Maturity Model
SECMM-95-01
CMU/SEI-95-MM-003

A Systems Engineering Capability Maturity Model[SM], Version 1.1

SE - CMM

Systems Engineering Improvement

Systems Engineering
Capability Maturity Model Project

November 1995

Maturity Model
SECMM-95-01
CMU/SEI-95-MM-003
November 1995

A Systems Engineering Capability Maturity Model, Version 1.1

SE - CMM

Systems Engineering Improvement

Roger Bate, Chief Architect

Dorothy Kuhn, Product Manager

Curt Wells, Product Manager

James Armitage

Gloria Clark

Kerinia Cusick

Suzanne Garcia

Mark Hanna

Robert Jones

Peter Malpass

Ilene Minnich

Hal Pierson

Tim Powell

Al Reichner

Unlimited distribution subject to the copyright.

Software Engineering Institute

This report was prepared for the

SEI Joint Program Office
HQ ESC/AXS
5 Eglin Street
Hanscom AFB, MA 01731-2116

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

FOR THE COMMANDER

(signature on file)

Thomas R. Miller, Lt Col, USAF
SEI Joint Program Office

This work is sponsored by the U.S. Department of Defense.

Copyright © 1995 by Carnegie Mellon University.

Permission to reproduce this document and to prepare derivative works from this document for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

Requests for permission to reproduce this document or to prepare derivative works of this document for external and commercial use should be addressed to the SEI Licensing Agent.

NO WARRANTY

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

This work was created in the performance of Federal Government Contract Number F19628-95-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 52.227-7013.

This document is available through Research Access, Inc., 800 Vinial Street, Pittsburgh, PA 15212.
Phone: 1-800-685-6510. FAX: (412) 321-2994. RAI also maintains a World Wide Web home page. The URL is http://www.rai.com

Copies of this document are available through the National Technical Information Service (NTIS). For information on ordering, please contact NTIS directly: National Technical Information Service, U.S. Department of Commerce, Springfield, VA 22161. Phone: (703) 487-4600.

This document is also available through the Defense Technical Information Center (DTIC). DTIC provides access to and transfer of scientific and technical information for DoD personnel, DoD contractors and potential contractors, and other U.S. Government agency personnel and their contractors. To obtain a copy, please contact DTIC directly: Defense Technical Information Center / 8725 John J. Kingman Road / Suite 0944 / Ft. Belvoir, VA 22060-6218. Phone: (703) 767-8222 or 1-800 225-3842.]

Use of any trademarks in this report is not intended in any way to infringe on the rights of the trademark holder.

Table of Contents

Acknowledgments....iii
To the Reader....v

Part 1: Overview Information

Chapter 1: Introduction....1-1
- 1.1 About This Document....1-2
- 1.2 About the SE-CMM Project....1-5

Chapter 2: Overview of the SE-CMM....2-1
- 2.1 SE-CMM Foundations....2-2
- 2.2 Key Concepts of the SE-CMM....2-8
- 2.3 SE-CMM Architecture Description....2-14
- 2.4 Process Capability Aspect of the SE-CMM....2-21
- 2.5 Capability Levels....2-25
- 2.6 Domain Aspect of the SE-CMM....2-27
- 2.7 A Path for Improving Systems Engineering Maturity....2-34

Chapter 3: Using the SE-CMM....3-1
- 3.1 Many Usage Contexts....3-2
- 3.2 Using the SE-CMM to Support Appraisal....3-5
- 3.3 Using the SE-CMM to Support Process Improvement....3-8
- 3.4 Using the SE-CMM in Process Design....3-13

Part 2: The SE-CMM Practices

Chapter 4: The SE-CMM Generic & Base Practices....4-1

Chapter 4A:....Generic Practices 4-2
- Capability Level 0 - Not Performed....4-3
- Capability Level 1 - Performed Informally....4-4
- Capability Level 2 - Planned and Tracked....4-5
- Capability Level 3 - Well Defined....4-9
- Capability Level 4 - Quantitatively Controlled....4-12
- Capability Level 5 - Continuously Improving....4-13

Chapter 4B:....Process Areas/Base Pract
- Process Area Format....4-16
- PA 01: Analyze Candidate Solutions....4-18
- PA 02: Derive and Allocate Requirements....4-23
- PA 03: Evolve System Architecture....4-34
- PA 04: Integrate Disciplines....4-42
- PA 05: Integrate System....4-47
- PA 06: Understand Customer Needs and Expectations....4-53
- PA 07: Verify and Validate System....4-59
- PA 08: Ensure Quality....4-66
- PA 09: Manage Configurations....4-72
- PA 10: Manage Risk....4-77
- PA 11: Monitor and Control Technical Effort....4-82
- PA 12: Plan Technical Effort....4-86
- PA 13: Define Organization's Systems Engineering Process....4-95
- PA 14: Improve Organization's Systems Engineering Processes....4-100
- PA 15: Manage Product Line Evolution....4-104
- PA 16: Manage Systems Engineering Support Environment....4-108
- PA 17: Provide Ongoing Skills and Knowledge....4-113
- PA 18: Coordinate with Suppliers....4-120

Part 3: Appendices

- Appendix A: Change History and Change Request Form....A-3
- Appendix B: Approved Model Requirements....A-7
- Appendix C: References....A-21
- Appendix D: Systems Engineering Glossary....A-25

Tables and Figures

Tables:

Table 1-1. SE-CMM Work Products....1-4
Table 1-2. SE-CMM Version 1.0 Authors....1-6
Table 1-3. SE-CMM Version 1.1 Authors....1-7
Table 2-1. Components of the Process Capability Aspect of the SE-CMM....2-15
Table 2-2. SE-CMM Capability Levels....2-17
Table 2-3. Components of the Domain Aspect of the SE-CMM....2-18
Table 2-4 SE-CMM Process Areas....2-19
Table 3-1. Customer Base....3-3
Table 3-2. Six Steps to Six Sigma....3-4
Table 3-3. Process Improvement Principles in the SE-CMM....3-11
Table A-1. Change History Table....A-3
Table A-2. Traceability Matrix....A-17

Figures:

Figure 1-1. SE-CMM Project Organization....1-4
Figure 2-1. Critical Dimensions of Organizational Capability....2-2
Figure 2-2. Model Architecture....2-5
Figure 2-3. Focus of the SE-CMM....2-7
Figure 2-4. Diagram of SE-CMM Architecture....2-14
Figure 2-5. Improvement Path for Process Capability....2-22
Figure 2-6. SE-CMM Process Areas....2-27
Figure 2-7. Iteration....2-32
Figure 2-8. Concurrency....2-32
Figure 2-9. Recursion....2-33
Figure 2-10. Capability Profile for Level 1 Engineering....2-35
Figure 2-11. Capability Profile for Level 2 Engineering....2-36
Figure 2-12. Capability Profile for Level 3 Engineering....2-37
Figure 2-13. Capability Profile for Level 4 Engineering....2-38
Figure 2-14. Capability Profile for Level 5 Engineering....2-39
Figure 3-1. Determining Process Capability....3-6
Figure 3-2. Factors for Successful Process Design....3-14
Figure 4-1. Process Area Format....4-17

Acknowledgments

Participants

The Systems Engineering Capability Maturity Model[SM] (SE-CMM[SM]) is the work of many individuals from industry, academia, and government. Their spirit of cooperation and willingness to give of themselves in a joint pursuit of excellence was remarkable. Although a few individuals who fulfilled specific reviews roles are mentioned below, there are many others who supported the points of contacts or otherwise participated. A listing of all known participants, their affiliation and role(s) is too extensive to be included here. A copy is available, by request, from the project.

The level of individual participation varied from a few hours to full time (and more). But, the project could not have been successfully completed without the active contribution of everyone concerned, and their efforts are truly appreciated.

Key reviewers who returned comments

The authors would like to extend their thanks to the key reviewers who returned comments on intermediate releases of the model description, including those who participated in our workshops. Their insights were always valued and considered seriously by the authors, and their contributions extended the reach of the authors far beyond their own experience. This group includes

R. Ade, E.R. Anderson, B. Andrews, J. Armento, K. Arunski, R. Bechtold, L. Bloodsworth, M. Brown, J. Burleson, J. Campbell, T. Carpenter, M. Carroll, E. Cherian, T. Concannon, J. Cooper, D. Corpman, D. Coskren, R. Czizik, R. Daniel, J. DeFoe, A. Dniestrowski, R. Elkin, W. Fabrycky, M. Falat, K. Ferraloio, D. Francis, S. Friedenthal, L. Gallagher, C. Giffen, M. Ginsberg, J. Gross, D. Gunther, J. Harbison, W. Hefley, G. Hingle, J. Huller, J. Jaggers, P. James, H. Jesse, D. Jester, J.P. Jones, K. Jones, G. Kasch, D. Kinney, K. Kirlin, M. Klien, M. Konrad, T. Kudlick, G. Lake, J. Lake, H. Lee, S. Levie, W. Mackey, B. Mar, J. Marciniak, D. Marquet, D. McCall, S. McCammon, D. McCauley, D. McConnell, B. McDonough, M. Merrill, J. Miller, C. Montgomery, J. Moon, R. Nakahara, C.L. Nelson, L. Nelson, W. Oran, G. Pandelios, T. Parker, B. Pencole, W. Peterson, J. Porter, T. Powell, D. Preklas, R. Ragano, T. Robertson, J. Romano, M. Ross, R. Sabouhi, N. Sanford, R. Schmidt, A. Shumskas, R. Sibner, J. Solomond, A. Sutton, T. Sweeney, Y. Trsonstad,.J. Velman, M. Ward-Callan, C. Wasson, A. Wilbur, A.M. Wilhite, R. Williams, H. Wilson, D. Zaugg, and C. Zumba.

continued on next page

[SM] CMM and Capability Maturity Model are service marks of Carnegie Mellon University.

Pilot appraisals

The authors would also like to extend their sincere thanks to the organizations who made the first set of pilot appraisals for the SE-CMM such successes. These organizations contributed to the success of the pilot appraisal both by providing data about the model and by helping us learn about how the SE-CMM architecture facilitates appraisal. Over 100 individuals participated in 9 appraisals of organizations in industry and government. The sponsors of these appraisals deserve special recognition for being early adopters of the SE-CMM. John Grimm from Texas Instruments, Steve Cunningham from Hughes, and Kenneth Rosen from United Technologies were among the earliest users of the SE-CMM appraisal. Their cooperation, insight, and patience contributed significantly to the quality of the first public version of the SE-CMM.

SE-CMM steering group members

The Steering Group for the SE-CMM Project has provided both traditional management oversight functions and extensive technical and strategic input to the project, and their individual and collected contributions to the project are appreciated beyond measure. The names and organization for the SE-CMM Steering Group members in the collaboration are provided in the table below:

Organization	Contacts
Colin Tully Associates	Colin Tully
Department of Defense	John Burt
Department of Defense	Mark Schaeffer
General Dynamics Corporation	Bob Fox
Hughes Aircraft Company	Ilene Minnich
Lockheed Martin Corporation	Douglas Bowman
Loral Federal Systems Company	Joan Weszka
National Institute of Standards and Technology	Thomas Rhodes
Software Engineering Institute	Julia Allen
Software Engineering Institute	Roger R. Bate
Software Productivity Consortium	Art Pyster
Texas Instruments Incorporated	Merle Whatley

SE-CMM Collaboration Contacts

To the Reader

What is the SE-CMM?

The Systems Engineering Capability Maturity Model (SE-CMM) describes the essential elements of an organization's systems engineering process that must exist to ensure good systems engineering. It does not specify a particular process or sequence. In addition, the SE-CMM provides a reference for comparing actual systems engineering practices against these essential elements.

This document provides an overall description of the principles and architecture upon which the SE-CMM is based, an overview of the model, suggestions for appropriate use of the model, the practices included in the model, and a description of the attributes of the model. It also includes the requirements used to develop the model.

Why was it developed?

Success in market driven and contractually negotiated market areas are often determined by how efficiently an organization translates customer needs into a product that effectively meets those needs. Good systems engineering is key to that activity, and the SE-CMM provides a way to measure and enhance performance in that arena.

Why is systems engineering important?

The following classic example backs up the need for good systems engineering.

The Denver International Airport (DIA) was built in the early 1990s, as the area's air transportation needs had outstripped the capacity of the city's Stapleton Airport. It was the first major U.S. airport built in 20 years, and its opening was delayed for 25 months at an estimated cost of $500,000 per day.

Initial construction delays gave way to lengthy delays due to the airport's faulty baggage system. The symptomatic jamming of the system's telecars, mis-directed telecars, and ripped and overdue luggage revealed errors in the system's software and hardware. Even though the baggage system's prime contractor was well experienced with baggage-handling systems, built-in redundancies, and tested components, the requirements of the DIA contract proved too much [Geppert 94].

One of the advantages of systems engineering based on a defined process is the precept of fully investigating the nature of the environment around the system and the effects that the environment will have on the system under all circumstances. Systems engineers using processes based on SE-CMM practices are not any more likely to know the parameters of a particular problem and to follow disciplined investigative methods that draw out the risk areas of a system.

continued on next page

What is the scope of the SE-CMM?

This version of the SE-CMM encompasses all phases of the system life cycle and focuses on process characteristics. Given sufficient community support, future expansions may encompass both personnel and technology characteristics.

How should it be used?

The SE-CMM is designed to help organizations improve their practice of systems engineering through self-assessment and guidance in the application of statistical process control principles. Use of the model for supplier selection is discouraged.

In conjunction with the model itself, a companion appraisal method has been developed, and is described in SECMM-94-06|CMU/SEI-94-HB-05, *SE-CMM Appraisal Method Description.*

Intended audience

The SE-CMM is focused on four primary groups: systems engineering practitioners from any business sector or government, process developers, individuals charged with appraising how specific systems engineering organizations implement their systems engineering processes, and systems engineering managers. Persons with five years or more of experience as a systems engineering practitioner or manager and exposure to formal methods of organization assessment will benefit most from the model.

Additional information-project office

If you have any questions about this model or about appraisals using this model, please contact the SE-CMM Project. The maintenance site for the project is the Software Engineering Institute of Carnegie Mellon University. The product manager, Dorothy Kuhn, may be contacted at

Texas Instruments
6500 Chase Oaks Boulevard
MS 8420
Plano, TX 75086

(214)575-2648 (voice)
(214)575-6807 (fax)
kuhnd@ti.com (email)

Data rights associated with the SE-CMM

The Enterprise Process Improvement Collaboration (EPIC) members are committed to encouraging free use of the SE-CMM as a reference for the systems engineering community. Members have agreed that this and future versions of this document, when released to the public, will retain the concept of free access via a permissive copyright notice.

Chapter 1: Introduction

Purpose of this chapter

The purpose of this chapter is to introduce the reader to the document and to the SE-CMM Project.

In this chapter

The following table provides a guide to the information found in this chapter.

Topic	See Page
1.1 About This Document	1-2
1.2 About the SE-CMM Project	1-5

1.1 About This Document

Purpose of this document

This document is designed to acquaint the reader with the SE-CMM Project as a whole and its major product–the Systems Engineering Capability Maturity Model (SE-CMM). This document is one in a series of the SE-CMM Project's work products. It consists of four chapters and appendices. The document contains only a brief section on using the model for appraisal. Please refer to SECMM-94-06|CMU/SEI-94-HB-05, *SE-CMM Appraisal Method Description*, for details in this area.

Basic organization

This document contains four chapters plus appendices:

- Introduction
- Overview of the SE-CMM
- Using the SE-CMM
- The SE-CMM Base and Generic Practices

These chapters are described in the blocks below.

Chapter 1: Introduction

This chapter provides the document overview and a brief description of the model, the need it is designed to meet, who wrote it, and how this version has been constructed to fit economic and time constraints.

Chapter 2: Overview

This chapter introduces the model and provides an overview of the requirements it is intended to satisfy. It introduces basic concepts that are key to understanding the details and architecture of the model. It also introduces the two-sided architecture of the model: the domain-specific side and the capability side. These and other underlying constructs and conventions used in expressing the model are explained to help readers understand and use the model.

Chapter 3: Using the SE-CMM

This chapter provides information that will be useful to individuals interested in adopting the model and adapting it to different organizational situations and contexts.

continued on next page

Chapter 4: SE-CMM Practices

This chapter contains a specific, comprehensive description of the model. In the domain-specific side of the discussion, base practices, which are characteristics considered essential to successful systems engineering, are grouped into specific process areas (PAs). Each process area is described in detail. In the capability side, generic practices, which are characteristics of how well the base practices are performed, are discussed. The concepts of increasing process capability are also described in the capability part of the chapter.

Appendices

The appendices include a change history for the document, a change request form, the requirements for the model description, the references, and a glossary of the terms used in project documents.

continued on next page

Related products

In addition to this document, the SE-CMM Project has produced the SE-CMM Appraisal Method (SAM) Description (SECMM-94-06 |CMU/SEI-94-HB-05). The SAM provides a description of the appraisal method developed for use with the SE-CMM when evaluating adherence to the principles and/or practices of the SE-CMM. It also contains the appraisal method requirements. This document was published for public release via the maintenance site for the SE-CMM Project, Carnegie Mellon University's Software Engineering Institute.

The documents shown in Table 1-1 are slated for public revision in the 1996 time frame.

Identifier	Name	Description
SECMM-94-08 CMU/SEI-94-TR-25	SE-CMM Pilot Appraisal Report	The *SE-CMM Pilot Appraisal Report* describes the results of piloting activity for the systems engineering community to use as they adopt and work with the SE-CMM and its associated appraisal method.
SECMM-94-09 CMU/SEI-94-TR-26	Relationships Between the SE-CMM and Other Products	The SE-CMM relationships document presents information on relationships between the process areas/common features of the SE-CMM and other products of interest to the SE-CMM author group. The first version includes relationships to the Air Force Software Development Capability Evaluation, IEEE P1220, draft of MIL-STD-499B, and the Capability Maturity Model for Software, v1.1.
	SE-CMM Workshop Report	The *SE-CMM Workshops Report* describes the nature and results of the two 1994 SE-CMM workshops.

Table 1-1. SE-CMM Work Products

1.2 About the SE-CMM Project

Project history

The Systems Engineering Capability Maturity Model (SE-CMM) was developed as a response to industry requests for assistance in coordinating and publishing a model that would foster improvement in the systems engineering process. In July 1993 Dr. Roger Bate, the SE-CMM chief architect, presented an approach to developing a Systems Engineering Capability Maturity Model to potential industry participants. The SE-CMM collaboration was subsequently formed, and specific project goals and requirements were defined by the SE-CMM Steering Group. Initial task completion was set at December 1994, when SE-CMM v1.0 was published.

In August 1995, the name of the industrial collaboration was changed to Enterprise Process Improvement Collaboration (EPIC) to reflect the broader membership and scope of work.

Project organization chart

Figure 1-1 illustrates the project organization chart. It is discussed in the blocks below.

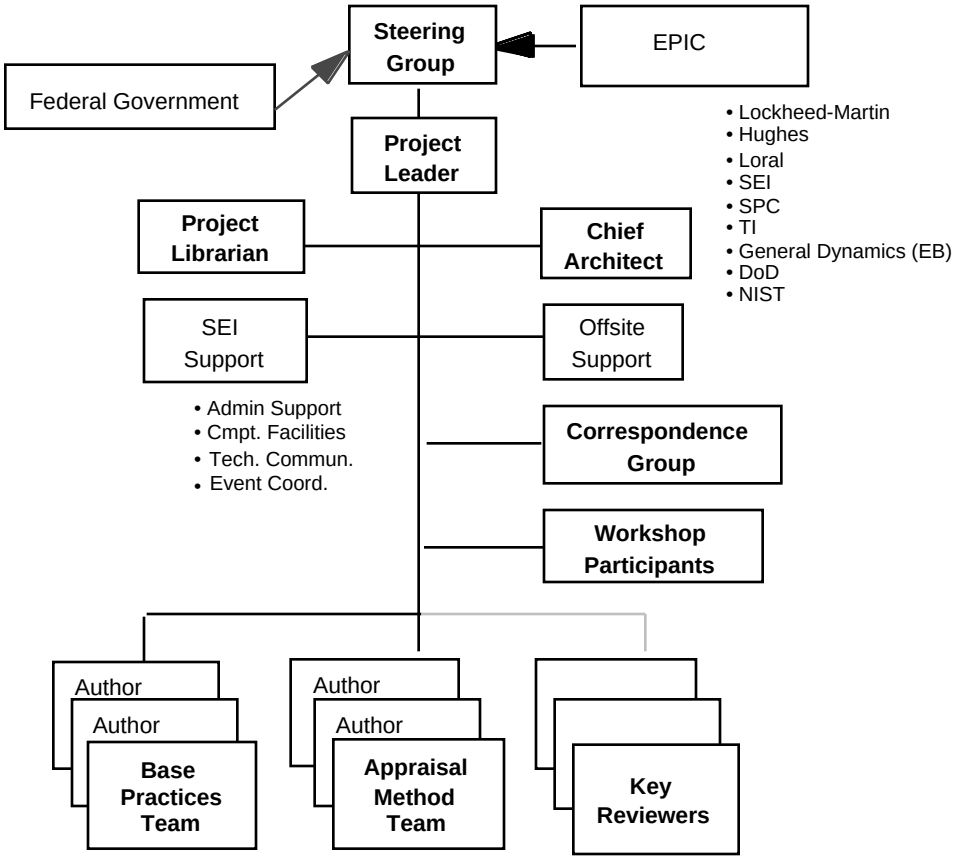

Figure 1-1. SE-CMM Project Organization

continued on next page

SE-CMM Project composition

The SE-CMM Project is guided by a steering group which is composed of people from the SE-CMM collaboration, with ex officio members from the federal government. SEI supplies the project leadership, chief architect, project librarian, and administrative support. The authors provide the systems engineering technical expertise and/or modeling and appraisal expertise necessary to support the model development. The key reviewers and workshop participants provide input to the author group who incorporate their comments into the model. Model development is also supported by the correspondence group and pilot appraisal sites. The authors have come from GTE, Hughes, Lockheed Martin, Loral, Software Engineering Institute, Software Productivity Consortium, and Texas Instruments. These are organizations with an established history of good systems engineering performance and/or modeling and assessment methodology.

SE-CMM v1.0 authors

The authors for version 1.0 are listed in alphabetical order in Table 1-2.

Author	Organization
James Armitage, Ph.D.	GTE Government Systems, Pittsburgh, PA
Roger Bate, Ph.D.	Software Engineering Institute, Pittsburgh, PA
Kerinia Cusick	GM Hughes Electronics Los Angeles, CA
Suzanne Garcia	Software Engineering Institute, Pittsburgh, PA
Robert Jones	Loral Federal Systems Company, Houston, TX
Dorothy Kuhn	Texas Instruments Incorporated, Dallas, TX
Ilene Minnich	Hughes Aircraft Company, Fullerton, CA
Hal Pierson, Ph.D.	Software Productivity Consortium, Herndon, VA
Tim Powell	Software Productivity Consortium, Herndon, VA
Al Reichner	Loral Space & Range Systems, Sunnyvale, CA
Curtis Wells	Lockheed Martin Corporation Valley Forge, PA

Table 1-2. SE-CMM Version 1.0 Authors

continued on next page

SE-CMM v1.1 authors

The authors for version 1.1 are listed in alphabetical order in Table 1-3.

Author	Organization
Roger Bate, Ph.D.	Software Engineering Institute, Pittsburgh, PA
Gloria Clark	GM Hughes Electronics Los Angeles, CA
Kerinia Cusick	GM Hughes Electronics Los Angeles, CA
Suzanne Garcia	Software Engineering Institute, Pittsburgh, PA
Mark Hanna	Software Productivity Consortium, Herndon, VA
Dorothy Kuhn	Texas Instruments Incorporated, Dallas, TX
Peter Malpass	Software Engineering Institute, Pittsburgh, PA
Hal Pierson, Ph.D.	Software Productivity Consortium, Herndon, VA
Curtis Wells	Lockheed Martin Corporation Valley Forge, PA

Table 1-3. SE-CMM Version 1.1 Authors

Incorporating community feedback

The SE-CMM v1.0 was developed by the collaboration of a group of companies with long and successful histories in building complex systems. Many of the principal authors have over 20 years experience in systems engineering and/or process improvement. The principal authors were supplemented by an extensive reviewer panel selected from academia, government, and industry for their systems engineering expertise. The SE-CMM v1.1 includes feedback from two 1994 public workshops, three pilot appraisals of organizations using early versions of the model, one 1995 workshop, and six 1995 pilot appraisals.

continued on next page

Changes from Version 1.0

Changes reflected in Version 1.1 of the SE-CMM are driven by the SE-CMM requirement (3.1) to achieve industry acceptance. (SE-CMM requirements can be found in Appendix B.) In the revision, we addressed comments received from industry and government and a workshop held in September 1995; specific changes include the following:

- Increase the scope of the model to better address the life cycle aspects of the system.
- Add a new process area to address the coordination of supplier contributions.
- Clarify the generic practices.
- Add to the base practice descriptive model.
- Improve the Model Overview chapter.
- Increase the emphasis on production and operations input into development practices.

Future plans outline

This version of the SE-CMM addresses the process aspects of systems engineering and the product development portion of the life cycle. There are several possible avenues for future work which are being considered by the steering group. They include

- Develop an integrated product development (IPD) framework that addresses common and unique aspects of IPD in relation to the systems engineering concepts embodied in the SE-CMM. This work was begun in 1995 and continues in 1996.
- Extend the model into addressing the people and technology aspects of product development.

Pilot appraisals of the SE-CMM continued throughout 1995, for a total of nine appraisals conducted. Given the success of the pilot appraisals, EPIC expects to end their piloting in early 1996. At present, two firms are available to assist organizations in conducting self-appraisals. More such firms may become available.

Chapter 2: Overview of the SE-CMM

Purpose of this chapter

The purpose of this chapter is to provide an overview of the concepts and constructs used in the SE-CMM. It provides information on the requirements that led to the design of the SE-CMM, a description of the architecture, and a section on key concepts and terms which are helpful in understanding the model. It serves as an introduction to the detailed discussion of the model in Chapter 4.

In this chapter

The following table provides a guide to the information found in this chapter.

Topic	See Page
2.1 SE-CMM Foundations	2-2
2.2 Key Concepts of the SE-CMM	2-8
2.3 SE-CMM Architecture Description	2-14
2.4 Process Capability Aspect of the SE-CMM	2-21
2.5 Capability Levels	2-25
2.6 Domain Aspect of the SE-CMM	2-27
2.7 A Path for Improving Systems Engineering Maturity	2-34

2.1 SE-CMM Foundations

Introduction

In this section, the fundamental concepts that have guided the development of the SE-CMM are presented. The SE-CMM requirements and their traceability can be found in Appendix B.

Critical dimensions of capability

The SE-CMM Project believes that the quality of a product is a direct function of (at least) the process and technology used to develop the product and the capability of the people assigned to do the work (see Figure 2-1, below). The initial efforts of the project focus on modeling characteristics of the process dimension, that is, processes used to implement and institutionalize sound systems engineering practices within an organization. The SE-CMM Project believes that the quality of a product is a direct function of the process capability, the technology capability, and the people capability used to develop the product.

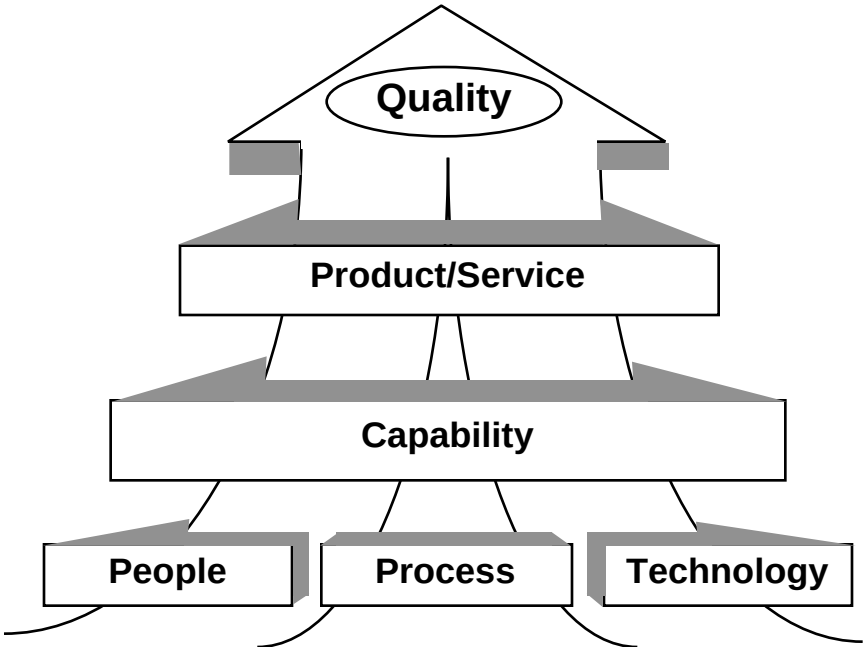

Figure 2-1. Critical Dimensions of Organizational Capability

continued on next page

Why process first?

There are several reasons that process is the first dimension of organizational capability addressed by the SE-CMM. A few of these include

- Process is an integrating function for people and technology.
- Process focus improves predictability of performance, as well as performance itself.
- Research in improving process capability translates well from other fields, such as software engineering, to systems engineering.

Definition of systems engineering

There are dozens of definitions of systems engineering published in various industry, academic, and government documents that address systems engineering topics. Rather than invent an additional definition, the authors chose to adopt the definition found in Army Field Manual 770-78, which reads as follows:

Systems engineering is the selective application of scientific and engineering efforts to

- transform an operational need into a description of the system configuration which best satisfies the operational need according to the measures of effectiveness;
- integrate related technical parameters and ensure compatibility of all physical, functional, and technical program interfaces in a manner which optimizes the total system definition and design;
- integrate the efforts of all engineering disciplines and specialties into the total engineering effort [FM 770-78].

Why this definition?

This definition was adopted over others primarily because it emphasizes the leadership role of system engineering in integrating other disciplines and does not contain terminology specific to a particular industry segment.

Depth and breadth of model coverage

SE-CMM coverage extends to, but does not include, various component implementation disciplines (e.g., hardware, firmware, and software development) and specialty engineering disciplines. Version 1.1 of the model covers the total system life cycle.

continued on next page

Specialty engineering disciplines

The SE-CMM does not specifically address specialty engineering disciplines such as reliability, human factors engineering, or manufacturing engineering. The model requires the integration of the engineering disciplines and specialties, whichever ones are necessary and appropriate for a particular product development.

The issue of whether specialty engineering should be more directly addressed in the SE-CMM (e.g., via additional process areas) has been debated at both the 1994 and 1995 SE-CMM workshops. The consensus at the 1995 workshop was that creating process areas for specific specialties in the SE-CMM would not serve the communities who work in those areas and that making improvements to Integrate Disciplines process area (PA04) was a better solution. The authors accepted this advice and have increased the descriptive language related to specialties, but did not add any new process areas.

Relationship of systems engineering to overall program/ project management

There is considerable debate within the systems engineering community as to systems engineering's role within the overall management of a project or program. Some argue that the systems engineering role encompasses all the program management functions. Systems engineering must have sufficient control over all the resources that are critical to balancing cost, schedule, quality, and functionality objectives. Others argue that the systems engineering role should be subservient to program management, to be able to provide the necessary engineering viewpoint into business decisions. The SE-CMM has taken the latter approach, although it recognizes that systems engineers commonly perform extensive program/project management roles in some environments. The project management practices expressed in the SE-CMM are those most commonly found as part of the technical management function of the systems engineer, and those supporting practices that are critical to the successful performance of systems engineering regardless of performer.

Flexible architecture

The model architecture, shown in Figure 2-2, below, separates systems engineering process areas (on the domain side) from generic characteristics (on the capability side) related to increasing process capability (See Section 2.3 for a more detailed description). This architecture, which separates domain-specific characteristics from capability-related characteristics, was deliberately chosen to enable the use of process capability criteria in other domain areas, e.g., software engineering. It also supports the expansion of the model into specialty engineering or other component engineering disciplines, should this be deemed appropriate by the organization using the model.

continued on next page

Diagram of Model Architecture

Figure 2-2 illustrates the dual-path of the SE-CMM architecture.

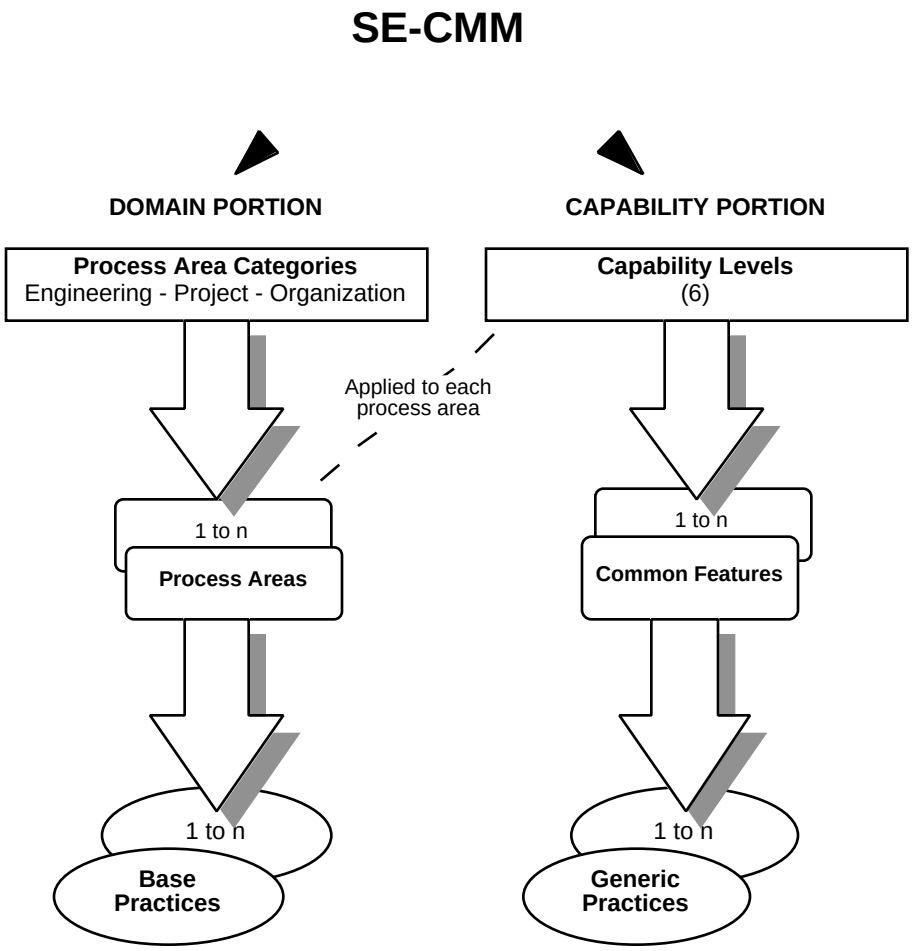

Figure 2-2. Model Architecture

Usability

The SE-CMM is specifically developed to support an organization's need to assess and improve their systems engineering capability. The structure of the model enables a consistent appraisal methodology to be used across diverse process areas. Because the model clearly distinguishes essential, basic systems engineering elements (the domain side) and process management-focused elements (the capability side), organizations should find it easier to improve their processes in response to their business needs.

continued on next page

Range of applicability

The SE-CMM has a wide range of applicability. The SE-CMM was developed to be valuable to market-driven project environments as well as negotiated-contract environments. By providing a multipurpose asset that can be used by (1) individual systems engineering practitioners as a guide, (2) their parent organizations for productivity improvement, and (3) any organization as an eventual supplier selection tool, the SE-CMM meets the needs of a wide range of users. Applicability will be enhanced by incorporating changes from applying the model.

Capture and gain leverage from existing & emerging standards

One of the design goals of the SE-CMM effort was to capture the salient concepts from emerging standards and initiatives (e.g., ISO 9001, draft MIL-STD- 499B [now being revised as EIA IS-632], IEEE P1220) and existing models. SE-CMM-94-09|CMU/SEI-94-TR-26, *Relationships Between the SE-CMM and Other Products,* provides a cross-reference between the SE-CMM and some of these standards and models.

Retain CMM interface

Although the architecture and syntax used to express the SE-CMM model are different from those used in the CMM for Software v1.1, it is envisioned that these two models can be used together effectively to improve and assess the systems and software engineering processes of a project or organization. SECMM-94-09|CMU/SEI-94-TR-26, *Relationship Between the SE-CMM and Other Products,* contains information on this interface.

continued on next page

SE-CMM application environment

Figure 2-3 illustrates the intended relationship of the SE-CMM to an organization's process design and improvement activities. The SE-CMM does not intend to imply or prescribe organizational issues such as organizational culture, role definitions, or structure, nor is it intended to imply any particular product or project context. It establishes characteristics essential to good systems engineering, but does not imply or define a specific, executable process. The major implication of this approach is that the SE-CMM will enhance the resulting systems engineering processes without necessarily driving changes in culture or products. This approach supports the desire to use the SE-CMM in a wide spectrum of organizational contexts.

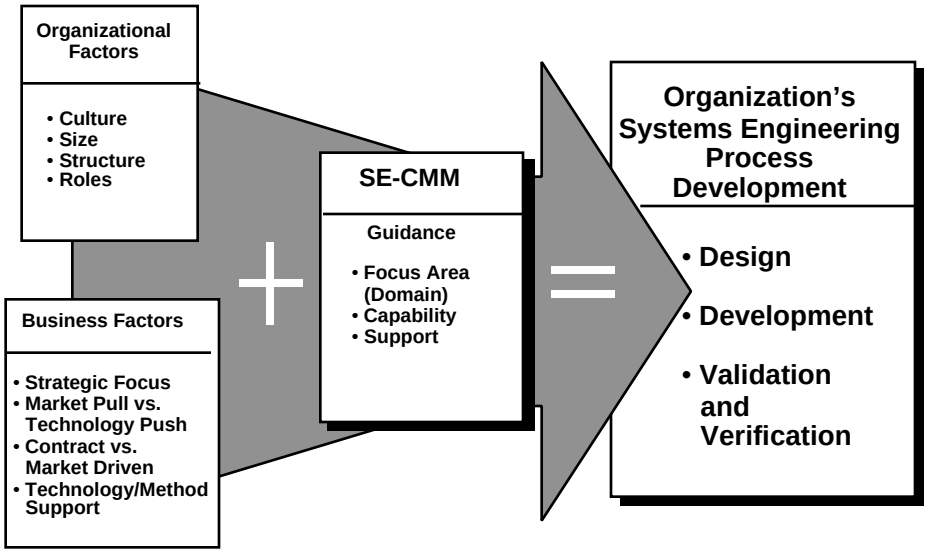

Figure 2-3. Focus of the SE-CMM

2.2 Key Concepts of the SE-CMM

Introduction

In the discussions above, and those which follow, terms are used and concepts are introduced that have particular meaning within the context of the SE-CMM. This section elaborates those concepts that are critical to effective understanding, interpretation, and use of the SE-CMM. Some concepts specific to the model, such as "generic practice" and "base practice," are defined and discussed in the sections of the model description that address them. Other terms and concepts are defined in the glossary (Appendix D). The concepts to be discussed in this section are listed below:

- organization
- project
- system
- work product
- customer
- process
- systems engineering process
- process area
- role independence
- process capability
- institutionalization
- process management
- capability maturity model

Organizations and projects

Two terms are used within the SE-CMM to differentiate aspects of organizational structure: organization and project. The authors realize that other constructs, such as teams, exist within business entities, but there is no commonly accepted terminology that spans all business contexts. These two terms were chosen because they are commonly used/understood by most of the anticipated audience of the SE-CMM.

Organization

For the purposes of the SE-CMM, an organization is defined as a unit within a company, the whole company or other entity (e.g., government agency or branch of service), within which many projects are managed as a whole. All projects within an organization typically share common policies at the top of the reporting structure. An organization may consist of co-located or geographically distributed projects and supporting infrastructures.

The term "organization" is used to connote an infrastructure to support common strategic, business, and process-related functions. The infrastructure exists and must be maintained for the business to be effective in producing, delivering, supporting, and marketing its products.

continued on next page

Project

The project is the aggregate of effort and other resources focused on developing and/or maintaining a specific product. The product may include hardware, software, and other components. Typically a project has its own funding, cost accounting, and delivery schedule. A project may constitute an organizational entity of its own, or it may be structured as a team, task force, or other entity used by the organization to produce products.

The process areas in the domain side of the SE-CMM have been divided into three categories, engineering, project, and organization, as discussed in Section 2.6. The categories of organization and project are distinguished based on typical ownership. The SE-CMM differentiates between project and organization categories by defining the project as focused on a specific product, versus the organization which encompasses one or more projects.

System

A system can be defined as

- an integrated composite of people, products, and processes that provide a capability to satisfy a need or objective [MIL-STD-499B]
- an assembly of things or parts forming a complex or unitary whole; a collection of components organized to accomplish a specific function or set of functions
- an interacting combination of elements, viewed in relation to function [INCOSE 95]

A system may be a product that is hardware only, hardware/software, software only, or a service. The term “system” is used throughout the model to indicate the sum of the products being delivered to the customer(s) or user(s) of the products. Denoting a product as a system is an acknowledgment of the need to treat all the elements of the product and their interfaces in a disciplined and systematic way, so as to achieve the overall cost, schedule, and performance objectives of the business entity developing the product.

continued on next page

Work product

Work products are all the documents, files, data, etc., generated in the course of performing any process. For example, work products of a review activity might be action item lists, whereas work products of a requirements process might be a database file containing all the elaborated requirements for the product. Rather than call out individual work products for each process area, the SE-CMM lists "typical work products" of a particular base practice, to elaborate further the intended scope of that base practice. These lists are not to be construed as "mandatory" work products; they are illustrative only, and reflect a range of organizational and product contexts.

Customer

A customer is the individual(s) or entity for whom a product is developed or service is rendered, and/or the individual or entity who uses the product or service.

In the context of the SE-CMM, a customer may be either negotiated or non-negotiated. A negotiated customer is an individual or entity who contracts with another entity to produce a specific product or set of products according to a set of specifications provided by the customer. A non-negotiated, or market-driven, customer is one of many individuals or business entities who have a real or perceived need for a product. The customer may also be represented by a customer surrogate such as marketing or product focus groups.

In most cases, the SE-CMM uses the term customer in the singular, as a grammatical convenience. However, the SE-CMM does intend to include the case of multiple customers.

Note that, in the context of the SE-CMM, the individual or entity using the product or service is also included in the notion of customer. This is relevant in the case of negotiated customers, since the entity to whom the product is delivered is not always the entity or individual who will actually use the product or service. The concept and usage of customer in the SE-CMM is intended to recognize the responsibility of the systems engineering function to address the entire concept of customer, which includes the user.

continued on next page

Process

A process is a set of activities performed to achieve a given purpose. Activities may be performed iteratively, recursively, and/or concurrently. (These sequencing concepts are discussed in Section 2.6). Some activities may transform input work products into output work products needed for other activities. The allowable sequence for performing activities is constrained by the availability of input work products and resources, and by management control. A well-defined process includes activities, input and output artifacts of each activity, and the mechanisms to control the performance of the activities.

Several types of processes are mentioned in the SE-CMM, including "defined" and "performed" processes. A defined process is formally described for or by an organization for use by its systems engineers. This description may be contained, for example, in a document or a process asset library. The defined process is what the organization's systems engineers are supposed to do. The performed process is what the systems engineers actually do.

Systems engineering process

The systems engineering process is defined as a comprehensive problem-solving process that is used to

- transform customer needs and requirements into a life-cycle balanced solution set of system product and process designs
- generate information for decision makers
- provide information for the next product development or acquisition phase

The systems engineering process is an instance of the general concept of process. Because of its relation to the general concept of process, the SE-CMM is able to adopt the generic practices of the Software Process Improvement Capability dEtermination (SPICE) Project (with slight modifications). This relationship between the SE-CMM and general process models is discussed in the description of process capability in this chapter (Section 2.4).

continued on next page

Process area

A process area (PA) is a defined set of related systems engineering process characteristics, which, when performed collectively, can achieve a defined purpose.

The process areas are composed of base practices, which are mandatory characteristics that must exist within an organization's implemented systems engineering process for the organization to claim satisfaction of that PA.

Role independence

The process areas of the SE-CMM are groups of practices that, when taken together, achieve a common purpose. However, the groupings are not intended to imply that all the base practices of a process are necessarily performed by a single individual or role. All base practices are written in verb-object format (i.e., without a specific subject) so as to minimize the perception that a particular base practice "belongs to" a particular role. This is one way in which the syntax of the model supports its use across a wide spectrum of organizational contexts.

Process capability

Process capability is defined as the quantifiable range of expected results that can be achieved by following a process. The SE-CMM Appraisal Method (SAM), which can be used to determine process capability levels for each process area within a project or organization, is based upon statistical process control concepts which define the use of process capability in many industrial environments. (The appraisal method is further described in Section 3.2) The capability side of the SE-CMM reflects these concepts and provides guidance in improving the process capability of the systems engineering practices which are referenced in the domain side of the SE-CMM.

The capability of an organization's process helps to predict a project's ability to meet its goals. Projects in low capability organizations experience wide variations in achieving cost, schedule, functionality, and quality targets. These concepts are further discussed in Chapter 3.

continued on next page

Institution-alization

Institutionalization is the building of infrastructure and corporate culture that support methods, practices, and procedures so that they are the ongoing way of doing business, even after those who originally defined them are gone. The process capability side of the SE-CMM supports institutionalization by providing practices and a path toward quantitative management and continuous improvement. In this way, the SE-CMM asserts that the organization needs to explicitly support process definition, management, and improvement. Institutionalization provides a path toward gaining maximum benefit from a process that exhibits sound systems engineering characteristics.

Process management

Process management is the set of activities and infrastructures used to predict, evaluate, and control the performance of a process. Process management implies that a process is defined (since one cannot predict or control something that is undefined). The focus on process management implies that a project or organization takes into account both product- and process-related factors in planning, performance, evaluation, monitoring, and corrective action.

Capability maturity model

A capability maturity model such as the SE-CMM describes the stages through which processes progress as they are defined, implemented, and improved. The model provides a guide for selecting process improvement strategies by determining the current capabilities of specific processes and identifying the issues most critical to quality and process improvement within a particular domain, such as software engineering or systems engineering. A capability maturity model (CMM) may take the form of a reference model to be used as a guide for developing and improving a mature, defined process.

A CMM may also be used to appraise the existence and institutionalization of a defined process that implements the referenced practices. A capability maturity model can cover the processes used to perform the tasks of the specified domain, (e.g., systems engineering). In addition, a CMM can cover the processes used to ensure effective development and use of human resources, and the insertion of appropriate technology into the products and into the tools used to produce the products. The latter aspects have not yet been elaborated for systems engineering.

2.3 SE-CMM Architecture Description

Introduction

Figure 2-4 illustrates the architecture of the model and provides the basis for the discussion in this section. Each of the major components of the model is briefly discussed, and intended interactions between the aspects of the model are introduced. Details of each aspect of the model are covered in the sections, Process Capability Aspect of the SE-CMM and Domain Aspect of the SE-CMM, found later in this chapter.

Diagram of the SE-CMM architecture

The following diagram illustrates the SE-CMM architecture. As stated earlier, the model is divided into two aspects: the domain aspect, focusing on characteristics that are specific to the systems engineering process, and the capability aspect, focusing on generic process characteristics that contribute to overall process management and institutionalization capability. The elements shown in this figure are explained in this section and Sections 2.4-2.6.

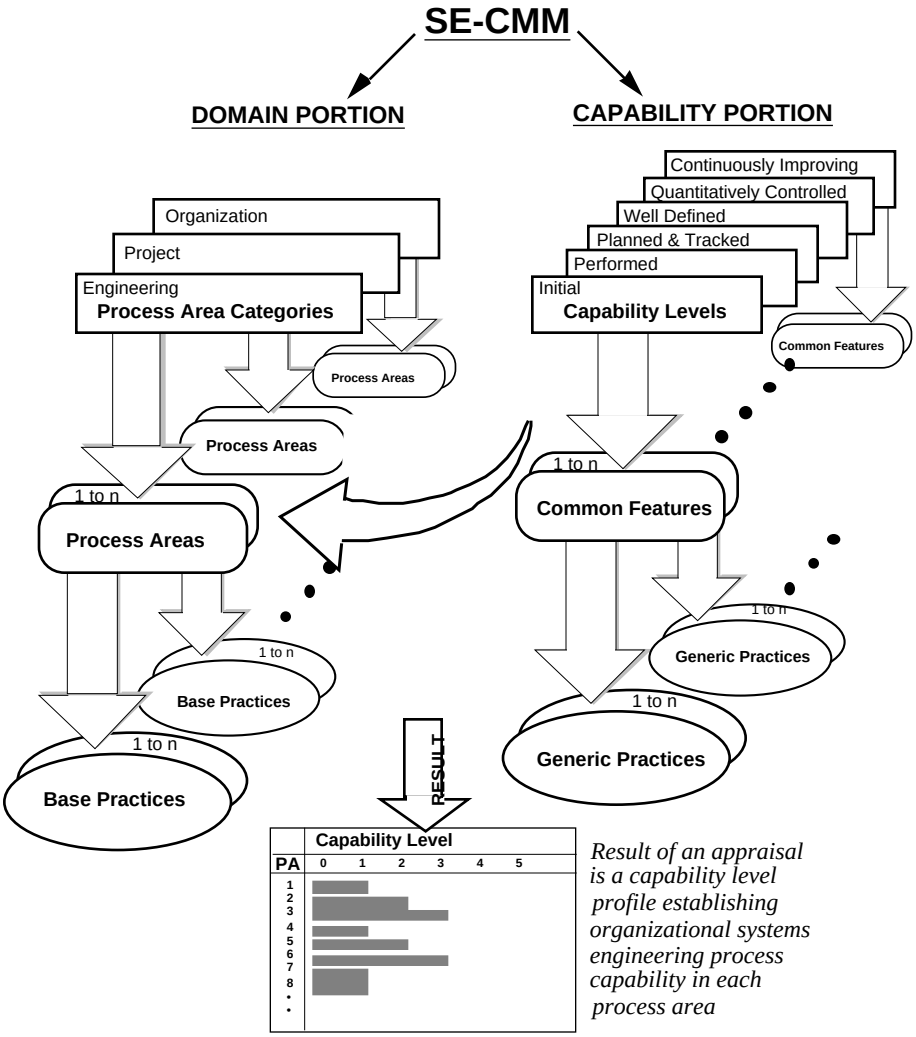

Figure 2-4. Diagram of SE-CMM Architecture

continued on next page

Dual-path architecture

The dual-path architecture shown in Figure 2-4 was adopted with only slight modification from an early version of the model chosen by the SPICE Project's Baseline Practices Guide. This approach was deemed particularly applicable to the SE-CMM because it clearly separates basic characteristics of the systems engineering process (the domain aspect) from process management and institutionalization characteristics of the systems engineering process (capability aspect).

Architectural components of the capability aspect

Table 2-1 contains the basic definitions of the components of the capability aspect of the SE-CMM. They are further explained in the process capability section (Section 2.4), as well as elaborated in Chapter 4a.

Architectural Component	Definition	Example
Capability Level	A set of common features (sets of activities) that work together to provide a major enhancement in the capability to perform a process (SPICE)	2 Planned and Tracked
Common Feature	A set of practices that address the same aspect of process management or institutionalization (SPICE)	2.1 Planning performance
Generic Practice	An implementation or institutionalization practice that enhances the capability to perform any process (SPICE)	2.1.3 Document the process. Document the approach to performing the process area in standards and/or procedures

Table 2-1. Components of the Process Capability Aspect of the SE-CMM

continued on next page

The process capability aspect of the SE-CMM

The SE-CMM groups process capability in three tiers: capability levels, common features, and generic practices. The capability levels indicate increasing levels of process maturity and are composed of one or more common features. Each common feature is further detailed by several generic practices.

The common features are designed to describe major shifts in an organization's characteristic manner of performing work processes (in this case, the systems engineering domain). Each common feature has one or more generic practices. With one exception, the generic practices can be applied to each of the process areas (from the domain side of the SE-CMM) in addition to the basic performance of the practice. The one exception is the first common feature, "Base practices are performed."

The first capability level has only one generic practice. It is the "do it" generic practice. It asks, "Does someone in your environment do each of the base practices as a part of their process for accomplishing the kind of work described in this process area?" Answering "yes" to this question for each base practice of a process area means that the process area is informally performed (level 1).

The subsequent common features have generic practices that help determine how well a project manages and improves each process area as a whole. The generic practices, described in Chapter 4A, are grouped to emphasize any major shift in an organization's characteristic manner of doing systems engineering.

continued on next page

Capability levels

Table 2-2 lists the capability levels and common features of the capability aspect of the SE-CMM.

Capability Level	Common Features
Continuously Improving	• Improving organizational capability • Improving process effectiveness
Quantitatively Controlled	• Establishing measurable quality goals • Objectively managing performance
Well Defined	• Defining a standard process • Perform the standard process
Planned and Tracked	• Planning performance • Disciplined performance • Verifying performance • Tracking performance
Performed Informally	• Base practices performed

Table 2-2. SE-CMM Capability Levels

Derived requirements

Because the architecture for the model was not expressed in the project requirements, there are several areas where, based on the selected architecture, we developed derived requirements that address particulars implied by the SPICE architecture. These derived requirements reflect mostly issues such as criteria for including/excluding process areas, or criteria for base or generic practices.

Derived requirements for generic practices

The following criteria express the derived requirements for a generic practice

- A generic practice can be applied to all process areas.
- Only one generic practice is necessary to achieve a Level 1 in each process area (i.e., generic practice 1.1, Perform the Practice.).
- Redundancy with base practices is allowed for special emphasis.
- Practices that are essential to a given level of process capability are included.
- Where generic practice topics overlap with process area topics, the generic practice focuses on the deployment and management aspect of the topic.

continued on next page

The domain aspect of the SE-CMM

The SE-CMM characterizes the systems engineering domain by using process areas. Each process area is further detailed by several base practices and explanatory notes. There are 18 process areas, which are grouped into 3 process categories: engineering, project, and organization.

The 18 process areas are designed to describe the major topic areas essential to effective systems engineering within an organization. In your home organization, your process will include base practices from the process areas that are executed by (or primarily by) individuals in the role of systems engineers. These are the practices primarily grouped in the *engineering* category. Other process areas may be included in processes that are executed by people who are performing other roles. These are the *project* and *organization* process areas, which can also be thought of as *support* process areas.

The authors included support process areas in the SE-CMM because effective systems engineering is unlikely unless these support tasks are performed. For example, it is unlikely that effective systems engineering will be executed if no one ensures that all the engineering staff is working to the same requirement and design baselines at a given period in time (an aspect of the Manage Configurations process area). The point of the SE-CMM is not to indicate "who" does the kinds of things described in a particular process area, but to indicate that the work needs to be performed by someone regardless of their role.

Architectural components of the domain aspect

Table 2-3 contains the basic definitions of the components of the domain aspect of the SE-CMM.

Architectural Component	Definition
Process Category	A set of process areas addressing the same general area of activity
Process Area	A set of related practices, which when performed collectively, can achieve the purpose of the process area (SPICE)
Base Practice	An engineering or management practice (activity) that addresses the purpose of a particular process area and thus belongs to it (SPICE)

Table 2-3. Components of the Domain Aspect of the SE-CMM

continued on next page

Process areas of the domain aspect

Table 2-4 lists the 18 process areas. To emphasize that the SE-CMM does not prescribe a specific process or sequence, the process areas are arranged alphabetically by title within each group.

Engineering Process Areas	Project Process Areas	Organizational Process Areas
Analyze Candidate Solutions	Ensure Quality	Coordinate with Suppliers
Derive and Allocate Requirements	Manage Configurations	Define Organization's Systems Engineering Process
Evolve System Architecture	Manage Risk	Improve Organization's Systems Engineering Processes
Integrate Disciplines	Monitor and Control Technical Effort	Manage Product Line Evolution
Integrate System	Plan Technical Effort	Manage Systems Engineering Support Environment
Understand Customer Needs and Expectations		Provide Ongoing Knowledge and Skills
Verify and Validate System		

Table 2-4. SE-CMM Process Areas

continued on next page

Life-cycle coverage

The SE-CMM process areas provide complete coverage of the life cycle of the system. This life-cycle coverage stems in part from the recognition that the same basic systems engineering activities are appropriately applied to each product life-cycle phase. The key aspect of detecting when each activity should be addressed for a given phase is addressed in the Ensure Quality process area (PA08).

Process area requirements

In developing the model, the authors needed to determine the basis for including or not including a process area within the model. The following criteria were developed for evaluating if a process area should be included:

- The process area is essential for effective systems engineering to exist within an organization.
- The process area's purpose is not addressed sufficiently in the generic practices.
- The process area's purpose is considered too important by the author team to be left out.
- The process area assembles key concepts in one area for ease of use.

Derived requirements for base practices

The following criteria express the derived requirements for a base practice:

- The base practice is considered by the authors to be essential to the practice of good systems engineering.
- The base practice is considered by the authors to be essential to achieve a capability level 1 within that process area.
- Redundancy with generic practices is allowed for special emphasis.
- Where base practice and generic practice topics overlap, the base practice focuses on the performance of the primary activities related to the topic.

2.4 Process Capability Aspect of the SE-CMM

Why address process capability?

There are dozens of sources of theory and practice that describe the benefits of improving process capability. (See the bibliography in the *CMM for Software v1.1* [Paulk 93a] for a starter list.) For most organizations, the ability to estimate and predict accurately the results of their product development activities from a viewpoint of cost, schedule, and quality is a fundamental business goal. Case studies from the software engineering community and elsewhere suggest that addressing issues of process management, measurement, and institutionalization improve the organization's ability to meet its cost, quality, and schedule goals [Herbsleb 94].

Why is process capability separated from the process areas?

As experience in applying process improvement principles in different environments has evolved, principles that contribute significantly to increasing capability have been noted and analyzed. The separation of the process capability practices from domain-specific practices as described in the previous section, provides two major benefits:

- Most product development activities encompass many disciplines and domains. The ability to use a set of focused process improvement principles as a guide for appraisal and improvement across those disciplines improves communication among them, and provides leveraging opportunities which are not present if the principles are embedded in discipline-specific expressions of capability, such as occurs in the CMM for Software v1.1.

- The separation of process capability practices from domain-specific practices provides an opportunity for guidance that transcends organizational and role-based boundaries. For example, the common feature on planning performance can be applied before the common feature on verifying performance. These common features, as detailed by their generic practices, are clearly independent of business area and application domain. This improves communication and adoption of these principles across a wide spectrum of industries.

continued on next page

Process capability level diagram

The following diagram illustrates the improvement path adopted by the SE-CMM Project.

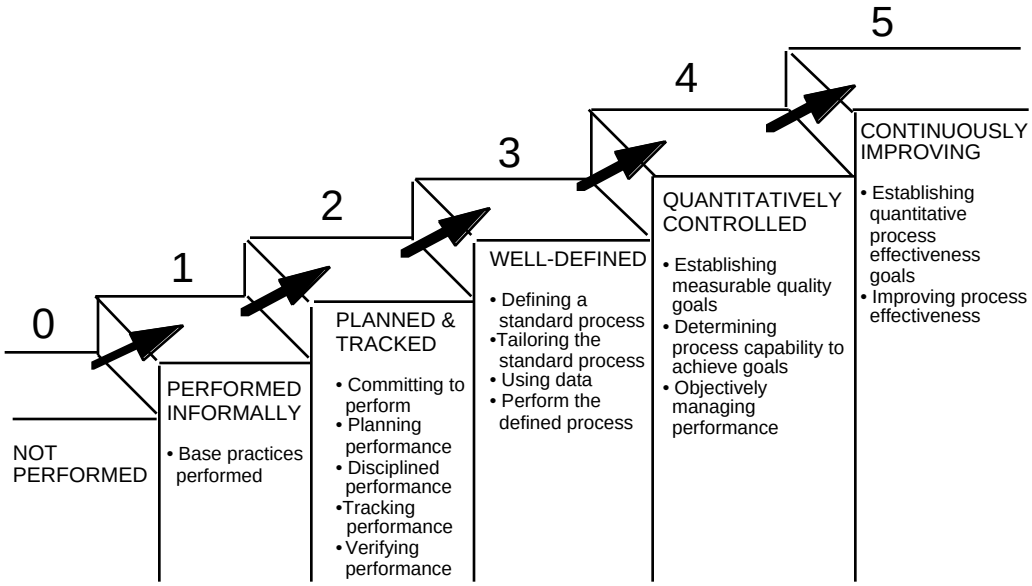

Figure 2-5. Improvement Path for Process Capability

continued on next page

Why group common features by capability level?

By their nature, there is more than one way to group practices into common features and common features into capability levels.

In order to separate "how well you do something" from "what you do," the SE-CMM based its approach on an early version of the SPICE Baseline Practices Guide.

The following discussion addresses these common features.

The ordering of the common features stems from the observation that implementation and institutionalization of some practices benefit from the presence of others. This is especially true if practices are well established. Before an organization can define, tailor, and use a process effectively, individual projects should have some experience managing the performance of that process. For example, before institutionalizing a specific estimation process for an entire organization, the organization should first attempt to use the estimation process on a project. However, some aspects of process implementation and institutionalization should be considered together (not one ordered before the other) since they work together toward enhancing capability.

Common features and capability levels are important both in performing an assessment and improving an organization's process capability. In the case of an assessment where an organization has some, but not all common features implemented at a particular capability level for a particular process, the organization usually is operating at the lowest completed capability level for that process. For example, at capability level 2, if the Tracking Performance common feature is lacking, it will be difficult to track project performance. If a common feature is in place, but not all its preceding ones (i.e., those at lower capability levels), the organization may not reap the full benefit of having implemented that common feature. An assessment team should take this into account in assessing an organization's individual processes.

In the case of improvement, organizing the practices into capability levels provides an organization with an "improvement road map" should it desire to enhance its capability for a specific process. For these reasons, the practices in the SE-CMM are grouped into common features which are ordered by capability levels.

In either case, an assessment should be performed to determine the capability levels for each of the process areas. This indicates that different process areas can and probably will exist at different levels of capability. The organization will then be able to use this process-specific information as a means to focus improvements to its processes. The priority and sequence of the organization's activities to improve its processes should take into account its business goals.

continued on next page

Common features

Common features are groupings of generic practices appropriate within capability levels. For example, common features included in the Planned and Tracked level (Level 2) are Planning Performance, Disciplined Performance, Tracking Performance, and Verifying Performance. An expansion of each feature is provided in Chapter 4A. See Table 2-2 for a complete list of common features.

Generic practices

Generic practices are a series of activities that apply to all processes. They address the management, measurement, and institutionalization aspects of a process. In general, they are used during an appraisal to determine the capability of any process. Generic practices are, as mentioned earlier, grouped by common feature and capability level.

A note on measurement throughout the SE-CMM

The SE-CMM addresses measurement in two ways. On the capability side, the definition of a standard process or process family necessitates the incorporation of measurement. At capability level 2, the generic practice Track with Measurement emphasizes the use of measurement in tracking the use of a process. The common feature Establishing Measurable Quality Goals adds emphasis in terms of quantitative quality goals for higher levels of maturity.

On the domain side, the process areas Plan Technical Effort (PA12) and Monitor and Control Technical Effort (PA11) describe basic measurements that support systems engineering. The base practices of the Ensure Quality process area (PA08) describe measurement of the quality of the systems engineering process and of the work products of all the process areas. References to measurement and measurement-related issues are embedded within the SE-CMM rather than addressed separately to emphasize the integration of measurement into the activities and processes being described or performed.

2.5 Capability Levels

Introduction

This section describes the six capability levels to provide the reader with a sense of the changes that would be expected as a process within a project or organization increases in capability.

The Not Performed level

The Not Performed level (Level 0) displays no common features. It is characteristic of an organization just entering the systems engineering field, or one that has not focused on the systematic application of systems engineering principles in their product development. They accomplish some of the tasks, but are not necessarily sure how. Performance is not generally consistent, particularly if key individuals are absent or the tasks become more complex.

The Not Performed level has no common features. There is general failure to perform the base practices in the process area. Where there are work products that result from performing the process, they are not easily identifiable or accessible.

The Performed Informally level

At this level, all base practices are performed somewhere in the project's or organization's implemented process. However, consistent planning and tracking of that performance is missing. Good performance, therefore, depends on individual knowledge and effort. Work products are generally adequate, but quality and efficiency of production depend on how well individuals within the organization perceive that tasks should be performed. Based on experience, there is general assurance that an action will be performed adequately when required. However, the capability to perform an activity is not generally repeatable or transferable.

The Planned & Tracked level

At the Planned and Tracked level, planning and tracking have been introduced. There is general recognition that the organization's performance is dependent on how efficiently the systems engineering base practices are implemented within the project's or organization's process. Therefore, work products related to base practice implementation are periodically reviewed and placed under version control. Corrective action is taken when indicated by variances in work products.

The primary distinction between the Performed Informally and the Planned and Tracked levels is that at the Planned and Tracked level, the execution of the base practices in the project's implemented process is planned and managed. Therefore, it is repeatable within the implementing project, though not necessarily transferable across the organization.

continued on next page

The Well Defined level

At this level, base practices are performed throughout the organization via the use of approved, tailored versions of standard, documented processes. Data from using the process are gathered and used to determine if the process should be modified or improved. This information is used in planning and managing the day-to-day execution of multiple projects within the organization and is used for short- and long-term process improvement.

The main difference between the Planned and Tacked and Well Defined levels is the use of organization-wide, accepted standard processes that implement the characteristics exhibited by the base practices. The capability to perform an activity is, therefore, directly transferable to new projects within the organization.

The Quantitatively Controlled level

At the Quantitatively Controlled level, measurable process goals are established for each defined process and associated work products, and detailed measures of performance are collected and analyzed. These data enable quantitative understanding of the process and an improved ability to predict performance. Performance, then, is objectively managed and defects are selectively identified and corrected.

The Continuously Improving level

This is the highest achievement level from the viewpoint of process capability. The organization has established quantitative, as well as qualitative, goals for process effectiveness and efficiency, based on long-range business strategies and goals. Continuous process improvement toward achievement of these goals using timely, quantitative performance feedback has been established. Further enhancements are achieved by pilot testing of innovative ideas and planned insertion of new technology.

2.6 Domain Aspect of the SE-CMM

Context of the process areas

The domain aspect of the SE-CMM is a collection of essential elements, called base practices, that are grouped into process areas, as described earlier. The seven process areas in the engineering category are shown below grouped within the organizational and project process areas which support their execution. How process areas were selected is discussed later in this section.

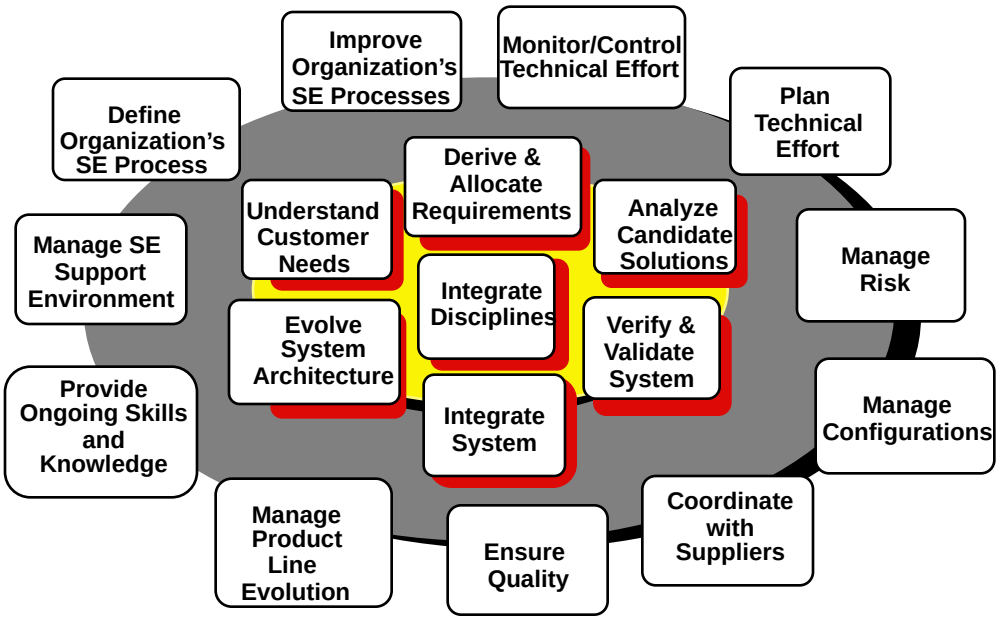

Figure 2-6. SE-CMM Process Areas

continued on next page

Logical vs. chrono-logical arrangement

The depiction of the process areas in Figure 2-6 without connecting lines is deliberate. It is meant to indicate that the process areas are not, by nature, chronologically established. While there is a logical initiation sequence, many are expected to be exhibited in the organization's product development process several times during the development of a product. For example, requirements are developed and refined at several different levels during the system or product development life cycle. The process area titled Derive and Allocate Requirements (PA02) would, therefore, be used as a guide to the implemented process whenever the work product is one or more requirements document or files.

Process categories of the SE-CMM

There are three process categories defined for the SE-CMM. They are

- engineering
- project
- organization

These three categories and their contents are discussed below. The process areas in each category are described in detail in Chapter 4B.

Process areas of the engineering category

The engineering category groups together those process areas that are primarily concerned with the technical and engineering aspects of product development. They are organized alphabetically within the category to discourage the reader from implying a particular sequencing of the process areas. They include

- Analyze Candidate Solutions
- Derive and Allocate Requirements
- Evolve System Architecture
- Integrate Disciplines
- Integrate System
- Understand Customer Needs and Expectations
- Verify and Validate system

continued on next page

Process areas of the project category

The project category groups together process areas that are primarily concerned with providing the technical management infrastructure needed to develop a product successfully. Like the process areas in the engineering category, they are organized alphabetically. They include

- Ensure Quality
- Manage Configurations
- Manage Risk
- Monitor and Control Technical Effort
- Plan Technical Effort

Process areas of the organization category

The organization category groups together process areas that are primarily concerned with providing a business infrastructure that directly supports the needs of systems engineering, but that are usually found concentrated at an organization, rather than a project level. Like the other categories, they are organized alphabetically. They include

- Coordinate with Suppliers
- Define Organization's Systems Engineering Process
- Improve Organization's Systems Engineering Processes
- Manage Product Line Evolution
- Manage Systems Engineering Support Environment
- Provide Ongoing Skills and Knowledge

continued on next page

Rationale for inclusion of selected process areas

Especially when looking at the support process areas of the SE-CMM, questions often arise as to why certain process areas are included or excluded from the model. Following is a brief discussion of the rationale for including process areas about which the author team has received such inquiries.

The process areas Manage Configurations (PA09) and Provide Ongoing Skills and Knowledge (PA17) were considered to be essential for effective systems engineering to exist within an organization, even though they may not be a primary systems engineering responsibility. The Plan Technical Effort process area (PA12) was included because it was believed that the generic practices did not provide sufficient guidance to the model user to be of significant value. The Ensure Quality process area (PA08) was considered too important by the author team to leave out even though there was significant discussion that the fundamental concepts were covered in the Define Organization's Systems Engineering process area (PA13). The Manage Risk process area (PA10) was included as a process area for ease of use, since the other alternative was to spread the concepts throughout the model, dispersing the practices throughout other process areas. The Coordinate with Suppliers process area (PA18) was requested by September 1995 workshop participants and companies contributing SE-CMM authors.

continued on next page

Balancing the process areas and capability levels

Selecting the process areas to be included within the SE-CMM was a compromise between completeness and having a reasonable number of process areas to deal with when improving and appraising processes. We selected process areas that are

- essential elements of systems engineering
- crucial to the success of a systems engineering activity even if they are not performed by systems engineering. For example, it would be difficult to appraise a systems engineering activity without knowing whether configuration management is consistently practiced and supported.
- activities covered in the generic practices, but requiring more detail specific to systems engineering
- common sources of difficulty in achieving quality results from the systems engineering activities, and thus require special emphasis
- the subject of intense concerns among managers and are needed to ensure that the area gets the amount of attention that management feels is appropriate. One example of this type of process is the Ensure Quality process area (PA08), which is included to meet management concerns and to assemble in one area essential activities that are crucial to high-quality outputs of the projects' and organization's processes.

Inclusion of support process areas among the process areas can provide the opportunity to describe the basic elements of support activities without having to include extra generic practices which would necessarily apply to all process areas.

Control and sequencing concepts

The SE-CMM specifies a number of practices that should occur in the implemented process of a project. It is silent on the control and sequencing of the implemented process activities that carry out these practices. Nevertheless, it is a general requirement of the SE-CMM that a well-defined process should describe the control and sequencing of process activities to accomplish the purposes of the process efficiently and to produce a quality product (See capability level 3 in Chapter 4A).

There are several types of sequencing that are common and expected to be seen in implementation: waterfall, iteration, concurrency, and recursion. These are briefly discussed below.

Waterfall

The waterfall sequence implies that activities are executed one-after-another until the last is reached. The outputs of one are furnished to the later ones in the sequence. This is a common way of describing processes, but is rarely implemented exactly as described.

continued on next page

Iteration

Iteration implies that some activities are executed over and over again until some exit criteria are satisfied. An example is a sequence of an activity that produces a work product, and a verification activity, which checks that requirements are satisfied. If the work product is acceptable, the iteration loop is exited; if not, the loop is executed again. Figure 2-7 illustrates iteration.

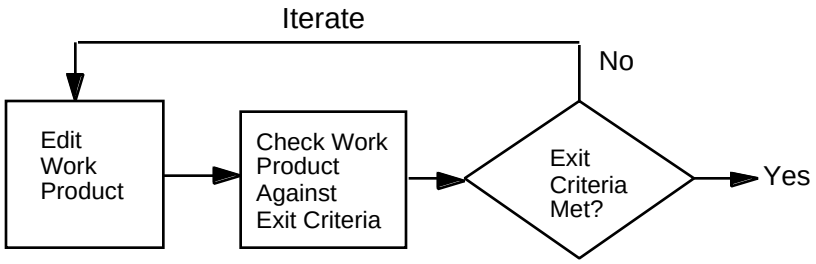

Figure 2-7. Iteration

Concurrency

Concurrency is appropriate when two or more activities are producing independent work products or when the results of two or more activities are closely coupled and interdependent. The activities are executed at the same time and appropriate interim data are passed back and forth between them as necessary. Concurrency may be an effective way to reduce cycle time and to make efficient use of resources. Control of concurrence should be specified in the project plan. Figure 2-8 illustrates concurrency.

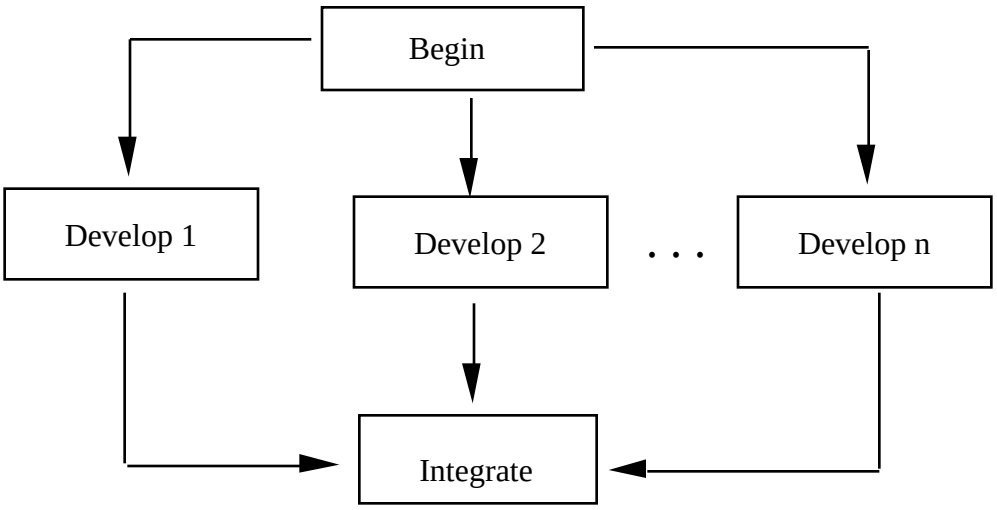

Figure 2-8. Concurrency

continued on next page

Recursion

Recursion is the invocation of an activity by the same activity in a new context to accomplish a task subordinate to the invoking task. It is useful in applying system engineering activities to subsystems resulting from decomposition of requirements. This form of recursion may continue to lower levels. Figure 2-9 illustrates recursion.

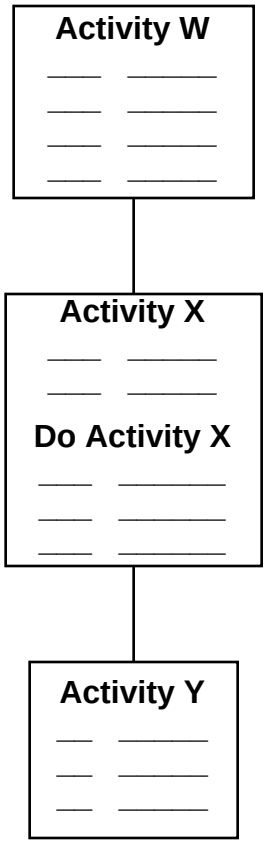

Figure 2-9. Recursion

2.7 A Path for Improving Systems Engineering Maturity

An example path through the SE-CMM

The following discussion addresses a possible path an organization might take in improving its systems engineering processes. This provides a context for describing some of the implied relationships between the capability levels and process areas that exist in the SE-CMM.

The assumption in the following discussion is that an organization's primary goal of an improvement effort is to maximize the utility of the systems engineering function. These functions may be perceived as being the direct "value-added" processes.

There may be temptations in some organizations to focus solely on improving their engineering processes. The drawback with such an approach is that it fails to consider the interaction between engineering, project, and organization process areas. For example, a mature engineering process requires a organization that provides a supporting structure to operate in. Correspondingly, the project maturity must be such that it fosters a cohesive working environment.

An organization using the SE-CMM for process improvement, can use the following building block approach.

continued on next page

Performed Informally Engineering

Figure 2-10 illustrates a summarized capability level profile for an organization, each of whose engineering processes are operating at capability level 1, the Performed Informally level.

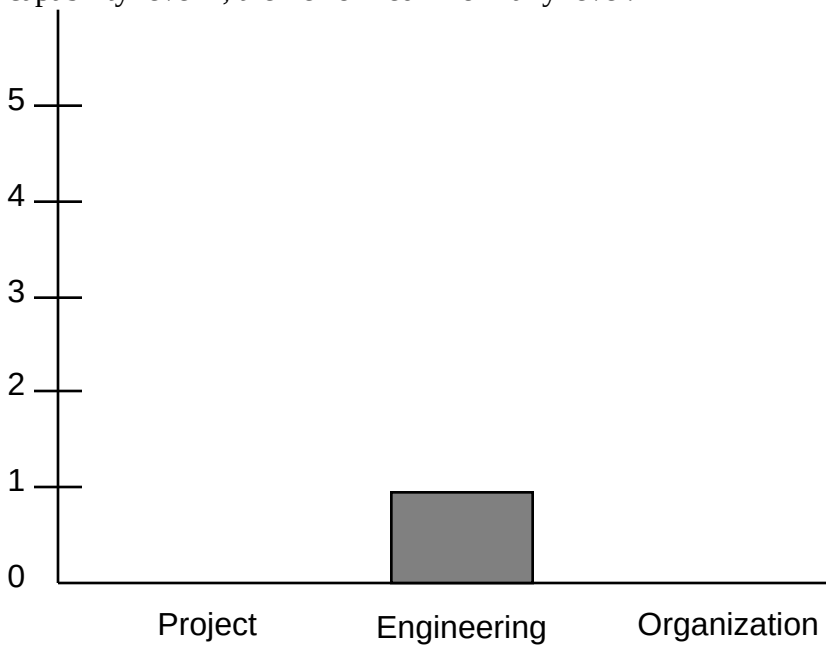

Figure 2-10. Capability Profile for Level 1 Engineering

Discussion of Level 1

The engineering processes in an organization can be matured to capability level 1–Performed Informally–without addressing the project management or organizational process areas. The benefit of addressing the engineering process areas is that the first benefits of process improvement are seen by an organization's systems engineering community. This helps build motivation for further improvement.

continued on next page

Planned and Tracked Engineering

Figure 2-11 illustrates a summarized capability level profile for an organization whose engineering processes are operating at capability level 2, the Planned and Tracked level.

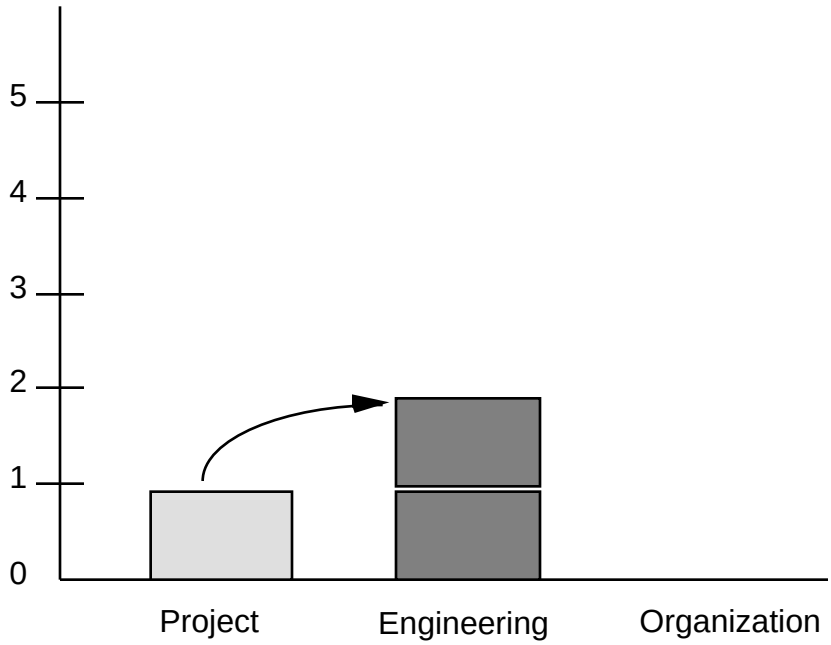

Figure 2-11. Capability Profile for Level 2 Engineering

Discussion of Level 2

It is most effective to mature the engineering processes in an organization to capability level 2–Planned and Tracked–after the project process areas have been brought to Level 1. The project process areas address the necessary support for good systems engineering that must be done, and that can be done effectively at the project-level to take advantage of commonalities in planning and control. The benefits of process improvement are seen at the project-level by systems engineers and those with whom they regularly interface.

continued on next page

Well Defined Engineering

Figure 2-12 illustrates a summarized capability level profile for an organization whose engineering processes are operating at capability level 3, the Well Defined level.

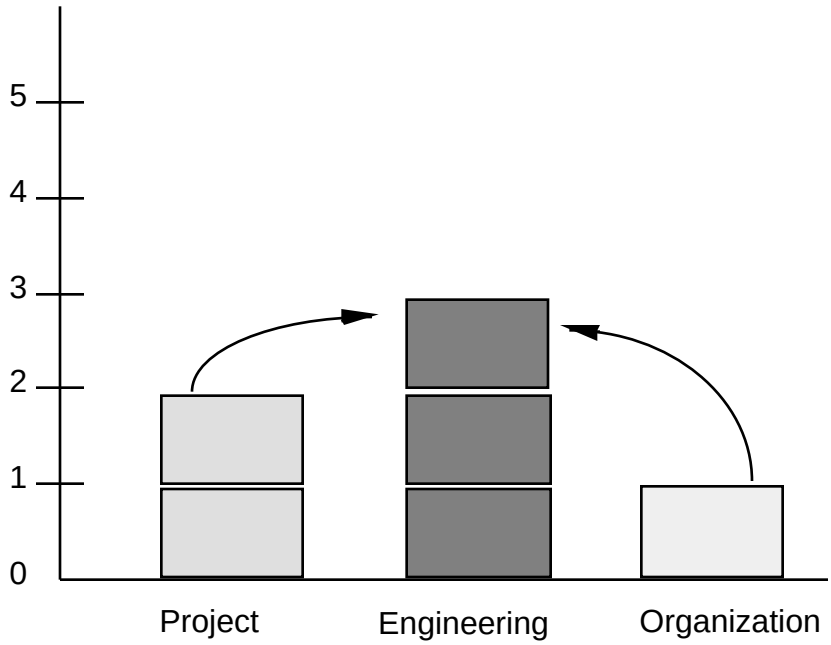

Figure 2-12. Capability Profile for Level 3 Engineering

Discussion of Level 3

Bringing the engineering processes in an organization up to capability level 3–Well Defined, can most effectively be done after the project process areas have been brought to Level 2 and after the organizational process areas have been brought to Level 1. Planning and tracking the project process areas at the project level provides the quality, configuration, risk, and planning control necessary to provide data to organizational process improvement activities. Performing the organizational process areas provides the necessary resources for use by the engineering process areas at Level 3. Such resources include organization's standard systems engineering process, the systems engineering support environment, and obtaining systems engineering skills and knowledge.

continued on next page

Quantitatively Controlled Engineering

Figure 2-13 illustrates a summarized capability level profile for an organization whose engineering processes are operating at capability level 4, the Quantitatively Controlled level.

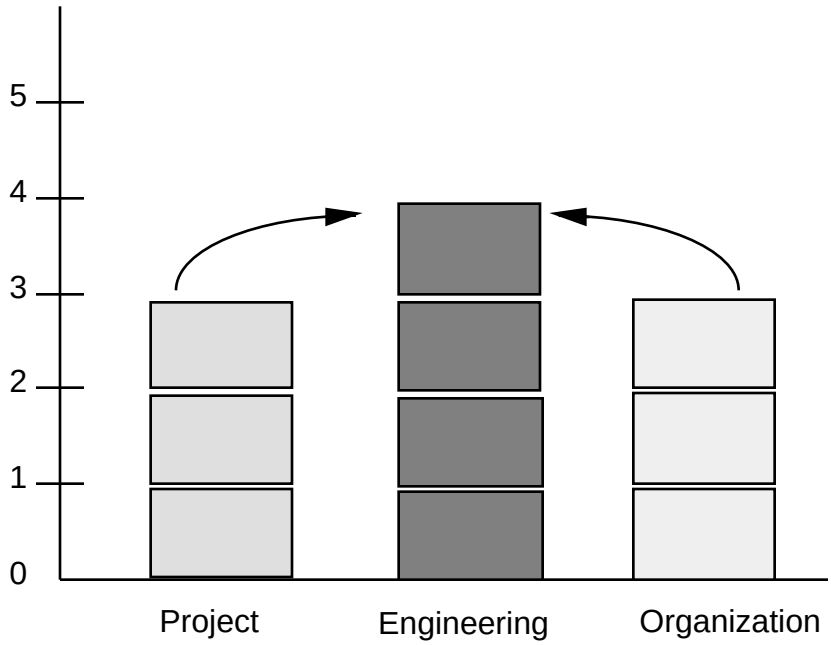

Figure 2-13. Capability Profile for Level 4 Engineering

Discussion of Level 4

Bringing the engineering processes in an organization up to capability level 4–Quantitatively Controlled, can most effectively be done after the project and organizational process areas have been brought to Level 3. Having well-defined engineering, project and organizational process areas provides the data (from Generic Practices 3.2.2 and 3.2.3) and the necessary supporting structures on which improvements in the engineering process areas can be based. At Level 4, an organization will be able to determine the process capability of its engineering processes. Knowing the capability of a process is the foundation of statistical process control.

continued on next page

Continuously Improving Engineering

Figure 2-14 illustrates a summarized capability level profile for an organization whose engineering processes are operating at capability level 5, the Continuously Improving level.

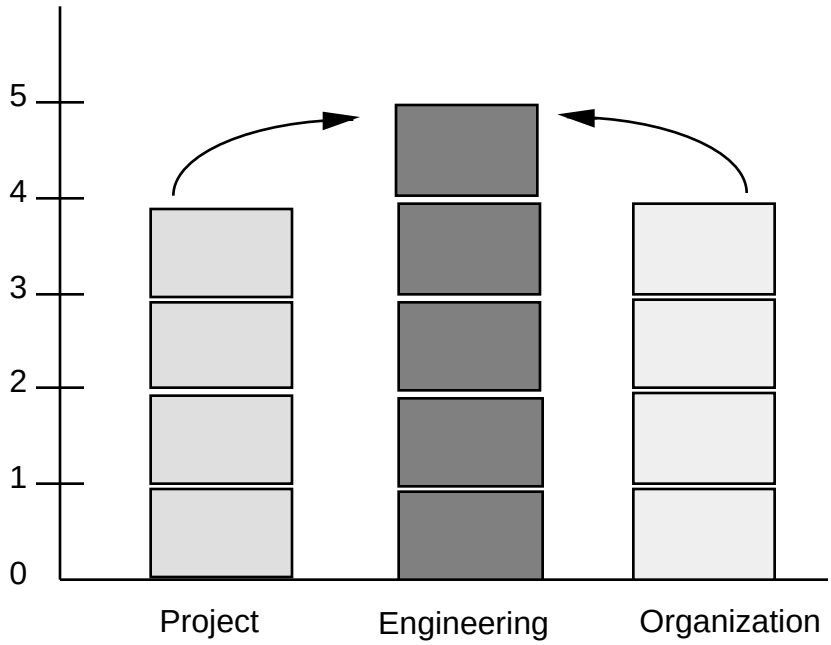

Figure 2-14. Capability Profile for Level 5 Engineering

Discussion of Level 5

Bringing the engineering processes in an organization up to capability level 5–Continuously Improving, can most effectively be done after the engineering, project and organizational process areas have been brought to Level 4. Having quantitatively controlled processes provides the data necessary to execute continuous improvement based on process capability data.

Chapter 3: Using the SE-CMM

Introduction

This chapter provides information on using the SE-CMM for organizational process improvement and design.

In this chapter

Topic	See Page
3.1 Many Usage Contexts	3-2
3.2 Using the SE-CMM to Support Appraisal	3-5
3.3 Using the SE-CMM to Support Process Improvement	3-8
3.4 Using the SE-CMM in Process Design	3-13

3.1 Many Usage Contexts

Product/ project context

Practitioners in systems engineering recognize that there are as many product contexts as there are products in the marketplace, and the methods used to develop products are as varied as the products themselves. However, there are some issues related to product and project context that are known to have an impact on the way products are conceived, produced, delivered, and maintained. Two issues in particular have significance for the SE-CMM

- type of customer base (negotiated vs. market driven)
- production cycle (small run, high value vs. large run, lower value)

The differences between two diverse customer bases and the impacts of those differences in the SE-CMM are discussed below. This is provided as an example of how an organization or industry segment might go about analyzing of how to use the SE-CMM appropriately in their environment.

SE-CMM not limited to a particular industry segment

Every industry reflects its own particular culture, nomenclature, and communication style. By minimizing the role dependencies and organization structure implications, the authors hope that practitioners from all industry segments will be able to easily translate the concepts expressed in the SE-CMM into their own language and culture. However, because of the makeup of the author team, it is natural that the language used to convey SE-CMM concepts has some flavor of the aerospace contractor industry, in which many of the authors have spent significant portions of their careers. Users are urged to look beyond specific terminology differences to the common concepts underneath the terminology. Users are also encouraged to communicate problems using the SE-CMM to the project, via the issue form attached to this document.

continued on next page

Type of customer base

The SE-CMM can be applied in both single-customer and multi-customer segments. The table below illustrates some differences that are evident in single vs. multi-customer segments that relate to the SE-CMM. Because of these differences, SE-CMM users may find it useful to tailor the terminology in the model to reflect their customer segment.

Aspect	Characteristics Seen with Single Customer	Characteristics Seen with Multiple Customers	SE-CMM Implications
Number of customers	One entity, either one individual or one organization	Many entities, either many individuals who can be segmented according to specific characteristics, or many organizations	Language related to customer, customer surrogates should be emphasized
Visibility of the customer	Customer is highly visible to the developer	Customer is not often directly visible to the developer: surrogates, such as focus groups or marketing departments, provide the interface to the developer	Understand Customer Needs process area (PA) must be interpreted to suit the context
Methods of measuring customer satisfaction	• Award of follow on work • Periodic reviews • Award fee • Incentive fee • Customer feedback	• Marketplace buying patterns • Creation of follow-on customer demands • Customer survey	Manage Product Line Evolution PA and other organizational PAs may be affected by how support functions are viewed in relation to customer-focused activities

Table 3-1. Customer Base

continued on next page

Relationship to the "Six Steps to Six Sigma"

The concepts of Six Sigma were pioneered by the Motorola Company. They are quality improvement concepts that have growing acceptance in U.S. industry. Elements of the SE-CMM correspond to the Six Sigma concepts. For example, step two and three correspond to the Understand Customer Needs and Expectations process area (PA06). The literature expresses the "Six Steps to Six Sigma" for both manufacturing and intellectual work. The steps for intellectual work are repeated here [Morgello 91].

The mapping of the Six Steps to Six Sigma to several parts of the SE-CMM, is shown in Table 3-2.

Six Steps to Six Sigma	SE-CMM
Identify your products/services	PA 15: Manage Product Line Evolution
Identify your customers	PA 6: Understand Customer Needs and Expectations
Identify needs	PA 6: Understand Customer Needs and Expectations
Define the process	PA 13: Define Organization's Systems Engineering Process PA 14: Improve Organization's Systems Engineering Processes
Mistake-proof the process and eliminate wasted effort	PA 14: Improve Organization's Systems Engineering Processes GP 2.4.1: Track with measurement GP 3.2.3: Use well-defined data GP 4.2.1: Determine process capability GP 4.2.2: Use process capability GP 5.2.3: Continuously improve the defined process
Ensure continuous effort	GP 5.1.2: Continuously improve the standard process GP 5.2.3: Continuously improve the defined process

Table 3-2. Six Steps to Six Sigma

3.2 Using the SE-CMM to Support Appraisal

Introduction

The SE-CMM is structured to support a wide variety of improvement activities, including self-administered appraisals or internal appraisals augmented by expert "facilitators" from inside or outside the organization. Although it is primarily intended for internal process improvement, it can also be used to evaluate a potential vendor's capability to perform its systems engineering process. (This use is not recommended by the SE-CMM Project at this time.)

The SE-CMM Appraisal Method

Although it is not required that any particular appraisal method be used with the SE-CMM, an appraisal method designed to maximize the utility of the model has been designed by the SE-CMM Project. The SE-CMM Appraisal Method (SAM) is fully described, along with some support materials for conducting appraisals, in SECMM-94-06|CMU/SEI-94-HB-05, *SE-CMM Appraisal Method Description.* The basic premises of the appraisal method are listed here to provide context for how the model might be used in an appraisal.

Features of the SAM

SAM is an organizational or project-level appraisal method that uses multiple data-gathering methods to obtain information on the processes being practiced within the organization or project selected for appraisal. The purposes of a SAM-style appraisal in its first release version are twofold:

- Obtain a baseline or benchmark of actual practice related to systems engineering within the organization or project.
- Create and support momentum for improvement within multiple levels of the organizational structure.

SAM is a method which is tailorable to meet the organization's or project's need, and some guidance on tailoring is provided in the SAM description document.

Data gathering is primarily via questionnaires that directly reflect the contents of the model, and a series of both structured and unstructured interviews with key personnel involved in the performance of the organization's processes. Some of these individuals would be considered systems engineers, while others would be in other roles (e.g., configuration managers) that support systems engineering tasks.

Multiple feedback sessions are conducted with the appraisal participants, culminating in a briefing to all participants plus the sponsor of the appraisal. Capability levels are assigned to each of the process areas that were appraised. The briefing also includes a set of prioritized strengths and weaknesses that support process improvement based on the organization's stated appraisal goals.

continued on next page

Determining capability to perform systems engineering processes

Figure 3-1 illustrates how the process areas (base practices) and the common features (generic practices) can be used to determine the process capability of systems engineering processes. A capability level of 0 to 5 can be determined for each process area.

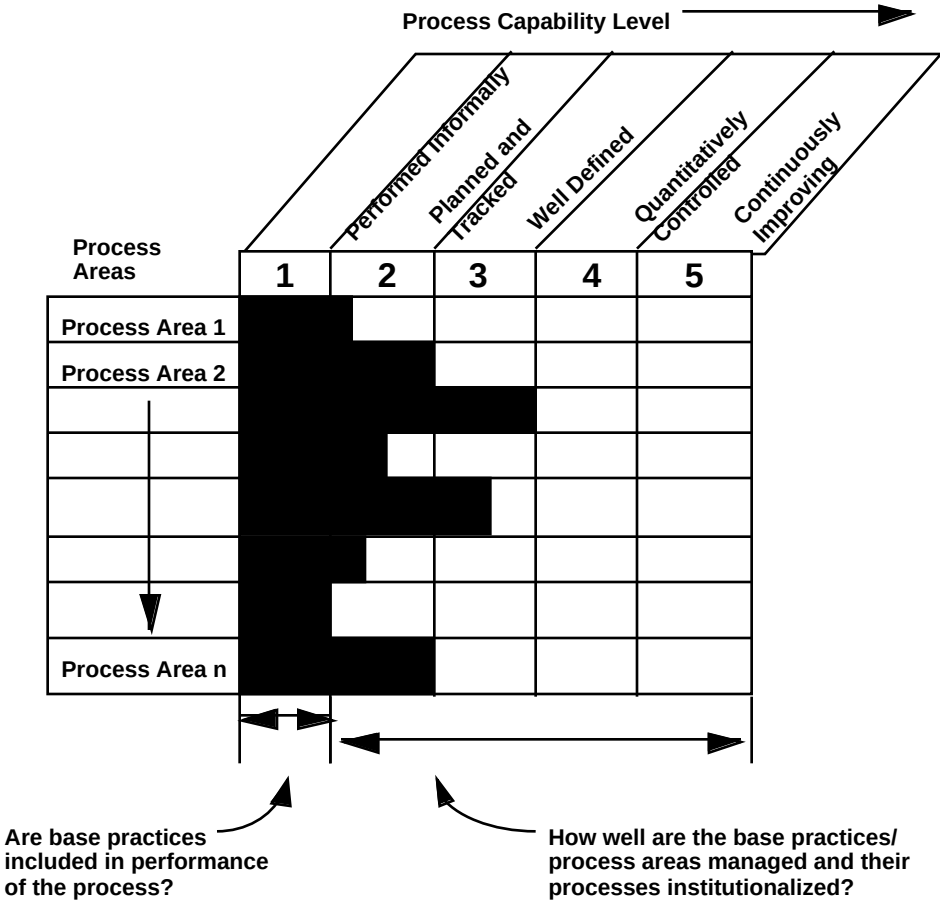

Figure 3-1. Determining Process Capability

Using both sides of the architecture in appraisal

The first step in developing a profile of an organization's capability to perform its systems engineering process is to determine whether the basic systems engineering process (all the base practices) is *implemented* within the organization (not just written down) via their performed process. The second step is to assess how well the characteristics (base practices) of the process that have been implemented are managed and institutionalized by looking at the base practices in the context of the generic practices. Consideration of both the base practices and generic practices in this way results in a process capability profile that can help the organization to determine the improvement activities that will be of most benefit in the context of its business goals.

continued on next page

Relationship between generic and base practices

Because process capability levels are primarily determined by applying the generic practices to the base practices, the SE-CMM may appear to contain a certain amount of redundancy between the generic practices and base practices. This is most visible when looking at some of the project and organizational process areas. Chapter 4A addresses how the generic practices are applied during an appraisal.

Example of relationship between generic/base practices

The SE-CMM contains both base practices and a generic practice that address configuration management: the Manage Configurations process area (PA09) and generic practice 2.2.2 ("Place work products of the process area under version control or configuration management, as appropriate"). However, the focus of Manage Configurations is the *process* being used for managing configurations. The generic practice, however, addresses whether the project's process for configuration management results in action.

In general, the base practices in cases such as this should be viewed as guidance on the basic aspects of the topics that need to be addressed, and the related generic practices deal with deployment of the base practices to the project. Keep in mind that the application of the generic practices to each process area results in a unique interpretation of the generic practice for the subject process area.

Sequencing

The practices of many of the process areas would be expected to be seen a number of times in the execution of an organization's process for the product life cycle. The process areas should be considered a source for practices whenever there is a need to incorporate the process area's purpose in a project's or organization's process. In an appraisal, always keep in mind that the SE-CMM does not imply a sequence: sequencing should be determined based on the organization's or project's selected life cycle and other business parameters (see Section 3.4, Using the SE-CMM in Process Design).

3.3 Using the SE-CMM to Support Process Improvement

Introduction

Either with or without an appraisal to benchmark an organization's systems engineering practices, an organization should consider several aspects of the SE-CMM when using it as the basis to design an improvement program. This section does not provide overall guidance on initiating and conducting an improvement program. There are many sources within industry for approaches to organizational improvement, and most should be able to be used with the SE-CMM or adapted for SE-CMM use.

Prioritizing improvement based on business goals

It should be emphasized that any process improvement effort, using any reference model, should be constructed to support the business goals of the organization. An organization using the SE-CMM should prioritize the process areas relative to their business goals and strive for improvement in the highest priority process areas first.

Tailoring

The model defines only those elements that the authors considered to be essential for the practice of good systems engineering. As such the model is not intended, in general, to be tailored. However, not all projects may need to use processes that exhibit all the characteristics associated with each process area. Under such circumstances, the project should follow a process to tailor out the activity related to the unnecessary process area from the organization's systems engineering process. Tailoring should, in all cases, be based on the organization's goals and customer needs.

continued on next page

Gaining leverage from other experiences

Empirical data are not readily available on the benefits of process improvement to systems engineering. However, because systems engineering has a strong influence on the success of other disciplines, the benefits from improving the systems engineering process are projected to equal or exceed the benefits of process improvement in other disciplines such as software engineering.

In the case of software process improvement, organizations that have done software process improvement for more than three years have gained substantial benefits [Herbsleb 94]:

- return on investment of 7:1
- 37% average gain per year in productivity
- 18% increase per year in the proportion of defects found in pre-test
- 19% reduction in time to market
- 45% reduction in filed error reports per year

This is comparable to published total quality management reports from other industries. Surveys and case studies on software process improvement are listed below to support model users who need to understand the potential analogies between software and systems engineering process improvement.

continued on next page

List of software process improvement references

Joe Besselman and Stan Rifkin, "The Effect of Software Process Improvement on the Economics of Procurement," *Proceedings of the 6th SEPG National Meeting*, Dallas, TX, 25-28 April 1994.

C. Billings, J. Clifton, B. Kolkhorst, E. Lee, and W.B. Wingert, "Journey to a Mature Software Process," *IBM Systems Journal*, Vol. 33, No. 1, 1994, pp. 46-61.

Raymond Dion, "Process Improvement and the Corporate Balance Sheet," *IEEE Software*, Vol. 10, No. 4, July 1993, pp. 28-35.

James Herbsleb, Anita Carleton, et al., *Benefits of CMM-Based Software Process Improvement: Initial Results* (CMU/SEI-94-TR-13). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, August 1994.

Watts S. Humphrey, Terry R. Snyder, and Ronald R. Willis, "Software Process Improvement at Hughes Aircraft," *IEEE Software*, Vol. 8, No. 4, July 1991, pp. 11-23.

A. Johnson, "Software Process Improvement Experience in the DP/MIS Function," *Proceedings of the 16th International Conference on Software Engineering*, IEEE Computer Society Press, Sorrento, Italy, 16-21 May 1994, pp. 323-330.

Jed Johnson, "How We Climbed to Maturity Level Two," *Application Development Trends*, April 1994, pp. 20-23.

W.H. Lipke and K.L. Butler, "Software Process Improvement: A Success Story," *Crosstalk: The Journal of Defense Software Engineering*, No. 38, November 1992, pp. 29-31.

R. A. Radice, J. T. Harding, P. E. Munnis, and R. W. Phillips, "A Programming Process Study," *IBM Systems Journal*, vol. 24, no. 2, 1985.

H. Wohlwend and S. Rosenbaum, "Software Improvements in an International Company," *Proceedings of the 15th International Conference on Software Engineering*, IEEE Computer Society Press, May 1993.

continued on next page

Walk before you run

Although the business goals are the primary driver in interpreting a model such as the SE-CMM, there is a fundamental order of activities and basic principles that drive the logical sequence of typical improvement efforts. This order of activities is expressed in the common features and generic practices of the capability level side of the SE-CMM architecture. These principles and order of activities are summarized in Table 3-3.

Principle	How Expressed in SE-CMM
You have to do it before you can manage it.	Performed Informally level focuses on whether an organization or project performs a process that incorporates the base practices.
Understand what's happening on the project (where the products are!) before defining organization-wide processes.	Planned and Tracked level focuses on project-level definition, planning, and performance issues.
Use the best of what you've learned from your projects to create organization-wide processes.	Well Defined level focuses on disciplined tailoring from defined processes at the organization level.
You can't measure it until you know what "it" is.	Although it is essential to begin collecting and using basic project measures early, i.e., at the Planned and Tracked level, measurement and use of data is not expected organization wide until the Well Defined and, particularly, the Quantitatively Controlled levels have been achieved.
Managing with measurement is only meaningful when you're measuring the right things.	Quantitatively Controlled level focuses on measurements being tied to the business goals of the organization.
A culture of continuous improvement requires a foundation of sound management practice, defined processes, and measurable goals.	Continuously Improving level gains leverage from all the management practice improvements seen in the earlier levels, then emphasizes the cultural shifts that will sustain the gains made.

Table 3-3. Process Improvement Principles in the SE-CMM

continued on next page

Some expected results

Based on analogies in the software and other communities, some results of process and product improvement can be predicted. These are discussed below.

Improving predictability

The first improvement expected as an organization matures is *predictability*. As capability increases, the difference between targeted results and actual results decreases across projects. For instance, Level 1 organizations often miss their originally scheduled delivery dates by a wide margin, whereas organizations at a higher capability level should be able to predict the outcome of cost and schedule aspects of a project with increased accuracy.

Improving control

The second improvement expected as an organization matures is *control*. As process capability increases, incremental results can be used to establish revised targets more accurately. Alternative corrective actions can be evaluated based on experience with the process and other projects' process results in order to select the best application of control measures. As a result, organizations with a higher capability level will be more effective in controlling performance within an acceptable range.

Improving effectiveness

The third improvement expected as an organization matures is *effectiveness*. Targeted results improve as the maturity of the organization increases. That is, as an organization matures, costs decrease, development time becomes shorter, and productivity and quality increase. In a Level 1 organization, development time can be quite long because of the amount of rework that must be performed to correct mistakes. In contrast, organizations at a higher maturity level have increased process effectiveness and have reduced costly rework, allowing overall development time to be shortened.

3.4 Using the SE-CMM in Process Design

Introduction

This section provides brief guidance on issues related to using the SE-CMM to support process design. This section sets a context for how the SE-CMM could be used in a software engineering design activity.

Analyzing your organiza-tional context

Organizations often overlook many internal and/or intermediate processes or products when first defining their processes. However, it is not necessary to address all of the possibilities when first defining a systems engineering process for an organization. The organization should describe with reasonable accuracy its current process as a baseline. It is best to focus on capturing a reasonable baseline process that be produced in six months to a year, and that can be improved over time.

An organization must have a stable baseline to determine whether future changes constitute improvements. There is no value in looking for improvements in a process that the organization does not perform. An organization may find it useful to include current "delays" and "queues" in the baseline process. During subsequent process improvement efforts, these allow a good starting point for cycle-time reduction.

A systems engineering organization may define its process from the point of view of what its systems engineers are responsible for. This may include interfaces with the implementing disciplines of software engineering, electrical engineering, mechanical engineering, and more.

The first step in designing processes that will meet the business needs of an enterprise is to understand the business, product, and organizational context that will be present when the process is being implemented. Some questions that need to be answered before the SE-CMM can be used for process design include

- What life cycle will be used as a framework for this process?
- How is the organization structured to support projects?
- How are support functions handled (e.g., by the project or the organization)?
- What are the management and practitioner roles used in this organization?
- How critical are these processes to organizational success?

Understanding the cultural and business contexts in which the SE-CMM will be used is a key to its successful application in process design.

continued on next page

Adding role and structure information

Figure 3-2 illustrates the factors that need to be added to the content of the SE-CMM process areas and common features to come up with a performable and sustainable process design. It is the organization's context regarding role assignments, organizational structure, systems engineering work products, and product life cycle that, combined with guidance from SE-CMM generic practices and base practices, produces sound organizational processes that have the potential for deliberate improvement.

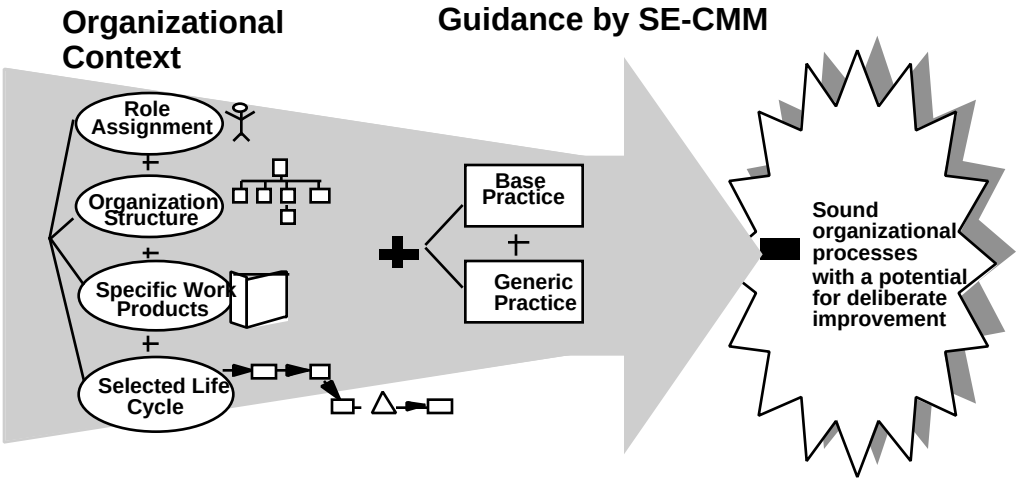

Figure 3-2. Factors for Successful Process Design

Chapter 4: The SE-CMM Generic & Base Practices

Introduction

This chapter contains the practices for both the process capability and domain aspects of the SE-CMM. Section 4A contains the generic practices (process capability aspect), organized by common feature and capability level. Section 4B contains the base practices (domain aspect), organized by process area. The process areas are sequenced alphabetically within each process category.

Chapter 4A: Generic Practices

Introduction

This chapter contains the generic practices, which express the process capability side of the model. These generic practices were adapted from an early version of the Baseline Practices Guide (BPG) SPICE's effort. The generic practices are grouped according to common feature and capability level.

The six capability levels are the numbered levels that apply to each process area. The common features are groups of generic practices within a capability level. They were useful in developing the generic practices, and can be useful in understanding the progression of generic practices.

"Process" vs. "process area"

The BPG uses the term "process" where the SE-CMM uses "process area." This change of terminology has been useful to users, who typically need to create or improve a process for their own organizations.

Source

This chapter is reproduced with minor adaptations for the SE-CMM from the SPICE Baseline Practices Guide v1.00a, with the permission of the Baseline Practices Guide technical center manager. The BPG version used was a work in progress.

Adaptations to the BPG

The "Notes" sections of the BPG generic practices were updated to reflect SE-CMM cross-references. In addition, cross-references among generic practices and between generic practices and process areas were added.

In this chapter

Chapter 4A is divided into the six process capability levels shown below:

Topic	See Page
The Not Performed level	4-3
The Performed Informally level	4-4
The Planned and Tracked level	4-5
The Well Defined level	4-9
The Quantitatively Controlled level	4-12
The Continuously Improving level	4-13

Capability Level 0 - Not Performed

Description

The Not Performed level has no common features. There is general failure to perform the base practices in the process area. Where there are work products that result from performing the process, they are not easily identifiable or accessible.

Capability Level 1 - Performed Informally

Description	Base practices of the process area are generally performed. However, the performance of these base practices may not be rigorously planned and tracked. Performance depends on individual knowledge and effort. Individuals within the organization recognize that an action should be performed, and there is general agreement that this action is performed as and when required. There are identifiable work products for the process, which demonstrate the performance of the base practices.
Common Feature 1.1: Base Practices are Performed	**1.1.1 Perform the process.** Perform a process that implements the base practices of the process area to provide work products and/or services to a customer. *Note:* This process may be termed an "informal process." The customer(s) of the process area may be internal or external to the organization.

Capability Level 2 - Planned and Tracked

Description

Base practices of the process area are planned and tracked. Performance according to specified procedures is verified. Work products conform to specified standards and requirements. The organization uses measurement to track process area performance, thus enabling the organization to manage its activities based on actual performance. The primary distinction from the Performed Informally level is that the performance of the process is planned and tracked.

Common Feature 2.1: Planning Performance

2.1.1 Allocate resources. Allocate adequate resources (including people) for performing the process area.

Relationship to process areas: Identification of critical resources is done in PA 12 - Plan Technical Effort.

2.1.2 Assign responsibilities. Assign responsibilities for developing the work products and/or providing the services of the process area.

Relationship to process areas: Assignment of responsibilities is related to PA 12 - Plan Technical Effort.

continued on next page

Common Feature 2.1: Planning Performance, continued

2.1.3 Document the process. Document the approach to performing the process area in standards and/or procedures.

Note: Participation of the people who perform a process (its owners) is essential to creating a usable process description. Processes in an organization or on a project need not correspond one-to-one with the process areas in the SE-CMM. Therefore, a project's process addressing a process area may be described in more than one way (e.g., policies, standards, and/or procedures). Similarly a project's process description may span more than one process area.

Relationship to other generic practices: Process descriptions evolve with increasing process capability. This is the "Level 2" process description. See 3.1.1, 3.1.2, 5.2.3 for descriptions of this practice at higher capability levels.

Relationship to process areas: This practice is related to PA 13 - Define Organization's Systems Engineering Process and PA 14 - Improve Organization's Systems Engineering Processes.

2.1.4 Provide tools. Provide appropriate tools to support performance of the process area.

Relationship to other generic practices: Tool changes may be part of process improvements (see 5.2.3 for practices on process improvements).

Relationship to process areas: Tools are managed in PA 16 - Manage Systems Engineering Support Environment.

2.1.5 Ensure training. Ensure that the individuals performing the process area are appropriately trained in how to perform the process.

Note: Training, and how it is delivered, will change with process capability due to changes in how the process(es) is performed and managed.

Relationship to process areas: Training and training management is addressed in PA 17 - Provide Ongoing Skills and Knowledge.

continued on next page

Common Feature 2.1: Planning Performance, continued

2.1.6 Plan the process. Plan the performance of the process area.

Note: Plans for process areas in the engineering and project categories may be in the form of a project plan, whereas plans for the organization category may be at the organizational level.

Relationship to process areas: Project planning is described in PA 12 - Plan Technical Effort.

Common Feature 2.2: Disciplined Performance

2.2.1 Use plans, standards, and procedures. Use documented plans, standards, and/or procedures in implementing the process area.

Note: A process performed according to its process descriptions is termed a "described process." Process measures should be defined in the standards, procedures, and plans.

Relationship to other generic practices: The standards and procedures used were documented in 2.1.3, and the plans used were documented in 2.1.6. This practice is an evolution of 1.1.1 and evolves to 3.2.1.

Relationship to process areas: Using plans implies applying them in practice as addressed in PA11 - Monitor and Control Technical Effort.

2.2.2 Do configuration management. Place work products of the process area under version control or configuration management, as appropriate.

Note: Where PA 09 - Manage Configurations focuses on the general practices of configuration management, this generic practice is focused on the use of these practices to produce the work products of the individual process area under consideration.

Relationship to process areas: The typical practices needed to support systems engineering in the configuration management discipline are described in PA 09 - Manage Configurations.

continued on next page

Common Feature 2.3: Verifying Performance

2.3.1 Verify process compliance. Verify compliance of the process with applicable standards and/or procedures.

Relationship to other generic practices: The applicable standards and procedures were documented in 2.1.3 and used in 2.2.1.

Relationship to process areas: The quality management and/or assurance process is described in PA 08 - Ensure Quality.

2.3.2 Audit work products. Verify compliance of work products with the applicable standards and/or requirements.

Relationship to other generic practices: The applicable standards and procedures were documented in 2.1.3 and used in 2.2.1.

Relationship to process areas: Product requirements are developed and managed in PA 02 - Derive and Allocate Requirements. Verification and validation is further addressed in PA 07 - Verify and Validate System.

Common Feature 2.4: Tracking Performance

2.4.1 Track with measurement. Track the status of the process area against the plan using measurement.

Note: Building a history of measures, such as cost and schedule variances, is a foundation for managing by data, and is begun here.

Relationship to other generic practices: The use of measurement implies that the measures have been defined and selected in 2.1.3 and 2.1.6, and data have been collected in 2.2.1. This practice evolves to 3.2.3 and 4.2.1.

Relationship to process areas: Project tracking is described in PA 11 - Monitor and Control Technical Effort.

2.4.2 Take Corrective Action. Take corrective action as appropriate when progress varies significantly from that planned.

Note: Progress may vary because estimates were inaccurate, performance was affected by external factors, or the requirements, on which the plan was based, have changed. Corrective action may involve changing the process(es), changing the plan, or both.

Relationship to other generic practices: This practice evolves to 4.2.2.

Relationship to process areas: Project control is described in PA 11 - Monitor and Control Technical Effort.

Capability Level 3 - Well Defined

Description

Base practices are performed according to a well-defined process using approved, tailored versions of standard, documented processes. The primary distinction from the Planned and Tracked level is that the process is planned and managed using an organization-wide standard process.

Common Feature 3.1: Defining a Standard Process

3.1.1 Standardize the process. Document a standard process or family of processes for the organization, that describes how to implement the base practices of the process area.

Note: The critical distinction between generic practices 2.1.3 and 3.1.1, the Level 2 and Level 3 process descriptions, is the *scope of application* of the policies, standards, and procedures. In 2.1.3, the standards and procedures may be in use in only a specific instance of the process, e.g., on a particular project. In 3.1.1, however, policies, standards, and procedures are being established at an organizational level for common use throughout the organization, and are termed the "standard process."

The processes in an organization need not correspond one-to-one with the process areas in this capability maturity model. An organization may create more than one standard process description to address a process area. Similarly, an organization's process may span multiple process areas. The SE-CMM does not dictate the organization or structure of an organization's process descriptions. Therefore, the organization may define more than one standard process to address differences among application domains, customer constraints, etc. These related standard processes are termed a "standard process family." The members of a standard process family are typically similar in their descriptions of how and in what order tasks are done. They typically differ in customer constraints, application domain (technology), etc.

Relationship to other generic practices: The Level 2 process description was documented in 2.1.3. The Level 3 process description is tailored in 3.1.2.

Relationship to process areas: The process for developing a process description is described in PA 13 - Define Organization's Systems Engineering Process.

continued on next page

Common Feature 3.1, Defining a Standard Process, continued

3.1.2 Tailor the standard process. Tailor the organization's standard process family to create a well-defined process that addresses the particular needs of a specific use.

Note: Tailoring the organization's standard process for use by a project or group within the organization creates a project's defined process. The tailoring addresses the particular needs of the project.

Relationship to other generic practices: The organization's standard process (family) is documented in 3.1.1. The tailored process definition is used in 3.2.1.

Relationship to process areas: Tailoring guidelines are defined in PA 13 - Define Organization's Systems Engineering Process.

continued on next page

Common Feature 3.2: Perform the Defined Process

3.2.1 Use a well-defined process. Use a well-defined process in implementing the process area.

Note: A well-defined process is characterized by readiness criteria, inputs, standards and procedures, verification mechanisms (such as defect reviews), outputs, and completion criteria. An organization's standard process (or processes) should be well defined. When the organization's standard process is well defined, a project may create a well-defined process for its use by tailoring the organization's well-defined process to meet project needs.

Relationship to other generic practices: This practice evolved from 2-2-1 and is related to the well-defined process created in 3.1.2.

3.2.2 Perform defect reviews. Perform defect reviews of appropriate work products of the process area.

Note: In SPICE's BPG and the CMM for Software, defect reviews are called peer reviews. The purpose of defect reviews is to use an inspection technique to identify sources of error in early or interim systems engineering work products. The inspection techniques described in the software literature are readily adaptable to other development aspects, such as systems engineering.

3.2.3 Use well-defined data. Use data from performing the defined process to manage it.

Note: An example would be classification of defects by the program phase (e.g., requirement, design) in which they were introduced, detected, and corrected.

In a well-defined process, measurements are tailored from those described by the organization's standard process. Other measures may be added, for example, as pilots for improvements. Data collection, analysis, and reporting are planned, and benefit both control and improvement activities. Data are used, as in 2.4.1 and 2.4.2, to track and initiate corrective action when deviations from planned performance are significant. Data are also used as a basis for identifying and prioritizing improvement opportunities. In addition, data should be collected on experiments or pilots of new or improved process elements, so as to understand their success or failure.

Relationship to other generic practices: This is an evolution of 2.4.1 and 2.4.2; corrective action taken here is based on a well-defined process, which has objective criteria for determining progress (see 3.2.1). In addition, all of Level 4 builds on this practice.

Capability Level 4 - Quantitatively Controlled

Description

Detailed measures of performance are collected and analyzed. This leads to a quantitative understanding of process capability and an improved ability to predict performance. Performance is objectively managed, and the quality of work products is quantitatively known. The primary distinction from the Well Defined level is that the defined process is quantitatively understood and controlled.

Common Feature 4.1: Establishing Measurable Quality Goals

4.1.1 Establish quality goals. Establish measurable quality goals for the work products of the organization's standard process family.

Note: These quality goals can be tied to the strategic quality goals of the organization, the particular needs and priorities of the customer, or the tactical needs of the project. The measurements referred to here go beyond the traditional end-product measurements. They are intended to imply sufficient understanding of the processes being used to enable the organization to set and use intermediate goals for work-product quality.

Relationship to other generic practices: Data gathered via defect reviews (3.2.2) can be particularly important in setting goals for work-product quality.

Common Feature 4.2: Objectively Managing Performance

4.2.1 Determine process capability. Determine the process capability of the defined process quantitatively.

Note: This is a quantitative process capability based on a well-defined (3.1.1) and measured process. Measurements are inherent in the process definition and are collected as the process is being performed.

Relationship to other generic practices: The defined process is established through tailoring in 3.1.2 and is performed in 3.2.1. Data are collected in 3.2.3 and used here.

4.2.2 Use process capability. Take corrective action as appropriate when the process is not performing within its process capability.

Note: Special causes of variation, identified based on an understanding of process capability, are used to understand when and what kind of corrective action is appropriate.

Relationship to other generic practices: This practice is an evolution of 3.2.3, with the addition of quantitative process capability to the defined process.

Relationship to process areas: When an out-of-control condition is defected, PA11 - Monitor and Control Technical Effort is involved.

Capability Level 5 - Continuously Improving

Description

The organization establishes quantitative performance goals (targets) for process effectiveness and efficiency, based on its business goals. The organization is able to continuously improve its process by gathering quantitative data from performing the defined processes and from piloting innovative ideas and technologies. The primary distinction from the Quantitatively Controlled level is that the defined process and the standard process undergo continuous refinement and improvement, based on a quantitative understanding of the impact of changes to these processes.

Common Feature 5.1: Improving Organizational Capability

5.1.1 Establish process effectiveness goals. Establish quantitative goals for improving process effectiveness of the standard process family, based on the business goals of the organization and the current process capability.

5.1.2 Continuously improve the standard process. Continuously improve the process by changing the organization's standard process family to increase its effectiveness.

Note: The information learned from managing individual projects is communicated back to the organization for analysis and deployment to other applicable areas. Changes to the organization's standard process family may come from innovations in technology or incremental improvements. Innovative improvements will usually be externally driven by new technologies. Incremental improvements will usually be internally driven by improvements made in tailoring for the defined process. The goal of improving the standard process is to reduce common causes of variation in the process.

Relationship to other generic practices: Special causes of variation are controlled in 4.2.2.

Relationship to process areas: Organizational process improvement is described in PA 14 - Improve Organization’s Systems Engineering Processes.

continued on next page

Common Feature 5.2: Improving Process Effectiveness

5.2.1 Perform causal analysis. Perform causal analysis of defects.

Note: Those who perform the process are typically participants in this analysis. This is a proactive causal analysis activity as well as reactive; defects from prior projects of similar attributes can be used to target improvement areas for the new effort.

Relationship to other generic practices: Results of these analyses are used in 5.2.2 and 5.2.3.

5.2.2 Eliminate defect causes. Eliminate the causes of defects in the defined process selectively.

Note: Defect causes are selectively eliminated because it may be impractical to perform causal analysis (5.2.1) *on all defects*. In this case, some screening may be used.

Relationship to other generic practices: Causes were identified in 5.2.1.

5.2.3 Continuously improve the defined process. Continuously improve process performance by changing the defined process to increase its effectiveness.

Note: The improvements may be based on incremental improvements (5.1.2) or innovative improvements such as new technologies (perhaps as part of pilot testing). Improvements will typically be driven by the goals established in 5.1.1.

Relationship to other generic practices: Practice 5.2.2 may be one source of improvements. Goals were established in 5.1.1.

Relationship to process areas: Improvements are addressed in PA 15 - Manage Product Line Evolution and PA14 - Improve Organization's Systems Engineering Processes.

Chapter 4B: Process Areas/Base Practices

In this chapter

This chapter contains the base practices, that is, the practices considered essential to the conduct of basic systems engineering.

Topic	See Page
Process Area (PA) Format	4-16
PA 01: Analyze Candidate Solutions	4-18
PA 02: Derive and Allocate Requirements	4-23
PA 03: Evolve System Architecture	4-34
PA 04: Integrate Disciplines	4-42
PA 05: Integrate System	4-47
PA 06: Understand Customer Needs and Expectations	4-53
PA 07: Verify and Validate System	4-59
PA 08: Ensure Quality	4-66
PA 09: Manage Configurations	4-72
PA 10: Manage Risk	4-77
PA 11: Monitor and Control Technical Effort	4-82
PA 12: Plan Technical Effort	4-86
PA 13: Define Organization's Systems Engineering Process	4-95
PA 14: Improve Organization's Systems Engineering Processes	4-100
PA 15: Manage Product Line Evolution	4-104
PA 16: Manage Systems Engineering Support Environment	4-108
PA 17: Provide Ongoing Skills and Knowledge	4-113
PA 18: Coordinate with Suppliers	4-120

Process Area Format

Current contents

At present, the SE-CMM domain aspect consists of 18 process areas (PAs), each of which contains a number of base practices. Each process area is described in the following subsections.

The general format of the process areas is shown in Figure 4-1. The summary description contains a brief overview of the purpose of the PA. Each PA is decomposed into a set of base practices (BPs). The BPs are considered mandatory items, (i.e., they must be successfully implemented to accomplish the purpose of the process area they support). Each base practice is described in detail following the PA summary.

Although the PAs are identified and discussed separately, they do not exist in a vacuum. Even the PAs in the engineering category (PA-01 through PA-07), are inextricably intertwined with all the others in the creation of good systems engineering processes, the implementation of which produces sound, customer-pleasing products.

continued on next page

Figure

The following figure provides the general format of the process areas and describes the content of each part.

PA #: PA Title

Summary Description	The purpose of <PA Title> is. . .<description of the purpose of the PA and summary of its major points>
Process Area Notes	<PA notes paragraphs>
Base Practices List	The following list contains the titles of the base practices that are essential elements of good systems engineering: • BP #: BP Title .....
	end of PA Summary Section
BP # **BP Title**	**<BP text: imperative, verb-object statement that describes an essential element for attaining the purpose of the PA>** *BP Description* <BP description text: provides elaborations of the base practice text.> *Typical Work Products* <List of work products> *BP Notes* <BP notes text: contains conceptual examples, potential techniques, methods, etc.; content of these will vary from base practice to base practice.>
	end of Process Area <PA Title>

Figure 4-1. Process Area Format

PA 01: Analyze Candidate Solutions

Summary description

The purpose of Analyze Candidate Solutions is to perform studies and analyses that will result in the selection of a solution to meet the identified problem and its defined constraints. Analyze Candidate Solutions involves defining the approach and evaluation criteria for the analysis, as well as for choosing, selecting, and studying the candidate solutions. It also involves communicating the rationale and results of the analysis.

Process area notes

Analyze Candidate Solutions may be invoked from any of the other process areas. This process area (PA) identifies the characteristics of a process for choosing a solution from several alternatives.

Candidate solutions may be provided by the invoking PA, but additional solutions may be generated in this PA when needed to further the analysis.

Analyze Candidate Solutions should be invoked throughout the life of a project. It may be used for the following types of decisions, among others

- design decision
- production decisions
- life-cycle cost decisions
- human factors decisions
- risk reduction decisions

Base practices list

The following list contains base practices that are essential elements of good systems engineering:

BP.01.01	Establish evaluation criteria based on the identified problem and its defined constraints.
BP.01.02	Define the general approach for the analysis, based on the established evaluation criteria.
BP.01.03	Identify alternatives for evaluation in addition to those provided with the problem statement.
BP.01.04	Analyze the competing candidate solutions against the established evaluation criteria.
BP.01.05	Select the solution that satisfies the established evaluation criteria.
BP.01.06	Capture the disposition of each alternative under consideration and the rationale for the disposition.

continued on next page

BP 01.01 Establish Evaluation Criteria

Establish evaluation criteria based on the identified problem and its defined constraints.

Description
The criteria used in the evaluation process may vary considerably, depending on the stated problem and the level and complexity of the analysis. The criteria are weighted or ranked in order of importance. For more complex analyses, there may be levels of criteria.

Typical Work Products
- captured evaluation criteria
- trade-study criteria
- defect data-related criteria

Notes
At the system level, parameters of primary importance include system performance, cost effectiveness, producibility, logistics, risk, and operational availability and maintainability.

BP 01.02 Define Analysis Approach

Define the general approach for the analysis, based on the established evaluation criteria.

Description
The general approach, resources, and procedures for performing the analysis should be defined based on the evaluation criteria, personnel, tools, facilities, special equipment, and related resources. The general approach for the analysis should be defined and documented to ensure that the procedures can be consistently repeated.

Typical Work Products
- trade-study approach
- problem solving process

Notes
Some example approaches that could be used to analyze candidate solutions are

- prototyping
- simulation
- modeling
- trade study
- decision tree
- literature search
- exploitation of prior analyses
- elicitation of expert judgment
- process quality improvement team

continued on next page

BP 01.03 Identify Additional Alternatives

Identify alternatives for evaluation in addition to those provided with the problem statement.

Description
Candidate solutions may be furnished with the need for analysis. As the analysis proceeds, other alternatives may be added to the list of candidate solutions.

Typical Work Products
- trade-study alternatives
- decision tree

Notes
Some requests for analysis may be made without supplying any candidate solutions; in these cases, the subject matter experts would need to identify all of the alternative candidate solutions.

On the other hand, some requests for analysis may be made that already supply every possible candidate solution. In that case, this practice would not be applicable.

BP 01.04 Analyze Candidate Solutions

Analyze the competing candidate solutions against the established evaluation criteria.

Description
Analyses should be defined, conducted, and documented at the various levels of functional or physical detail to support the decision needs of the systems engineering process. The level of detail of a study should be commensurate with cost, schedule, performance, and risk impacts.

Typical Work Products
- analyses of candidate solutions

Notes
An example activity: Perform a sensitivity analysis on candidate solutions to determine if small variations in parameters will affect the outcome.

continued on next page

BP 01.05 Select Solution

Select the solution that satisfies the established evaluation criteria.

Description
Zero, one, or several solutions may be found that satisfy the evaluation criteria. The objective is to arrive at a decision where the selected approach is preferred among the alternatives, based on the evaluation criteria.

Typical Work Products
- trade study
- rationale for preferred solution
- description of the preferred solution

Notes
The following questions will usually arise when selecting among alternative solutions:
- How much better is the selected approach than the next best alternative?
- Is there a significant difference between the results of the comparative evaluation?
- Have all feasible alternatives been considered?
- What are the areas of risk and uncertainty?

continued on next page

BP 01.06 Capture Results

Capture the disposition of each alternative under consideration and the rationale for the disposition.

Description
The results from all system analysis activities should be captured and maintained in a decision database. The disposition of each alternative under consideration and the rationale for the disposition should also be documented in the decision database.

Typical Work Products
- evaluation of alternatives for the trade study
- mathematical models of appropriate solutions
- reports of prototype operation
- results of tradeoff studies
- other supporting data of all studies

Notes
Examples of ways to capture results include
- formal, deliverable documentation
- informal, internal documentation
- computer files
- a prototyped product
- an engineering log book
- change request database

End of PA 01: Analyze Candidate Solutions

PA 02: Derive and Allocate Requirements

Summary description

The purpose of Derive and Allocate Requirements is to analyze the system and other requirements and derive a more detailed and precise set of requirements. These derived requirements are allocated to system functions', objects', people', and supporting processes, products, and services, which can be used to synthesize solutions. This process area addresses both the analysis of system-level requirements and the allocation of system-level or derived requirements to lower level functions or objects. This involves addressing the concept of operations, functional partitioning, object identification, and performance allocation, as well as capturing the status and traceability of requirements. The derived and allocated requirements will evolve as the systems requirements evolve over time. When corrective actions have an impact on requirements, it may be necessary to revise the derived and allocated requirements.

Process area notes

The practices in the Derive and Allocate Requirements process area operate in parallel with the practices in the Evolve System Architecture process area (PA 03). Potential derived requirements are evaluated for feasibility against the functional partitions or objects, and are evaluated iteratively against the components of the architecture. It is important to note that the terms "function" and "functional" do not preclude object-oriented methods. Objects perform functions, and functions may be performed by objects. When conflicts or issues are identified with customer or derived requirements (e.g., requirements are not verifiable per the verification and validation practices), the issues may be referred to the practices of the process areas Understand Customer Needs and Expectations (PA06) or Analyze Candidate Solutions (PA01).

continued on next page

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.02.01	Develop a detailed operational concept of the interaction of the system, the user, and the environment, that satisfies the operational need.
BP.02.02	Identify key requirements that have a strong influence on cost, schedule, functionality, risk, or performance.
BP.02.03	Partition requirements into groups based on established criteria (such as similar functionality, performance, or coupling) to facilitate and focus the requirements analysis.
BP.02.04	Derive, from the system and other (e.g., environmental) requirements, requirements that may be logically inferred and implied as essential to system effectiveness.
BP.02.05	Identify the requirements associated with external interfaces to the system and interfaces between functional partitions or objects.
BP.02.06	Allocate requirements to functional partitions, objects, people, or support elements to support synthesis of solutions.
BP.02.07	Analyze requirements to ensure that they are verifiable by the methods available to the development effort.
BP.02.08	Maintain requirements traceability to ensure that lower level (derived) requirements are necessary and sufficient to meet the objectives of higher level requirements.
BP.02.09	Capture system and other requirements, derived requirements, derivation rationale, allocations, traceability, and requirements status.

continued on next page

BP 02.01 Develop Detailed Operational Concept

Develop a detailed operational concept of the interaction of the system, the user, and the environment, that satisfies the operational need.

Description

This practice adds detail to the operational concept used to develop system requirements. The operational concept includes scenarios and timelines of the system's responses to stimuli. The behavior of the system and its elements is organized by states, modes, and time sequences. The behavior is flowed down to subsystem elements as required to fully discover the derived and allocated requirements for each system element. The operational behavior of the system and subsystem includes the behavior required to meet the customer's operational need and any exceptional behavior that may be caused by the environment or system faults.

Typical Work Products

- operational concept
- user interaction sequences
- maintenance operational sequences
- timelines
- simulations
- usability analysis

Notes

Examples of activities to develop a detailed operational concept include

- Develop a prototype of the user interface and capture vignettes of user interaction.
- Develop a system simulation.

Development and analysis of operational concepts are valuable tools used in the practices of the process areas Understand Customer Needs and Expectations (PA06) and Derive and Allocate Requirements (PA02). They help the analyst to discover new requirements and to verify and validate existing or potential requirements. Operational concepts, simulations, and prototypes are key to user-centered development and maintenance processes.

continued on next page

BP 02.02 Identify Key Requirement Issues

Identify key requirements that have a strong influence on cost, schedule, functionality, risk, or performance.

Description

In analyzing system and derived requirements, requirements are often identified that have an especially strong influence on the cost, development schedule, risk, or performance of a product. The total set of requirements is screened for potential key requirements. A cost-benefit analysis is then performed on these requirements using the process areas Analyze Candidate Solutions (PA01) and Evolve System Architecture (PA03). The results of analyzing key requirements may be reviewed with the customer using the methods of the Understand Customer Needs and Expectations process area (PA06). Key requirements that show a relatively low benefit-to-cost ratio, high risk, or long development schedule are candidates for negotiation with the customer. Key requirements are a primary input to the activities of the Manage Risk process area (PA10).

Typical Work Products

- key requirements issues
- benefit-to-cost sensitivity analyses for key requirements

Notes

An example activity: Identify performance requirements that are near the limits of what has been achieved before (near the state of the art).

continued on next page

BP 02.03 Partition Functions

Partition requirements into groups based on established criteria (such as similar functionality, performance, or coupling) to facilitate and focus the requirements analysis.

Description
Requirements are evaluated for similarity in function and grouped into appropriate partitions. Criteria for appropriate functional partitions are established and may include, in addition to similarity, high coupling within functional partition and low coupling between functional partitions. Functional partitions are chosen so that overall performance requirements can be budgeted to the functions.

Typical Work Products
- identified functional partitions
- functional performance budgets

Notes
Examples of activities to partition requirements include
- Group all requirements that apply to user interaction.
- Group all requirements that apply to data storage and retrieval.
- Use affinity diagrams.

Functional partitions include functions and subfunctions whose requirements are ultimately allocated to physical architecture elements.

continued on next page

BP 02.04 Derive Requirements

Derive, from the system and other (e.g., environmental) requirements, requirements that may be logically inferred and implied as essential to system effectiveness.

Description
Derived requirements are those requirements that are explicitly identified or discovered as necessary implications of stated system and other top-level requirements. A system requirement's derived requirements "represent" the system requirement in terms of development constraints and verification. Typically, a system requirement may have to be decomposed into one or more derived requirements in order to allocate responsibility and to provide for feasible verification. Derived requirements apply to all aspects of the developed system, including the development, production, environmental, and operational parameters. Derived requirements may result from a single higher level requirement or partitions of higher level requirements.

Typical Work Products
- derived operational requirements assigned to a functional partition
- derived performance requirements

Notes
Examples include
- Assess system requirements for derived requirements relating to the operational environment.
- Produce derived requirements necessary to render system requirements testable.
- Produce derived requirements necessary to allocate system timing budgets to functional partitions.
- Produce rationale for derived requirements.

Derived functional and performance requirements are allocated directly, or as appropriate, to functional partitions, derived requirements, and ultimately to physical architecture elements.

continued on next page

BP 02.05 Develop Interface Requirements

Identify the requirements associated with external interfaces to the system and interfaces between functional partitions or objects.

Description

External and internal interfaces are identified throughout the analysis of system requirements. Identification of these interfaces is essential to the development of a complete set of requirements for the physical architecture. The early and complete definition of external interfaces is especially important in characterizing the overall functionality of the system because the interfaces are typically independent of the internal architecture. Also, external interfaces may be a driver of the internal architecture and functionality. This is especially true of the user interface. The internal interfaces and their related derived requirements are identified in conjunction with the functional or object partitioning. After partitions are identified, their interfaces and logical data flows are defined.

Typical Work Products

- interface requirements

Notes

Examples include

- Identify the input and output data for each user interface function.
- Identify the input and output data of all external systems that must interface to the subject system.
- Identify the physical requirements of all external system interfaces.
- Identify need for physical mounting requirements
- Identify operator stimuli and control points.
- Identify signal and control structures.
- Identify interfaces to the environment.

External stimuli identified in Develop Detailed Operational Concept (BP02.01) are candidates for external interfaces. The identification of external interfaces is facilitated by the development and understanding of the detailed operational concept. In addition, the identification of external interfaces forms the basis for derived external interface requirements, as well as many derived functional and performance requirements. Interfaces are captured and controlled according to the practices of the Integrate System process area (PA05).

continued on next page

BP 02.06 Allocate Requirements

Allocate requirements to functional partitions, objects, people, or support elements to support synthesis of solutions.

Description
The purpose of this practice is to facilitate development of the functional architecture at successively lower partitions. Requirements are initially allocated to functional partitions (which may include functions or objects, and subfunctions) and ultimately to system elements and components. The allocations are performed so that the derived requirements can be implemented to satisfy the higher level requirements. Where it appears that a requirement is to be satisfied jointly by several system elements, it is necessary to derive separate requirements for each system element.

Alternatives should be considered regarding the allocation of requirements to people versus systems. Support elements (including processes, production, maintenance, and environmental constraints) should be evaluated for allocation of derived requirements.

Typical Work Products
- derived requirements
- attributes of allocated requirements

Notes
Examples include
- Identify the requirements and derived requirements that apply to all functions or objects and allocate these requirements to all elements.
- Identify requirements and derived requirements that constitute a performance partition and uniquely allocate these requirements to the appropriate function or object.

Allocations of functional and performance requirements facilitate the division of responsibilities for development and testing. The practices of the process areas Understand Customer Needs and Expectations (PA06), Derive and Allocate Requirements (PA02), and Evolve System Architecture (PA03) iterate the allocation of requirements.

continued on next page

BP 02.07 Ensure Requirement Verifiability

Analyze requirements to ensure that they are verifiable by the methods available to the development effort.

Description
The method and feasibility of verifying requirements is established early in the development cycle. It is essential for a system or derived requirement to have characteristics that can be verified to prove that the resulting product meets the intended purpose. Evaluating the feasibility of verifying a potential requirement facilitates the production of good requirements. Throughout the life cycle, requirements are continually assessed to ensure the feasibility of verification, especially in connection with evaluating changes to requirements. Methods of verification include inspection, test, demonstration, and analysis.

Typical Work Products
- verifiability status of requirements
- captured verification method

Notes
An example activity: Assess the feasibility of verifying each requirement.

It is important to ensure that requirements verification is performed iteratively and recursively with the practices of the Verify and Validate System process area (PA07).

continued on next page

BP 02.08 Maintain Requirement Sufficiency and Traceability

Maintain requirements traceability to ensure that lower level (derived) requirements are necessary and sufficient to meet the objectives of higher level requirements.

Description

This practice captures, maintains, and controls the traceability and status of requirements throughout the product life cycle. Of particular importance is the relationship between higher level requirements and their associated derived requirements, which in effect represent the higher level requirement. This dependence of derived requirements on other requirements or design features is referred to as traceability and is recorded and maintained from the highest level (most general) to the lowest level (most specific) as the requirements and design evolve. A continuous assessment of the lower level requirements and the validity of their traceability is conducted to ensure that the developed system or product meets all the requirements, but does not have features beyond what is necessary to meet the requirements.

Typical Work Products

- requirement exception report
- requirement traceability tables
- requirements databases
- traceability exception report

Notes

Examples include

- Perform analyses to ensure that related sets of derived requirements, taken as a whole, meet the intent of the parent requirement.
- Perform analyses to ensure that there are no unnecessary requirements.
- Verify requirements traceability.

All practices involving creating, changing, or verifying of requirements (especially those of the process areas Understand Customer Needs and Expectations (PA06), Derive and Allocate Requirements (PA02), Evolve System Architecture (PA03), and Verify and Validate System (PA07)) must maintain requirements traceability.

continued on next page

BP 02.09 Capture Results and Rationale

Capture system and other requirements, derived requirements, derivation rationale, allocations, traceability, and requirements status.

Description
This base practice forms the basis for systematically developing and verifying a system that meets the customer's operational and performance expectations within acceptable constraints of cost and schedule. Captured results also include other attributes of requirements such as a unique requirement number, interpretation, test method, issues, and acceptance/change status.

Typical Work Products
- requirement document
- requirements databases
- interface requirements document
- functional architecture
- requirement allocation sheet

Notes
Examples of activities for capturing results and rationale include
- Enter requirements, their traceability, allocation, and status into a requirements database.
- Distribute, review and coordinate requirements data with the development team.

The collection of work products from this process area is sometimes called the functional architecture.

The capture of results and rationale applies to all the practices associated with the derivation and allocation of requirements as well as the analysis of candidate solutions and design decisions.

End of PA 02: Derive and Allocate Requirements

PA 03: Evolve System Architecture

Summary description

The purpose of Evolve System Architecture is to provide a basis for establishing and evolving a system design. It involves deriving the architecture requirements, identifying key design issues, determining the functional and physical structure and interfaces, and allocating the architecture requirements to system elements. The practices described herein are expected to be performed iteratively with other systems engineering practices until the architecture is handed off to the implementing or component engineering disciplines.

System architecture comprises functional (or logical), physical (tangible), and foundation architectures. Evolve System Architecture activities are applicable to all life-cycle phases of a product and may be initiated either by new development, changes in requirements, or corrective actions.

Process area notes

This process area generates candidate solutions and then makes use of the Analyze Candidate Solutions process area (PA01) to choose an alternative that meets established criteria for the system architecture. This process area is performed iteratively with the process areas Understand Customer Needs and Expectations (PA06) and Derive and Allocate Requirements (PA02).

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.03.01 Derive the requirements for the system architecture.

BP.03.02 Identify the key design issues that must be resolved to support successful development of the system.

BP.03.03 Generate alternative(s) and constraints for the architecture and select a solution in accordance with the Analyze Candidate Solutions process area (PA01).

BP.03.04 Develop the interface requirements for the selected architecture components.

BP.03.05 Allocate the system and derived requirements to the chosen architecture components and interfaces.

BP.03.06 Maintain requirement traceability for the architecture's requirements to ensure that lower level (derived) requirements are necessary and sufficient to meet the needs of higher level requirements or design.

BP.03.07 Describe the system architecture by capturing the design results and rationale.

BP.03.08 Identify appropriate derived requirements that address the effectiveness and cost of life-cycle phases following development, such as production and operation.

continued on next page

BP 03.01 Derive System Architecture Requirements

Derive the requirements for the system architecture.

Description

This activity makes use of and iterates with a number of other activities, including development and evolution of system requirements, integration, and verification. Derived requirements may include requirements taken directly from the system requirements, as well as requirements that are inferred from the system requirements, either directly or as constrained by the current architectures. Types of derived requirements include performance, human interaction, production, maintenance, etc. Derived requirements may apply broadly or they may apply only to specific subsystems or support elements. These requirements provide a basis for the selection criteria used when analyzing architecture alternatives.

Typical Work Products

- derived maintenance requirements
- derived human interface requirements

Notes

Derived requirements for the system's architecture apply to the actual (tangible) subsystems, configuration items, or components and to the functional or notional architecture.

continued on next page

BP 03.02 Identify Key Design Issues

Identify the key design issues that must be resolved to support successful development of the system.

Description
The design activity must begin with an awareness of the many issues facing the system development. An evaluation must take place to determine what subset of the many issues are the design drivers for the system. This subset of key design issues then becomes a constraint on the system design and development. This activity also identifies analyses and trade studies which need to be performed in order to select appropriate architecture and design alternatives.

Typical Work Products
- list of key design issues
- analyses to be performed
- trade studies to be performed

Notes
Key design issues may include cost drivers, performance drivers, risk, or technology. In an integrated product development team environment, key design issues may identify the need for "specialty engineers" to be a part of the design team. There may be issues seemingly unrelated to the system that become key design issues. An example of such an issue is compliance with governmental laws governing the manufacturing or disposal of a product.

continued on next page

BP 03.03 Develop Architectural Structure

Generate alternative(s) and constraints for the architecture and select an architectural solution in accordance with the practices of the Analyze Candidate Solutions process area (PA01).

Description
A structure for the system architecture is developed that satisfies the system requirements, derived requirements, and architecture requirements. The system's architectural structure includes subsystems, configuration items, or components, as well as their interrelationships, which are to be developed to meet the requirements.

Typical Work Products
- architecture structure
- subsystems
- major assemblies
- identified interfaces
- engineering drawings

Notes
The identified elements of the system's architectural structure constitute the major "pieces" of the system to be developed, upgraded, maintained, or integrated. For new development, these elements are optimally selected through the analysis of alternatives against established requirements or criteria. In the case of reuse or upgrades of existing systems, an existing architectural structure or its elements may be a requirement.

continued on next page

BP 03.04 Develop Architecture Interface Requirements

Develop the interface requirements for the selected architecture components.

Description
External and internal interfaces are identified to develop a complete set of architecture requirements. Alternative solutions are developed and a solution is selected in accordance with the practices of the Analyze Candidate Solutions process area (PA01).

Typical Work Products
- interface requirements
- user interface requirements
- environmental interface requirements
- subsystem interface requirements

Notes
The system architecture's interface requirements can be broadly classified as those interface requirements between system elements and entities external to the system, and those among elements of the selected architecture. Generally, all or part of the external interface requirements may be known before selecting the system architecture. Internal interface requirements are typically deferred until after the architectural structure is selected.

BP 03.05 Allocate Architecture Requirements

Allocate the system and derived requirements to the chosen architecture components and interfaces.

Description
Derived requirements, functions, or objects are allocated to system elements, as well as interfaces. Performance of the design is analyzed, and the system architecture is refined and modified as necessary.

Typical Work Products
- allocated requirements
- requirements traceability data

Notes
Examples of activities for allocating architecture requirements include
- Identify the requirements and derived requirements that apply to all system elements and allocate these requirements to all elements.
- Identify requirements and derived requirements that constitute a performance partition and allocate these requirements to the appropriate system element.

continued on next page

BP 03.06 Maintain Requirement Traceability

Maintain requirement traceability for the architecture's requirements to ensure that lower level (derived) requirements are necessary and sufficient to meet the needs of higher level requirements or design.

Description
In this practice, requirement traceability and status is captured, maintained, and controlled throughout the product life cycle. Derived requirements levied on the architecture must result from, and trace to, higher level system requirements, functional requirements derived from the higher level requirements, or higher level design decisions. This traceability is recorded and maintained from the highest level (most general) to the lowest level (most specific) as the requirements and design evolve. The lower level requirements and the validity of their traceability are assessed periodically to ensure that the developed system or product meets all the requirements, but does not have features beyond what is necessary to meet the requirements.

Typical Work Products
- requirement traceability tables
- requirement exception report
- traceability exception report
- requirement database

Notes
The complete requirements traceability relationships include all requirements levied on the system and its parts as the solution evolves. Thus, requirements derived from a valid functional analysis and the more detailed requirements derived for the architecture are captured in the same set of traceability data.

Examples of activities to maintain requirement traceability include
- Perform analysis to ensure that related sets of derived requirements, taken as a whole, meet the intent of the parent requirement.
- Perform analysis to ensure there are no unnecessary requirements.

continued on next page

BP 03.07 Capture Results and Rationale

Describe the system architecture by capturing the design results and rationale.

Description
The system architecture includes the functional and physical (tangible) architecture elements, their relationships, interfaces, allocated derived requirements, requirements traceability, and the rationale supporting the selected solution. The rationale for the design and architectural decisions draws heavily on the results of analyzing alternatives against established criteria and requirements. In developing the system, it is essential to capture, baseline, and disseminate the architecture description to verify that the system meets the customers' operational and performance expectations.

Typical Work Products
- physical architecture
- interface requirements
- requirement allocations
- design documents
- requirements traceability table

Notes
Examples of ways to capture the design results and rationale include
- design document
- specification
- interface control drawing
- engineering notebook entries
- block diagrams
- data flow or control flow diagrams

continued on next page

BP 03.08 Identify Requirements Related to Production and Operations

Identify appropriate derived requirements that address the effectiveness and cost of life-cycle phases following development, such as production and operation.

Description

This practice derives those requirements necessary to ensure that the developed product can be produced economically, operated reliably, and maintained cost effectively. A producibility analysis, as described in the Manage Risk process area (PA10), is performed to identify any critical or production engineering requirements that constrain the design. Requirements and constraints are also derived from the operational concept and system mission to ensure that the customer needs are met by providing for reliable and cost-effective operation and maintenance. These new requirements are included in the applicable requirements documentation.

Typical Work Products

- producibility related design constraints
- reliability goals for program phases
- quantified maintainability requirements
- operation-related derived requirements

Notes

Examples of requirements related to production and operations include

- mechanical or electrical design-related requirements to ensure systems can be manufactured efficiently at low risk
- quantified maintainability requirements that are necessary to allocate to components
- derived requirements by program phases that are necessary to meet the system mission and to allocate to components
- operational requirements that address educational and skill levels of system operators/users

End of PA 03: Evolve System Architecture

PA 04: Integrate Disciplines

Summary description

The purpose of Integrate Disciplines is to identify those disciplines necessary for effective system development and create an environment in which they jointly and effectively work together toward a common agenda. Each discipline's unique expertise and concerns are brought forward and considered, but the focus on total system development is maintained. These disciplines may include, but are not limited to, problem domain, marketing, manufacturing, component design, development, reliability, maintainability, operations, quality, supportability, human factors, logistics, safety, and security. It is critical to be able to meld such disciplines without sacrificing their parochial interests concerning issues important to and unique to each discipline. This cooperative environment must persist throughout the system life cycle.

Process area notes

It is essential to sustain a focus on the human interaction activities and issues related to cooperative group dynamics during the development, synthesis, and integration efforts. In many cases, the systems engineer role, in this environment, is to function as an "information broker," coordinating and distributing information through the project/operations staff. The goal is to eliminate nonessential information while providing essential information to members of the development staff, on a timely basis.

The practices of concurrent engineering, interdisciplinary teams, or integrated product development may meet the requirements of the process area, if they include the base practices described below.

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.04.01	Involve the disciplines that are essential to system development in a timely manner.
BP.04.02	Promote cross-discipline understanding among the developers.
BP.04.03	Establish methods for interdisciplinary coordination.
BP.04.04	Establish and use methods for identifying and resolving interdisciplinary issues, and creating integrated solutions.
BP.04.05	Communicate results of interdisciplinary activities to affected groups.
BP.04.06	Develop project goals and ensure that all affected groups and individuals are fully aware of them.

continued on next page

BP 04.01 Involve Essential Disciplines

Involve the disciplines that are essential to system development in a timely manner.

Description
Efficient and effective systems result from a blending of the efforts of people from many disciplines. These people should be identified and involved in the processes that affect them, in time for effective collaboration.

Typical Work Products
- roster of essential disciplines
- list of representatives from each discipline
- agendas and schedules of collaboration activities

Notes
As the development effort proceeds through its life cycle, the number of critical disciplines varies. The initial focus should be on attaining complete coverage, not limiting participants. The systems engineer must be cognizant enough of the concerns of all disciplines so that he or she can recall specialists when needed throughout the product life cycle.

continued on next page

BP 04.02 Promote Cross-Discipline Understand-ing

Promote cross-discipline understanding among the developers.

Description
Developers need to become familiar with the issues that are important to those disciplines essential to the use and development of the system and the effect that each discipline has on the quality of the product. The systems engineer is a natural avenue to provide an overview of the primary focus of and the issues of concern to each discipline involved with the product.

Typical Work Products
- pamphlets describing each discipline
- briefings to familiarize the developers with lessons learned

Notes
This is often one of the most overlooked areas in the list of systems engineering tasks; yet it often produces the highest return on investment in terms of cost-effective solutions to development problems. Understanding the other individuals' concerns is the first step to achieving a cooperative, harmonious work environment, so it is difficult to focus too much effort in this area. However, the objective is not to create a group who are experts in all the disciplines; rather, it is to create a group of individuals who are aware of each others' technical concerns and understand how proper consideration of each concern has a positive impact on the quality of the group's product.

To illustrate that consideration of the specialty disciplines is key to product success, it may help to show the time-critical nature of some of the decisions made early in the development life cycle and how they can produce positive or negative customer impressions when the system is introduced to its intended environment.

Example activities include
- holding a meeting at the inception of the project/program at which representatives of the identified disciplines share their issues
- summarizing the issues of each discipline in a one- or two-page paper
- distributing this paper to all

continued on next page

BP 04.03 Establish Coordination Methods

Establish methods for interdisciplinary coordination.

Description
In addition to understanding the roles and what information to share, the product developers must know how to share knowledge, i.e., the particular methods of getting information from an individual or group to others who need it. In addition, they must recognize that specialties may have their own process that should be integrated with systems engineering.

Typical Work Products
- methods for coordinating integrated development

Notes
Knowledge sharing may center around an automation strategy, in which case individuals would share knowledge through the automation tool suite.

Alternatively, knowledge sharing may center around a teaming strategy, in which case individuals would share knowledge in accordance with the particular teaming structures used.

BP 04.04 Establish Resolution Methods

Establish and use methods for identifying and resolving interdisciplinary issues, and creating integrated solutions.

Description
Issues will arise between the disciplines during product development. Thus, product developers must have available several predetermined techniques for resolving these issues. The technique used would depend on several factors, including the time available to come to resolution, the severity of the issue, and the related consequences of the issue.

Typical Work Products
- issue resolution methods
- integrated solutions

Notes
Examples of methods for resolving interdisciplinary issues include
- Pugh's Controlled Convergence technique
- Quality Function Deployment technique
- autocratic ediction
- arbitration and rules

continued on next page

BP 04.05 Communicate Results

Communicate results of interdisciplinary activities to affected groups.

Description
The results of interdisciplinary activities will include alternatives considered, the decisions made, and the rationale for the decisions. These results must be communicated promptly to affected groups and individuals.

Typical Work Products
- results of interdisciplinary activities
- meeting minutes
- decision database

Notes
Examples of methods to communicate results include
- electronic mail decisions with rationale
- use of the project's selected automation tool set

BP 04.06 Develop and Communicate Project Goals

Develop project goals and ensure that all affected groups and individuals are fully aware of them.

Description
For the product development to proceed with reasonable smoothness, each project member and the direct support staff must know and work toward the same goals. These goals must be clearly developed and communicated to every member of the staff and other affected groups and individuals.

Typical Work Products
- project objectives
- excerpts from the technical management plan

Notes
Examples of project goals include
- a cost/schedule goal
- a quality/cost goal
- a quality/schedule goal

End of PA 04: Integrate Disciplines

PA 05: Integrate System

Summary description

The purpose of Integrate System is to ensure that system elements will function as a whole. This primarily involves identifying, defining, and controlling interfaces, as well as verifying system functions that require multiple system elements. The activities associated with Integrate System occur throughout the entire product life cycle.

Process area notes

The Integrate System activities begin early in the development effort, when interface requirements can be influenced by all engineering disciplines and applicable interface standards can be invoked. They continue through design and checkout. During design, emphasis is on ensuring that interface specifications are documented and communicated. During system element checkout, both prior to assembly and in the assembled configuration, emphasis is on verifying the implemented interfaces. Throughout the integration activities, interface baselines are controlled to ensure that changes in the design of system elements have minimal impact on other elements to which they interface. During testing, or other validation and verification activities, multiple system elements are checked out as integrated subsystems or systems.

There is some redundancy between the process characteristics captured in this process area and some of those in the Evolve System Architecture process area (PA 03). However, the emphasis in the Evolve System Architecture process area is to generate alternatives and select a solution, while the emphasis in this process area is to develop a detailed description of interfaces. The importance of interfaces is also emphasized in this process area.

continued on next page

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.05.01	Develop detailed specifications of the interfaces implied by the system architecture.
BP.05.02	Coordinate interface specifications and changes with all affected groups and individuals.
BP.05.03	Verify the receipt of each system element required to assemble the system in accordance with the physical architecture.
BP.05.04	Verify the implemented design features of developed or purchased system elements against their requirements.
BP.05.05	Verify that the system element interfaces comply with the interface specifications prior to assembly.
BP.05.06	Assemble aggregates of system elements in accordance with the established integration strategy.
BP.05.07	Check the integrated system interfaces in accordance with the established integration strategy.
BP.05.08	Develop an integration strategy and supporting documentation that identify the optimal sequence for receipt, assembly, and activation of the various components that make up the system.

BP 05.01 Define Interfaces

Develop detailed specifications of the interfaces implied by the system architecture.

Description

The bulk of integration problems arise from unknown or uncontrolled aspects of interfaces. Therefore, system and subsystem interfaces are specified as early as possible in the development effort. Interface specifications address logical, physical, electrical, mechanical, human, and environmental parameters as appropriate. Intra-system interfaces are the first design consideration for developers of the system's subsystems. Interfaces are used from previous development efforts or are developed in accordance with interface standards for the given discipline or technology. Novel interfaces are constructed only for compelling reasons. Interface specifications are verified against interface requirements.

Typical Work Products

- interface descriptions
- interface control documents
- interface requirements specifications

Notes

Examples of components of data interface specifications include data element description, direction, and frequency. Mechanical and environmental interface requirements may also be appropriate at the architecture phase, especially for interfaces to existing systems or subsystems.

continued on next page

BP 05.02 Coordinate Interfaces

Coordinate interface specifications and changes with all affected groups and individuals.

Description
This practice is intended to ensure that the interfaces of each element of the system or subsystem are controlled and known to the developers. Additionally, when changes to the interfaces are needed, the changes must at least be evaluated for possible impact to other interfacing elements and then communicated to the affected developers. Although all affected developers are part of the group that makes changes, such changes need to be captured in a readily accessible place so that the current state of the interfaces can be known to all.

Typical Work Products
- interface control documents
- exception reports

Notes
The mechanism for controlling and coordinating changes could take the form of an interface change control board with direct feed to configuration management services.

BP 05.03 Verify Receipt of System Elements

Verify the receipt of each system element required to assemble the system in accordance with the physical architecture.

Description
This practice is intended to ensure that each element of the system or subsystem is received. The elements are checked for quantity, obvious damage, and consistency between the element description and a list of element requirements. In addition, there needs to be some method to assess the timeliness of receipt of system elements.

Typical Work Products
- acceptance documents
- delivery receipts
- checked packing list

Notes
An example activity is to check the packing list against the received items.

continued on next page

BP 05.04 Verify System Element Correctness

Verify the implemented design features of developed or purchased system elements against their requirements.

Description
This practice is intended to ensure that each element of the system or subsystem functions in its intended environment. Such verification may be by test, inspection, analysis, etc., and may be executed either by the organization that will assemble the system or subsystem or by the producing organization. Some method of discerning the elements that "passed" verification from those elements that "failed" will need to be in place.

Typical Work Products
- verified system features
- exception reports

Notes
Examples of verification activities include
- Inspect and/or test elements.
- Prepare deficiency or compliance reports.
- Use regression testing as a tool as subsystems/elements are combined.
- Verify that elements meet requirements before shipping by manufacturer/supplier.

BP 05.05 Verify System Element Interfaces

Verify that the system element interfaces comply with the interface specifications prior to assembly.

Description
This practice is intended to ensure that the interface of each element of the system or subsystem is verified against its corresponding interface specification. Such verification may be by test, inspection, analysis, etc., and may be executed either by the organization that will assemble the system or subsystem or by another organization. Some method of discerning the elements that "passed" verification from those elements that "failed" will need to be in place.

Typical Work Products
- verified system element interfaces
- test reports
- exception reports

Notes
Examples of verification activities include

- Inspect and/or test elements to verify that the interfaces were implemented in accordance with the defined interface specifications.
- Prepare compliance or deficiency reports.

---

continued on next page

BP 05.06 Assemble Aggregates of System Elements

Assemble aggregates of system elements in accordance with the established integration strategy.

Description
This practice is intended to ensure that the assembly of the system elements into larger or more complex assemblies is conducted in accordance with the planned strategy. Testing of the aggregates is explicitly addressed in the Verify and Validate System process area (PA07), and is to occur as needed here.

Typical Work Products
- integration reports
- exception reports

Notes
Examples of system element assembly include
- subsystem build
- subsystem test

BP 05.07 Check Aggregate of System Elements

Check the integrated system interfaces in accordance with the established integration strategy.

Description
This practice is intended to ensure that a planned strategy is followed to assemble the system elements into the final system and test the assembled system. System testing is explicitly addressed in the Verify and Validate System process area (PA07), and is to occur as needed here.

Typical Work Products
- integration reports
- integrated system

Notes
An example activities integration testing, which includes assembling the system according to the integration plan or strategy. This may include practice of system verification procedures.

continued on next page

BP 05.08 Develop Integration Strategy

Develop an integration strategy and supporting documentation that identify the optimal sequence for receipt, assembly, and activation of the various components that make up the system.

Description
Using business as well as technical factors an integration strategy is developed for an assembly, activation, and loading sequence that minimizes cost and assembly difficulties. The larger or more complex the system or the more delicate its elements, the more critical the proper sequence becomes, as small changes can cause large impacts on project results.

The optimal sequence of assembly is built from the bottom up as components become subelements, elements, and subsystems, each of which must be checked prior to fitting into the next higher assembly. The sequence will encompass any effort needed to establish and equip the assembly facilities (e.g., raised floor, hoists, jigs, test equipment, I/O, and power connections). Once established, the sequence must be periodically reviewed to ensure that variations in production and delivery schedules have not had an adverse impact on the sequence or compromised the factors on which earlier decisions were made.

Typical Work Products
- integration strategy document
- assembly/check area drawings
- system/component documentation
- sequence and rationale for selected assembly

Notes
Example contents of a strategy document include
- personnel requirements
- assembly area drawings
- special handling
- system documentation for systems engineering users
- shipping schedule
- assembly sequence and rationale
- test equipment and drivers

End of PA 05: Integrate System

PA 06: Understand Customer Needs and Expectations

Summary description

The purpose of Understand Customer Needs and Expectations is to elicit, stimulate, analyze, and communicate customer needs and expectations to obtain a better understanding of what will satisfy the customer. Understand Customer Needs and Expectations involves engaging the customer or surrogate in ongoing dialogue designed to translate his/her needs and expectations into a verifiable set of requirements which the customer understands and which provide the basis for agreements between the customer and the systems engineering effort.

Customer needs typically change over time. Organizations need to have a workable way to incorporate such changes into current and future version of the product.

Process area notes

Since this process area supports the dialogue between systems engineering and the customer, all other process areas will use it to keep the customer informed throughout the project life cycle.

Customer, as used here, denotes either a directly contracted customer or a customer surrogate who represents a particular market segment in a market-driven, multi-customer industry.

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.06.01	Elicit the customer's needs, expectations, and measures of effectiveness.
BP.06.02	Analyze the customer's needs and expectations to develop a preliminary operational concept of the system.
BP.06.03	Develop a statement of system requirements.
BP.06.04	Obtain the customers' agreement that system requirements satisfy their needs and expectations.
BP.06.05	Inform the customer on a regular basis about the status and disposition of needs, expectations, and measures of effectiveness.

continued on next page

BP 06.01
Elicit Needs

Elicit the customer's needs, expectations, and measures of effectiveness.

Description
Frequently, customer needs and expectations are poorly identified or conflicting. The needs and expectations, as well as customer limitations, must be clearly identified and prioritized. An iterative process is used throughout the life of the project to accomplish this. During this process, an effort is made to identify any unique end-user needs and expectations and to obtain customer approval to include them, or customer justification to omit them. In the case of non-negotiated situations, the surrogate for the end user or customer is frequently the customer relations or marketing part of the organization.

Typical Work Products
• technical performance parameters
• needs statement

Notes
Examples of techniques to elicit needs include
- Joint Applications Design (JAD) meetings
- interface control working groups
- technical control working groups
- interim program reviews
- questionnaires, interviews, operational scenarios obtained from users
- prototypes and models
- brainstorming
- Quality Function Development (QFD)
- market surveys
- beta testing
- extraction from documents, standards, specs., etc.
- observation of existing systems, environments, and workflow patterns

Environmental, legal, and other constraints which may be external to the customer must also be applied when creating and resolving the set of system requirements.

continued on next page

BP 06.02 Analyze Needs

Analyze the customer's needs and expectations to develop a preliminary operational concept of the system.

Description
An analysis is performed to determine what impact the intended operational environment will have on the ability to satisfy the customer's needs and expectations. Feasibility, mission needs, cost constraints, potential market size, etc., must all be taken into account, depending on the product context. The objective of the analysis is to determine system concepts that will satisfy the customer needs and expectations, and then translate these concepts into top-level system requirements. In parallel with this activity, the parameters that will be used to evaluate system effectiveness are determined based on customer input and the preliminary system concept.

Typical Work Products
- operational concept
- system concept
- system cost
- technical parameters
- market-segment description

Notes
Systems engineers must often help the customer formulate complete concepts. The customer's needs and expectations should be probed to ensure that the developer adequately understands them and has prioritized them correctly.

Expression of the logistics, support, maintenance, training, etc., are ways to capture system needs for feedback to the customer.

Examples of formal methodologies used to analyze needs include
- Quality Function Deployment (QFD)
- trade studies
- mathematical techniques (design of experiments, sensitivity analysis, timing, sizing, Monte Carlo simulation)
- prototypes
- customer value determination cycle

continued on next page

BP 06.03 Develop System Requirements

Develop a statement of system requirements.

Description
Once a complete set of customer needs and expectations and a preliminary operational and system concept are available, they are translated into top-level system requirements.

Typical Work Products
- system requirements

Notes
System requirements may be initially provided by the customer. In this case, systems engineering performs a "validation" of these requirements, finding the inconsistencies or holes, and adds to them as necessary. In other cases, the system engineering effort creates the entire set of system requirements.

System requirements may be documented formally using a customer-specified format or internal organization, or they may be captured informally.

continued on next page

BP 06.04 Obtain Customer Agreement

Obtain customers' agreement that system requirements satisfy their needs and expectations.

Description
Customer concurrence on interpretation of needs, operations concept, results of analyses, and translation of needs into system requirements is obtained initially via extensive communication. These understandings to which the customer committed are then updated throughout the life of the project.

Typical Work Products
- validated system requirements
- storyboards
- models

Notes
Examples of forums to obtain customer concurrence include
- working groups
- formal program reviews
- payment milestones
- in-process reviews
- status meetings
- weekly telephone conferences
- focus groups
- beta tests

Results of trade studies and/or feasibility studies can be presented to the customer to elicit their preferred approach.

continued on next page

BP 06.05 Inform Customer

Inform the customer on a regular basis about the status and disposition of needs, expectations, and measures of effectiveness.

Description
Communication with the customer is particularly crucial while analyzing customer needs and deciding on general approaches. A key aspect of refining the common understanding of customer needs and expectations is communicating the results of preliminary analysis and obtaining the customer's feedback. Informing the customer continues throughout the life of the project. Another aspect of building customer understanding could be eliciting and stimulating new needs.

Typical Work Products
- technical interchange minutes
- prototypes
- requirement traceability tables

Notes
Examples of forums to inform the customer include
- working groups
- formal program reviews
- payment milestones
- in-process reviews
- status meetings
- weekly telephone conferences
- focus groups
- beta tests

End of PA 06: Understand Customer Needs and Expectations

PA 07: Verify and Validate System

Summary description

The purpose of Verify and Validate System is to ensure that the developer/supplier team performs increasingly comprehensive evaluations to ensure that evolving work products will meet all requirements. The activities associated with Verify and Validate System begin early in the development, address all work products (including requirements and design), and continue through development and integration of system elements into production, use, and disposal of the system. The scope of verification covers development of the full system, as well as its production, operation, and support. Validation is a measure of customer satisfaction, given the customer's operational need. Validation should continue throughout product use.

Process area notes

Means of evaluation associated with verification include inspection, analysis, demonstration, prototyping, simulation, and testing. Evaluation begins early in the development process to ensure that requirements and specifications are correct from the highest levels as they are allocated downward (top-down); later, it becomes a bottom-up integration from the lowest level through each higher level of integration to cover the full system and its associated manufacturing processes and procedures.

Verification primarily address the work products of the process areas Understand Customer Needs and Expectations (PA06), Analyze Candidate Solutions (PA01), Derive and Allocate Requirements (PA02), Evolve System Architecture (PA03), and Integrate System (PA05). In many environments, the term “test” is used to encompass the concepts included in verification and validation. Corrective actions resulting from verification and validation are monitored in the Monitor and Control Technical Effort process area (PA11).

Validation is an evaluation of the customer's satisfaction with the product, at its current stage of development. The customer's evaluation should be done in a realistic operational environment. Such an environment could include personnel, procedures, data packages, and logistical support.

continued on next page

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.07.01	Establish plans for verification and validation that identify the overall requirements, objectives, resources, facilities, special equipment, and schedule applicable to the system development.
BP.07.02	Define the methods, process, reviews, inspections, and tests by which incremental products that were verified against established criteria or requirements that were established in a previous phase.
BP.07.03	Define the methods, process, and evaluation criteria by which the system or product is verified against the system or product requirements.
BP.07.04	Define the methods, process, and evaluation criteria by which the system or product will be validated against the customer's needs and expectations.
BP.07.05	Perform the verification and validation activities that are specified by the verification and validation plans and procedures, and capture the results.
BP.07.06	Compare the collected test, inspection, or review results with established evaluation criteria to assess the degree of success.

continued on next page

BP 07.01 Establish Verification and Validation Plans

Establish plans for verification and validation that identify the overall requirements, objectives, resources, facilities, special equipment, and schedule applicable to the system development.

Description

The purpose of developing plans for verification and validation activities is to establish the requirements, objectives, resources, facilities, special equipment, and schedule for coordination among the development team and with the customer. Plans for verification of incrementally developed products address evaluation of identified work products such as in-progress requirement, design, and component specifications; formal and informal reviews and audits; and inspection of completed or received (procured) components or subsystems. System-level verification plans also address integration requirements, incremental builds, and reverification activities. Development of validation plans involves the customer (or surrogate) in determining the approach, schedule, system configuration, environment, and resource requirements for operational evaluation of the system. Include verification, configuration control, and maintenance of the test equipment and environment.

Typical Work Products

- master test and evaluation plan
- system test plan
- operational test and evaluation plan

Notes

Example practices include

- Develop master test and evaluation plan.
- Develop system test plan.
- Develop operational test and evaluation plan.
- Use regression testing, especially where modifications are being incorporated.

continued on next page

BP 07.02 Define Incremental Verification

Define the methods, process, reviews, inspections, and tests by which incremental products are verified against established criteria or requirements that were established in a previous phase.

Description
Define incremental verification involves identifying the incremental work products, (such as requirements, designs, software code, or hardware components) to be verified. It also involves defining the methods, procedures, reviews, inspections or tests, and evaluation criteria by which the work products are to be evaluated.

Typical Work Products
- requirements inspection procedure and acceptance criteria
- design inspection procedure and acceptance criteria
- component test procedure and acceptance criteria
- operational use reports

Notes
The level of verification should range from the lowest units to the overall system and should include usability. Methods should include analysis, prototyping, and simulation, as well as evaluation of the deliverable product.

Examples of process activities related to the practice include
- Conduct formal and informal technical reviews and audits.
- Define the procedures, checklists, and evaluation criteria for in-progress design reviews.
- Define the test equipment, test data, procedures, and evaluation criteria for component tests.

continued on next page

BP 07.03 Define System Verification

Define the methods, process, and evaluation criteria by which the system or product is verified against the system or product requirements.

Description
Define system verification consists of defining the methods (test, analysis, demonstration, inspection), verification conditions, environmental conditions, and system configuration under which the system will be verified. In the case of testing, it also includes defining inputs, outputs, expected results, and evaluation criteria for each product requirement or group of requirements against which the system is to be evaluated.

Typical Work Products
- system test procedures

Notes
Example practices include
- Define the environment, test cases, inputs, expected results, and evaluation criteria for system test.
- Capture traceability between system requirements and test requirements.

BP 07.04 Define System Validation

Define the methods, process, and evaluation criteria by which the system or product will be validated against the customer's needs and expectations.

Description
Define system validation consists of defining the test environment, operational scenario, test procedures, inputs, outputs, expected results, and evaluation criteria for validation of the developed system. Defining system validation takes into account the customer as user/operator of the system during testing. It includes both structured and unstructured use and operation of the system or product by the user, and defines the type of data to be collected, analyzed, and reported.

Typical Work Products
- test environment definition
- simulation requirements
- validation procedures

Notes
Example practices include
- Define realistic operational environment.
- Identify requirements for the operational environment.

continued on next page

BP 07.05 Perform and Capture Verification and Validation

Perform the verification and validation activities that are specified by the verification and validation plans and procedures, and capture the results.

Description

Incremental work products, subsystems, components, and systems are verified and validated. Verification and validation start early in project and are performed according to defined procedures. The results are captured to support analysis and comparison with expected results defined in the verification procedures. Verification of requirements, design, and as-built components involves both comparison with established standards and criteria, and comparison with the parent work product from a prior phase (e.g., comparison of the requirements with the design). Validation is performed to ensure the customer's expectations have been captured or realized in the work product or system. The verification or validation environment is carefully controlled to provide for replication, analysis of results, and reverification of problem areas.

Typical Work Products

- inspection results
- test results
- system validation data
- validation exception reports
- trouble reports

Notes

Example practices include

- Validate system requirements.
- Review requirements specifications.
- Perform receiving inspection of procured components.
- Perform formal and informal technical reviews.
- Perform system test.
- Perform operational test and evaluation.

continued on next page

BP 07.06 Assess Verification and Validation Success

Compare the collected test, inspection, or review results with established evaluation criteria to assess the degree of success.

Description
Verification and validation activities are executed and the resulting data collected according to established plans and procedures. The data resulting from tests, inspections, or evaluations are then analyzed against the defined verification or validation criteria. Analysis reports indicate whether or not requirements were met and, in the case of deficiencies, assess the degree of success or failure and categorize the probable cause of failure.

The collected test, inspection, or review results are compared with established evaluation criteria, to determine whether to proceed or to rework and retest.

Typical Work Products
- test deficiency reports
- test incident reports

Notes
Example practices include
- Capture inspection results.
- Assess inspection results for root causes.
- Capture test results.
- Analyze test anomalies.

End of PA 07: Verify & Validate System

PA 08: Ensure Quality

Summary description

The purpose of Ensure Quality is to address not only the quality of the system, but also the quality of the process being used to create the system and the degree to which the project follows the defined process. The underlying concept of this process area is that high-quality systems can only be consistently produced on a continuous basis if a process exists to continuously measure and improve quality. In addition, this process must be adhered to rigorously and throughout the system life cycle. Key aspects of the process required to develop high-quality systems are measurement, analysis, and corrective action.

Process area notes

A successful quality program requires integration of the quality efforts throughout the project team and support elements. Effective processes provide a mechanism for building in quality and reduce dependence on end-item inspections and rework cycles.

This is not meant to imply that those managing and/or assuring the quality of work products and processes are solely responsible for the quality of the work product outputs. On the contrary, the primary responsibility for "building in" quality lies with the builders. A quality management process helps to ensure that all aspects of quality management are seriously considered and acted upon by the organization and reflected in its products. This increases the confidence of developers, management, and customers in the system's quality.

The kinds of quality variances that may be addressed by this process area include technical content, such as the particular values of derived or allocated requirements; and form issues, such as whether the customer prefers instructions on product use to be in paper or electronic form. Cost and schedule variances can also be considered defects and would be dealt with as are other defects.

Organizations may wish to determine the variances, from expected values, of technical and other issues in increments that correspond to the schedule commitments of the organization. For example, if the organization has committed to deliver or roll-out a product during a given week, then it would be wise to measure or determine its progress, by measuring variances, on a weekly basis. If the commitment is monthly, then monthly measurements would likely be appropriate.

continued on next page

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.08.01	Ensure the defined system engineering process is adhered to during the system life cycle.
BP.08.02	Evaluate work product measures against the requirements for work product quality.
BP.08.03	Measure the quality of the systems engineering process used by the project.
BP.08.04	Analyze quality measurements to develop recommendations for quality improvement or corrective action as appropriate.
BP.08.05	Obtain employee participation in identifying and reporting quality issues.
BP.08.06	Initiate activities that address identified quality issues or quality improvement opportunities.
BP.08.07	Establish a mechanism or a set of mechanisms to detect the need for corrective actions to processes or products.

BP 08.01 Monitor Conformance to the Defined Process

Ensure the defined system engineering process is adhered to during the system life cycle.

Description

Ensure that the project's execution follows the defined system engineering process. Compliance should be checked at useful intervals. Deviations from the defined process and the impact of the deviation should be recorded.

Typical Work Products

- recorded deviations from defined systems engineering process
- recorded impact of deviations from defined systems engineering process
- quality handbook (paper or on-line)

Notes

The defined process can be monitored in a number of ways. For example, a designated auditor/reviewer can participate in or observe all (or a sample percentage of) process activities, or an auditor/reviewer may inspect all (or a sample percentage of) in-process work products.

continued on next page

BP 08.02 Measure Quality of the Work Product

Evaluate work product measures against the requirements for work product quality.

Description

Measuring the characteristics of the work product provides an indication of the quality of the system. Measurements should be designed to assess whether the work product will meet customer and engineering requirements. Product measurements should also be designed to help isolate problems with the system development process.

Typical Work Products

- assessment of the quality of the product
- product quality certification

Notes

Example approaches to measurement of work product quality include

- statistical process control of product measurements at various points in the development process
- measurement of a complete set of work product requirements such as
  - specification value
  - planned value
  - tolerance band
  - demonstrated value
  - demonstrated technical variance
  - current estimate
  - predicted technical variance

continued on next page

BP 08.03 Measure Quality of the Process

Measure the quality of the systems engineering process used by the project.

Description
The process that is used to create a quality product is as important as the quality of the product. It is important to have a system development process that is checked by measurement so that degrading conditions are caught early, before the final work product is produced and found to not meet requirements. Thus, having a process that is measured may lead to less waste and higher productivity.

Typical Work Products
- process quality certification

Notes
Examples of tools to use in measuring the process include
- process flow chart: can be used to determine which characteristics should be measured and to identify potential sources of variation, in addition to defining the process
- statistical process control on process parameters
- design of experiments

BP 08.04 Analyze Quality Measure-ments

Analyze quality measurements to develop recommendations for quality improvement or corrective action, as appropriate.

Description
Careful examination of all of the available data on product, process, and project performance can reveal causes of problems. This information will then enable improvement of the process and product quality.

Typical Work Products
- analysis of deviations
- failure analysis
- defect reports
- system quality trends
- corrective action recommendations
- cause and effect diagrams

Notes
Examples of measurements that support quality improvement include

- trend analysis, such as the identification of equipment calibration issues causing a slow creep in the product parameters
- standards evaluation, such as determining if specific standards are still applicable due to technology or process changes

---

continued on next page

BP 08.05 Obtain Participation

Obtain employee participation in identifying and reporting quality issues.

Description
The development of a quality work product, using a quality process that is adhered to, requires the focus and attention of all of the people involved. Ideas for improving quality need to be encouraged, and a forum needs to exist that allows each employee to raise process-quality issues freely.

Typical Work Products
- environment that promotes quality
- captured inputs and resolutions from workers

Notes
A quality environment can be fostered by
- process action teams
- a quality assurance group with a reporting chain of command that is independent of the project
- an independent channel for reporting quality issues

BP 08.06 Initiate Quality Improvement Activities

Initiate activities that address identified quality issues or quality improvement opportunities.

Description
In order to continuously improve quality, specific actions must be planned and executed. Specific aspects of the system development process that jeopardize product or process quality need to be identified and corrected. This would include minimizing cumbersome or bureaucratic systems.

Typical Work Products
- recommendations for improving the systems engineering process
- quality improvement plan
- process revisions

Notes
Effective implementation of quality improvement activities requires input and buy-in by the work product team.

continued on next page

BP 08.07 Detect Need for Corrective Actions

Establish a mechanism or a set of mechanisms to detect the need for corrective actions to processes or products.

Description
Such a mechanism must be available throughout the life cycle of the product (development through manufacturing through customer use). Mechanisms may include online reporting systems, workshops, periodic reviews, customer focus groups, etc. Mechanisms must be available to all affected groups, including design, manufacturing, customers, customer support, etc.

Typical Work Products
- ongoing database or repository containing identified needs, process improvements, and product improvements
- clearly described processes, methods, and avenues for getting identified needs into a database or repository
- identified needs for process improvement
- identified needs for product improvement
- trouble reports

Notes
This base practice is critical to the effective use of systems engineering in the production, operations, and maintenance life-cycle phases.

Needs for corrective action are detected in this base practice. Corrective actions are directed in the Monitor and Control Technical Effort process area (PA11).

Trouble reports also flow into this base practice from the Verify and Validate System process area (PA07).

End of PA 08: Ensure Quality

PA 09: Manage Configurations

Summary description

The purpose of Manage Configurations is to maintain data on and status of identified configuration units, and to analyze and control changes to the system and its configuration units. Managing the system configuration involves providing accurate and current configuration data and status to developers and customers.

This process area is applicable to all work products that are placed under configuration management. An example set of work products that may be placed under configuration management could include hardware and software configuration items, design rationale, requirements, product data files, or trade studies.

Process area notes

The configuration management function supports traceability by allowing the configuration to be traced back through the hierarchy of system requirements at any point in the configuration life cycle. Traceability is established as part of the practices in the Derive and Allocate Requirements process area (PA02).

When the practices of this process area are used to manage requirements, changes to those requirements need to be iterated through the Understand Customer Needs and Expectations process area (PA06) to communicate the impact of changes to the customer or their surrogate.

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.09.01	Decide among candidate methods for configuration management.
BP.09.02	Identify configuration units that constitute identified baselines.
BP.09.03	Maintain a repository of work product baselines.
BP.09.04	Control changes to established configuration units.
BP.09.05	Communicate status of configuration data, proposed changes, and access information to affected groups.

continued on next page

BP 09.01 Establish Configuration Management Methodology

Decide among candidate methods for configuration management.

Description
Three primary trade-off considerations will have an impact on the structure and cost of configuration management, including
- the level of detail at which the configuration units are identified
- when the configuration units are placed under configuration management
- the level of formalization required for the configuration management process

The Analyze Candidate Solutions process area (PA01) should be used as guidance to perform the trade studies.

Typical Work Products
- guidelines for identifying configuration units
- timeline for placing configuration units under configuration management
- selected configuration management process
- selected configuration management process description

Notes
Example criteria for selecting configuration units at the appropriate work product level include
- need to maintain interfaces at a manageable level
- unique user requirements such as field replaceable units
- new versus modified design
- expected rate of change

These criteria will affect the level of visibility into the design effort.

Example criteria for determining when to place work products under configuration management include
- portion of the development life cycle that the project is in
- if system element is ready for test
- degree of formalization selected
- cost and schedule limitations
- customer requirements

Example criteria for selecting a configuration management process include

- portion of the development life cycle
- impact of change in system on other work products
- impact of change in system on procured or subcontracted work products
- impact of change in system on program schedule and funding
- requirements management

continued on next page

BP 09.02 Identify Configuration Units

Identify configuration units that constitute identified baselines.

Description
A configuration unit is one or more work products that are baselined together. The selection of work products for configuration management should be based on criteria established in the selected configuration management strategy. Configuration units should be selected at a level that benefits the developers and customers, but that does not place an unreasonable administrative burden on the developers.

Typical Work Products
- baselined work product configuration
- identified configuration units

Notes
Configuration units in the area of requirements management could vary from individual requirements to groupings of requirements documents.

Configuration units for a system that has requirements on field replacement should have an identified configuration unit at the field-replaceable unit level.

continued on next page

BP 09.03 Maintain Work Product Baselines

Maintain a repository of work product baselines.

Description

This practice involves establishing and maintaining a repository of information about the work product configuration. Typically, this consists of capturing data or describing the configuration units. This could also include an established procedure for additions, deletions, and modifications to the baseline, as well as procedures for tracking/ monitoring, auditing, and the accounting of configuration data. Another objective of maintaining the configuration data is to provide an audit trail back to source documents at any point in the system life cycle.

Typical Work Products

- decision database
- baselined configuration
- traceability matrix

Notes

In the case of hardware configuration units, the configuration data would consist of specifications, drawings, trade study data, etc. Optimally, configuration data can be maintained in electronic format to facilitate updates and changes to supporting documentation.

Software configuration units typically include source code files, requirements and design data, and test plans and results.

continued on next page

BP 09.04 Control Changes

Control changes to established configuration units.

Description
Control is maintained over the configuration of the baselined work product. This includes tracking the configuration of each of the configuration units, approving a new configuration, if necessary, and updating the baseline.

Identified problems with the work product or requests to change the work product are analyzed to determine the impact that the change will have on the work product, program schedule and cost, and other work products. If, based upon analysis, the proposed change to the work product is accepted, a schedule is identified for incorporating the change into the work product and other affected areas.

Changed configuration units are released after review and formal approval of configuration changes. Changes are not official until they are released.

Typical Work Products
- new work-product baselines

Notes
Change control mechanisms can be tailored to categories of changes. For example, the approval process should be shorter for component changes that do not affect other components.

BP 09.05 Communicate Configuration Status

Communicate status of configuration data, proposed changes, and access information to affected groups.

Description
Inform affected groups of the status of configuration data whenever there are any status changes. The status reports should include information on when accepted changes to configuration units will be processed, and the associated work products that are affected by the change. Access to configuration data and status should be provided to developers, customers, and other affected groups.

Typical Work Products
- status reports

Notes
Examples of activities for communicating configuration status include
- Provide access permissions to authorized users.
- Make baseline copies readily available to authorized users.

End of PA 09: Manage Configurations

PA 10: Manage Risk

Summary description

The purpose of Manage Risk is to identify, assess, monitor, and mitigate risks to the success of both the systems engineering activities and the overall technical effort. This process area continues throughout the life of the project. Similar to the Plan Technical Effort (PA12) and Monitor and Control Technical Effort (PA11) process areas, the scope of this process area includes both the systems engineering activities and the overall technical project effort, as the systems engineering effort on the project cannot be considered successful unless the overall technical effort is successful.

Process area notes

All system development efforts have inherent risks, some of which are not easily recognized. Especially early on, the likelihood of known risks and the existence of unknown risks should be sought out. Poor risk management is often cited as a primary reason for unsatisfied customers, and cost or schedule overruns. Early detection and reduction of risks avoid the increased costs of reducing risks at a more advanced state of system development.

It is important to note the distinction among risk types, analysis, and management approach. Good risk management operates on all three dimensions. For example, analyzing developer risk primarily deals with the management approach, i.e., profit and market building; whereas analyzing user risk primarily is concerned with types and analysis, i.e., mission and goal satisfaction.

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.10.01	Develop a plan for risk-management activities that is the basis for identifying, assessing, mitigating, and monitoring risks for the life of the project.
BP.10.02	Identify project risks by examining project objectives with respect to the alternatives and constraints, and identifying what can go wrong.
BP.10.03	Assess risks and determine the probability of occurrence and consequence of realization.
BP.10.04	Obtain formal recognition of the project risk assessment.
BP.10.05	Implement the risk-mitigation activities.
BP.10.06	Monitor risk-mitigation activities to ensure that the desired results are being obtained.

continued on next page

BP 10.01 Develop Risk Management Approach

Develop a plan for risk-management activities that is the basis for identifying, assessing, mitigating, and monitoring risks for the life of the project.

Description
The purpose of this base practice is to develop an effective plan to guide the risk-management activities of the project. Elements of the plan should include identification of members of the risk-management team and their responsibilities; a schedule of regular risk-management activities, methods, and tools to be employed in risk identification and mitigation; and methods of tracking and controlling risk-mitigation activities. The plan should also provide for the assessment of risk-management results.

Typical Work Products
- risk-management plan

Notes
Examples of risk-management approaches include
- Use a spiral management approach where the objectives for the next cycle and the objectives for the overall project are clarified and documented periodically.
- Formally identify and review risks at the beginning of each cycle and develop mitigation approaches.
- At the end of each cycle, review progress made in reducing each risk.

continued on next page

BP 10.02 Identify Risks

Identify project risks by examining project objectives with respect to the alternatives and constraints, and identifying what can go wrong.

Description
Examine the project objectives, the project plans (including activity or event dependencies), and the system requirements in an orderly way to identify probable areas of difficulties and what can go wrong in these areas. Sources of risk based on past experience should be considered to identify potential risks. This activity is enacted during the Plan Technical Effort process area (PA12). Establishing critical development dependencies and providing tracking and corrective action is performed in the Monitor and Control Technical Effort process area (PA11).

Typical Work Products
- list of identified risks

Notes
Examples of activities to identify risks include
- Develop a common risk classification scheme or risk taxonomy to categorize risks. This taxonomy contains the history of risks for each category, including probabilities of occurrence (which system elements contribute most to risk), estimated cost of occurrence, and mitigation strategies. This practice is very useful in improving risk estimates and in reusing successful risk-mitigations [Charette 89].
- Focus mitigation resources and controls on system elements which contribute most to risk.
- Collect all the information specifying project and systems engineering objectives, alternative technical strategies, constraints, and success criteria. Ensure that the objectives for the project and the systems engineering effort are clearly defined. For each alternative approach suggested to meet the objectives, document items that may prevent attainment of the objectives: these items are risks. Following this procedure results in a list of risks per alternative approach. Note, some risks will be common across all the alternatives.
- Interview technical and management personnel to uncover assumptions and decisions leading to risk. Use historical data from similar projects to find out where problems have arisen in similar contexts.

continued on next page

BP 10.03
Assess Risks

Assess risks and determine the probability of occurrence and consequence of realization.

Description
Estimate the chance of potential loss (or gain) and the consequence if the previously identified risks occur. Analyze the risks independently of one another and understand the relationships between different individual risks. The analysis methodology should take into account factors such as the probability of failure due to the maturity and complexity of the technology.

Typical Work Products
- risk assessment

Notes
Examples of activities to assess risks include
- Develop standards for estimating the probability and cost of risk occurrence. Possible standards range from a simple high-moderate-low qualitative scale to quantitative scales in dollars and probability to the nearest tenth of a percent.
- Establish a practical standard based on the project's size, duration, overall risk exposure, system domain, and customer environment [Charette 89].

BP 10.04
Review Risk Assessment

Obtain formal recognition of the project risk assessment.

Description
Review adequacy of the risk assessment and obtain a decision to proceed, modify, or cancel the effort based on risks. This review should include the potential risk-mitigation efforts and their probability of success.

Typical Work Products
- risk-mitigation strategy

Notes

Examples of activities to review the risk assessment include

- Hold a meeting of all stakeholders of the project internal to the company to present the risk assessment. To help communicate a sense of control over the risks, present possible mitigation strategies along with each risk.
- Obtain agreement from the attendees that the risk estimates are reasonable and that no obvious mitigation strategies are being overlooked.

---

continued on next page

BP 10.05 Execute Risk Mitigations

Implement the risk-mitigation activities.

Description
Risk-mitigation activities may address lowering the probability that the risk will occur or lowering the extent of the damage the risk causes when it does occur. For risks that are of particular concern, several risk-mitigation activities may be initiated at the same time.

Typical Work Products
- risk-mitigation plan

Notes
Examples of activities to mitigate risks include the following:
- To address the risk that the delivered system will not meet a specific performance requirement, build a prototype of the system or a model that can be tested against this requirement. This type of mitigation strategy lowers the probability of risk occurrence.
- To address the risk that the delivery schedule will slip due to a subsystem not being available for integration, develop alternative integration plans with different integration times for the risky subsystem. If the risk occurs (i.e., the subsystem is not ready on time), the impact of the risk on the overall schedule will be less. This type of mitigation strategy lowers the consequence of risk occurrence.
- Use predetermined baselines (risk referents) to trigger risk-mitigation actions [Charette 89].

BP 10.06 Track Risk Mitigations

Monitor risk-mitigation activities to ensure that the desired results are being obtained.

Description
On a regular basis, examine the results of the risk mitigations that have been put into effect, to measure the results, and determine whether the mitigations have been successful.

Typical Work Products
- risk status
- risk taxonomy

Notes
For a project with a development schedule of about six months, re-assess risks every two weeks. Re-estimate the probability and consequence of each risk occurrence.

End of PA 10: Manage Risk

PA 11: Monitor and Control Technical Effort

Summary description

The purpose of Monitor and Control Technical Effort is to provide adequate visibility of actual progress and risks. Visibility encourages timely corrective action when performance deviates significantly from plans.

Monitor and Control Technical Effort involves directing, tracking and reviewing the project's accomplishments, results, and risks against its documented estimates, commitments, and plans. A documented plan is used as the basis for tracking the activities and risks, communicating status, and revising plans.

Process area notes

Similar to the Plan Technical Effort process area (PA12), this process area applies to the project's technical activities as well as to the systems engineering effort.

Progress is primarily determined by comparing the actual effort, work product sizes, cost, and schedule to the plan when selected work products are completed and at selected milestones. When it is determined that the plans are not being met, corrective actions are taken. These actions may include revising the plans to reflect the actual accomplishments and replanning the remaining work, or taking actions to improve performance or reduce risks.

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.11.01	Direct technical effort in accordance with technical management plans.
BP.11.02	Track actual use of resources against technical management plans.
BP.11.03	Track performance against the established technical parameters.
BP.11.04	Review performance against the technical management plans.
BP.11.05	Analyze issues resulting from the tracking and review of technical parameters to determine corrective actions.
BP.11.06	Take corrective actions when actual results deviate from plans.

continued on next page

BP 11.01 Direct Technical Effort

Direct technical effort in accordance with technical management plans.

Description
Carry out the technical management plans created in the Plan Technical Effort process area. This practice involves technical direction of all of the engineering activities of the project.

Typical Work Products
- matrix of responsibilities
- work authorizations

Notes
Effective technical direction includes the use of appropriate communication mechanisms and timely distribution of technical information to all affected parties. All technical direction must be captured to preserve the basis for decisions and actions.

BP 11.02 Track Project Resources

Track actual use of resources against technical management plans.

Description
Provide current information on the use of resources during the project to help adjust the effort and plans when needed.

Typical Work Products
- resource usage

Notes
Tracking cost includes comparing the actual costs to the estimates documented in the project plan to identify potential overruns and underruns.

continued on next page

BP 11.03 Track Technical Parameters

Track performance against the established technical parameters.

Description
The actual performance of the project and its products is tracked by measuring the technical parameters established in the technical management plan. These measurements are compared to the thresholds established in the technical management plan so that warnings of problems can be communicated to management.

Typical Work Products
- profile of technical performance management

Notes
An example of a performance tracking scenario follows:
For each technical parameter, define a benchmarking activity that will be used to obtain the measurement. Use persons from outside the control of the project manager to perform the benchmarking activities to ensure objective measurements. Periodically perform the benchmarking activity and compare the actual measurement with the planned values of the parameters.

BP 11.04 Review Project Performance

Review performance against the technical management plans.

Description
The performance of the project and its products is reviewed periodically and when technical parameter thresholds are exceeded. The results of analyzing the measurements of technical performance are reviewed, along with other indicators of technical performance, and corrective action plans are approved.

Typical Work Products
- change requests for the technical management plan
- approved corrective actions

Notes
Examples of reviewing performance include
- Holding a meeting of all stakeholders of the project internal to the organization to present analyses of performance and suggested corrective actions.
- Writing a status report which forms the basis of a project review meeting.

continued on next page

BP 11.05 Analyze Project Issues

Analyze issues resulting from the tracking and review of technical parameters to determine corrective actions.

Description
New project issues surface frequently and continuously through the project life cycle. Timely identification, analysis, and tracking of issues is crucial to controlling project performance.

Typical Work Products
- analysis of project performance issues
- approved corrective actions

Notes
New information is integrated with historical project data. Trends that are hurting the project are identified, along with new issues that indicate risks to the project's success. Obtain more detailed data, as needed, for issues and trends that are inconclusive. Analysis frequently requires modeling and simulation tools as well as outside expert opinions.

BP 11.06 Take Corrective Action

Take corrective actions when technical parameters indicate future problems or when actual results deviate from plans.

Description
When corrective actions are approved, take the corrective actions by reallocating resources, changing methods and procedures, or increasing adherence to the existing plans. When changes to the technical management plan are necessary, employ the practices of the Plan Technical Effort process area (PA12) to revise the plan.

Typical Work Products
- resource reallocations
- changes to methods and procedures
- change orders

Notes
This base practice covers whatever actions are needed to prevent anticipated problems or to correct the problems discovered. The possible actions taken under this base practice are varied and numerous.

End of PA 11: Monitor and Control Technical Effort

PA 12: Plan Technical Effort

Summary description

The purpose of Plan Technical Effort is to establish plans that provide the basis for scheduling, costing, controlling, tracking, and negotiating the nature and scope of the technical work involved in system development, manufacturing, use, and disposal. System engineering activities must be integrated into comprehensive technical planning for the entire project.

Plan technical effort involves developing estimates for the work to be performed, obtaining necessary commitments from interfacing groups, and defining the plan to perform the work.

Process area notes

Planning begins with an understanding of the scope of the work to be performed, along with the constraints, risks, and goals that define and bound the project. The planning process includes steps to estimate the size of work products, estimate the resources needed, produce a schedule, consider risks, and negotiate commitments. Iterating through these steps may be necessary to establish a plan that balances quality, cost, and schedule goals.

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.12.01	Identify resources that are critical to the technical success of the project.
BP.12.02	Develop estimates for the factors that affect the magnitude and technical feasibility of the project.
BP.12.03	Develop cost estimates for all technical resources required by the project.
BP.12.04	Determine the technical process to be used on the project.
BP.12.05	Identify technical activities for the entire life cycle of the project.
BP.12.06	Define specific processes to support effective interaction with the customer(s) and supplier(s).
BP.12.07	Develop technical schedules for the entire project life cycle.
BP.12.08	Establish technical parameters with thresholds for the project and the system.
BP.12.09	Use the information gathered in planning activities to develop technical management plans that will serve as the basis for tracking the salient aspects of the project and the systems engineering effort.
BP.12.10	Review the technical management plans with all affected groups and individuals, and obtain group commitment.

continued on next page

BP 12.01 Identify Critical Resources

Identify resources that are critical to the technical success of the project.

Description
Critical resources are resources that are essential to the success of the project and that may not be available for the project. Critical resources may include personnel with special skills, tools, facilities, or data. Critical resources can be identified by analyzing project tasks and schedules, and by comparing this project with similar projects.

Typical Work Products
- identified critical resources

Notes
Example practice: Examine the project schedule and think of the types of resources required at each point in time. List resources that are not easily obtainable. Cross check and augment this list by thinking of engineering skills that are required to synthesize the system and work products.

BP 12.02 Estimate Project Scope

Develop estimates for the factors that affect the magnitude and technical feasibility of the project.

Description

The project's scope and size can be estimated by decomposing the system into component elements that are similar to those of other projects. The size estimate can then be adjusted for factors such as differences in complexity or other parameters.

Historical sources often provide the best available information to use for initial size estimates. These estimates will be refined as more information on the current system becomes available.

Typical Work Products

- estimates of the scope of the system
  - number of source lines of code
  - number of cards of electronics
  - number of large forgings
  - number of cubic yards of material to be moved

Notes

Example practice: Analyze the available project documentation, and interview project personnel to determine the main technical constraints and assumptions. Identify the possible highest level technical approaches and the factors that may keep the project or the systems engineering effort from being successful. Identify the major technical parameters and estimate the acceptable range for each parameter.

continued on next page

BP 12.03 Estimate Project Costs

Develop cost estimates for all technical resources required by the project.

Description
A detailed estimate of project costs is essential to good project management, whether or not it is required by a customer. Estimates of project costs are made by determining the labor costs, material costs, and subcontractor costs based on the schedule and the identified scope of the effort. Both direct costs and indirect costs (such as the cost of tools, training, special test and support items) are included. For labor costs, historical parameters or cost models are employed to convert hours to dollars based on job complexity, tools, available skills and experience, schedules, and direct and overhead rates. Appropriate reserves are established, based on identified risks.

Typical Work Products
- total labor cost by skill level and schedule
- cost of material by item, vendor, and schedule
- cost of subcontracts by vendor and schedule
- cost of tools
- cost of training
- supporting rationale

Notes
A considerable amount of project data such as scope, schedule, and material items must be collected prior to estimating costs. Checklists and historical data from other projects can be used to identify cost items which may otherwise be overlooked. Variance reports and "lessons-learned" documents are typically good sources of this type of information.

continued on next page

BP 12.04 Determine Project's Process

Determine the technical process to be used on the project.

Description
At the highest level, the technical process should follow a life-cycle model based on the characteristics of the project, the characteristics of the organization, and the organization's standard process. Typical life-cycle models include waterfall, evolutionary spiral, and incremental. In the process definition, include process activities, inputs, outputs, sequences, and quality measures for process and work products.

Typical Work Products
- selected systems engineering process for the project

Notes
Establish and maintain an integrated management plan that defines the project's interaction with all internal and external organizations (e.g., the subcontractor) performing the technical effort. Include the planned project life-cycle model for the project and specific project activities.

BP 12.05 Identify Technical Activities

Identify technical activities for the entire life cycle of the project.

Description
Project and systems engineering activities may be selected from applicable standards, known best practice within the industry segment, reference models such as the SE-CMM, or the organization's historical experience.

Typical Work Products
- identified technical activities

Notes
Use historical records from similar projects, where possible, to develop the list of activities and to gain confidence that the list is complete. Use the "rolling wave" paradigm for planning. The "rolling wave" paradigm is used to define near-term activities more precisely than activities that start later in the project.

For example, the systems engineering activities would be decomposed into activities planned for the next three months until each activity is approximately two weeks in duration. Activities 3 to 12 months away should be planned at approximately a month in duration. Activities starting more than a year away can be described at a very high level, approximately two months in duration. For the nonsystems-engineering technical activities, use this same method while working with other disciplines according to the Integrate Disciplines process area (PA04).

continued on next page

BP 12.06 Define Project Interface

Define specific processes to support effective interaction with customer(s) and supplier(s).

Description
Project interfaces include all those with organizations and individuals who are necessary to successful project execution, whether they are inside or outside the project group. Types of interaction include information exchange, tasking, and deliveries. Methods and processes (including controls) for interaction are established as appropriate for the parties that are interacting.

Typical Work Products
- defined processes for project interfaces

Notes
For the project, identify the groups internal and external to your organization that the project needs to interact with in order to be successful. For each group, perform the base practices of the Integrate Disciplines process area (PA04) to define and implement each interface in terms of interaction mechanisms, interaction frequency, and problem resolution mechanisms.

continued on next page

BP 12.07 Develop Project Schedules

Develop technical schedules for the entire project life cycle.

Description
Project schedules include system and component development, obtaining procured items, training, and preparing the engineering support environment. Schedules are based on verifiable effort models or data for identified tasks, and they must allow for task interdependencies and the availability of procured items. Schedules should also include slack time appropriate for identified risks. All affected parties must review and commit to the schedule.

Typical Work Products
- project schedules

Notes
Schedules typically include both customer and technical milestones.

Example: Within project constraints (contractual, market timing, customer-provided inputs, etc.), define system increments consistent with the overall technical approach. Each increment should provide more system capability from the user's point of view. Estimate the additional staff hours required to develop each increment.

To create a schedule that uses resources at a level rate, select dates for completion of each increment proportional to the amount of work required to develop the increment. Derive detailed schedules for technical activities within each increment by sequencing the activities from the start of the increment and taking into account dependencies between activities.

For an event-driven schedule, the loading is typically not level. For noncritical-path activities, it may be necessary to adjust the activity duration, activity sequencing, or activity start dates to avoid unacceptable resource peaking.

continued on next page

BP 12.08 Establish Technical Parameters

Establish technical parameters with thresholds for the project and the system.

Description
Establish key technical parameters that can be traced over the life of the project and that will serve as in-progress indicators for meeting the ultimate technical objectives. Key technical parameters can be identified through interaction with the customer, customer requirements, market research, prototypes, identified risks, or historical experience on similar projects. Each technical parameter to be tracked should have a threshold or tolerance beyond which some corrective action would be expected. Key technical parameters should have pre-planned assessments scheduled at useful points in the project schedule.

Typical Work Products
- technical parameters
- technical parameter thresholds

Examples of technical parameters include
- payload capacity of cargo aircraft
- sensor resolution
- portable stereo weight
- automobile gas mileage
- video monitor distortion

Notes
Example: Identify aspects of the system that are primary drivers of system performance. Develop a metric for each aspect that can be tracked over time while the system is being developed.

continued on next page

BP 12.09 Develop Technical Management Plan

Use the information gathered in planning activities to develop technical management plans that will serve as the basis for tracking the salient aspects of the project and the systems engineering effort.

Description
Establish and maintain an integrated management plan that defines project interaction with all internal and external organizations (e.g., the subcontractor) performing the technical effort.

Typical Work Products
- technical management plan

Notes
Technical management plans typically include
- plans for developing the system
- plans for interacting with other organizations (e.g., subcontractors) performing the technical effort

continued on next page

BP 12.10 Review and Approve Project Plans

Review the technical management plans with all affected groups and individuals, and obtain group commitment.

Description

The objective of project plan reviews is to ensure a bottom-up, common understanding of the process, resources, schedule, and information requirements by affected groups and individuals throughout the project. Inputs on the project plan are solicited from all responsible organizational elements and project staff. Whenever possible, these inputs are incorporated to build team ownership of the plans. If an input is rejected or modified, feedback is provided to the individual who gave the input. Interim and completed project plans are distributed for review. A commitment to the project plans should be obtained from all groups comprising the project team.

Typical Work Products

- interface issues between disciplines/groups
- risks
- project plan inputs
- project plan comments
- project plan issues and resolutions

Notes

Affected groups and individuals typically include

- software engineering
- hardware engineering
- manufacturing
- management
- customers
- users
- partners
- subcontractors

Example activity: Identify questions that each group should answer as part of their review. (The questions may be different for different groups.) Communicate to the groups how the review will be conducted. Provide the technical management plans to the groups and, at the pre-arranged time, meet with them to discuss their comments. Produce a list of issues from the reviewers' comments and work on each issue until it is resolved.

End of PA 12: Plan Technical Effort

PA 13: Define Organization's Systems Engineering Process

Summary description

The purpose of Define Organization's Systems Engineering Process is to create and manage the organization's standard systems engineering processes, which can subsequently be tailored by a project to form the unique processes that it will follow in developing its systems or products.

Define Organization's Systems Engineering Process involves defining, collecting, and maintaining the process that will meet the business goals of the organization, as well as designing, developing, and documenting systems-engineering process assets. Assets include example processes, process fragments, process-related documentation, process architectures, process-tailoring rules and tools, and process measurements.

Process area notes

This process area covers the initial activities required to collect and maintain process assets, including the organization's standard systems engineering process. The improvement of the process assets and the organization's standard systems engineering process are covered in the process area Improve Organization's Systems Engineering Processes (PA14).

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.13.01	Establish goals for the organization's systems engineering process from the organization's business goals.
BP.13.02	Collect and maintain systems-engineering process assets.
BP.13.03	Develop a well-defined standard systems engineering process for the organization.
BP.13.04	Define guidelines for tailoring the organization's standard systems engineering process for project use in developing the project's defined process.

continued on next page

BP 13.01 Establish Process Goals

Establish goals for the organization's systems engineering process from the organization's business goals.

Description
The systems engineering process operates in a business context, and this must be explicitly recognized in order to institutionalize the organization's standard practice. The process goals should consider the financial, quality, human resource, and marketing issues important to the success of the business.

Typical Work Products
- goals of the organization's systems engineering process
- requirements for the organization's standard systems engineering process
- requirements for the organization's process asset library
- process asset library

Notes
Establishing goals may include determining the tradeoff criteria for process performance based on time-to-market, quality, and productivity business issues.

continued on next page

BP 13.02 Collect Process Assets

Collect and maintain systems-engineering process assets.

Description

The information generated by the process definition activity, both at the organization and project levels, needs to be stored (e.g., in a process asset library), made accessible to those who are involved in tailoring and process design efforts, and maintained so as to remain current.

Typical Work Products

- instructions for use of a process asset library
- design specifications for a process asset library
- process assets

Notes

The purpose of a process asset library is to store and make available process assets that projects will find useful in defining the process for developing the system. It should contain examples of processes that have been defined, and the measurements of the process. When the organization's standard systems engineering process has been defined, it should be added to the process asset library, along with guidelines for projects to tailor the organization's standard systems engineering process when defining the project's process.

Process assets typically include

- the organization's standard systems engineering process
- the approved or recommended development life cycles
- project processes together with measurements collected during the execution of the processes
- guidelines and criteria for tailoring the organization's standard systems engineering process
- process-related reference documentation
- measurements of the project's process

continued on next page

BP 13.03 Develop Organization's Systems Engineering Process

Develop a well-defined standard systems engineering process for the organization.

Description
The organization's standard systems engineering process is developed using the facilities of the process asset library. New process assets may be necessary during the development task and should be added to the process asset library. The organization's standard systems engineering process should be placed in the process asset library.

Typical Work Products
- organization's standard systems engineering process
- inputs to training
- inputs to systems engineering process improvement

Notes
The standard systems engineering process should include the interfaces to the organization's other defined processes. In addition, references used to define the systems engineering process (e.g., military standards, IEEE standards) should be cited and maintained.

To develop the standard systems engineering process, an organization can identify all the process elements or activities of the organization's system engineering process. The organization must evaluate the process elements for consistency of inputs and outputs, redundant activities, and missing activities. Inconsistencies must be resolved between process elements and provision made for appropriate sequencing and verification features. The resulting process should be well defined.

A well-defined process includes
- readiness criteria
- inputs
- standards and procedures
- verification mechanisms
  - peer reviews
  - outputs
  - completion criteria [SPICE]

continued on next page

BP 13.04 Define Tailoring Guidelines

Define guidelines for tailoring the organization's standard systems engineering process for project use in developing the project's defined process.

Description

Since the organization's standard systems engineering process may not be suitable for every project's situation, guidelines for tailoring it are needed. The guidelines should be designed to fit a variety of situations, while not allowing projects to bypass standards that must be followed or substantial and important practices prescribed by organization policy.

Typical Work Products

- tailoring guidelines for the organization's standard systems engineering process

Notes

Guidelines should enable the organization’s standard systems engineering process to be tailored to address contextual variables such as the domain of the project; the cost, schedule, and quality tradeoffs; the experience of the project's staff; the nature of the customer; the technical difficulty of the project, etc.

End of PA 13: Define Organization's Systems Engineering Process

PA 14: Improve Organization's Systems Engineering Processes

Summary description

The purpose of Improve Organization's Systems Engineering Processes is to gain competitive advantage by continuously improving the effectiveness and efficiency of the systems engineering processes used by the organization. It involves developing an understanding of the organization's processes in the context of the organization's business goals, analyzing the performance of the processes, and explicitly planning and deploying improvements to those processes.

Process area notes

This process area covers the continuing activities to measure and improve the performance of systems engineering processes in the organization. The initial collection of the organization's process assets and the definition of the organization's standard system engineering process is covered in the process area Define Organization's Systems Engineering Process (PA13).

Guidance on improving the standard process may be obtained from several sources, including lessons learned, application of the generic practices, and appraisals of the standard process against the SE-CMM. The resulting profile of capability levels against process areas will point to the most needed areas for improvement. Incorporating the generic practices in these process areas will be useful.

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.14.01	Appraise the existing processes being performed in the organization to understand their strengths and weaknesses.
BP.14.02	Plan improvements to the organization's processes based on analyzing the impact of potential improvements on achieving the goals of the processes.
BP.14.03	Change the organization's standard systems engineering process to reflect targeted improvements.
BP.14.04	Communicate process improvements to existing projects and to other affected groups, as appropriate.

continued on next page

BP 14.01 Appraise the Process

Appraise the existing processes being performed in the organization to understand their strengths and weaknesses.

Description
Understanding the strengths and weaknesses of the processes currently being performed in the organization is a key to establishing a baseline for improvement activities. Measurements of process performance and lessons learned should be considered in the appraisal. Appraisal can occur in many forms, and appraisal methods should be selected to match the culture and needs of the organization.

Typical Work Products
- process maturity profiles
- process performance analyses
- appraisal findings
- gap analyses

Notes
An example appraisal scenario: Appraise the organization's current systems engineering processes using the SE-CMM and its associated appraisal method. Use the results of the appraisal to establish or update process performance goals.

If delays and queues occur in the execution of the existing systems engineering process, then an organization may focus on them as starting points for cycle-time reduction. Recheck such process features as readiness criteria, inputs, and verification mechanisms.

continued on next page

BP 14.02 Plan Process Improvements

Plan improvements to the organization's processes based on analyzing the impact of potential improvements on achieving the goals of the processes.

Description
Appraising the process provides momentum for change. This momentum must be harnessed by planning improvements that will provide the most payback for the organization in relation to its business goals. The improvement plans provide a framework for taking advantage of the momentum gained in appraisal. The planning should include targets for improvement that will lead to high-payoff improvements in the process.

Organizations may take this opportunity to "mistake-proof" the process and eliminate wasted effort. It is important to make the process *stable*– that is, performed consistently by everyone. Deployment is commonly a challenge. In making improvements, be careful to avoid optimizing locally, and thereby creating problems in other areas.

Typical Work Products
- process improvement plan

Notes
Perform tradeoffs on proposed process improvements against estimated returns in cycle time, productivity, and quality. Use the techniques of the Analyze Candidate Solutions process area (PA01).

continued on next page

BP 14.03 Change the Standard Process

Change the organization's standard systems engineering process to reflect targeted improvements.

Description
Improvements to the organization's standard systems engineering process, along with necessary changes to the tailoring guidelines in the process asset library, will preserve the improved process and encourage projects to incorporate the improvements for new products.

Typical Work Products
- organization's standard systems engineering process
- tailoring guidelines for the organization's standard systems engineering process

Notes
As improvements to the standard systems engineering process are implemented and evaluated, the organization should adopt the successful improvements as permanent changes to the standard systems engineering process.

BP 14.04 Communicate Process Improvements

Communicate process improvements to existing projects and to other affected groups, as appropriate.

Description
Some process improvements may be useful to existing projects, and they can incorporate the useful improvements into their current project's process depending upon the status of the project. Others who are responsible for training, quality assurance, measurement, etc., should be informed of the process improvements.

Typical Work Products
- instructions for use of the process asset library
- tailoring guidelines for the organization's standard systems engineering process
- enumeration and rationale for changes made to the systems engineering process
- schedule for incorporating the process changes

Notes
Process improvements, as well as the rationale and expected benefits of the changes, should be communicated to all affected projects and groups. The organization should develop a deployment plan for the updated processes and monitor conformance to that deployment plan.

PA 15: Manage Product Line Evolution

Summary description

The purpose of Manage Product Line Evolution is to introduce services, equipment, and new technology to achieve the optimal benefits in product evolution, cost, schedule, and performance over time as the product line evolves toward its ultimate objectives.

An organization must first determine the evolution of a product. Then the organization has to decide how it will design and build those products including critical components, cost-effective tools, and efficient and effective processes.

Process area notes

The Manage Product Line Evolution process area is needed ". . . to ensure that product development efforts converge to achieve strategic business purposes, and to create and improve the capabilities needed to make research and product development a competitive advantage over the long term." from p. 34 of [Wheelwright 92].

This process area covers the practices associated with managing a product line, but not the engineering of the products themselves.

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.15.01	Define the types of products to be offered.
BP.15.02	Identify new product technologies or enabling infrastructure that will help the organization acquire, develop, and apply technology for competitive advantage.
BP.15.03	Make the necessary changes in the product development cycle to support the development of new products.
BP.15.04	Ensure critical components are available to support planned product evolution.
BP.15.05	Insert new technology into product development, marketing, and manufacturing.

continued on next page

BP 15.01 Define Product Evolution

Define the types of products to be offered.

Description
Define the product lines that support the organization's strategic vision. Consider the organization's strengths and weaknesses, the competition, potential market size, and available technologies.

Typical Work Products
- product line definition

Notes
Defined product lines enable a more effective reuse approach and allow investments with high potential payoff.

BP 15.02 Identify New Product Technologies

Identify new product technologies or enabling infrastructure that will help the organization acquire, develop, and apply technology for competitive advantage.

Description
Identify new product technologies for potential introduction into the product line. Establish and maintain sources and methods for identifying new technology and infrastructure improvements, such as facilities or maintenance services.

Typical Work Products
- reviews of product-line technology
- improvements recommended by process teams

Notes
This practice involves identifying, selecting, evaluating, and pilot testing new technologies. By maintaining an awareness of technology innovations and systematically evaluating and experimenting with them, the organization selects appropriate technologies to improve the quality of its product lines and the productivity of its engineering and manufacturing activities. Pilot efforts are performed to assess new and unproven technologies before they are incorporated into the product line. Infrastructure improvements such as facilities upgrades or enhancements to the service of the distribution chain may also provide opportunities for evolving a product line toward its future objectives.

continued on next page

BP 15.03 Adapt Development Processes

Make the necessary changes in the product development cycle to support the development of new products.

Description
Adapt the organization's product development processes to take advantage of components intended for future use.

Typical Work Products
- adapted development processes

Notes
This practice can include establishing a library of reusable components, which includes the mechanisms for identifying and retrieving components.

BP 15.04 Ensure Critical Component Availability

Ensure critical components are available to support planned product evolution.

Description
The organization must determine the critical components of the product line and plan for their availability.

Typical Work Products
- product-line components

Notes
The availability of critical components can be ensured by incorporating considerations for the future use of these components into the product line requirements. Appropriate resources must be allocated by the organization to maintain the components on a continuous basis.

continued on next page

BP 15.05 Insert Product Technology

Insert new technology into product development, marketing, and manufacturing.

Description
Manage the introduction of new technology into the product lines, including both modifications of existing product-line components and the introduction of new components. Identify and manage risks associated with product design changes.

Typical Work Products
- new product-line definition

Notes
The objective of this practice is to improve product quality, increase productivity, decrease life-cycle cost, and decrease the cycle time for product development.

End of PA 15: Manage Product Line Evolution

PA 16: Manage Systems Engineering Support Environment

Summary description

The purpose of Manage Systems Engineering Support Environment is to provide the technology environment needed to develop the product and perform the process. Development and process technology is inserted into the environment with a goal of minimizing disruption of development activities while upgrading to make new technology available.

The technology needs of an organization change over time, and the efforts described in this process area must be re-executed as the needs evolve.

Process area notes

This process area addresses issues pertaining to the systems engineering support environment at both a project level and at an organizational level. The elements of a support environment consist of all the surroundings of the systems engineering activities, including

- computing resources
- communications channels
- analysis methods
- the organization's structures, policies and procedures
- machine shops
- chemical process facilities
- environment stress facilities
- systems engineering simulation tools
- software productivity tools
- proprietary systems engineering tools
- work space

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.16.01	Maintain awareness of the technologies that support the organization's goals.
BP.16.02	Determine requirements for the organization's systems engineering support environment based on organizational needs.
BP.16.03	Obtain a systems engineering support environment that meets the requirements established in Determine Support Requirements by using the practices in the Analyze Candidate Solutions process area.
BP.16.04	Tailor the systems engineering support environment to individual project's needs.
BP.16.05	Insert new technologies into the systems engineering support environment based on the organization's business goals and the projects' needs.
BP.16.06	Maintain the systems engineering support environment to continuously support the projects dependent on it.
BP.16.07	Monitor the systems engineering support environment for improvement opportunities.

continued on next page

BP 16.01 Maintain Technical Awareness

Maintain awareness of the technologies that support the organization's goals.

Description
Awareness of the current state of the art or state of the practice is a necessary element for assessing improvement options. Therefore, to insert new technology, a sufficient awareness of new technology must be present in the organization. Such awareness may be maintained internally or acquired.

Typical Work Products
- reviews of support environment technology

Notes
Maintaining awareness may be accomplished by reading industry journals, participating in professional societies, and establishing and maintaining a technical library.

BP 16.02 Determine Support Requirements

Determine requirements for the organization's systems engineering support environment based on organizational needs.

Description
An organization's needs are primarily determined by assessing competitiveness issues. For example, does the organization's support environment hinder the organization's competitive position? Does each major element of the organization's support environment allow systems engineering to operate with sufficient speed and accuracy?

Typical Work Products
- requirements for systems engineering support environment

Notes
Determine the organization's needs for computer network performance, improved analysis methods, computer software, and process restructuring.

continued on next page

BP 16.03 Obtain Systems Engineering Support Environment

Obtain a systems engineering support environment that meets the requirements established in Determine Support Requirements by using the practices in the Analyze Candidate Solutions process area.

Description
Determine the evaluation criteria and potential candidate solutions for the needed systems engineering support environment. Then, select a solution using the practices in the Analyze Candidate Solutions process area (PA01). Finally, obtain and implement the chosen systems engineering support environment.

Typical Work Products
- systems engineering support environment

Notes
The systems engineering support environment may include many of the following: software productivity tools, tools for simulating systems engineering, proprietary in-house tools, customized commercially available tools, special test equipment, and new facilities.

BP 16.04 Tailor Systems Engineering Support Environment

Tailor the systems engineering support environment to individual project's needs.

Description
The total support environment represents the needs of the organization as a whole. An individual project, however, may have unique needs for selected elements of this environment. In this case, tailoring the elements of the systems engineering support environment elements can allow the project to operate more efficiently.

Typical Work Products
- tailored systems engineering support environment

Notes
Tailoring allows an individual project to customize its systems engineering support environment. For example, project A does not involve signal processing, so signal processing automation tools are tailored out of (i.e., not provided to) this project's automation tool set. Conversely, project B is the only project in the organization that has a need for automated requirements tracing, so the appropriate tools are tailored into (i.e., provided in addition to) this project's automated tool set.

continued on next page

BP 16.05 Insert New Technology

Insert new technologies into the systems engineering support environment based on the organization's business goals and the projects' needs.

Description
The organization's systems engineering support environment must be updated with new technologies as they emerge and are found to support the organization's business goals and the projects' needs.

Training in the use of the new technology in the systems engineering support environment must be provided.

Typical Work Products
- new systems engineering support environment

Notes
Inserting new technologies into the organization's support environment presents several difficulties. To minimize these difficulties, follow the steps below:

1. Test the new technology thoroughly.
2. Decide whether to insert the improvement across the entire organization or in selected portions of the organization.
3. Provide early notification of the impending change to those who will be affected.
4. Provide any necessary "how to use" training for the new technology.
5. Monitor the acceptance of the new technology.

continued on next page

BP 16.06 Maintain Environment

Maintain the systems engineering support environment to continuously support the projects dependent on it.

Description
Maintain the systems engineering support environment at a level of performance consistent with its expected performance. Maintenance activities could include computer system administration, training, hotline support, availability of experts, evolving/expanding a technical library, etc.

Typical Work Products
- performance report for the systems engineering support environment

Notes
Maintenance of the systems engineering support environment could be accomplished several ways, including
- hire or train computer system administrators
- develop expert users for selected automation tools
- develop methodology experts who can be used on a variety of projects
- develop process experts who can be used on a variety of projects

BP 16.07 Monitor Systems Engineering Support Environment

Monitor the systems engineering support environment for improvement opportunities.

Description
Determine the factors that influence the usefulness of the systems engineering support environment, including any newly inserted technology. Monitor the acceptance of the new technology and of the entire systems engineering support environment.

Typical Work Products
- reviews of the technology used in the systems engineering support environment

Notes
Design some monitoring to be an automated, background activity, so that users of the support environment do not need to provide data consciously. Also provide a way for users of the systems engineering support environment to consciously provide inputs on the usefulness of the current systems engineering support environment and to suggest improvements.

End of PA 16: Manage Systems Engineering Support Environment

PA 17: Provide Ongoing Skills and Knowledge

Summary description

The purpose of Provide Ongoing Skills and Knowledge is to ensure that projects and the organization have the necessary knowledge and skills to achieve project and organizational objectives. To ensure the effective application of these critical resources that are predominantly available only from people, the knowledge and skill requirements within the organization need to be identified, as well as the specific project's or organization's needs (such as those relating to emergent programs or technology, and new products, processes, and policies).

Needed skills and knowledge can be provided both by training within the organization and by timely acquisition from sources external to the organization. Acquisition from external sources may include customer resources, temporary hires, new hires, consultants, and subcontractors. In addition, knowledge may be acquired from subject matter experts.

Process area notes

The choice of training or external sourcing for the need skill and knowledge is often determined by the availability of training expertise, the project's schedule, and business goals. Successful training programs result from an organization's commitment. In addition, they are administered in a manner that optimizes the learning process, and that is repeatable, assessable, and easily changeable to meet new needs of the organization. Training is not limited to "classroom" events: it includes the many vehicles that support the enhancement of skills and the building of knowledge. When training is not a viable approach due to schedule or availability of training resources, external sources of the needed skills and knowledge are pursued.

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.17.01	Identify needed improvements in skill and knowledge throughout the organization using the projects' needs, organizational strategic plan, and existing employee skills as guidance.
BP.17.02	Evaluate and select the appropriate mode of acquiring knowledge or skills with respect to training or other sources.
BP.17.03	Ensure that appropriate skill and knowledge are available to the systems engineering effort.
BP.17.04	Prepare training materials based upon the identified training needs.
BP.17.05	Train personnel to have the skills and knowledge needed to perform their assigned roles.
BP.17.06	Assess the effectiveness of the training to meet the identified training needs.
BP.17.07	Maintain records of training and experience.
BP.17.08	Maintain training materials in an accessible repository.

continued on next page

BP 17.01 Identify Training Needs

Identify needed improvements in skill and knowledge throughout the organization using the projects' needs, organizational strategic plan, and existing employee skills as guidance.

Description
This base practice determines the improvements that are needed in skill and knowledge within the organization. The needs are determined using inputs from existing programs, the organizational strategic plan, and a compilation of existing employee skills. Project inputs help to identify existing deficiencies which may be remedied through training or acquisition of skills and knowledge by other means. The organizational strategic plan is used to help identify emerging technologies, and the existing skill level is used to assess current capability.

Identification of skill and knowledge needs should also determine training that can be consolidated to achieve efficiencies of scale, and increase communication via the use of common tools within the organization. Training should be offered in the organization's systems engineering process and in tailoring the process for specific projects.

Typical Work Products
- organization’s training needs
- project skill or knowledge

Notes
The organization should identify additional training needs as determined from appraisal findings and as identified by the defect prevention process. The organization's training plan should be developed and revised according to a documented procedure. Each project should develop and maintain a training plan that specifies its training needs.

continued on next page

BP 17.02 Select Mode of Knowledge or Skill Acquisition

Evaluate and select the appropriate mode of acquiring knowledge or skills with respect to training or other sources.

Description

The purpose of this practice is to ensure that the most effective method is chosen to make needed skill and knowledge available to projects in a timely manner. Project and organizational needs are analyzed, and the methods of the Analyze Candidate Solutions process area (PA01) are employed to choose among alternatives such as consultants, subcontracts, knowledge acquisition from identified subject matter experts, or training.

Typical Work Products

- survey of needed skills or knowledge
- trade-study results indicating the most effective mode of skill or knowledge acquisition

Notes

Example criteria which may be used to determine the most effective mode of acquiring knowledge or skill acquisition include

- time available to prepare for project execution
- business objectives
- availability of in-house expertise
- availability of training

continued on next page

BP 17.03 Assure Availability of Skill and Knowledge

Ensure that appropriate skill and knowledge are available to the systems engineering effort.

Description
This practice addresses acquisition of the full range of skill and knowledge which must be made available to the project systems engineering effort. Through deliberate assessment and preparation, plans can be developed and executed to make available the range of required knowledge and skills, including functional engineering skills, application problem-domain knowledge, interpersonal skills, multidisciplinary skills, and process-related skills. After the needed skills have been identified, evaluations of the appropriate mode of knowledge or skill acquisition can be used to select the most effective approach.

Typical Work Products
- assessment of skill types needed by skill category
- project knowledge acquisition plan
- training plan
- list of identified and available subject matter experts

Notes
Appropriate coverage of the full range of skill and knowledge types can be addressed with a checklist of knowledge types (e.g., functional engineering, problem domain, etc.) against each element of the work breakdown structure.

An example of ensuring the availability of the appropriate application-problem domain knowledge (e.g., satellite weather data processing), would be a plan to interview identified subject matter experts in connection with requirements interpretation or system design. Such an approach would be appropriate when an organization does not have the required expertise available (as with the first program in a new line of business).

continued on next page

BP 17.04 Prepare Training Materials

Prepare training materials based upon the identified training needs.

Description
Develop the training material for each class that is being developed and facilitated by people within the organization, or obtain the training material for each class that is being procured.

Typical Work Products
- course descriptions and requirements
- training material

Notes
Course description should include
- intended audience
- preparation for participation
- training objective
- length of training
- lesson plans
- criteria for determining the students' satisfactory completion

Prepare
- procedures for periodically evaluating the effectiveness of the training and special considerations, such as piloting and field testing the training course
- needs for refresher training, and opportunities for follow-up training
- materials for training a specific practice to be used as part of the process (e.g., method technique)
- materials for training a process
- materials for training in process skills such as statistical techniques, statistical process control, quality tools and techniques, descriptive process modeling, process definition, and process measurement

Review the training material with some or all of the following instructional experts, subject matters experts, and students from the pilot programs.

continued on next page

BP 17.05 Train Personnel

Train personnel to have the skills and knowledge needed to perform their assigned roles.

Description
Personnel are trained in accordance with the training plan and developed material.

Typical Work Products
- trained personnel

Notes
Offer the training in a timely manner (just-in-time training) to ensure optimal retention and the highest possible skill level.
- A procedure should exist to determine the skill level of the employee prior to receiving the training to determine if the training is appropriate (i.e., if a trainer waiver or equivalent should be administered to the employee).
- A process exists to provide incentives and motivate the students to participate in the training.
- Online training/customized instruction modules accommodate different learning styles and cultures, in addition to transferring smaller units of knowledge.

BP 17.06 Assess Training Effectiveness

Assess the effectiveness of the training to meet the identified training needs.

Description
A key aspect of training is determining its effectiveness. Methods of evaluating effectiveness need to be addressed concurrent with the development of the training plan and training material; in some cases, these methods need to be an integral part of the training material. The results of the effectiveness assessment must be reported in a timely manner so that adjustments can be made to the training.

Typical Work Products
- analysis of training effectiveness
- modification to training

Notes
A procedure should exist to determine the skill level of the employee after receiving the training to determine the success of the training. This could be accomplished via formal testing, on-the-job skills demonstration, or assessment mechanisms embedded in the courseware.

continued on next page

BP 17.07 Maintain Training Records

Maintain records of training and experience.

Description
Records are maintained to track the training that each employee has received and the employee's skills and capabilities.

Typical Work Products
- training and experience records

Notes
Records are kept of all students who successfully complete each training course or other approved training activity. Also, records of successfully completed training are made available for consideration in the assignment of the staff and managers.

BP 17.08 Maintain Training Materials

Maintain training materials in an accessible repository.

Description
Courseware material is maintained in a repository for future access by employees and for maintaining traceability in changes in course material.

Typical Work Products
- baselined training materials
- revisions to training materials

Notes
Maintain a repository of training materials and make it available to all employees. (For example, the organization's library could make books, notebooks, videotapes, etc., available; soft-copy training materials could be maintained in a public file server.) Incorporate lessons learned into process training materials and the training program. Update process training materials with all process changes and improvements.

End of PA 17: Provide Ongoing Skills and Knowledge

PA 18: Coordinate with Suppliers

Summary description

The purpose of Coordinate with Suppliers is to address the needs of organizations to effectively manage the portions of product work that are conducted by other organizations. Decisions made as a part of this process area should be made in accordance with the Analyze Candidate Solutions process area (PA01). The general term *supplier* is used to identify an organization that develops, manufactures, tests, supports, etc., a component of the system. Suppliers may take the form of vendors, subcontractors, partnerships, etc., as the business organization warrants.

In addition to coordination of schedules, processes, and deliveries of work products, affected organizations must have a shared a vision of the working relationship. Relationships can range from integrated developer / supplier product teams, to prime-contractor / subcontractor, to vendors, and more. A successful relationship between an organization and a supplier depends on the capability of both organizations, and on a mutual understanding of the relationship and expectations.

Process area notes

When suppliers deliver products that do not meet an organization's needs, the organization has the option to change to another supplier, lower its standards and accept the delivered products, or help the supplier or vendor meet the organization's needs.

The organization acts as the customer when the supplier executes the Understand Customer Needs and Expectations process area (PA06). The organization should help the supplier to achieve full understanding. If the supplier does not have the processes to execute this process area, the organization should coach the supplier in getting the necessary information.

Base practices list

The following list contains the base practices that are essential elements of good systems engineering:

BP.18.01	Identify needed system components or services that must be provided by other/outside organizations.
BP.18.02	Identify suppliers that have shown expertise in the identified areas.
BP.18.03	Choose suppliers in accordance with the Analyze Candidate Solutions process area (PA01).
BP.18.04	Provide to suppliers the needs, expectations, and measures of effectiveness held by the organization for the system components or services that are to be delivered.
BP.18.05	Maintain timely two-way communication with suppliers.

continued on next page

BP 18.01 Identify Systems Components or Services

Identify needed system components or services that must be provided by other/outside organizations.

Description
Rarely does an organization make every component of the system. Make-vs.-buy analyses and decisions determine which items will be procured. System needs that will be satisfied outside the organization are generally those in which the organization has little expertise or interest.

Typical Work Products
- make-vs.-buy trade study
- list of system components
- sub set of system components for outside organizations to address
- list of potential suppliers
- beginnings of criteria for completion of needed work

Notes
Example practices include
- Perform trade study.
- Examine own organization to determine missing expertise needed to address system requirements.

BP 18.02 Identify Competent Suppliers or Vendors

Identify suppliers that have shown expertise in the identified areas.

Description
The capabilities of the supplier should be complementary and compatible with those of the organization. Issues that may be of concern include competent development processes, manufacturing processes, responsibilities for verification, on-time delivery, life-cycle support processes, and ability to communicate effectively over long distances (video teleconferencing, electronic file transfers, e-mail and the like).

Typical Work Products
- list of suppliers
- advantages and disadvantages of each supplier
- potential ways of working over physical distances with suppliers

Notes
Example practices include
- Read trade journals.
- Use available library services.
- Use organizational knowledge-base (perhaps an online system).

continued on next page

BP 18.03 Choose Supplier or Vendors

Choose suppliers in accordance with the Analyze Candidate Solutions process area (PA01).

Description
Suppliers are selected in a logical and equitable manner to meet product objectives. The characteristics of a supplier which would best complement the organization's abilities are determined, and qualified candidates are identified. The practices of the Analyze Candidate Solutions process area (PA01) are applied to select the appropriate supplier.

Typical Work Products
- organization weaknesses which might be mitigated by a supplier
- characteristics of the desired working relationships with the supplier
- supplier requirements
- customer requirements to be "flowed down" to supplier
- selected supplier
- captured rationale for selected supplier

Notes
An important consideration in the selection of the supplier is the expected working relationship. This could range from a highly integrated product team to a classical "meet the requirements" relationship. The selection criteria are likely different, depending of the desired relationship.

continued on next page

BP 18.04 Provide Expectations

Provide to suppliers the needs, expectations, and measures of effectiveness held by the developing organization for the system components or services that are to be delivered.

Description
The contracting organization must clearly identify and prioritize its needs and expectations, as well as any limitations on the part of the suppliers. The organization works closely with suppliers to achieve a mutual understanding of product requirements, responsibilities, and processes which will be applied to achieve program objectives.

Typical Work Products
- needs statement
- technical performance parameters
- verification specifications

Notes
Examples of techniques and forums for providing needs, expectations, and measures of effectiveness to suppliers or vendors include
- trade studies
- formal contracts
- in-process reviews
- joint meetings
- payment milestones

continued on next page

BP 18.05 Maintain Communications

Maintain timely two-way communications with suppliers.

Description
The organization and supplier establish a mutual understanding of expected and needed communications. Characteristics of communications that are established include the types of information that are considered open and subject to no restrictions, the types of information subject to restrictions (e.g., policy or contractual relationships), the expected timeliness of information requests and responses, tools and methods to be used for communications, security, privacy, and distribution expectations. The need for "face-to-face" versus "at-a-distance" communications, and the need and mechanism for archiving communications are also considered.

Typical Work Products
- contractually required communication
- communications tools
- communications plans
- communications distribution lists

Notes
An effective communications environment between the organization and supplier is highly desirable. E-mail and voice-mail tools are effective for simple communications where two-way communication is not required.

Communications that affect schedule cost or scope should be restricted to authorized parties.

End of PA 18: Coordinate with Suppliers

Appendices

Introduction

The appendices contain information of interest to specific target audiences, or supplemental information which might prove distracting to the overall flow of the model description were it included in the main body of the document.

In this section

Topic	See Page
Appendix A: Change History and Change Request Form	A-3
Appendix B: Approved Model Requirements	A-7
Appendix C: References	A-21
Appendix D: Systems Engineering Glossary	A-25

Appendix A: Change History and Change Request Form

Introduction

This appendix contains the change history for the SE-CMM and a change request form. Significant changes in focus or content from one release to another are highlighted.

Change History Table

Table A-1 provides the change history for the SE-CMM:

Version Designator	Content	Change Notes
Release 1	• architecture rationale • Process Areas • ISO (SPICE) BPG 0.05 summary • Glossary	
Release 2 Workshop Version	• Executive Summary • Overview of the SE-CMM • Using the SE-CMM • Process Areas • BPG 0.06 with SE-CMM notes • Model Requirements • Appendices	• Front matter, overview added • PA descriptions, boundaries and base practices revised based on Workshop #1 comments
Release 2.02	• Same as release 2 Workshop version	• Many TBS's (to be supplied) filled in

Table A-1. Change History Table

continued on next page

Change History Table, continued

Version Designator	Content	Change Notes
Release 2.03	• Same as 2.02 minus Appendix E and F, which were pulled out and now constitute SECMM-94-09 (CMU/SEI-94-TR-26)	• TBS's filled in • Author review comments incorporated • Workshop #2 comments completed • Early key reviewer comments incorporated
Release 2.04	• Same as 2.03 minus App A (Practices Summary was moved to an Appendix of SECMM-94-06 CMU/SEI-94-HB-05) • PAs 4 and 10 were substantially rewritten and enhanced	• TBS's filled in • Pilot appraisal comments/lessons learned incorporated • Key reviewer comments incorporated
v1.0	• Official release for public review, use, and comment • Same contents as 2.04 plus requirements traceability table	• Chs 1-3 reorganized and edited for readability, flow • BP 10.07 deleted (was supposed to be deleted in v2.04) • BP 12.02 “historically proven” clause removed • Technical editor comments incorporated

Table A-1. Change History Table

continued on next page

Change History Table, continued

Version Designator	Content	Change Notes
v1.1	Version 1.0 content plus • process area addressing suppliers • extension of model to include production and operations life-cycle phases	• Model Overview enhanced for readability • Improvements to process area descriptions • Emphasis on production and operation impacts to development practices

Table A-1. Change History Table, continued

Issues Form for SECMM-95-01 Version 1.1

Reviewer Information Please provide your name and organizational affiliation.

Reviewer Name	Reviewer Orgn	Contact Phone #

If using hardcopy, you may attach several forms together with the name on just the first one.

Issue Reference

Please list the page #(s) or other reference (e.g., "global," "Chapter 3," "Glossary") to which this issue applies. Attach the page for reference if appropriate.

Issue Statement

Please characterize the issue as a *problem* (e.g., the glossary is not detailed enough to support...) vs. a *solution* (e.g., add more detail to the glossary), so that the authors can understand the cause of the issue, not just the suggested fix. Include your rationale for highlighting the issue, if appropriate.

Prioritization

This issue is _____ out of my top 10 issues with the SE-CMM Version 1.1.

Impact Assessment

Please evaluate the impact the stated problem has on your use of the SE-CMM according to this scale:

____ **High Impact**: can't use model as intended without problem being fixed.
____ **Medium Impact:** misleading or otherwise incorrect content of significance to the reviewer.
____ **Low Impact:** content error of low significance to reviewer.

Note: editorial issues

For typographical/grammatical/punctuation edits, please forward the redlined pages without the issue form attachment.

Appendix B: Approved Model Requirements

Introduction

This appendix consists of the requirements document for the SE-CMM that was approved by the SE-CMM Steering Group for v1.0 of the SE-CMM.

Requirements traceability

The requirements traceability matrix for this product is included at the end of this appendix.

Requirements changes

Requests for requirements changes may be submitted directly to a member of the SE-CMM Steering Group or to the SE-CMM Project Office for consideration. An "issues form" is included at the back of the SE-CMM. The SE-CMM Steering Group is the approval authority for any requirements changes.

As a result of the meeting held in October 1994, the following requirements changes were approved. The new requirement is what appears in this version of the model.

- Requirement 5.3.5.2.2 was deleted (example practices).
- Requirements 5.3.4 and 6.2.1.2 were deleted as requirements of the model. However, they are the guiding requirements for a new document approved by the steering group, SECMM-94-09 (CMU/SEI-94-HB-06), *Relationships Between the SE-CMM and Other Products.*
- Requirement 6.1.2 was modified to permit v1.0 to cover only the product development portion of the product life cycle.

continued on next page

1.0 Document Overview

1.1. Introduction

A fundamental assumption of maturity models is that the quality of a product depends upon the process used for development, the technology and tools used in development, and the capabilities of the people who do the work. The CMM for Software primarily covered the process for development, although aspects of people, facility and training issues were also covered to a certain extent. Eventually the SE-CMM should cover all three areas thoroughly. However, the initial version of the SE-CMM will only have coverage of non-process issues similar to that in the CMM for Software.

Approach

To have merit, a validated appraisal methodology must be used in conjunction with a representative model in order to effectively measure the capability and maturity of a systems engineering project or organization. This document identifies the requirements that one half of that methodology, a Systems Engineering-Capability Maturity Model (SE-CMM), must meet.

Growth

The quality of a product is a direct function of the process, technology, and tools used and the capability of the people assigned to do the work. The SE-CMM Project recognizes and supports the validity and interconnectivity of that assumption. However, the initial efforts of the project have been focused on modeling the characteristics of processes used to implement and institutionalize sound systems engineering practices within an organization. Until a follow-on activity expands the SE-CMM to fully address the technology, tools, and people elements cited, a sense of their impact will be captured by using "base practices" which address primarily process-related elements, but will overlap, in some cases, into non-process areas.

continued on next page

1.2. Requirements terminology

In the following sections, the term 'will' indicates a mandatory requirement. The usage of "will" in this document corresponds to the use of the term "shall" in Government requirements.

Elements which are not mandatory, but which have sufficient merit to warrant that the Project include them to the extent possible, are identified by the term "should."

1.3. Scope of this document

Section 2.0 outlines the overall Project goal. With that exception, this document is strictly limited to requirements imposed on the model portion of the SE-CMM Project. Information on the appraisal portion can be found in a separate document titled, *SE-CMM Appraisal Method Description* (SE-CMM-94-06).

2.0 Goal

2.1 Model and appraisal method

The overall goal of the SE-CMM Project is to provide a Systems Engineering Capability Maturity Model and appraisal methodology that:

1) Supports industry-wide improvement of systems engineering activities, and
2) Provides an accepted frame of reference for the appraisal of an organization's systems engineering capabilities.

3.0 Objectives

Introduction

In support of the Project goals, the model should seek to achieve the following objectives.

3.1. Industry acceptance

The SE-CMM should seek to obtain and maintain acceptance of the model by both industry and government organizations.

continued on next page

3.2. Compatibility	The SE-CMM should seek to avoid conflict with existing and emerging standards and initiatives (e.g., ISO 9001, draft Mil-Std-499B). In this context, "avoid conflict" means that the SE-CMM should not knowingly encourage activities or provide process guidance which contradicts appropriate emerging standards.

4.0 Scope of the Model

4.1 Focus	The SE-CMM will focus on the systems engineering processes executed by systems engineering practitioners and managers. Support areas will be considered where necessary.
4.2 Applicability	The SE-CMM will be applicable to a generalized, rather than a specifically instantiated, process.
4.3 Incremental development	
4.3.1 Initial version	Version 1.0 of the SE-CMM will focus on process capability improvement and assessment.
4.3.2 Growth	Subsequent versions of the SE-CMM will evolve and refine process coverage, based on field experience, and expand the ability of the model to assess additional dimensions of a project or organization's capability and maturity, such as human resource capacity and the effectiveness of available tools.
4.4 Depth of coverage	The Model will address systems engineering down to, but not including, the various implementation disciplines (e.g., hardware, firmware, and software development).

continued on next page

4.5 Applicability

4.5.1 Number of projects

The SE-CMM will be applicable regardless of the number (one, or more than one) of projects being implemented by a systems engineering organization.

4.5.2 Scaling, or size

The SE-CMM will be applicable to the assessment or evaluation of a systems engineering organization, regardless of size.

5.0 Model Description

Purpose

This section describes the content of a specific Project Product/Deliverable titled, *SE-CMM Model Description* (SECMM-94-04). The names of the sections of the document shown here may change in the final document to improve its readability.

5.1 Executive summary

This section will contain a brief overview of the model, its history and purpose, advantages, and constraints coupled with a brief, basic outline of how the document is constructed and how topics are linked.

5.2 Introduction

This section will formally introduce the reader to the document. It will contain a brief history of the Project, a short discussion of how the Project is organized, and an outline of future plans. Project work products (and their content) will be identified and their relationship to the model described.

5.3 Model description

This section describes the model in detail. It will contain, as a minimum, the following elements.

continued on next page

5.3.1 Applicability	In this section, a brief description of the scope of the model and its intended audience will be provided.
5.3.2 Architecture	A detailed description of model components will be provided. Relationships and interactions between and among the various components of the model will be shown. Constraints and cautions, if any, will also be provided in this section.
5.3.3 Interaction with similar maturity models	<deleted per Steering Group 10/12 - moved to SECMM-94-09> (CMU/SEI-94-HB-06)
5.3.4 SE-CMM practices	The term "practices" will, with specific adjectives, designate those characteristics which are considered essential and those which provide an advisory function.
5.3.4.1 Practice dependencies	Following are general characteristics applicable to all practices.
5.3.4.1.1 Organization dependencies	Practices will be organizationally independent.
5.3.4.1.2 Product dependencies	Practices will be product independent.

continued on next page

5.3.4.2 Base practices	The model will identify, as a minimum, a set of specific tasks which must be accomplished in order to achieve a satisfactory systems engineering outcome. These tasks will be identified as "Base Practices" and grouped according to the specific Process Area with which they are associated.
5.3.4.2.1 Usage/ interpretation guidelines	A description of each Base Practice will be provided which should describe the practice, provide interpretation guidelines, clearly identify the intended usage, and show how the practice interacts with others.
5.4 Glossary	A glossary of all systems engineering terms used in the SE-CMM will be provided as an appendix.
5.5 Appendix	Subsequent appendices will be provided on an as needed basis.

6.0 Constraints

6.1 Model characteristics	
6.1.1 Management characteristics	The SE-CMM will include practices to identify good system engineering management characteristics. Overall program/project management techniques should be considered only to the extent they impact systems engineering task execution.

continued on next page

6.1.2 Life-cycle coverage	The SE-CMM will eventually address planning and performance over the entire range of systems engineering activities throughout the complete systems engineering life cycle. Version 1.0 covers the product development cycle only.
6.1.3 Structure	The SE-CMM will be structured so the decomposition of each level downward is readily apparent and traceable either from top down, or bottom up.
6.1.4 Functionality	The SE-CMM will be functionally decomposed into areas directly relatable to management, process designers, and practitioners.
6.2. Relationships to other capability/ maturity models	
6.2.1 CMM for software	<requirement moved to SECMM-94-09 (CMU/SEI-94-HB-06)>
6.2.1.1 Terminology	<requirement moved to SECMM-94-09 (CMU/SEI-94-HB-06)>
6.2.1.2 Interfaces	The SE-CMM should be easily relatable to the CMM for Software.

continued on next page

7.0 Validation

Model validation will be in two phases. Initial validation will be through use of pilot appraisals. Final validation will be through industry/government acceptance, based on field experience.

7.1. Pilot appraisals

Initial validation will be through pilot appraisals conducted at a minimum of two separate organizations. If validation is accomplished using only two appraisals, the organizations will be of diverse size and product focus. Additional appraisals should be accomplished at every opportunity.

As part of the validation, an ad hoc, independently derived assessment should be made of the organization being evaluated and the results compared to those produced by the SE-CMM. Any discrepancies should be noted and the rationale for the differences should be determined.

7.1.1 Pilot diversity

The SE-CMM pilot appraisals should seek maximum diversity in applicability.

7.1.1.1 Maturity

The SE-CMM should be used as the basis for appraising at least one project or organization perceived to have a mature process capability.

7.1.1.2 Focus

The SE-CMM should be used as the basis for appraising at least one project or organization with a contract-driven product environment and at least one organization with a market-driven product development environment.

continued on next page

Derivation and Traceability of SE-CMM Requirements

Instruction

The requirements herein contained were produced using material garnered from project participants as recorded in the documents listed below. A specific listing of author's meetings and copies of the minutes are available, upon request. Following the sources list is a traceability matrix of SE-CMM requirements to the sections of the model that generally cover the requirement.

Sources list

1. Minutes, Potential Project Participants Meeting, September 27, 1993
2. NCOSE Request for Information on Capability Assessments
3. Minutes, SE-CMM Steering Group Meeting, January 27, 1994
4. Minutes, several SE-CMM Authors Meetings
5. Minutes, October 10-12, 1994 Steering Group Meeting

continued on next page

Traceability Matrix

Req. Number	Requirement Name	Text Location
1.0	**Document Overview**	N/A
1.1	Introduction	N/A
1.2	Requirements Terminology	Appendix B
1.3	Scope of This Document	1.1 About this Document, SECMM-94-06 (CMU/SEI-94-HB-05)
2.0	**Goal**	N/A
2.1	Model and Appraisal Method	Throughout
3.0	**Objectives**	N/A
3.1	Industry Acceptance	1.2 About the SE-CMM Project
3.2	Compatibility	Chapter 4: The SE-CMM Generic & Base Practices SECMM-94-09 (CMU/SEI-94-HB-06)
4.0	**Scope of Model**	N/A
4.1	Focus	Chapter 4: The SE-CMM Generic & Base Practices
4.2	Applicability	Chapter 4: The SE-CMM Generic & Base Practices 2.3 SE-CMM Architecture Description
4.3	Incremental Development	N/A
4.3.1	Initial Version	2.1 SE-CMM Foundations

Table A-2. Traceability Matrix

continued on next page

Traceability Matrix, continued

Req. Number	Requirement Name	Text Location
4.3.2	Growth	2.1 SE-CMM Foundations
4.4	Depth of Coverage	2.1 SE-CMM Foundations
4.5	Applicability	N/A
4.5.1	Number of Projects	2.2 Key Concepts of the SE-CMM
4.5.2	Scaling, or Size	3.2 Many Usage Contexts
5.0	**Model Description**	**N/A**
5.1	Executive Summary	To the Reader
5.2	Introduction	Chapter 1: Introduction
5.3	Model Description	N/A
5.3.1	Applicability	To the Reader Chapter 1: Introduction 2.1 SE-CMM Foundations
5.3.2	Architecture	Ch 2: Overview of SE-CMM Architecture
5.3.3	Interaction with Similar Maturity Models	moved to SECMM-94-09 (CMU/SEI-94-HB-06)
5.3.4	SE-CMM Practices	Chapter 4: The SE-CMM Generic & Base Practices
5.3.4.1	Practice Dependencies	N/A
5.3.4.1.1	Organization Dependencies	Chapter 3: Using the SE-CMM Chapter 4: The SE-CMM Generic & Base Practices
5.3.4.1.2	Product Dependencies	Chapter 3: Using the SE-CMM Chapter 4: The SE-CMM Generic & Base Practices
5.3.4.2	Base Practices	Chapter 4: The SE-CMM Generic & Base Practices

Table A-2. Traceability Matrix, continued

continued on next page

Traceability Matrix, continued

Req. Number	Requirement Name	Text Location
5.3.4.2.1	Usage/Interpretation Guidelines	Chapter 4: The SE-CMM Generic & Base Practices
5.4	Glossary	Appendix D: Glossary
5.5	Appendix	Appendices A-C
6.0	**Constraints**	**N/A**
6.1	Model Characteristics	N/A
6.1.1	Management Characteristics	2.1 SE-CMM Foundations Chapter 4: The SE-CMM Generic & Base Practices
6.1.2	Life-cycle coverage	2.1 SE-CMM Foundations Chapter 4: The SE-CMM Generic & Base Practices
6.1.3	Structure	2.3 SE-CMM Architecture Description Ch. 4: The SE-CMM Generic & Base Practices
6.1.4	Functionality	Chapter 4: The SE-CMM Generic & Base Practices
6.2	Relationships to Other CMMs	N/A
6.2.1	CMM for Software	N/A
6.2.1.1	Terminology	Whole document
6.2.1.2	Interfaces	SECMM-94-09 (CMU/SEI-94-HB-06)
7.0	**Validation**	**N/A**
7.1	Pilot Appraisals	See SE-CMM Pilot Appraisal Report
7.1.1	Pilot Diversity	See SE-CMM Pilot Appraisal Report
7.1.1.1	Maturity	See SE-CMM Pilot Appraisal Report
7.1.1.2	Focus	See SE-CMM Pilot Appraisal Report

Table A-2. Traceability Matrix, continued

Appendix C: References

Introduction

This appendix provides the references for documents cited within the SE-CMM, as well as selected bibliographic sources for concepts.

Reference List

[AFMC] *AF 800-Software Development Capability Evaluation* (SDCE).

[Blanchard 81] Blanchard, Benjamin S.; & Fabrycky, Walter J. *Systems Engineering and Analysis.* Englewood Cliffs, N.J.: Prentice-Hall, 1981.

[Charette 89] Charette, Robert N. *Software Engineering, Risk Analysis and Management.* New York: Intertext Publications, McGraw-Hill, 1989.

[Chestnut 67] Chestnut, Harold. *Systems Engineering Methods.* New York, NY: John Wiley & Sons, 1967.

[Crab 93] Crab, Don. “The New PCs.” *PC Magazine 12* (June 15, 1993): 109-170.

[Defense 86] Defense Systems Management College. *Systems Engineering Management Guide.* Washington, D.C.: U. S. Government Printing Office, 1986.

[Eisner 88] Eisner, Howard. *Computer Aided Systems Engineering.* New Jersey: Prentice Hall, 1988.

[Fagan 76] Fagan, Michael. "Software Inspections." *IBM Systems Research Journal* (September 1976).

[FM 770-78] USALMC Army Field Manual, "Systems Engineering," Training Support Center, Ft. Eustis (804-878-4668).

[Foster 93] Foster, Kenneth R. “Math, Visualization, and Date Acquisition.” *IEEE Spectrum 30* (November 1993): 42-59.

[Geppert 94] Geppert, Linda. "Faults and Failures." *IEEE Spectrum.* (August 1994): 17.

[Hall 62] Hall, Arthur D. *A Methodology for Systems Engineering.* Princeton, N.J.: Van Nostrand Company. 1962.

continued on next page

Reference list, continued

[Herbsleb 94] Herbsleb, James, et al. *Benefits of CMM-Based Software Process Improvement: Initial Results*, (CMU/SEI-94-TR-13), Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, August, 1994.

[Humphrey 87] Humphrey, Watts. *Characterizing the Software Process Maturity of Contractors Preliminary Report*, (CMU/SEI-87-TR-23, ADA 187230), Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, September, 1987.

[IEEE 90] *Dictionary of Computing Terms*, IEEE 630-90, 1990.

[IEEE 93] IEEE P1220. *IEEE Standard for Systems Engineering*, Preliminary, 1993.

[INCOSE 92b] International Council on Systems Engineering (INCOSE), INSIGHT (Issue 9, September 1995): 6.

[Kornbluh 93] Kornbluh, Ken. “Seeing Data in Action," *IEEE Spectrum 30* (November 1993): 60-75.

[Lacy 92] Lacy, James A. *Systems Engineering Management, Achieving Total Quality*. New York: McGraw-Hill, Inc. 1992.

[Lano 90] Lano, R. J., “The N2 Chart,” 244-27. *System and Software Requirements Engineering.* Thayer, Richard & Dorfman, Merlin, eds. Washington: IEEE Computer Society Press, 1990.

[McAuley 93] McAuley, James E.; & McCumber, William H. eds. *Proceedings of the Third Annual International Symposium of the National Council on Systems Engineering*, 1993.

[MIL-STD-499B] Draft Systems Engineering Standard, AFMC, 1994.

[Miller 56] Miller, George. “The Magical Number Seven, Plus or Minus Two.” *Psychological Review 63* (1956): 81-97.

[Mish 86] Mish, Frederick C., ed. *Webster’s Ninth New Collegiate Dictionary*. Springfield: Merriam-Webster. 1986.

continued on next page

Reference list, continued

[Morgello 91] Morgello, Clem. "George Fisher of Motorola: the Quest for Quality." *Institutional Investor* (1991): Volume 25, Number 9, 45-46.

[NCOSE 92a] National Council on Systems Engineering (NCOSE), *Membership flier*, 1992.

[NCOSE 92b] NCOSE. *Introduction to NCOSE*, 1992.

[NCOSE 93] NCOSE *Capability Assessment Working Group (CAWG) Request for Information for Systems Engineering Capability Maturity Models*, 1993.

[O'Boyle 90] O'Boyle, Thomas F. "GE Refrigerator Woes Illustrate the Hazards in Changing a Product," *Wall Street Journal 1* (May 7, 1990).

[Paulk 91] Paulk, Mark, et. al. *A Capability Maturity Model for Software*, (CMU/SEI-91-TR-24, ADA 240603). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, August 1991.

[Paulk 93a] Paulk, Mark; Curtis, William; & Chrissis, Mary Beth. *A Capability Maturity Model for Software v1.1*, (CMU/SEI-93-TR-24, ADA 263403). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, February 1993.

[Paulk 93b] Paulk, Mark; Weber, Charles; Garcia, Suzanne; Bush, Marilyn; & Chrissis, Mary Beth. *Key Practices for the Capability Maturity Model for Software v1.1*, (CMU/SEI-93-TR-25, ADA 263432). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, February 1993.

[Sage 77] Sage, Andrew P. *Methodology for Large-Scale Systems*. New York: McGraw-Hill, 1977.

[Sailor 90] Sailor, J. Douglas. "System Engineering: An Introduction," *System and Software Requirements Engineering*. Thayer, Richard; & Dorfman, Merlin, eds. Washington: IEEE Computer Society Press, 1990: 35-47.

continued on next page

Reference list, continued

[Schmidt 92] Schmidt, Stephen R.; & Launsby, Robert G. *Understanding Industrial Designed Experiments.* Colorado Springs, CO: Air Academy Press, 1992.

[Sedrick 91] Sedrick, Greg, ed. "A Commitment to Success," *Proceedings of the American Society for Engineering Management and the National Council on Systems Engineering.* NCOSE, 1991.

[Small 93] Small, Charles H. "Workstations vs PCs," *EDN 38* (March 18, 1993):164-174.

[Weber 91] Weber, Charles, et al., *Key Practices for the Capability Maturity Model for Software,* (CMU/SEI-91-TR-25, ADA 2640604) Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, August 1991.

[Wheelwright 92] Wheelwright, Steven C.; & Clark, Kim B. *Revolutionizing Product Development: Quantum Leaps in Speed, Efficiency and Quality*. New York: New York Press, 1992.

[Wood 91] Wood, David P. *Extending Process Modeling Accuracy* (CMU/SEI-91-SR-10) Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1991.

[Wymore 76] Wymore, A. Wayne. *Systems Engineering Methodology for Interdisciplinary Teams.* New York: John Wiley & Sons, 1976.

[Zajak 92] Zajak, Blair. "Enhanced System Engineering Efficiency in the 21st Century," *Proceedings of the Second Annual International Symposium of the National Council on Systems Engineering*. Morrison, Arthur F.; & Wirth, John M., eds. NCOSE, 1992.

Appendix D: Systems Engineering Glossary

Introduction

We developed the following glossary for use with all SE-CMM work products. Therefore, some terms are defined which are not, at present, included in this document. A common glossary approach was chosen because many terms used in the systems engineering world look the same, but convey differing and sometimes conflicting meanings, depending on the background of the author and reader. By placing all the terms in a common location, in a common context, we hope to help you understand the systems engineering concepts, while promoting continuity across the product line.

We chose these definitions from a wide spectrum of sources, including industrial, government, and societal standards, modified only to the extent needed to place them in the SE-CMM context. The source of the information has been provided whenever possible.

Definitions with a reference of [SECMM] indicate definitions that we produced as part of the SE-CMM Project.

Issues Form for SECMM-95-01 Version 1.1

Reviewer Information Please provide your name and organizational affiliation.

Reviewer Name	Reviewer Orgn	Contact Phone #

If using hardcopy, you may attach several forms together with the name on just the first one.

Issue Reference Please list the page #(s) or other reference (e.g., "global," "Chapter 3," "Glossary") to which this issue applies. Attach the page for reference if appropriate.

Issue Statement Please characterize the issue as a *problem* (e.g., the glossary is not detailed enough to support...) vs. a *solution* (e.g., add more detail to the glossary), so that the authors can understand the cause of the issue, not just the suggested fix. Include your rationale for highlighting the issue, if appropriate.

Prioritization This issue is _____ out of my top 10 issues with the SE-CMM Version 1.1.

Impact Assessment Please evaluate the impact the stated problem has on your use of the SE-CMM according to this scale:

____ **High Impact**: can't use model as intended without problem being fixed.
____ **Medium Impact:** misleading or otherwise incorrect content of significance to the reviewer.
____ **Low Impact:** content error of low significance to reviewer.

Note: editorial issues For typographical/grammatical/punctuation edits, please forward the redlined pages without the issue form attachment.

Submission Please return Issues Forms to:

Pete Malpass
Software Engineering Institute
4500 Fifth Avenue
Pittsburgh, PA 15213

acceptance criteria
The criteria that a system or component must satisfy in order to be accepted by a user, customer, or other authorized entity.
[IEEE 90]

action item
(1) A task assigned to an individual or group for disposition.
(2) An action proposal that has been accepted.
[SECMM]

activities performed
A description of the tasks necessary to implement a key process area. Activities Performed typically involve establishing plans and procedures, performing the work, tracking it, and taking corrective actions as necessary.
[SECMM]

activity
Any step taken or function performed (mental, physical, or both) toward achieving an intended objective.
[SECMM]

allocation
(1) The process of distributing requirements, resources, or other entities among the components of a system or program.
(2) The results of the distribution in (1).
[IEEE 90]

application domain
A bounded set of related systems, i.e., systems that address a particular type of problem. Development and maintenance in an application domain usually requires special skills and/or resources. Examples include payroll and personnel systems, command and control systems, compilers, and expert systems.
[Paulk 93b]

appraisal
A comparison of an implemented process to a process maturity model. Software process assessments and software capability evaluations are examples.
[SECMM]

architecture
The organizational structure of a system or component.
[IEEE 90]

attribute
A characteristic of an item; for example, the item's color, size, or type.
[IEEE 90]

audit	An independent examination of a work product or set of work products to assess compliance with specifications, standards, contractual agreements, or other criteria. [IEEE 90]
base practice	An engineering or management activity that addresses the purpose of a particular process area and thus belongs to it. [SPICE]
baseline	(1) A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and that can be changed only through formal change control procedures. (2) A document or a set of such documents formally designated and fixed at a specific time during the life cycle of a configuration item. (3) Any agreement or result designated and fixed at a given time, from which changes require justification and approval. [IEEE 90]
build	An operational version of a system or component that incorporates a specified subset of the capabilities that the final product will provide. [IEEE 90]
candidate solution	A solution that is developed for consideration when seeking an optimal solution. [SECMM]
capability	A measure of the system's ability to achieve the mission objectives, given that the system is dependable and suitable. Examples of capability measures are accuracy, range, payload, lethality, information rates, number of engagements, and destructiveness. Capability measures can be used as performance requirements, design constraints. and/or technical exit criteria. Capability is a systems engineering metric. [MIL-STD 499B]
capability evaluation	An appraisal made by a trained team of professionals, using an established method (e.g., the SEI software capability evaluation method) to (1) identify contractors qualified to perform specific task(s), or (2) monitor the state of the process used on an existing effort. [SECMM]

capability level	A set of common features (sets of generic practices) that work together to provide a major enhancement in the capability to perform a process area. [SPICE]
capability maturity model	A descriptive model of the stages through which organizations progress as they define, implement, evolve, and improve their processes. This model serves a guide for selecting process improvement strategies by facilitating the determination of the current process capabilities and the identification of issues most critical to quality and process improvement within a particular domain, such as software engineering or systems engineering. [Paulk 93b]
causal analysis	The analysis of defects to determine their underlying root cause. [Paulk 93b]
certification	Acknowledgement, based on a formal demonstration, that a system or component complies with its specified requirements and is acceptable for operational use. [IEEE 90]
change control	An element of configuration management, consisting of the evaluation, coordination, approval or disapproval, and implementation of changes to work products. [SECMM]
change control board	A group of people responsible for evaluating and approving or disapproving proposed changes to work products, and for ensuring implementation of approved changes. [SECMM]
change request	A formal request to change some aspect of an established baseline. [SECMM]
commitment	A pact that is freely assumed, visible, and expected to be kept by all parties. [Paulk 93b]
common feature	A set of practices that address an aspect of the process implementation or institutionalization. [SPICE]

compatibility	(1) The ability of two or more systems or components to perform their required functions while sharing the same environment. (2) The ability of two or more systems or components to exchange information. [IEEE 90]
complexity	(1) The degree to which a system or component has a design or implementation that is difficult to understand and verify. (2) Pertaining to any of a set of structure-based metrics that measure the attribute in (1). [IEEE 90]
component	One of the parts that make up a system. A component may be hardware or software and may be subdivided into other components. [IEEE 90]
configuration	In configuration management, the functional and physical characteristics of hardware or software as set forth in technical documentation or achieved in a product. [IEEE 90]
configuration data	Data that reflect the current configuration or state of the system or its components. [SECMM]
configuration item	An aggregation of hardware, software, or both, that is designated for configuration management and treated as a single entity in the configuration management process. [IEEE 90]
configuration management	A discipline applying technical and administrative direction and surveillance to identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report change processing and implementation status, and verify compliance with specified requirements. [IEEE 90]
configuration management library system	The tools and procedures to access the contents of the baseline library. [Paulk 93b]

configuration unit
The lowest level entity of a configuration item or component that can be placed into, and retrieved from, a configuration management library system.
[Paulk 93b]

corrective action recommendations
Proposed method (s) designed to correct a specific defect.
[SECMM]

corrective actions
Planned activities initiated to correct a defect.
[SECMM]

cost requirements
The financial thresholds and objectives expressed in terms of design-to-cost targets, research, development, test & evaluation (RDT&E) operating and support costs, and flyaway, weapon system, unit procurement, program acquisition, and life-cycle costs.
[MIL-STD 499B]

critical components
Components that are indispensable.
[SECMM]

critical design review
A review conducted to verify that the detailed design of one or more configuration items satisfy specified requirements; to establish the compatibility among the configuration items and other risk areas for each configuration item; and, as applicable, to assess the results of producibility analyses, review preliminary hardware product specification, evaluate preliminary test planning, and evaluate the adequacy of preliminary operation and support documents.
[IEEE 90]

current estimate
The value of a technical parameter that is predicted to be achieved with existing resources by the end of the contract.
[MIL-STD 499B]

customer
Individual(s) or organizational entity(ies) for whom the product or service is rendered; also one who uses the product or service.
[SECMM]

customer feedback
Information provided by the customer indicating the degree of satisfaction with the product or service.
[SECMM]

customer needs

What a customer believes that he needs to perform some activity of interest to him.

[SECMM]

customer satisfaction

An indicator of the degree to which a delivered product or service meets or exceeds the customer's expectations.
[SPICE]

decision database

A repository for storing all information pertinent to the systems engineering process. This repository is used to preserve a historical view into the tradeoffs and decisions that evolved the system architecture and design into a given state.
[IEEE 93]

defect

A flaw in a system or system component that has the potential to cause that system or component to fail to perform its required function during execution,
[SECMM]

defect prevention

The activities involved in identifying defects or potential defects and preventing them from being introduced into a product.
[Paulk 93b]

defined process

The operational definition of a set of activities. A defined process is well characterized and understood, and is described in terms of standards, tools, and methods.
Note: A defined process is developed by tailoring the organization's standard process to fit the specific characteristics of its intended use. (See also *standard process*)
[SPICE]

delivery

Release of a system or component to its customer or intended user.
[IEEE 90]

derived requirements

Requirements which may or may not be explicitly stated in the customer requirements, and which may be inferred from contextual requirements, e.g., applicable standards, laws, policy, common practice, and management decisions. Derived requirements can also arise during analysis and design from partitions of the system.
[SECMM]

design	(1) The process of defining the architecture, components, interfaces, and other characteristics of a system or component. (2) The result of the process in (1). [IEEE 90]
design constraints	Design limitations or implied requirements which constrain the design solution. A form of requirement which constrains the design solution set to a single or limited array of choices. This may include limitations on the logical execution, the physical characteristics, or performance of a system which are implied by a requirement statement, or derived from the analysis of conflicting or overlapping requirements. [IEEE 93]
design requirement	A requirement that specifies or constrains the design of a system or system component. [IEEE 90]
design review	A process or meeting during which a system, hardware, or software design is presented to project personnel, managers, users, customers, or other interested parties for comment or approval. Types include critical design review, preliminary design review, system design review. [IEEE 90]
detailed operational concept	A detailed description, derived from the preliminary operational concept, of the user's interaction with the system that satisfies the operational need. [SECMM]
development	The process of translating a design into hardware and/or software components. [SECMM]
deviation	A departure from the appropriate requirement, plan, standard, or procedure. [SECMM]
documented procedure	A written description of a course of action to be taken to perform a given task. [IEEE 90]

effectiveness analysis An analytical approach used to determine how well a system performs in its intended utilization environment.
[MIL-STD 499B]

efficiency The degree to which a system or component performs its designated functions with minimum consumption of resources.
[IEEE 90]

end user The individual or groups who will use the system for its intended operational use when it is deployed in its environment.
[Paulk 93b]

engineering change In configuration management, an alteration in the configuration of a configuration item or other designated item after formal establishment of its configuration identification.
[IEEE 90]

engineering requirements A translation of the set of essential customer needs into engineering language, specific to the domain expertise of the engineering staff that is charged with executing the design of the system. Engineering requirements are product requirements that are restated in engineering terms and are suitable for system development.
[SECMM]

engineering staff The technical people (e.g., analysts, programmers, and engineers, including task leaders), who are not managers and who perform the product development and maintenance activities for the project.
[SECMM]

enterprise A unit within a company or spanning several companies within which many projects are managed as a whole. All projects within an enterprise, at the top of the reporting structure, share a common manager and common policies.
[SECMM]

environment

The circumstances or conditions that will surround the system when it is in use. Examples include the natural environment (weather, climate, ocean conditions, terrain, vegetation, space conditions); combat environment (dust, fog, nuclear-chemical-biological); threat environment (effects of existing and potential threat systems to include electronic warfare and communications interception); operations environment (thermal, shock, vibration, power variations); transportation and storage environment; maintenance environment; test environments; manufacturing environments (critical process conditions, clean room, stress); and other environments (e.g., software engineering environment, electromagnetic) related to system utilization.
[MIL-STD 499B]

environment performance report

A summary of the performance of the systems engineering support environment compared to its expected performance.
[SECMM]

evaluation criteria

The criteria against which a selection, decision, or set of decisions will be made.
[SECMM]

exception report

A report that describes differences between requirement or design specifications and the measured properties of a system or system elements.
[SECMM]

exit criteria

The specific accomplishments or conditions that must be satisfactorily demonstrated before an effort can progress further in the current acquisition phase or transition to the next acquisition phase. Technical exit criteria are used for SEMS events and for acquisition phase milestone reviews.
[MIL-STD 499B]

external system (interfaces)

The system or product interfaces to other systems, communication networks, power supplies, resource connectors, etc., that affect the design of the product under consideration.
[IEEE 93]

failure

The inability of a system or component to perform its required functions within specified performance requirements.
[IEEE 90]

fault	(1) A defect in a hardware device or component; for example, a short circuit or broken wire. (2) An incorrect step, process, or data definition in a computer program. [IEEE 90]
feasibility	The degree to which the requirements, design, or plans for a system or component can be implemented under existing constraints. [IEEE 90]
findings	(1) The conclusions of an assessment, evaluation, audit, or review that identify the most important issues, problems, or opportunities within the area of investigation. (2) The issues, problems, or opportunities so identified. [SECMM]
formal review	A formal meeting at which a product is presented to the end user, customer, or other interested parties for comment and approval. It can also be a review of the management and technical activities and of the progress of the project. [Paulk 93b]
function	A task, action, or activity that must be accomplished to achieve a desired outcome or provide a desired capability. [IEEE 93]
functional architecture	The arrangement of functions, their decomposition, and interfaces (internal and external) that defines the execution sequencing, conditions for control or data flow, and the relative performance levels of achievement for a desired outcome, or that provides a desired capability. [IEEE 93]
functional interface requirement	The functional and performance requirements and constraints that exist at a common boundary between two or more functions in a functional architecture. [SECMM]
functional requirement	A requirement that specifies a task, action, or activity that a system or system component must be able to perform. [SECMM]

generic practice	An implementation or institutionalization practice that enhances the capability to perform any process. Generic practices are used during process appraisals to determine capability in any process area. [SECMM]
group	The collection of people who have responsibility for a set of tasks or activities. [SECMM]
implementation	(1) The process of translating a design into hardware components, software components, or both. (2) The result of the process in (1). [IEEE 90]
inspection	A method used to verify requirements. It involves the visual examination of documentation or a physical product (e.g., software code, hardware equipment) against predefined criteria or characteristics. An internal process of examining and evaluating the technical content of a work product against a set of predefined criteria. [SECMM]
institutionalization	The building of infrastructure and corporate culture that support methods, practices, and procedures so that they are the ongoing way of doing business, even after those who originally defined them are gone. [Paulk 93b]
integrated management	The unification and integration of the engineering and management activities into a coherent defined process based on the organization's standard process and related process assets. [SECMM]
integrated product development	A systematic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause developers, from the outset, to consider all elements of the product life cycle from conception through disposal (including quality, cost, schedule, and user requirements). [Adapted from Inst for Def Analysis]

integrated system	The product that the engineering staff builds to satisfy the product requirements. [SECMM]
integration	The merger or combining of one or more components, parts, or configuration items into a higher level system for ensuring that the logical and physical interfaces can be satisfied, and the integrated system satisfies its intended purpose. [IEEE 93]
integration plan	A plan describing the schedule, resources and approach to integrating the system elements. [SECMM]
integration report	Report describing the compliance of integration efforts with integration plans, the observed successes of integration efforts, and the observed failures of integration efforts. [SECMM]
interface control document	A specification, derived from the physical interface requirements, that details the physical interface between two system elements, including the number and types of wires, connectors and pins, electrical parameters, mechanical properties, and environmental constraints. [SECMM]
interface requirement	The functional, performance, electrical, environmental, human, and physical requirements and constraints that exist at a common boundary between two or more functions, system elements, configuration items, or systems. [MIL-STD 499B]
interface specification	A specification, derived from the interface requirements, that details the mechanical properties and/or logical connection between system elements, including the exact format and structure of the data and/or electrical signal communicated across the interface. [SECMM]
interfacing groups	Separate groups that must communicate in order to accomplish a unified set of goals or objectives. [SECMM]

item

A nonspecific term used to denote any product, including systems, subsystems, assemblies. subassemblies, units, sets, parts, accessories, computer programs. or computer software. In this standard, it also denotes any process that includes a series of actions, changes, or functions to achieve an end or result.
[MIL-STD 499B]

key design issues

A set of issues that, once decided, determine the technical direction of major portions of the system design.
[SECMM]

key practices

The infrastructures and activities that contribute most to the effective implementation and institutionalization of a key process area.
[Paulk 93b]

life cycle

The scope of the system or product evolution beginning with the identification of a perceived customer need, addressing development, test, manufacturing, operation, support and training activities, continuing through various upgrades or evolutions, until the product and its related processes are disposed of.
[IEEE 93]

life-cycle cost

The total investment in product development, test, manufacturing, distribution, operation, refining, and disposal. This investment typically is allocated across the anticipated number of units to be produced over the production life cycle, thus providing a per-unit view of life-cycle cost.
[IEEE 93]

maintenance

The process of modifying a system or component after delivery to correct faults, improve performance or other attributes, or adapt to a changed environment.
[SECMM]

manager

A person who provides technical and administrative direction and control to individuals performing tasks or activities within the manager's area of responsibility. The traditional functions of a manager include planning, allocating resources, organizing, directing, and controlling work within an area of responsibility.
[SECMM]

market survey	Investigation that focuses on a set of potential customers to help identify the customer requirements for a product or service. [SECMM]
maturity level	A well-defined evolutionary plateau toward achieving a mature software process. The five maturity levels in the SEI Capability Maturity Model are initial, repeatable, defined, managed, and optimizing. [Paulk 93b]
maturity model	A model of the stages through which organizations progress as they define, implement, evolve, and improve their processes. This model serves as a guide for selecting process improvement strategies by facilitating the determination of current process capabilities and identification of the issues most critical to quality and process improvement. [SECMM]
measure	A unit of measurement such as source lines of code or document pages of design. [Paulk 93b]
measurement	A raw data item collected on a process. The basic quantitative value that describes the magnitude of an element of the process. [SECMM]
measures of effectiveness	The figures-of-merit which provide a quantitative means for comparing alternative system solutions. [IEEE 93]
method	A reasonably complete set of rules and criteria that establish a precise and repeatable way of performing a task to provide a desired result. [SECMM]
methodology	A collection of methods, procedures, and standards that defines an integrated synthesis of engineering approaches to the development of a product. [Paulk 93b]
metric	A composite of two or more measurements resulting in a value that defines a characteristic of the process. [SECMM]

milestone	A scheduled event for which some project member or manager is held accountable and at which progress toward a defined goal is measured. [SECMM]
modification	The act of changing a system or component after delivery to improve performance or some other attribute, or to adapt the system or component to function in a changed environment [SECMM]
need	A user related capability shortfall (such as those documented in a Mission Need Statement, field deficiency report, or engineering change proposal), or an opportunity to satisfy a capability requirement because of a new technology application or breakthrough, or to reduce costs. Also a statement of capability required for each supplier related primary function including disposal. [MIL-STD 499B]
operational environment	The natural or induced environmental conditions, and user interactions, within which the system is expected to be operated. [IEEE 93]
operational requirements	The statements that identify the essential capabilities (the process or series of actions performed to effect a purpose or result) that are desired in the system under development. [IEEE 93]
organization	A unit within an entity (e.g., company, government agency, or branch of service) within which many projects are managed as a whole. All projects within an organization, at the top of the reporting structure, share a common manager and common policies. [SECMM]
organization's business goals	The reasons for an organization’s existence. Such goals may include reducing the number of change requests during a system's integration phase, reducing development cycle time, increasing the number of errors found in a system's first or second phase of development, reducing the number of customer-reported defects, etc., when applied to systems engineering activities. [SECMM]

organization's process assets

A collection of entities, maintained by an organization, for use by the projects and others in developing, tailoring, maintaining, and implementing their product development processes. These process assets typically include the organization's standard product development processes, descriptions of the product life cycles approved for use, the guidelines and criteria for tailoring the organization's standard product development process, the organization's product development process database, and a library of product development process-related documentation. Any entity that the organization considers useful in performing the activities of process definition and maintenance could be included as a process asset.
[SECMM]

organization's product development process database

A database established to collect and make available data on the product development processes and resulting work products, particularly as they relate to the organization's standard product development process. The database also contains or references the actual measurement data and related information and data that are needed to understand and interpret the measurement data and assess it for reasonableness and applicability. Examples of process and work product data include estimates of product size, effort, and cost; actual data on product size, effort, and cost; productivity data; peer review coverage and efficiency; and number and severity of defects found in the product.
[SECMM]

organization's standard product development process

The operational definition of the basic process that guides the establishment of a common product development process across the product development projects in the organization. It describes the fundamental elements of the product development process that each project is expected to incorporate into its defined process. It also describes the relationships, e.g., ordering and interfaces between these elements of the product development process.
[SECMM]

organization's standard systems engineering process

The operational definition of the basic process that guides the establishment of a common systems engineering process across the projects in the organization. It describes the fundamental elements of the systems engineering process that each product development project is expected to incorporate into its defined systems engineering process. It also describes the relationships e.g., ordering and interfaces, between these systems engineering process elements.
[SECMM]

peer review

A review of a work product, following defined procedures, by peers of the product's producers for the purpose of identifying defects and improvements. In the SE-CMM questionnaire, this is called a *defect review*.
[SECMM]

performance

The degree to which a system or component accomplishes its designated functions within given constraints, such as speed, accuracy, or memory usage.
[IEEE 90]

performance requirement

A requirement that imposes conditions on a functional requirement, for example, a requirement that specifies the speed, accuracy, or memory usage with which a given function must be performed.
[IEEE 90]

physical architecture

The hierarchical arrangement of product and process solutions, their functional and performance requirements, their internal and external (external to the aggregation itself) functional and physical interfaces and requirements, and the physical constraints that form the basis of design requirements. The physical architecture provides the basis for system/configuration item baselines as a function of the acquisition phase. It documents one or more physical designs as required to (1) accomplish effectiveness analysis, risk analysis, and technology transition planning; (2) establish the feasibility of physically realizing the functional architecture; (3) identify manufacturing verification, support, and training requirements; (4) document the configuration of prototypes and other test articles; and (5) define in increasing detail the solution to identified needs.
[MIL-STD 499B]

physical characteristics

The physical attributes or distinguishing features that pertain to a distinctive quality.
[IEEE 93]

physical interface requirement

The performance, electrical, environmental, human, and physical requirements and constraints that exist at a common boundary between two or more system elements, configuration items, or systems.
[SECMM]

physical requirement	A requirement that specifies a physical characteristic that a system or system component must possess (for example, material, shape, size, weight). [IEEE 90]
planned profile	Profile representing the projected time-phased demonstration of a technical parameter requirement. [MIL-STD 499B]
planned value	Predicted value of the technical parameter for the time of measurement based on the planned profile. [MIL-STD 499B]
policy	A guiding principle, typically established by senior management, that is adopted by an organization or project to influence and determine decisions. [Paulk 93b]
pratice	An activity, or set of activities, that contributes to the achievement of a process area purpose. These practices are of two types: base practices and generic practices. (See also *base practice and generic practice.*) [SECMM]
preliminary design	The process of analyzing design alternatives and defining the architecture, components, interfaces, and timing and sizing estimates for a system or component. [IEEE 90]
preliminary design review	A review conducted to evaluate the progress, technical adequacy, and risk resolution of the selected design approach for one or more configuration items; to determine each design's compatibility with the requirements for the configuration item; to evaluate the degree of definition and assess the technical risk associated with the selected manufacturing methods and processes; to establish the existence and compatibility of the physical and functional interfaces among the configuration items and other items of equipment, facilities, software and personnel; and, as applicable, to evaluate the preliminary operational an support documents. [IEEE 90]

preliminary operational concept	A conceptual description of how the customer envisions using or how the customer might use the product. This concept gives insight into the reason behind customer desires. [SECMM]
primary functions	Those essential tasks, actions, or activities that must be accomplished to ensure that the system will satisfy customer needs from a system life-cycle perspective. The eight primary system life-cycle functions are development, manufacturing, verification, deployment, operations, support, training, and disposal. [MIL-STD 499B]
procedure	A written description of a sequence of actions to be taken to perform a given task. [SECMM]
process	1. A set of activities (ISO 12207). 2. A set of practices that address the same purpose. [BPG\TP\BPG.101]
process area	A grouping of a purpose and a set of related practices that, when performed collectively, can achieve the purpose of the process area. [SECMM]
process asset library	A library of process assets that exist within a defined architecture that gives structure to the example processes, process fragments, process-related documentation, process architectures, process tailoring rules and tools, and process measurements. [SECMM]
process assets	Example processes, process fragments, process-related documentation, process architectures, process tailoring rules and tools, and process measurements. These assets are to be tailored by a project to form the specific process that it will follow in developing its system. [SECMM]
process capability	The range of expected results that can be achieved by following a process. [Paulk 93b]

process description The operational definition of the major components of a process. Documentation that specifies, in a complete, precise, verifiable manner, the requirements, design, behavior, or other characteristics of a process. It may also include the procedures for determining whether these provisions have been satisfied. Process descriptions may be found at the activity, project, or organizational level.
[Paulk 93b]

process element The constituent elements of a process. Each process element covers a well-defined, bounded, closely related set of tasks (e.g., estimating element, design element, coding element, and peer review element). The descriptions of the process elements may be templates to be filled in, fragments to be completed, abstractions to be refined, or complete descriptions to be modified or used unmodified.
[Paulk 93b]

process enactment technology A specific method of process implementation that involves automation of the transfer and collection of information from entities charged with executing subprocesses or tasks.
[SECMM]

process evaluation Analysis of process measurements to understand and improve the process.
[SECMM]

process measurement The set of definitions, methods, and activities used to take measurements of a process and its resulting products for the purpose of characterizing and understanding the process.
[Paulk 93b]

process performance A measure of actual results achieved by following a process.
[SECMM]

process tailoring The activity of creating a process description by elaborating or adapting process elements or other incomplete specifications of a process. Specific business needs for a project will usually be addressed during process tailoring.
[SECMM]

process technology The application of a science and/or engineering technology (e.g., tools or methodology) to a process or subprocess.
[SECMM]

product	The result of a human, mechanical or natural effort or process, such as, a manufacturing process. [IEEE 93]
product baseline	In configuration management, the initial approved technical documentation (including, for software, the source code listing) defining a configuration item during the production, operation, maintenance, and logistic support of its life cycle. [IEEE 90]
product development cycle	The time required to execute the product development process. [SECMM]
product development process	The process by which new products are created and brought to market. [SECMM]
product line requirements	The requirements for a family of products that can satisfy the organization's strategic vision. Requirements for a set of development projects chosen to provide superior products and processes. [SECMM]
product quality certification	A formal demonstration that a system or component complies with its specified quality requirements and the product is acceptable for operational use. [SECMM]
product requirements	The translation of customer needs and expectations into a set of requirements for the system to be built in terms that the customer understands and upon which any desired agreements between the customer and systems engineering organization can be based. [SECMM]
profile	A comparison, usually in graphical form, of plans or projections versus actual data, typically over time. [Paulk 93b]

program

An initiative, prescribed plan, or course of action, such as a training program or process improvement program, which is usually undertaken at the organizational level. A program typically specifies the objective, methods, activities, plans, and success measures for the target of the program.
[Paulk 93b]

project

The aggregate of effort and other resources focused on developing and/or maintaining a specific product. The product may include hardware, software, and other components. Typically a project has its own funding, cost accounting, and delivery schedule.
[Paulk 93b]

project plan

A document that describes the technical and management approach to be followed for a project. The plan typically describes the work to be done, the resources required, the methods to be used, the procedures to be followed, the schedules to be met, and the way that the project will be organized (for example, a software development plan).
[IEEE 90]

project's defined process

The operational definition of the process as used by a specific project. Well characterized and understood, it is described in terms of standards, procedures, tools, and methods. It is developed by tailoring the organization's standard process to fit the specific characteristics of the project.
[SECMM]

prototype

A preliminary type, form, or instance of a system that serves as a model for later stages or for the final, complete version of the system.
[IEEE 90]

prototyping

A hardware and software development technique in which a preliminary version of part or all of the hardware or software is developed to permit user feedback, determine feasibility, or investigate timing or other issues in support of the development process.
[IEEE 90]

quality assurance

A planned and systematic means for assuring management that defined standards, practices, procedures, and methods of the process are applied.
[SECMM]

records of training and experience	A mapping of an organization's personnel to the training and experience that each individual has completed or accomplished. [SECMM]
reliability	The ability of a system or component to perform its required functions under stated conditions for a specified period of time. [IEEE 90]
requirements	Statements which identify the essential needs for a system in order for it to have value and utility. Requirements may be derived or based upon interpretation of stated requirements to assist in providing a common understanding of the desired operational characteristics of a system. [IEEE 93]
requirements analysis	The process of studying user needs to arrive at a definition of system, hardware, or software requirements. [IEEE 90]
requirements for systems engineering support environment	An environment in which development activities are supported with needed development and process enactment technology. Included are computer software, computer hardware, test equipment, etc. (See also *systems engineering support environment*.) [SECMM]
risk	A measure of the uncertainty of attaining a goal, objective, or requirement pertaining to technical performance, cost, and schedule. Risk level is categorized by the probability of occurrence and the consequences of occurrence. Risk is assessed for program, product, and process aspects of the system. This includes the adverse consequences of process variability. The sources of risk include technical (e.g., feasibility, operability, producibility, testability, and system effectiveness); cost (e.g., estimates, goals); schedule (e.g., technology/material availability, technical achievements, milestones); and programmatic (e.g., resources, contractual). [MIL-STD 499B]
risk management	An organized, analytic process to identify what can go wrong, to quantify and assess associated risks, and to implement/control the appropriate approach for preventing or handling each risk identified. [MIL-STD 499B]

risk management plan	A document which describes the risk management activities to be performed on a project. [SECMM]
risk mitigation activities	Actions taken to reduce the impact or likelihood of a risk. [SECMM]
risk mitigation strategies	The principles used to identify the order in which risk mitigation activities are implemented. [SECMM]
role	Defined responsibilities that may be assumed by one or more individuals. [SECMM]
sensitivity analysis	A technique for discovering the behavior of a system by changing one input at a time by a small amount and determining the changes in the outputs. A matrix of the quotients of the output changes over the input changes is called a sensitivity matrix. [SECMM]
software capability evaluation	An appraisal by a trained team of professionals, using a method such as the SEI software capability evaluation method, to (1) identify contractors who are qualified to perform the software work, or (2) monitor the state of the software process used on an existing software effort. [Paulk 93b]
software development plan	The collection of plans that describe the activities to be performed for the software project. It governs the management of the activities performed by the software engineering group for a software project. It is not limited to the scope of any particular planning standard, such as DOD STD 2167A and IEEE-STD-1058, which may use similar terminology. [Paulk 93b]
software process	A set of activities, methods, practices, and transformations that people use to develop and maintain software and the associated products e.g., project plans, design documents, code, test cases, and user manuals. [Paulk 93b]

software product	The complete set, or any of the individual items of the set, of computer programs, procedures, and associated documentation and data designated for delivery to a customer or end user. [IEEE 90]
software requirements	A condition or capability that must be met by software needed by a user to solve a problem or achieve an objective. [IEEE 90]
solution (or solution set)	The selected candidate solution(s) that best satisfies the analysis requirements. [SECMM]
specification	A document prepared to support acquisition and life-cycle management that clearly and accurately describes essential technical requirements and verification procedures for items, materials, and services. When invoked by a contract, it is legally enforceable and its requirements are contractually binding. [MIL-STD 499B]
staff	The people, including task leaders, who are not managers and who are responsible for accomplishing the assigned business function. [Paulk 93b]
standard	An approved, documented, and available set of criteria used to determine the adequacy of an action or object. [Paulk 93b]
standard process	The operational definition of the basic process that guides the establishment of a common process in an organization. It describes the fundamental process elements that are expected to be incorporated into any defined process. It also describes the relationships (e.g., ordering and interfaces) between these process elements. (See also *defined process*.) [SPICE]

standard process family

A group of standard processes within an organization that share some common characteristics, but that are different enough in their domain of applicability to be considered as separate standard processes. Organizations that find they are constantly tailoring the same areas of their standard process to meet the needs of a specific group within the organization may find the concept of a standard process family a useful way of characterizing their standard processes.
[SECMM]

statistical process control

A statistically based technique appropriate to analyze a process, identify special causes of variations in the performance of the process, and bring the performance of the process within well-defined limits.
[SECMM]

strategic vision

The political, economic, and psychological forces of an organization that ensure the maximum support for the adopted market goals of the organization. In this context, strategic vision can be expressed as the architecture of a family of products.
[SECMM]

subcontract manager

A person who has direct responsibility for administering and managing a subcontract.
[SECMM]

subcontractor

An individual, partnership, corporation, or association who contracts with an organization to design, develop, and/or manufacture items.
[Paulk 93b]

subprocess

A process that is part of a higher level process.
[SECMM]

subsystem

A grouping of items satisfying a logical group of functions within a particular system.
[MIL-STD 499B]

suppliers

The development, manufacturing, verification, and deployment personnel that define, design, code, fabricate, assemble, integrate, verify, test, deliver and/or install system end items, and safely dispose of the by-products of their activities.
[MIL-STD 499B]

support environment technology reviews	Reviews of the available support technology, including literature reviews, in-house demos, and trial usage of support technology. Such technology includes computer software, computer hardware, test equipment, etc. [SECMM]
support function	The tasks, actions, and activities to be performed and the system elements required to provide operations, maintenance, logistics (including training) and materiel management support. It provides for the definition of tasks, equipment, skills, personnel, facilities, materials, services, supplies, and procedures required to ensure the proper supply, storage, and maintenance of a system end item. [MIL-STD 499B]
synthesis	The combining of information, concepts, constraints, components, or elements to establish a complete and consistent system architecture, or to identify conflicts or deficiencies in established requirements or design solutions. [IEEE 93]
system	An integrated composite of people, products, and processes that provide a capability to satisfy a stated need or objective. [MIL-STD 499B]
system architecture	The composite of the functional, physical, and foundation architectures, which form the basis for establishing a system design. The system architecture includes the supporting requirement traceability and allocation matrices which identifies the relationship between the system design, and the elements of the functional, physical, and foundation architectures. [IEEE 93]
system configurations	Configuration data and status on the current state of the system. [SECMM]
system design	The product of the development process which provides sufficient details, drawings, or other pertinent information, on the system components, elements, parts, interfaces, etc., to permit the fabrication, production, assembly, integration and testing of the system under development. [IEEE 93]

system design review A review conducted to evaluate the manner in which the requirements for a system have been allocated to configuration items, the system engineering process that produced the allocation, the engineering planning for the next phase of the effort, manufacturing considerations, and the planning for production engineering.
[IEEE 90]

system development The summation of the creative actions taken during the system development cycle.
[SECMM]

system development cycle The period of time that begins with the decision to develop a system and ends when the system is delivered to its end user.
[IEEE 90]

system development process The engineering process employed to develop a new system. The process by which new products are created and brought to market.
[SECMM]

system effectiveness A measure of the ability of a system to satisfy its intended operational uses when called upon to do so. System effectiveness is a composite view of how the system performs under anticipated environmental conditions, the reliability and maintainability of system parts and components, and the ability to produce, distribute, support, train, operate and dispose of the system throughout its life cycle.
[IEEE 93]

system elements The basic constituents (hardware, software, facilities, personnel, data, material, services, or techniques) that comprise a system and satisfy one or more requirements in the lowest levels of the functional architecture.
[MIL-STD 499B]

system end item A deployed system product and/or process that is ready for its intended use.
[MIL-STD 499B]

system requirements

A description of desired capabilities, constraints, and other details which pertain to the product's functional, performance, and physical characteristics. These descriptions provide the stimulus for investigating product alternatives, and for making trade-offs among requirement sets. These requirements should establish the capabilities, physical characteristics, and customer quality attributes which define a quality product offering within the marketplace.
[IEEE 93]

system testing

Testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.
[IEEE 90]

systems analysis and control

The assessment and control mechanisms, including performance based progress measurements, to
• Establish system effectiveness.
• Balance cost, schedule, performance, and risk.
• Control the system configuration.
[MIL-STD 499B]

systems engineering

The selective application of scientific and engineering efforts to:
1. transform an operational need into a description of a system configuration which best satisfies the operational need according to the measures of effectiveness;
2. integrate related technical parameters and ensure compatibility of all physical, functional, and technical program interfaces in a manner which optimizes the total system definition and design;
3. integrate the efforts of all engineering disciplines and specialities into the total engineering effort.
[FM770-78]

systems engineering process	A comprehensive, iterative problem solving process that is used to: (a) transform validated customer needs and requirements into a life-cycle balanced solution set of system product and process designs, (b) generate information for decision makers, and (c) provide information for the next acquisition phase. The problem and success criteria are defined through requirements analysis, functional analysis/allocation, and systems analysis and control. Alternative solutions, evaluation of those alternatives, selection of the best life-cycle balanced solution, and the description of the solution through the design package are accomplished through synthesis and systems analysis and control. [MIL-STD 499B]
systems engineering support environment	An environment in which development activities are supported with needed development and process enactment technology. These include computer software, computer hardware, test equipment, etc. [SECMM]
tailor	To adapt a process or a set of standards or procedures to better match process or product requirements. [Paulk 93b]
task	Well-defined unit of work in the process that provides management with visible checkpoints into the status of the project. Tasks have readiness criteria and completion criteria. [SECMM]
task leader	A team leader for a specific task who has technical responsibility and provides the technical direction to the staff working on that task (including him/herself). [SECMM]
team	A collection of people, drawn from diverse, but related, groups, to perform a well-defined function for an organization or a project. Team members may have other primary responsibilities. [SECMM]

technical effort	The total engineering, test, manufacturing, and specialty engineering effort associated with the development of a product offering which encompasses all of the system, equipment, facilities, etc., necessary for the Enterprise to develop, produce, distribute, operate, test, support, train, and dispose of the product. [IEEE 93]
technical management plan	A plan that describes how the technical effort will be managed and conducted. [MIL-STD 499B]
technical objectives	The "target" values for the development effort when insufficient data is available for stating binding technical requirements. Also can be used to define capabilities beyond established technical requirements when opportunities have been identified for substantial increases in effectiveness, decreases in cost. or additional flexibility. Includes cost, schedule, and performance attributes deemed important. [MIL-STD 499B]
technical parameters	A selected subset of the system's technical metrics tracked in TPM. Critical technical parameters are identified from risk analyses and contract specification or incentivization, and are designated by management. Example of Technical Parameters include: a. Specification Requirements. b. Metrics associated with technical objectives and other key decision metrics used to guide and control progressive development. c. Design to cost requirements. d. Parameters identified in the acquisition program baseline or user requirements documentation. [MIL-STD 499B]
technical requirements	Those requirements that describe what the product must do. Examples of technical requirements include functions, performance, and interface requirements. [Paulk 93b]

technical reviews

A series of systems engineering activities by which the technical progress of a program is assessed relative to its technical or contractual requirements. Conducted at logical transition points in the development effort to reduce risk by identifying and correcting problems/issues resulting from the work completed before the program is disrupted or delayed. Provide a method for the contractor and Government to determine that the development of a system and/or configuration item and its documentation have met contract requirements. Includes incremental reviews (functional, subsystem, and interim system) and major system level technical reviews.
[MIL-STD 499B]

technology

The tools, techniques, training, and methods that can be applied by people in accomplishing some particular result.
[Paulk 93b-revised]

test

An activity in which a system or component is executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component.
[IEEE 90]

test plan

A plan describing the schedule, resources, and approach to verify the compliance of a system or its elements with the requirements.
[SECMM]

test report

Report that describes the compliance of test efforts with test plans, and the behavior and faults of the objects under test.
[SECMM]

testability

The degree to which a requirement is stated in terms that permit establishment of test criteria and performance of tests to determine whether those criteria have been met.
[IEEE 90]

threshold

The limiting acceptable value of a technical parameter, usually a contractual performance requirement
[MIL-STD 499B]

tolerance band

Management alert limits placed on each side of the planned profile to indicate the envelope or degree of variation allowed. The tolerance band represents the projected level of estimating error.
[MIL-STD 499B]

traceability
The degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another.
[IEEE 90]

traceability matrix
a matrix that records the relationship between two or more products of the development process; for example, a matrix that records the relationship between the requirements and the design of a given component.
[IEEE 90]

train
To make proficient with specialized instruction and practice.
[Paulk 93b]

training materials
Developed or acquired materials that are or will be used in building the needed skills among the organization's employees. These may include books, manuals, computer hardware, computer software, video tapes, audio tapes, etc.
[SECMM]

training program
An initiative that includes the organization's training plan, training materials, development of training, conduct of training, training facilities, evaluation of training, and maintenance records of training.
[Paulk 93b]

trend analysis
An analysis technique that relies on a collection of history for making future projections.
[SECMM]

users
The operators and supporters of system end items, and the trainers that train the operations and support personnel. Users execute the operations, support, training, and disposal functions associated with system end items.
[MIL-STD 499B]

validation
Validation involves evaluation of the customer requirements against customer needs and expectations, and evaluation of the delivered system to meet the customer's operational need in the most representative environment achievable.
[SECMM]

variation	Difference between the planned value of the technical parameter and the achievement-to-date value derived from analysis, test, or demonstration. [MIL-STD 499B]
verification	The process of determining whether or not the products of a given phase of development fulfill the requirements established during the previous phase. [IEEE 93]
well-defined process	A process with inputs, entry criteria, tasks, verification, outputs, and exit criteria that are documented, consistent, and complete. [SPICE - modified]
work product	All the data, files, documents, assemblies, components, etc., generated in the course of performing any process. [SECMM]