stakeholders speak different languages, conversations and shared project documents such as scope, requirements, budgets, and contracts must be translated. The challenge is to make sure that every translation faithfully 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 may have different meanings; add to that 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 the Americans call the sidewalk; tell the Americans to walk on the pavement, they will walk down the middle of what the English call the tarmac."[6] US managers often 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 out important messages and documents, ask several people to interpret them. Napoleon did something like this: before issuing military orders he always had a corporal read them, reasoning that if someone of low rank could understand them, then certainly so would his officers.[7]

Often locals will claim to understand English when in fact their grasp of it is poor at best. When they pepper their responses with "yes, yes, yes," it is a sure bet they don't understand what's being said. When giving verbal directives, always follow it up in writing.

The manager of an overseas project should learn at least enough of the local language to conduct simple daily transactions. Doing so also shows respect for people of the host country, who are appreciative of visitors' (perhaps awkward attempts) to communicate in the local language.

Managers sometimes create a project glossary of terms that can be extensive and can even include pictures. For the 1960s project to develop the Anglo-French Concorde supersonic airplane, a special French–English project dictionary was created.

Formality

Whereas business associates in North America tend to address one another—subordinates, immediate superiors, and even senior managers—by first name, most everywhere else in the world they use some variant of sir, mister, or madam. Such formality extends to the way people 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 faxed, emailed, or verbally communicated, such practices in international projects pose questions regarding the country where agreements are made or contracts concluded, and, hence, whose contract law and court of law applies.

In some areas of the world, practices of little import elsewhere are raised to a high art. In Japan, for example, exchange of business cards is an essential part of business etiquette and comprises, what amounts to, a business card "ceremony."

Attitudes about Age

Many cultures associate wisdom with age. Older people automatically garner greater respect, reverence, and credibility than younger managers, regardless of experience. 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.

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 suitable or even expected elsewhere—like bringing a spouse to a gathering or talking to another's spouse—could be embarrassing and potentially ruin a business relationship.

Of course, offensive behavior and dress should always be avoided, although 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. People in Rome tend to dress more smartly than, say, in US cities, and a tourist from the US who would not draw any attention at home might come across as somewhat slovenly in Rome. When working in Rome (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. Certain gifts are considered acceptable, others not, and discretion is necessary to avoid violating local customs of etiquette or laws.

Food and Drink

Newly arriving expatriates often will scan local menus looking for familiar items —not knowing that the foods listed won't be the same as back home. Home-based or well-known restaurant franchises are more reliable and sometimes provide welcome familiarity. Meat portions in Europe and Asia tend to be small—miniscule by US standards. Meat and martinis might not be on the menu—or on any menu anywhere in the country—and to even ask about them is utterly inappropriate. The rule of thumb concerning food and drink—but applicable to everything about local customs—is to be respectful, polite, and accepting, even when the customs do not suit your taste or predisposition.

Attitudes about Time

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. A Western manager accustomed to filling every minute with work will be annoyed by the many "time-wasting" gatherings organized by his Asian or Middle-Eastern business associates; they, in turn, will be insulted by *his* angst to get on with business and will question his motives and loyalty to them.

Holidays, Weekends, Vacations

Every country has its own non-work holidays. The US has seven national public holidays; most European countries have, but Germany has 16. A project that involves participants from, say, four countries, each with five national holidays could conceivably face 20 days of holiday downtime. The Ramadan and Chinese New Year holidays affect the schedules of many projects. Even when different counties share the same holidays, exact dates may differ. The Christmas holiday runs in the US December 23 through January 2, but in Russia and some Eastern European countries it is December 31 through January 8—sometimes later. 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. While these differences create problems for some projects, they offer opportunities to others by enabling work to continue at different places around the world seven days per week.

Vacation time-off also varies by country and region. Whereas in the US two- or three-week vacations are standard, Australian law prescribes four weeks, as does the European Union—usually the whole month of August. Some countries mandate by law six weeks' vacation plus another six for sick leave.

Labor Time

What constitutes a "usual" workday and workweek also varies. French law mandates and enforces a notto-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 the "work day" is between

6 am and 2 pm, the manager of a 9-to-5 project will probably see her local workforce falling asleep around 3 pm.

Layoffs

Although commonly in the US employees are terminated when a project ends and there is no follow-up work, in some countries such termination is *not* automatic, especially for workers who served 12 continuous months on the project. What is a manager to do with these employees? In many European countries labor laws dictate who an employer can lay off and how the employer must 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 very difficult to get new jobs."[8] French employers often 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 home country of the developer or contractor—although US contractors working overseas must confusingly also comply with US law, and the trick is to not violate laws in 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 just to ignore them (of course, risking the possibility that at any time the rules *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, formalities, terminology, regulations, and laws, international contracts take longer to finalize than domestic contracts. Getting the wording and terminology right on contracts is extremely important, and even the littlest 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 his 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 in 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."[9]

The contractor must be sure to include stipulations and actions in the contract to protect its intellectual rights and be prepared to take action should it discover that its ideas, products, or technology are being pirated.

Litigation, Payment, Meeting Contract Terms

Contracting in international projects is a whole topic onto its own and beyond the scope of this book. While some projects employ standard-format contracts (e.g. FIDIC or NEC; see note 10, Chapter 5), many 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 any legal disputes would be litigated in a neutral country, i.e., neither the host nor the contractor's home country. US contractors often specify England.

See Chapter 5

Each contract should provide stipulations to assure that the customer will receive its deliverables and the contractor its payment. This would seem customary even in single-country, domestic projects, 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 the contractor, the contract might specify a large first payment followed by payments upon meeting frequent time-phased targets. Payments are often delayed, not by the customer but because international funds transfer typically requires approval by an agency of the host country, which can take 60 days. Sometimes payments to foreigners must be made via tax agents, further complicating the payment process.

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 very good and customer creditworthiness is difficult to ascertain.

Politics

National and local political stability and the government's position regarding the project are potential risk factors. Radical 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 such as the US, they are common elsewhere. But such events rarely materialize at short notice and without warning signs. A contractor in an international project must have reliable people in the host region to monitor these signs and keep project management 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—plus what happens when cross-cultural teams ignore integration.

Example 19.1 The Chunnel Project[10]

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 teams represented two different cultures and the competition only aggravated the differences and exacerbated problems.

For starters, ideally contracts are written in one language and governed by one legal system, but the Chunnel 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 differ with regard to 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. It was decided that where a difference existed between the standards of the two countries, the higher should prevail—though it was not always obvious which standard was the higher (e.g. the 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 door width from 600 mm to 700 mm took 9 months.

19.4 Local Stakeholders

Contractors[11]

Project teams operating in foreign countries are often required to hire local contractors. Although sub-contracting 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, thus erasing any potential savings (many countries like India, however, have low labor costs yet productivity as high as in Western nations). A local contractor who is familiar with local customs and bureaucracy can sometimes cut through red tape and avoid hassles that would stymie a contractor from the outside.

Selecting a local contractor goes beyond the usual criteria of skill, experience, resources, and financial stability. One consideration is the likely quality of the contractor's communications as determined by language and culture. Another is the contractor's familiarity with common business practices. Practices taken for granted in most countries (e.g. RFPs, proposals, SOWs, change controls, and status reporting) may be unfamiliar to a local contractor and challenging for it 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 is always important, but even more so in international projects. In general, whereas Westerners tend to first set contractual agreements and then build relationships, Easterners build relationships first and then reach agreements. Regardless of the professional track record of the project manager and his company, local businesses, subcontractors, vendors, 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.

Example 19.2 How to Ruin a Business Relationship[12]

Negotiations between a US company and a firm in India to finalize the contract on a promising project began with a series of informal meetings. Soon after arriving in India, the American project manager sensed that his customers were unnecessarily dragging their feet, so he tried to urge them along. But the more he tried, the more the Indians doubted his motives and the less they trusted him. As is their custom, they had planned to delay serious talks until after becoming acquainted with the American—a trust-building process intended to occur during a few days of after-hour dinners and social gatherings. The project manager, however, was expecting serious talks to begin soon after his arrival and conclude after no more than a few days. Because the negotiations were in English and most of the project work was to be done in the US by a US team—and only later to be transferred to the customer's site in India, the project manager hadn't bothered to familiarize himself with Indian social customs; in other words, he blew it. The negotiations failed and the manager flew home without a contract.

19.5 Geo-National Issues

Many issues regarding international projects arise from the simple fact that the stakeholders are dispersed across different nations or geographic regions.

Exchange Risk and Currency

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/$US; by December 12 it fell to R15.89; and by January 12, 2016 it traded at R16.16. For any South African project that depended on imported items this exchange rate change could have posed serious consequences.

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 19.3 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 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 = €928,571. Instead of a tidy €100,000, the contractor profits only € (900,0000 − 928,571) = €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 €900,000(1.3) = $1,170,000 to €900,000(1.4) = $1,260,000). However you look at it, the contractor made less profit.

One way to reduce exchange risk is to lock into the contract today's price for a payment that will not occur until later. Called *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.[13] Another way to reduce risk is for both parties to agree upon and specify the exchange rate in the contract. The amount of payment is thereby determined by the rate set in the contract, not the rate at the time of payment. A third and the most common way to reduce exchange risk is to "*forward cover,*" or transfer the risk of an unfavorable change in currency value to an insurance company.

Offsets[14]

Foreign contractors on large government-funded projects are often subject to requirements 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 supplied 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 an "indirect offset," requires the contractor to contribute to non-project endeavors such as business enterprises or improvements to roads, communications, or other infrastructure, the purpose: to reduce the net amount of 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. Sometimes the trick is for the contractor to satisfy the offset requirement, yet still make a profit.

Offset requirements are specified in the RFP, and sometimes a contractor wins the job based primarily on the offset plan as described in the proposal. In essence, the offset is the deal-clincher, exceeding in importance the principle work of the project.

Export/Import Restrictions

The export/import of certain US technology, software, and hardware are 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.

Time Zones

Project stakeholders located in different time zones might have no overlapping normal business hours, and messages between them might take days to read or respond to. Avoiding communication delays is largely a matter of planning, such as scheduling work hours in the zones so as to allow 2–3 hours overlap, and assuring easy accessibility of the project manager and other key participants via cell phone messaging and email during critical stages of the project.

In projects that require frequent travel across multiple time zones, jet-lagged managers and team members need more time to get up to speed.

19.6 Project Manager

Typical problems in an international project:[15]

- Team members need travel visas.
- Someone on the project team does not have a valid passport.
- Someone on the team 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.

At times like these the first place people go is to the project manager, expecting her to be able to handle the predicament personally or know where to get help. While dealing with such issues the project manager must continue to deal with project-related problems both on-site and back home.

The project manager must be largely self-sufficient. Faced with unique challenges and often 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. A sense of humor helps, as does prior work experience in international projects.

Sensitivity and Acceptance

The project manager must understand local norms and customs and be able to develop trusting relationships with business associates and customers in the host country. 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 be able to 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, and 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 require careful instruction and explanation. 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!

Nor should it be assumed that, because a process or method succeeded in one country, it will do so in another, or that local laborers and suppliers will automatically accept the process or method. Making assumptions without considering the local sentiments and attitudes can create resentment and resistance among local staff.

The project manager might need to adjust her leadership style according to the culture. For example, people in Hofstede's high-power distribution cultures might need or expect more coaching than those from low-power distribution cultures.

Among the challenges of managing a cross-cultural team are being aware of and dealing with biases when appraising team members. In general, the tendency is to appraise people from one's own culture higher than those of 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, possesses skills to bridge cultural differences. Such skills include:[16]

- 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 is in the middle of everything, managing the project not from a remote office but at the project site. She is always or frequently on hand to see what is happening and discuss problems with local managers, staff, and workers. She is fully committed to the project and remains at the site until the project is completed and the customer has signed off.

Members of the team witness the project manager making decisions that affect the project and them personally. The project manager must be in constant touch with her team and available to assist them when they need help—not only with project decisions but with documents, currency, housing, or medical assistance; in this way she earns their gratitude, respect, and commitment. When the project manager cannot be on-site or works with a virtual team, she must remain engaged through frequent emails and instant messaging, 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 someone who workers see as visibly engaged and fully in charge, a *local project manager*. 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 responsible for on-site, detailed planning and daily management. The local manager reports to the global manager; the responsibilities and authority of the two are clearly delineated and understood by the project team.

At time of hiring, the local project manager should be informed about expectations, responsibilities, and performance targets, and then periodically reminded. Hiring and training a good local project manager is not easy, so when a problem arises she should be given every opportunity to work it out. If the problem is serious and thought to be getting worse, the global project manager should "parachute in" a trusted person to assess the situation and offer assistance. Only when the situation is deemed hopeless should the local project manager be replaced. But that can cause a 6-month delay as the new manager settles in and attends to family and other (survival) issues.

19.7 local Representative[17]

Every international project needs 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. This person—the *local representative*—is responsible for:

- Representing the project manager and company to the customer, and vice versa.
- Keeping the project sold to customers and supporters.
- Arranging for in-country services (hotel and car reservations, local communications, interpreters, office staff and space).
- Arranging meetings with government officials, attaches, and consulates.
- Educating the customer about home–country government requirements; e.g. the transfer of technology and technical knowledge.
- Helping arrange local housing for project personnel.
- Assisting in locating local subcontractors.
- Informing 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 sensitive to and comfortable with local customs and culture. Also, the local rep must be committed to the project and not eager to race off as it nears completion.

When the project has a local project manager, ideally that person also serves as the local rep unless, however, she is not familiar with local culture, customs, and stakeholders, in which case she should have a local rep.

One way to find a local rep is by partnering with a local company for a portion of the project work. In effect, the partner becomes the local rep. Qualifications of the partner combine those described in section 19.4 with capability to perform the contracted work, ability to communicate, and ethical reputation.

19.8 Top Management, Steering Committee, and PMO

Practically everything about an international project is more difficult and takes longer. 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, top management should create a committee to guide the project and assign the PMO a role to help manage it.

Steering Committee[18]

The steering committee (or review board) for an international project includes senior managers and sponsors from both company headquarters and the host country/region of the project. For a global project comprised of multiple project sites, the manager in charge of the overall project (i.e., 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 a "local," steering committee. This committee is comprised of local sponsors and managers and, for a global project comprised of multiple project sites, the local project manager. This committee plans and executes details of the project, and handles problems originating at the project site or host country. Serious issues that it cannot resolve are forwarded to the executive committee.

Role of PMO[19]

In addition to the functions described in Chapter 17, the PMO for international projects does the following:

- Assists senior management in assessing and selecting international projects.
- Collects lessons learned from international projects, and incorporates them into templates, checklists, and training sessions.
- Follows up on issues and problems identified by management that require coordination among multiple international projects.
- Manages files and documentation for international projects.
- Identifies project managers 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.

In general, project personnel going overseas should be well informed about the project and know 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, the local project rep, and numerous other matters, big and small.

See Chapter 17

19.9 Team and Relationship Building

The project manager kicks off the international project with a team-building session for key members from 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 possible 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.[20] A useful exercise is for each participant to express how much he assumes people from other cultures are willing to adapt to his culture, and how much he is willing to adapt to theirs.

The contractor should also hold a session with each local subcontractor to discuss issues that might arise and prepare a plan to prevent or mitigate them. At the session they determine which tasks they will do individually and which together. Ideally a large portion of the work packages (20–30 percent) will be performed jointly by teams from both the host and the home countries. This will encourage local workers to take ownership in the project yet allow the contractor to retain control over the work.

Beyond building relationships with local project team members, the project manager must develop relationships with stakeholders in the host country. If the project becomes embroiled in serious problems, having personal ties with local and national government, trade, labor officials and vendors will come in handy. To this end the project manager should make time to attend social events with local officials and celebrate local holidays and cultural events.

19.10 Project Definition

An international project cannot be approached in the same way as a domestic project. Potential issues in the project associated with culture, country, laws, people, and politics must first be identified.

Where to Start

How do project managers learn what they need to know in each international project? Here are common ways:[21]

1. *Look at examples of similar projects* done in the country by your company or others and try to learn what they did. Seek out project managers with experience in the host country or region and ask for advice.
2. *Hire a credible consultant or freelance expatriate* to provide guidance and serve as a cultural intermediary with local stakeholders. Seek those who have project experience and have developed a social network and local connections in the host region.
3. *Ask trusted guides, professionals, or international advisory groups* for advice about local politics, norms, customs, business practices, and the economic environment. Although they might not be familiar with the business or technology of a particular project, they will know about local labor, resources, and laws.
4. *Attend formal training programs* devoted to coping with foreign stakeholders, institutions, and environments.
5. *Start with a small pilot project* in the country to allow time to become familiar with the culture and laws before committing to larger, more risky projects.
6. *Create a culture risk management team* to identify potential cross-cultural and cross-national issues and steps to reduce or avoid them. The membership of this team should mirror the national and ethnic groups

of the project stakeholders.

Customer Requirements

Most projects begin with a list of customer needs and wants, which the contractor later expands and converts into a list of technical requirements. In a multi-language project the process gets complicated because the customer's list must be translated into the contractor's language, then the contractor's list must be translated back into the customer's language for approval. The process can be lengthy though, typically, Western managers are eager to get it done as quickly as possible. But non-Western managers, taking a different stance, may prefer to hold off on defining the details and first build relationships and establish areas of agreement. The attitude is, not to worry, disagreements over details are inevitable but will be worked out. Building trust and establishing areas of agreement are key responsibilities for the project manager; they must not be delegated to someone in business development, sales, or marketing, as happens in domestic projects.

Scope and SOW in Global Projects[22]

For a global project that consists of sub-projects at multiple international locations, a 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 country and subproject, 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. They also make suggestions to the global 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 19.3.

Because of differences in culture, norms, and languages, sub-projects that start out with almost identical purpose, scope and SOW often end up varying substantially. To accommodate differences in purposes, goals, etc. (Table 19.3, rows 1–8), the global steering committee adjusts the scope and SOW (rows 9 and 10) for each subproject. In the course of back and forth iterations between the global and local steering committees, the scopes and SOWs of the subprojects and the global project (row 11) are mutually adjusted and made compatible.

Table 19.3 Impacts of Country Differences on Global and Local 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			

The intended outcomes of the process are that:

1. Local project managers and teams are involved in and 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 are in agreement.
5. Goals, scope, and SOW of the sub-projects align with those of the global project.

Work Definition and WBS[23]

Work definition must account for the many additional 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 starting template lays out the first-level breakdown of activities or end-items, general areas of work, and resources needed, and might look not much different than for a one-country, domestic project. Then, each first-level activity is assigned to a team member who will be responsible for managing it (presumably the person who knows the most about it). This person, who might be the local project manager, subdivides the activity into detailed task definitions with estimates for resources, time, and cost.

Thus far the work-definition process is not much different than for a domestic project. In an international project, however, 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 truly unique. Although a generic kind of project repeated in each of several countries might look the same in terms of high-level technical 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. Table 19.4) must be addressed with detailed tasks within work packages or by additional work packages.

Table 19.4 Issues in International Projects

• Team members speak different languages.
• Expatriate team members need vaccinations, passports, visas, etc.
• Expatriate team members need local room, board, 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 project due to other higher-priority needs.

Besides discovering issues through the WBS process described above, a separate cross-cultural/cross-national WBS devoted entirely to international issues (Figure 19.1) can be created. This is done by a special "culture risk team" whose sole purpose is to identify and deal with cultural/international issues. The first-level breakdown of this WBS might consist of the following work packages:[24]

1. Identify important international and local issues and factors in the project.
2. Assess risks associated with these issues and prepare plans to address them.
3. Provide support for overseas personnel on the project.
4. Provide team-building and relationship-building support.
5. Manage knowledge obtained for this and other international projects.

As Figure 19.1 illustrates, the two WBSs provide a dual-pronged approach to help assure that no important international issues are overlooked; any redundancies that appear in both WBSs are simply consolidated.

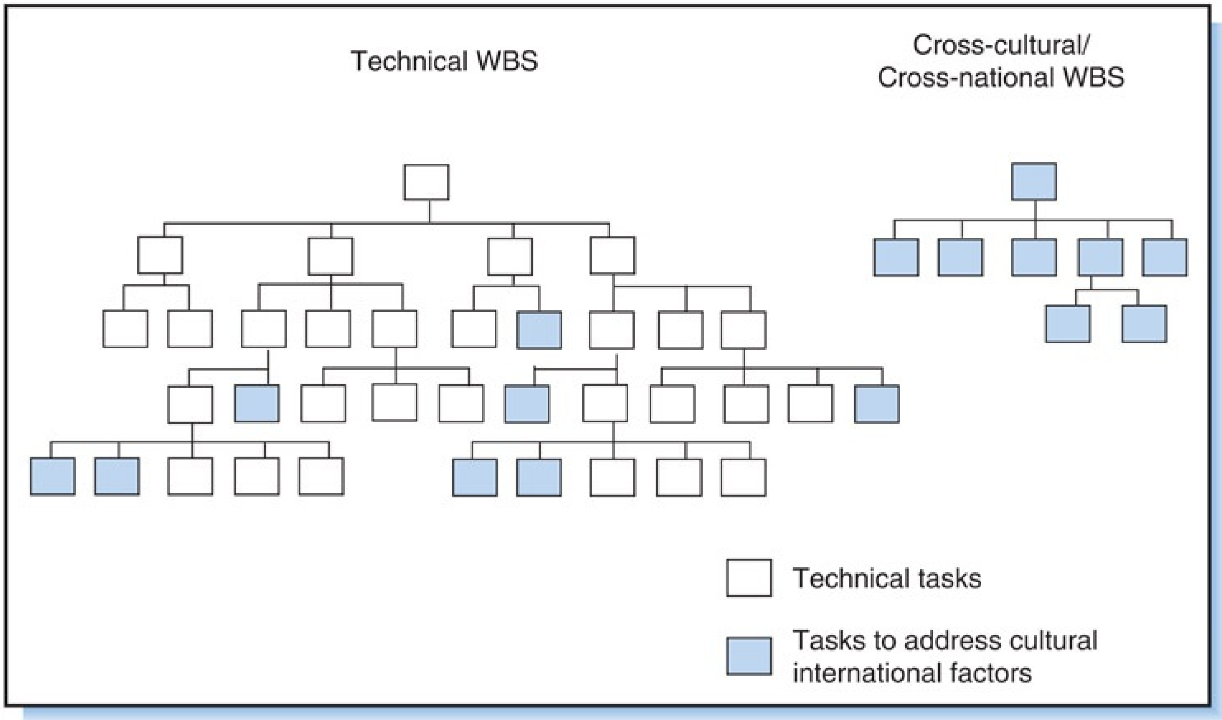


Figure 19.1 WBSs for an international project.

One way to keep track of the detailed tasks and work packages in a global project is with a summary matrix, shown in Table 19.5. The matrix reveals which tasks are unique to certain sub-projects and countries and which are common among many or all. It also suggests places where knowledge gained from one sub-project might be used in another, and helps ensure that important tasks or issues are not overlooked.

Work Packages and Responsibilities

Since in general, smaller work packages are easier to track and control than larger ones, the technical WBS should ultimately be subdivided into small packages of short duration and measurable outcomes. Early in the definition process, however, such a detailed breakdown will neither be possible nor—because of the many unknowns—desirable. Nonetheless, once the project is underway and the picture of pending activities becomes clearer and the unknowns fade, the work can be defined in greater detail. As with phased project planning, the WBS and plan are continually reviewed, and the immediate, upcoming work packages are subdivided into detailed, short-duration tasks, ideally of no more than 2 weeks each.

While the WBS is being created, so is the responsibility matrix to show the responsibilities of all parties working on or supporting the project—customer, subcontractors, and other stakeholders, both at home and at the project site/host country.

Table 19.5 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	
Tasks Addressing Local Issues			
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. Downloaded September 9, 2005 from http://www.etpint.com/globalproject.htm.

Resources, Schedule, and Budget[25]

Any estimates for resources, time, and cost based upon domestic experience must be revised when applied to overseas projects. Planned resources must be adjusted for differences in equipment and labor productivity levels, and schedules and budgets adjusted for the time and costs for communication (fax, phone, courier, translators), travel (air fares, car rentals, taxi and limo fares) and arrangements for conferences and local services. The budget must include fees and costs for insurance, licenses, governmental reviews, local housing, overseas work salary incentives, automobile, daycare, schooling, security, and medical care. Expenses and lead times for obtaining passports and visas, and transporting managers, workers, and replacements in accordance with the project schedule must also be accounted for.

Besides those factors already mentioned, adding to time and cost in international projects are shipping preparation, transport between countries, customs inspection and clearance, and transportation in the host country. Transport time in the host country depends on the quality of roads and on available airport, harbor, trucking and other local services. If the only available transport to or from the project site departs only once a week, missing it by a minute could result in a week's delay. Any material or equipment to be brought in from the US but deemed as "transfer of technology" must first be approved by the Department of State, which can take months. Fluctuating exchange rates should also be anticipated, for example by forecasting the impact of, say, a change of + 10 percent in the euro on the project estimated cost at completion. All of these extra activities make international projects, *ceteris paribus*, more costly, lengthy, and risky than domestic projects.

Example 19.4 Added Time and Cost of an International Project

A contractor working on an overseas project encountered bad weather that fouled the equipment and stopped the project. Back home, the contractor simply would have brought in other equipment more appropriate for the weather, but in the host country that equipment was not available 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 relatively 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 that of, say, an American engineer and would be compensated for by extending the project's engineering work schedule. On the other hand, if lower labor costs of local engineers would allow hiring several of them to replace one American, then extending the project schedule might not be necessary. But rarely are such tradeoffs easy to determine in advance.

Example 19.5 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, in all cases these engineers needed significant time before they became as productive as the local engineers due to factors such as time to "settle in," lack of personal networks, lack of knowledge about local companies and processes, poor understanding of the cultural environment, and communication problems; these put expatriate engineers at a disadvantage and reduced their productivity, at least initially, and restricted them from working at their full potential. As a consequence, expatriate engineers were given only technical assignments, whereas South African engineers with similar qualifications and experience were given assignments that included management responsibility.

Training

Often, much preparation goes into training and coaching expatriate managers and staff in the culture, traditions, and regulations of the host country. Typically overlooked, but sometimes as important, is training local managers and staff in the culture, common business practices, and technical procedures of the contractor and the home country. Cultural adjustment is a two-way street. For training of locals, the strategy and setting must be carefully designed since the Western mode of classroom lecture-discussion is not very effective in some cultures.

19.11 Project Monitoring[26]

The project manager should make certain that every local subcontractor understands her expectations for communication and progress reporting. She should require that the local project manager and team leaders submit weekly task updates; in an international project this is simplified by posting project plans and updates on the Internet. Assuming that technical work packages have been defined to be of relatively short duration—no longer than 2 or 3 weeks—the project manager will be able to readily discern whether work has been completed, is on schedule, or is behind.

When a local subcontractor starts to fall behind or miss requirements, the project manager needs to step in and take a more 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 than to resort to legal action.

The project manager who cannot be on site will rely heavily on telephone and teleconferencing to communicate with locals. Good practice is to precede all verbal communication with written communication so local workers will know what to expect and be prepared, and then follow up with any written directives or action plans. This will help reduce misunderstandings among parties, common in international projects.

The project manager of an international project must make her presence known; if she cannot always be on-site, then she should make frequent visits—unannounced. Nowhere is the value of site visits and visibility more important than in international projects.

19.12 Communication

Communication Plan[27]

Like other projects, an international project should have a communication plan. Besides the contents described in Chapter 12, this plan must address difficulties arising from differences in languages and time zones. Also, it should specify important contact persons (Who's Who) in the host country, home country, and elsewhere. Everyone—domestic and foreign project staff and subcontractors—must understand the required reports and written communication, and the content and format of each. Foreign contractors and local project staff might not be familiar with "common" project documents and have to be taught why they are important and how they will be used.

A common "working language" should be adopted for all or specific portions of the project. Those not familiar with the working language should be given accelerated language lessons; everyone using the common language 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 of the key project stakeholders.

See Chapter 12

Meetings

The communication plan should include a tentative schedule for all formal reviews and milestone meetings, and describe the meetings' format, expected content, advance preparations, time limits, attendance policy, and who will lead. Since formal meetings in international projects can be difficult to schedule, require much preparation, and expose people to cultural gaffes or imbroglios, it is best to have as few of them as possible. Also, the project manager should meet with local customers or officials *before* formal meetings to report any major problems; no one should be shocked by what they hear in a meeting.

The primary method for tracking status and identifying problems should be one-on-one communication and frequent *informal* meetings, convened as needed, the time and place determined by urgency and purpose, e.g. alternate weeks if everything is okay, more often if not, and at the location experiencing the problems or issues. Attendance should be restricted to contributors to the meeting or people who would benefit from being there. As with domestic projects, the project manager should be the person who takes notes, writes them up, and distributes them.

See Chapter 16

The team in an international project might be dispersed across multiple locations around the globe—be it a distributed or virtual team—and most meetings occur via electronic meeting technology or audio- or web-conferencing. The recommendations for "virtual meetings" in Section 16.8 in Chapter 16 apply as well to international projects, with the addition of a few more. In the interest of building personal relations and team trust, every meeting should begin with casual conversation. Devote a few minutes for members to talk about their families, hobbies, interests, etc., and allow time before a holiday for the local team to explain the customs of the holiday. It's not so important that members know much about each other but rather that they show they *care enough about each other* 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."[28] 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. Scheduling London–Cape Town calls and meetings will pose few problems; scheduling London–San Francisco meetings, that's something else! Meetings held during mealtimes should be avoided, although what's "mealtime" varies around the world. In North America dinner is around 6 pm; in Europe, India, and elsewhere, 8 or 9 pm is more common.

When time differences are big, share the pain in scheduling meetings.[29]

Chicago and Mumbai are 10.5 hours apart and allow no convenient meeting time for teams 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). The rule should hold even if Chicago is the project home office with 30 people and Mumbai is a satellite with only six people. Rotation reduces perceived "power" differences among members and helps build trust and respect. If the project has people dispersed around the world, the number affected can be minimized by requiring only a few members or a rep from each place to participate in meetings (again, rotating times). This alternative is never as good as everyone participating (just as virtual meetings are never as good as face-to-face) but compromises are sometimes necessary.

To raise awareness, the project manager should distribute a guide to all team members showing each member's country and the hours difference between time zones. It should also list times, days, mealtimes, and holidays when team members have said they cannot or prefer not to meet.[30]

19.13 Risks and Contingencies

International projects are fraught with risk, though often the risks are subtle or hidden and can be exposed only by viewing the project from the perspectives of the different cultures and countries of the project stakeholders. Any standing risk policies of the contractor or customer (described in Chapter 10) 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 constant, no matter the project or country.

As discussed in Chapter 10, risk analysis begins in project conception and definition by imagining different scenarios about what could go wrong. Project risk is associated with level of uncertainty: the less certain you are about something, the greater the risk. In an international project much of the uncertainty relates to ignorance about local culture, customs, language, institutions, infrastructure, and stakeholders. Thus, learning is an important strategy for reducing risks in international projects: the more you know about these matters, the better you can identify and mitigate the risks.

See Chapter 10

Another strategy, however, is to decrease the amount of learning necessary to deal with local regulations, laws, and resources. This is done in the following ways:[31]

- *Outsource activities that are heavily restricted by local regulations.* Purchasing land, obtaining permits, hiring locals, and moving materials through customs are risky because they require knowledge about local laws and customs. By outsourcing these activities to knowledgeable subcontractors, the burden of responsibility (and much of the risk) is shifted to the subcontractors.
- *Perform technology-intensive work at home.* Rather than dealing with the uncertainties of local labor, materials, and infrastructure, do most of the work on major hardware and software deliverables at home and then

transport them abroad to the site 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 in a neutral country where the laws are more familiar. This practice is mandatory in countries where local laws are unclear or enforcement unpredictable.

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. The mix depends on the kind of project. In general, the more a project requires the contractor to be "imbedded" in a foreign country, the more the contractor must learn about the country, its laws, and culture. Contractors such as Fluor and Bechtel performing large construction projects *are* heavily imbedded in the local environment because the projects take years, have large scope, and rely somewhat on local resources. Hence, the firms must learn about the country or region of the project, which they do by hiring local contractors, local laborers, and expatriates who thoroughly know the country. They also methodically manage all knowledge gained about the host country. At the same time, they reduce their need to learn about *everything* by prefabricating deliverables at home wherever possible, outsourcing to local suppliers and contractors, and hiring local representatives to deal with local stakeholders and freelance expatriates to manage technology and contracts.

Of course, the on-site project manager of an international project is always "imbedded" in the host country—even when the contractor (his employer) is not. Although knowing the local ways and protocols might not matter to his firm, it does matter to the manager who has to live and work in the host country for as long as the project takes. 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 trusting relationships with leaders, subcontractors, laborers, and officials in the host country.

19.14 Summary

A project that is international in scope 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 tracking and control.

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. Thus, gaining personal familiarity and building relationships is a fundamental aspect of managing international projects. Besides "domain competency" over technical aspects of the project, the project manager must be self-sufficient, adaptable in unfamiliar environments, and able to understand and respect local culture and customs.

When the project manager cannot always be on site, a local project manager should be appointed to handle detailed planning and daily management. In addition, a permanent "local representative" should be appointed to update the project manager on local matters, mediate with local stakeholders, and help resolve local issues.

Each global project should have an executive steering committee to oversee governance and funding, and set goals and coordinate work among sub-projects at different sites. Each sub-project should have a local steering committee to plan and execute details and handle local problems.

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. The project might have two WBSs, one for technical aspects of the project, another for cultural or international aspects. The fact that most everything takes more effort, time, and cost must be factored into tasks, schedules, and budgets.

The project manager must provide firm goals and direction to local managers and subcontractors. Ideally she is on-site; if not, she makes frequent visits, unannounced.

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.

Review Questions

1. Consider the analogy of an international project to 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, which unknowns 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 you are familiar with. 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 following the project cause legal problems for the contractor or employer?
7. For an overseas project, whose laws prevail, the host country or the home country?
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”? Compare 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)? 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 and host countries?
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 about 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 similar to or different from a technical WBS for a single-country, domestic project?
28. Name some of the issues the WBS in an international project might have

to address.

29. Describe the purpose and content of the summary matrix in Table 19.5.
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, and in establishing budgets and schedules for an international project.
32. What special issues should the communication plan for an international project address?
33. What are some strategies for handling risks in international projects?

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 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 make to resources, time, and cost estimates to account for differences in countries supplying labor and materials to the project?
8. What strategies were employed to identify and reduce project risks?

Case 19.1 Mozal Project—International Investment in an Undeveloped Country[32]

Mozal is a $1.4 billion project launched in 1998 to construct a 250,000 tons per annum (tpa) aluminum smelter in Mozambique (Figure 19.2). The idea of such a project at first seemed preposterous. To build such a large, modern, state-of-the-art production facility would require international financing and stable supplies of raw materials and labor, but Mozambique was one of the world's poorest nations with an infrastructure in ruins after two decades of civil war. Yet the project was a success, completed months ahead of schedule and well under budget. It is worthwhile seeing how that happened.

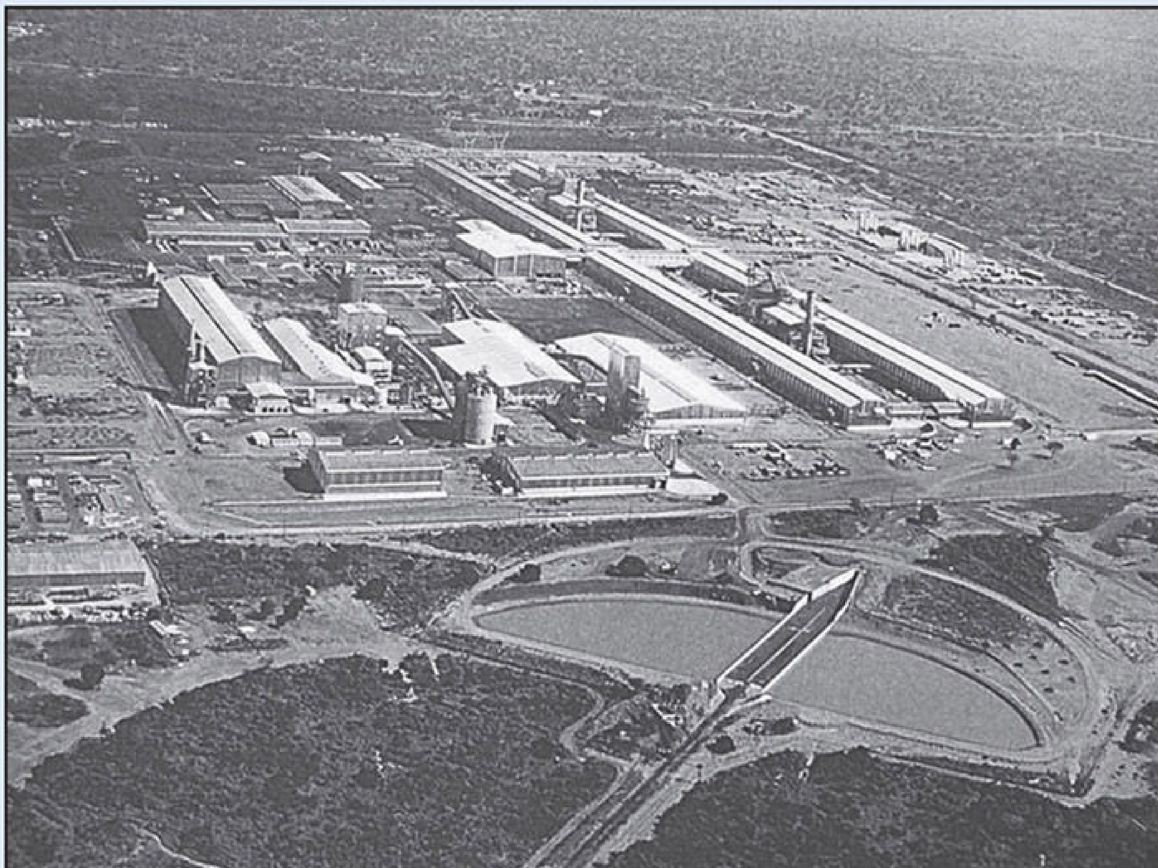


figure 19.2 Mozal aluminum smelter.

Mozal's primary promoting and controlling shareholder was Gencor, a large South African mining firm (later a part of BHP Billiton) that had recently completed the world's largest (500,000 tpa) Hillside smelter in Richards Bay, 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 the site for another smelter.

Mozambique

The team chose Mozambique for several reasons (Figure 19.3). Its capital Maputo offered a suitable (though run-down) harbor for importing alumina and exporting aluminum, plus abundant low cost (though largely unskilled) labor. Also, the South African power utility Eskom saw an opportunity to extend its power grid into Mozambique. The grid would provide Swaziland with reliable power and, later, would 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; this enabled them to build relationships with stakeholders throughout government and the community.

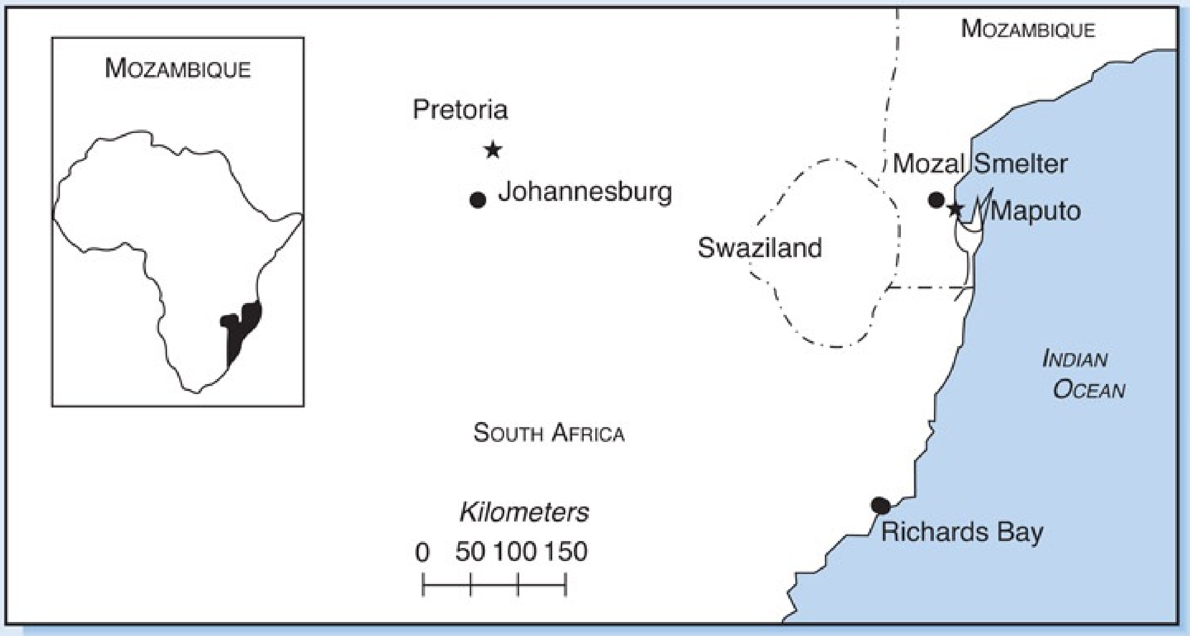


figure 19.3 Mozal smelter and surroundings.

Financing

Another sponsoring shareholder for the project was the Industrial Development Corporation (IDC), a development bank of the RSA government created to seek investment opportunities that promote economic stability. IDC agreed to provide low cost financing, export credit, and guarantees to South African 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 offered important benefits to the region. All major cash inputs and outputs were set in US dollars to minimize currency exposure.

Risk Mitigation and Go-ahead

The project's production costs 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 the go-ahead.

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, 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 crossing 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, though 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 parts of the site and service corridors, land mines laid during the civil war had to be cleared. Construction of cross-country power supply lines and, consequently, commissioning of 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.

Study Assignments

1. You are the newly appointed director for the proposed Mozal project. The feasibility study is complete and you must convince the international sponsors and lenders to commit to the project. Develop a presentation to a special board of stakeholders asking for the go-ahead to commit $1.4 billion to the project; address their expectations and how you will deal with the perceived risks.
2. The project has received the go-ahead and you now face the reality of mobilizing your team and starting 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 19.2 Spirit Electronics' Puerto Rico Office[33]

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; this is 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 received from US firms seemed extremely high; this plus the fact that labor laws in Puerto Rico require that some jobs be performed by local vendors led Susan to select mostly Puerto Rican vendors.

Spirit wanted the project completed within 30 weeks. Since cost bids from the 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 since 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 threatened to cancel 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 must be included in the schedule to account for unknowns. She proposed delaying the target completion 8 weeks but Spirit objected. She was able to create a schedule to meet the original target by paying the government $20,000 to rush the permits.

As the project progressed Susan had to respond to several other 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 room dimensions known. To avoid this, the building design was changed so millwork could be premade.

- Long lead times on permits (3–16 weeks). She submitted drawings and permit applications far in advance, showing dates when permits would be needed.
- Disorganized furniture installation vendors. Susan made the vendors create a plan (from which she estimated 8 weeks completion time) and then held them to it.
- Local labor pool dichotomy: extremely high-cost (five times more expensive than in US but able to reliably meet expectations) or extremely low-cost (uncertain ability to do quality, on-time work). Typically Susan hired the first.
- Added cost and time for imported materials due to import tax and shipping costs, and 6 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 (site superintendent, IT personnel, carpenters and laborers). For tasks requiring 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?

Endnotes

1. Jubail City - 2: SR 200bn Industrial Hub in the Making, *Zawya.* http://www.zawya.com/story.cfm/sidZAWYA20050321123628. Downloaded June 12, 2006.

2. Adopted 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.

3. 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.

4. 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.

5. Carmel E. *Global Software Teams: Collaborating Across Borders and Time Zones.* Upper Saddle River, NJ: Prentice Hall; 1998.

6. Turner J. *Handbook of Project-Based Management.* New York, NY: McGraw-Hill; 1998, p. 483.

7. Murphy O. *International Project Management.* Mason, OH: Thompson; 2005, p. 23.

8. 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.

9. International Chamber of Commerce, *Policy and Business Practices, Incoterms 2000.* http://www.iccwbo.org/policy/law/id315/index.html. Accessed April 11, 2007.

10. 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.

11. Murphy, *International Project Management*, p. 63.

12. Adapted from Mathew M. *Doing Business in India: A Cultural Perspective.* www.stylusinc.com/business/india/cultural_tips.htm. Downloaded September 25, 2006.

13. Foreign Exchange. *TD Commercial Banking.* www.tdcommercialbanking.com/foreignx/products/forward.jsp. Accessed March 10, 2006.

14. Murphy, *International Project Management*, p. 63.

15. Lientz B. and Rea K. *International Project Management.* Amsterdam: Academic Press; 2003, p. 49.

16. Duarte D. and Snyder N. *Mastering Virtual Teams*. San Francisco, CA: Jossey-Bass; 2006, p. 134

17. Murphy, *International Project Management*, pp. 65–66.

18. Lientz and Rea, *International Project Management,* pp. 44–45.

19. Ibid., p. 56

20. Ibid., pp. 71–72.

21. Orr, *Strategies to succeed in foreign environments*, p. 5.

22. Similar approaches discussed Lientz and Rea, *International Project Management*, Chapter 2.

23. Lientz and Rea, *International Project Management*, Chapter 2.

24. Grove C., Hallawell W. and Smith C. *A Parallel WBS for International Projects*. Professional Knowledge Center, Grovewell LLC, 1999. www.grovewell.com/pub-project-wbs.html. Downloaded January 25, 2006.

25. These and other considerations discussed in Murphy, *International Project Management*, pp. 72–73, 88, 91–94, 178.

26. Lientz and Rea, *International Project Management*, pp. 124–128

27. Ibid., pp.155–170

28. Cohen M. *Succeeding with Agile.* Upper Saddle River, NJ: Addison-Wesley, 2010, p. 375.

29. Ibid, p. 377.

30. Duarte and Snyder, *Mastering Virtual Teams*, p. 68.

31. Orr, *Strategies to succeed in foreign environments*, pp. 6–7.

32. Personal communication, Rob A. Barbour, former chairman, Mozal.

33. 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
RFP for Midwest Parcel Distribution Company

Following is the RFP for the proposed LOGON system sent by Midwest Parcel Distribution Company (MPD) to contractors perceived as most capable of meeting the requirements. Only partial entries are shown to minimize the length of the example. 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 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 for proposal.

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 discussion of the Chicago distribution facility, projected future needs, and purpose and objectives 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 operational capability for the LOGON system. All necessary testing of systems and subsystems designed and installed by the contractor, as well as of current 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 discussion of 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 … (*Discussion of services, equipment, and material provided by MPD or other contractors and for which the contractor is NOT responsible.*)

Scheduled Delivery Date

LOGON system is to be fully operational on or before April 30, 2021. All hardware, software, and support services necessary for full system operation will be supplied and/or completed by April 30, 2021. Site installation will initiate no later than November 30, 2020.

Subcontractors

With the proposal, the contractor will shall submit a list of subcontractors and work to be assigned to each. Subcontractors will be subject to MPD approval prior to placement of a 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 April 30, 2021.

Section 3 Proposal Content and Format

Proposal will include the following sections and conform to specific 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) copies of the completed proposal along with all MPD confidential information to:

Lynn Joffrey
Administrative Assistant
Midwest Parcel Distribution
Company 13257
N. Wavelength Avenue
Chicago, IL 60699, USA
(773) 773-7733

Deadline

Proposal must be received by MPD by 5 pm August 15, 2019.

Selection Date and Criteria

Selection and Award Date

September 5, 2019.

Section 5 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 requested drawings, specifications, requirements, and addenda 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 are Appendices containing forms, agreements, and supporting technical data, standards, and performance requirements necessary for preparing and submitting a proposal.)

Supporting Technical Data

1. Technical data attached in Appendix C of 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
   N. Wavelength Avenue
   Chicago, IL 60699, USA
   773-7733

Appendix B

Proposal for Logistical Online System Project (LOGON): *Submitted to Midwest Parcel Distribution Company from Iron Butterfly 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**

    1. Cover Sheet
    2. Executive Summary X
    3. Statement of Work

        A. Background Statement of Need X
        B. Technical Approach and Distinguishing Features X
        C. Project Plan and Schedule (Forms from RFP: II. Work packages; III. Deliverables; IV. Work schedule; V. Subcontractors) X

    4. Budget and Price (Project Price: $14,413,905)

        A. Budget and Price (Form VI from RFP) X
        B. Variations, Changes, Contingencies X
        C. Billing and Payments X

    5. Project Organization and Management Plan X
    6. Qualifications and Key Personnel

        A. Company and Prior Projects X

B. Resumes of Project Manager and Project Engineer X

7. 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 in 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 project management of 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 administration 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

We recognize that 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 is to 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 and 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 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. (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 4 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 containing 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 unloads it into the vacant slot. Shipping containers are stacked three high in seven rows of racks (Items 2 and 3). The facility storage capacity 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 keyed in at the dock terminal workstation and the database system identifies all containers with buckets with 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 system uses six robot drone transporters that operate independently and simultaneously. The drone transporters retrieve the buckets and transport them to the loading dock for placement of parcels into departing truck. The longest specified retrieval time is 6 minutes. A seventh drone transporter will be included as backup.

(*Discussion continues about features of the robotic 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 (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

1.	Commence basic design	May 2020
2.	Basic design review	July 2020

3.	Process/Track design approval	September 2020
4.	Computer system specs review	October 2020
5.	Hardware Groups A and B received	December 2020
6.	Begin installation at site	January 2021
7.	Finish installation of complete system1/2	March 2021
8.	Final user approval	May 2021

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., 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: 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, Mr. Wesley, 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 fulfilment of system technical requirements at the site. Ms. Melissa has worked at IBC for 7 years and on IBC's three most recent robotic 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 in the business of 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 Company to design the first fully automated robotic warehousing system, and in 2011 we installed the first operational 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 design of the Model EZ was modified for warehousing application 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,* *and certifications. 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

Memorandum

Iron Butterfly, Corp.
Elegant design. Built to last.

To: SEE DISTRIBUTION	Ref. Job No.: 904-01
From: Frank Wesley, Project Manager	Date: January 3, 2020
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

Logistical Online System Project Execution Plan

I Management Summary

On September 5, 2019, 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 and a computerized database 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 subcontractors 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, 2020 and final system approval by MPD Co. will happen on or before May 2, 2021. 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, 2021. 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 Project Description

On September 5, 2019, IBC was awarded the contract for the LOGON System Project. The award followed a 1-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.

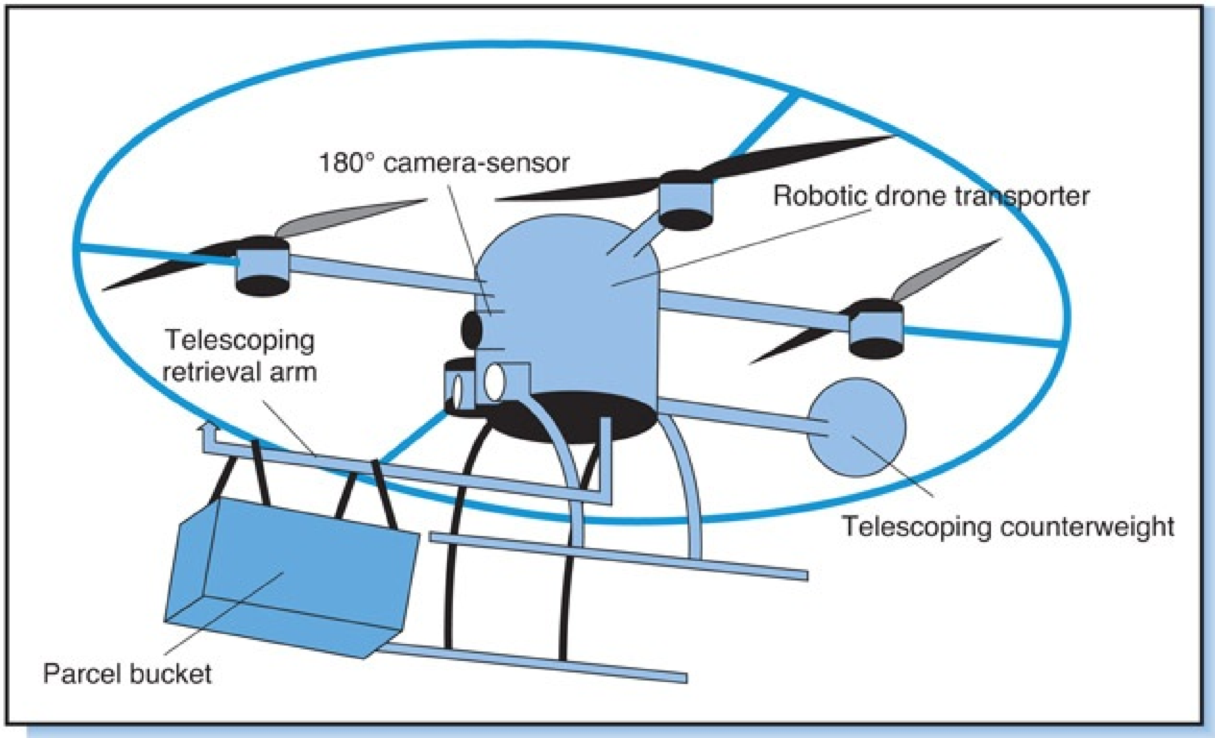
The project consists of designing, fabricating, and installing a parcel transport, storage, and database system (LOGON) for placement, storage, and retrieval of standardized shipping containers. The system will substantially improve the speed of parcel handling, increase the utilization of storage facility 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 for automatic placement and retrieval of parcels and record keeping. The system works as follows:

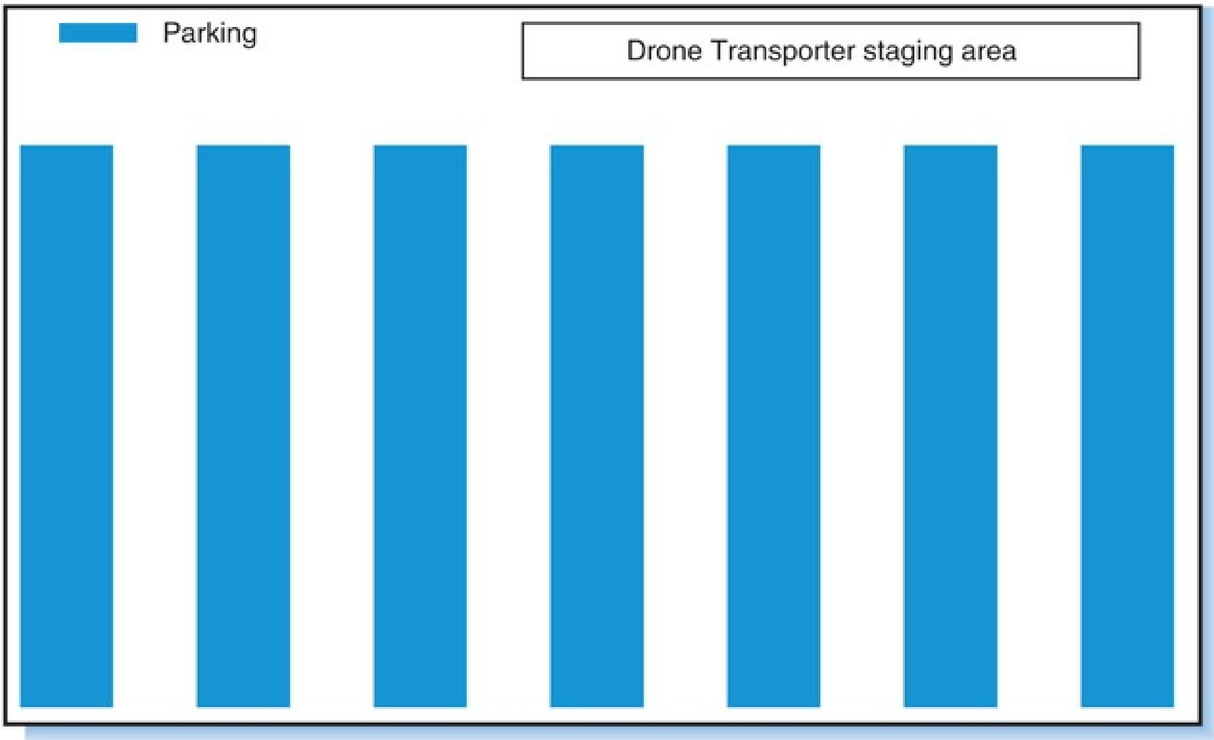
Upon a parcel's arrival at the distribution center receiving dock, it is placed into one of three standard-sized parcel "buckets" that are electronically coded as to parcel item and shipping destination. This code is relayed to a master database from any of four terminal workstations. The workstations 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 and 350,000 hours of industrial service with no accidents.

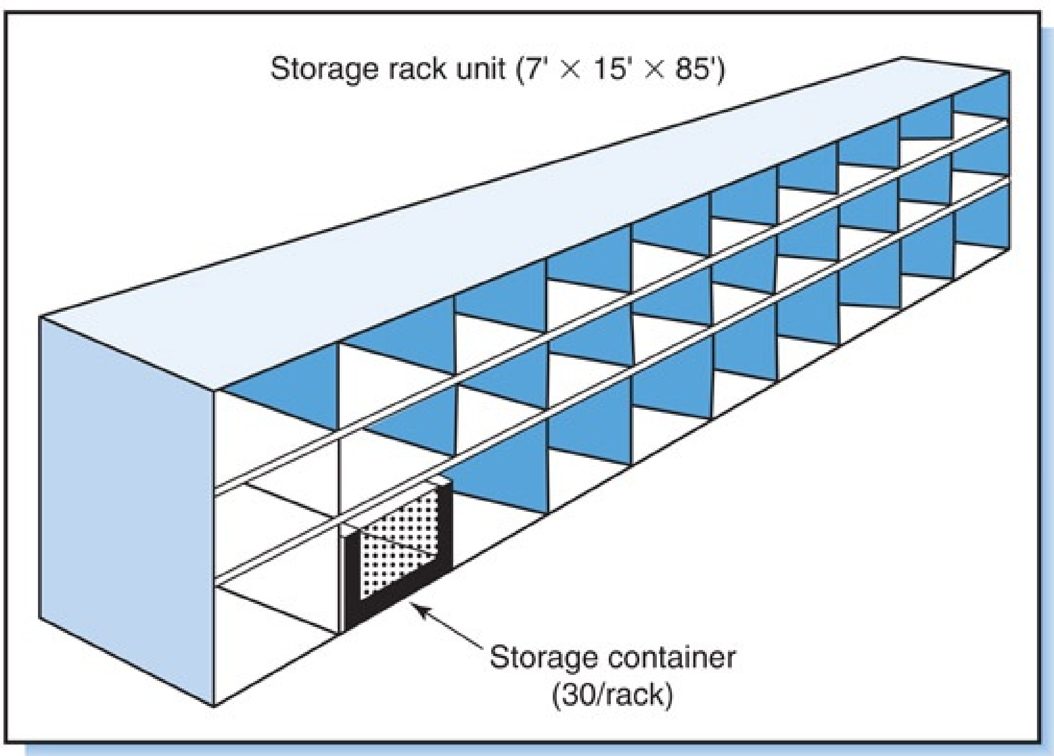
The transporter carries the bucket to a "suitable" vacant storage slot within a shipping container located on a rack in the facility. The computer determines which shipping container has a vacant slot of sufficient size and containing parcels going to the same or nearby destination as parcels in the drone transporter's parcel bucket. The transporter then conveys the bucket to the appropriate shipping container and unloads 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 1 Robotic drone transporter.



Item 2 MPD site layout.



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 on 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 try to initially move the project to an advanced stage. Remaining design work will use as much as possible of 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 the project manager for IBC and the project director for MPD. Project personnel may correspond directly with the client or subcontractors for information, providing the project manager and project director with copies of memos and conversations.

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, reporting of progress, 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

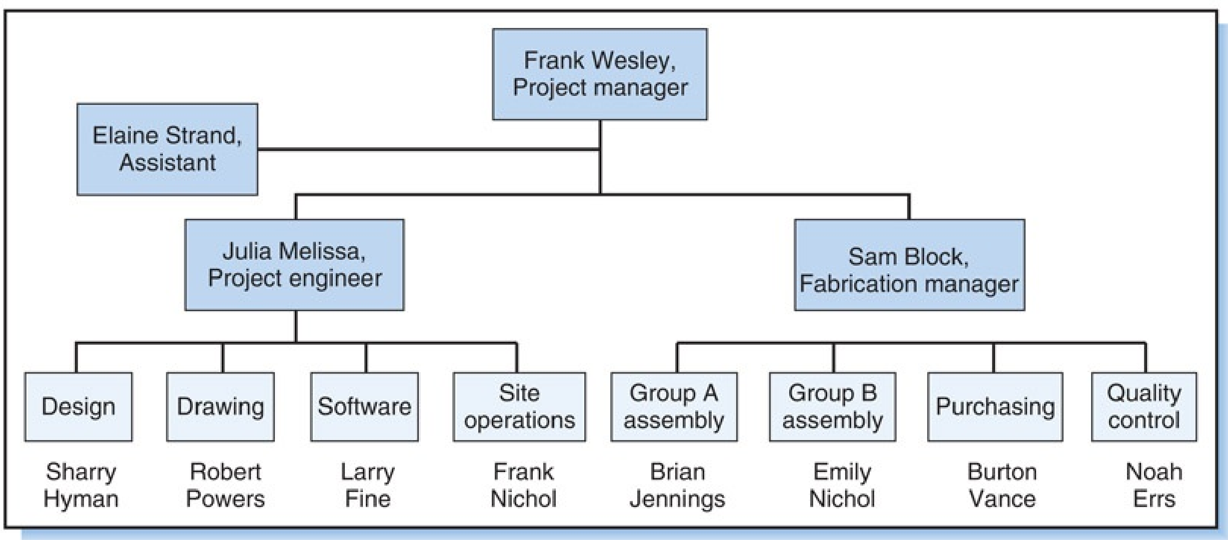
Key personnel at the four primary subcontractors CRI, SEI, UPC, and CRC are:

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

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 previously named first four parties or their substitutes. Software specifications related work with CRC will be coordinated by the CRC customer representative. Project telephone conversations between IBC and subcontractors shall be noted in handwritten memos and copies sent to the IBC project engineer.



Item 4 LOGON organization chart.

Responsibility code

P Primary responsibility
S Secondary responsibility
N Must be notified
A Must give approval

	Persons responsible																												
		Project manager																											
			Project engineer																		Fabrication manager								
			Design				Drawing					Software				Site operations						Assembly A				Assembly B			
Project task or Activity	F.W.	J.M.	S.E.H.	R.L.Q.	P.J.	D.V.R.	R.I.P.	O.E.M.	P.V.P.R.	D.M.N.	R.L.	L.S.F.	L.L.L.	J.R.S.	D.V.Q.	F.W.N.	J.M.M.N.	L.O.T.	A.U.A.	D.A.R.	S.O.B.	E.N.	G.G.F.	R.T.T.	B.V.L.	B.J.	T.T.Y.	H.R.D.	B.V.-Purchasing
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									

Item 5 Project responsibilities.

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 formal meetings are included in the respective subcontractor agreements.

CRI	S 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
Lynn Joffrey	Administrative assistant, Chicago
Cecil Party	Financial manager, Chicago
Mary Marquart	Operations manager, New York

Changes or modifications to the agreement requested by MPD or by IBC will be acted upon by the operations manager 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 the MPD New York office. Fifteen meetings are included in the agreement. MPD will provide information and perform services on the project 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 manager. 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 robotics seminar 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.

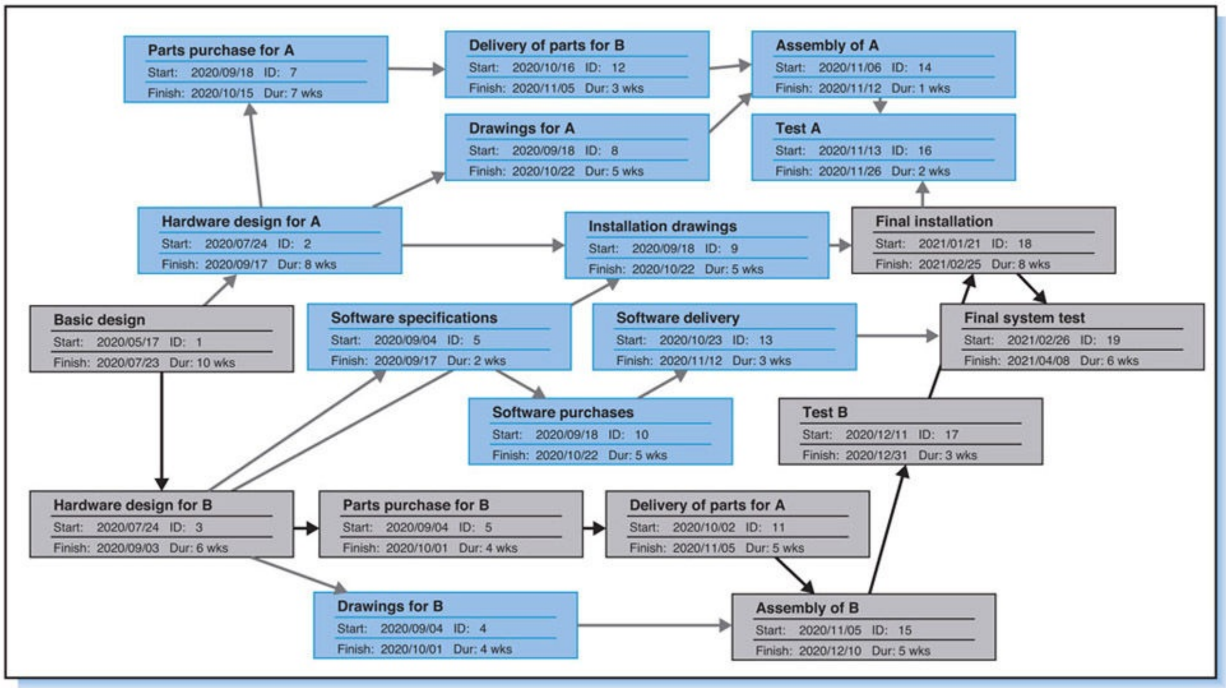
IV Technical Section

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 6 (letters refer to task designations on Item 6):

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).



Item 6 Principal subtasks.

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 Calendar

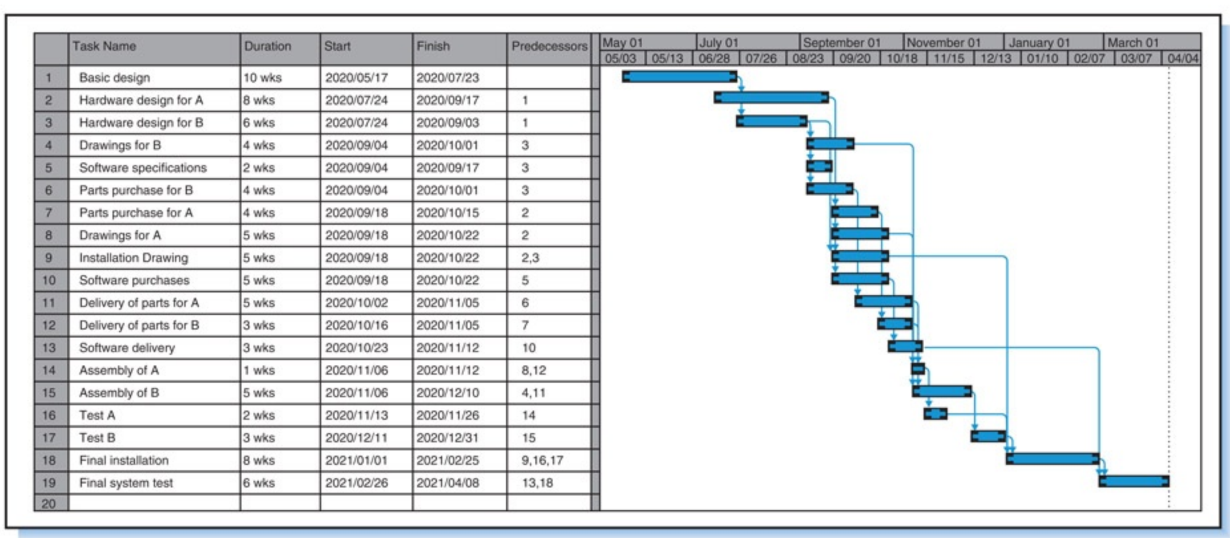
The project will commence with basic design on or before May 11, 2020; installation at the site will begin on or before January 10, 2021; and final system approval by MPD Co. will be on or before May 2, 2021. The schedule for significant aspects of the project is in Item 7. The indicated milestones are:

1. Commence basic design	May 11, 2020
2. Basic design review	July 26, 2020
3. Transporter and conveyor design review	September 6, 2020
4. Computer system specs review	September 20, 2020
S. Hardware group A and B review	November 29, 2020
6. Begin installation at site	January 10, 2021
7. Final user approval	May 2, 2021

Starting dates for activities 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.

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.



	Task Name	Duration	Start	Finish	Predecessors
1	Basic design	10 wks	2020/05/17	2020/07/23	
2	Hardware design for A	8 wks	2020/07/24	2020/09/17	1
3	Hardware design for B	6 wks	2020/07/24	2020/09/03	1
4	Drawings for B	4 wks	2020/09/04	2020/10/01	3
5	Software specifications	2 wks	2020/09/04	2020/09/17	3
6	Parts purchase for B	4 wks	2020/09/04	2020/10/01	3
7	Parts purchase for A	4 wks	2020/09/18	2020/10/15	2
8	Drawings for A	5 wks	2020/09/18	2020/10/22	2
9	Installation Drawing	5 wks	2020/09/18	2020/10/22	2,3
10	Software purchases	5 wks	2020/09/18	2020/10/22	5
11	Delivery of parts for A	5 wks	2020/10/02	2020/11/05	6
12	Delivery of parts for B	3 wks	2020/10/16	2020/11/05	7
13	Software delivery	3 wks	2020/10/23	2020/11/12	10
14	Assembly of A	1 wks	2020/11/06	2020/11/12	8,12
15	Assembly of B	5 wks	2020/11/06	2020/12/10	4,11
16	Test A	2 wks	2020/11/13	2020/11/26	14
17	Test B	3 wks	2020/12/11	2020/12/31	15
18	Final installation	8 wks	2021/01/01	2021/02/25	9,16,17
19	Final system test	6 wks	2021/02/26	2021/04/08	13,18
20					

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, 2021. 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 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 is a 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.

Item 8 LOGON project cost estimate.

Task	Labor time	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				
	5,000	48	240,000	60,000				
		Total	800,000	200,000	20,000		102,00	1,122,000
Project development	1,000	112	112,000	28,000				
	1,000	80	80,000	20,000				
		Total	192,000	48,000	45,000		28,500	313,500
System design	125	112	14,000	3,500				
	375	96	36,000	9,000				
	375	48	18,000	4,500				
		Total	68,000	17,000	6,000	1,550,000	164,100	1,805,100
H Basic hardware	750	120	90,000	22,500				
	4,000	96	384,000	96,000				
	3,500	60	210,000	52,500				
		Total	684,000	171,000	54,100		90,910	1,000,010
I Hardware design A	450	104	46,800	11,700				
	2,750	96	264,000	66,000				
	2,250	60	135,000	33,750				
		Total	445,800	111,450	24,500		58,175	639,925
J Hardware design B	625	104	65,000	16,250				
	3,375	96	324,000	81,000				
	3,250	80	260,000	65,000				
		Total	649,000	162,250	61,500		87,275	960,025
K Drawings B	400	104	41,600	10,400				
	400	72	28,800	7,200				
		Total	70,400	17,600	57,400		14,540	159,940
L Software specs	400	112	44,800	11,200				
	600	96	57,600	14,400				
	600	80	48,000	12,000				
		Total	150,400	37,600	23,300	116,000	32,730	360,030
M Parts purchase B	5	112	560	140				
	40	96	3,840	960				
		Total	4,400	1,100	250	758,000	76,375	840,125
N Parts purchase A	10	112	1,120	280				
	50	96	4,800	1,200				
		Total	5,920	1,480	350	719,500	72,725	799,975

O Drawings A	1,625	104	169,000	42,250				
	1,750	72	126,000	31,500				
		Total	295,000	73,750	85,800		45,455	500,005
P Installation drawings	1,125	112	126,000	31,500				
	1,500	104	156,000	39,000				
	1,750	72	126,000	31,500				
		Total	408,000	102,000	35,400		54,540	599,940
Q Software purchase	20	112	2,240	560				
	40	96	3,840	960				
		Total	6,080	1,520	1,600	717,500	72,670	799,370
U Assembly A	25	112	2,800	700				
	250	96	24,000	6,000				
	300	80	24,000	6,000				
		Total	50,800	12,700	64,000		12,750	140,250
V Assembly B	250	112	28,000	7,000				
	2,750	96	264,000	66,000				
	2,750	80	220,000	55,000				
		Total	512,000	128,000	87,000		72,700	799,700
W Test A	50	104	5,200	1,300				
	750	96	72,000	18,000				
	750	80	60,000	15,000				
		Total	137,200	34,300	47,000		21,850	240,350
X Test B	75	104	7,800	1,950				
	1,125	96	108,000	27,000				
	1,125	80	90,000	22,500				
		Total	205,800	51,450	70,000		32,725	359,975
Y Final installation	800	112	89,600	22,400				
	3,000	96	288,000	72,000				
	2,250	88	198,000	49,500				
		Total	575,600	143,900	121,000	105,000	94,550	1,040,050
Z Final test	500	112	56,000	14,000				
	2,500	96	240,000	60,000				
	1,500	84	126,000	31,500				
		Total	422,000	105,500	12,500	60,000	60,000	660,000
Totals			5,682,400	1,420,600	816,700	4,026,000	1,194,570	13,140,270

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.

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, 2020
Parts and components for storage rack systems from SEI	November 4, 2020
One shipping container and one each of three-size parcel buckets from UPC	November 10, 2020
Robotic drone transporter system control software from CRC	October 25, 2020

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, 2020
Final structural, functional checkout	November 29, 2020
Delivered 400 shipping containers installed at site	December 6, 2020
Delivered 1,000 size D43A parcel buckets	December 13, 2020
Delivered 600 size D25B parcel buckets	December 13, 2020
Delivered 600 size D12C parcel buckets	December 13, 2020
Final structural, functional checkout	November

	8, 2020
Hardware (Group B)	
Seven robot transporter units (each 80 pounds maximum load capacity compatible with three-size parcel buckets)	
Installed at site	November 8, 2020
Seven unit functional checkout	November 10, 2020
Integration checkout, groups A and B	January 3, 2021
Software Group	
Submission of software specifications to CRC	September 19, 2020
(Installation of DEM-LAN network, four CRC 2950 workstation terminals,and CRC 4000 server, performed by CRC)	February 7, 2021
Software-integration checkout, performed by CRC	March 7, 2021
Final checkout	
Two copies, system operation/maintenance manuals	March 7, 2021
Robotic drone transporter/CRC 4000 integration	April 4, 2021
Benchmark systems test, with parcels	April 8, 2021
User training	April 11–12, 2021
Final system checkout, user	Latest, May 2, 2021

Index