Handbook
SECMM-94-04
CMU/SEI-94-HB-04

Carnegie Mellon University
Software Engineering Institute

A Systems Engineering Capability Maturity Model, Version 1.0

SE - CMM

Systems Engineering Improvement

Systems Engineering Capability Maturity Model Project

December 1994

Handbook
CMU/SEI-94-HB-04

A Systems Engineering Capability Maturity Model, Version 1.0

R. Bate, S. Garcia, J. Armitage, K. Cusick,
R. Jones, D. Kuhn, I. Minnich,
H., Pierson, T. Powell, A. Reichner

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

This report was prepared for the

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

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.

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.]

Table of Contents

Tables and Figures

Tables:

Figures:

Acknowledgments

Participants

The model herein described 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 are mentioned below who fulfilled specific review roles, we are aware that, beyond those whose names we know, there are many who supported the points of contacts who turned in review comments 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: A. Dniestrowski, A. Shumskas, A. Sutton, A. Wilbur, B. Andrews, B. Mar, C. Giffen, C. Montgomery, C. Zumba, C.L. Nelson, D. Francis, D. Gunther, D. Jester, D. Kinney, D. Marquet, D. McCauley, D. McConnell, D. Preklas, D. Zaugg, E. Cherian, E.R. Anderson, G. Hingle, G. Lake, G. Pandelios, H. Jesse, H. Lee, H. Wilson, J. Burleson, J. Cooper, J. Gross, J. Harbison, J. Huller, J. Jaggers, J. Lake, J. Marciniak, J. Miller, J. Moon, J. Porter, J. Velman, J.P. Jones, K. Arunski, K. Ferraloio, K. Jones, L. Bloodsworth, L. Gallagher, L. Nelson, M. Brown, M. Carroll, M. Falat, M. Ginsberg, M. Klien, M. Konrad, M. Merrill, M. Ross, M. Ross, M. Ward-Callan, N. Sanford, P. James, R. Ade, R. Bechtold, R. Czizik, R. Daniel, R. Elkin, R. Nakahara, R. Ragano, R. Sabouhi, R. Schmidt, S. Friedenthal, S. Levie, S. McCammon, T. Carpenter, T. Concannon, T. Kudlick, T. Parker, T. Robertson, T. Sweeney, W. Hefley, W. Mackey, W. Oran, W. Peterson, K. Kirlin, and Y. Trsonstad.

continued on next page

Acknowledgments, Continued

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, from the viewpoint of both gathering data about the model, and learning about how the SE-CMM architecture facilitates appraisal. Over 100 individuals participated in 3 appraisals scattered between Texas Instruments, Hughes, and United Technologies. 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. 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
Department of Defense	John Burt
Hughes Aircraft Company	Paul Stevens, Ph.D.
Lockheed Corporation	Chris Caren, Ph.D.
Loral Federal Systems Company	Virginia Lentz
Loral Space & Range Systems	Dorothy McKinney
National Institute of Standards and Technology	Roger Martin
National Council on Systems Engineering	Don Crocker
Software Engineering Institute	Floyd Hollister
Software Productivity Consortium	Art Pyster, Ph.D.
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.

The *SE-CMM Model Description* provides an overall description of the principles and architecture upon which the SE-CMM is based, an executive 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 is 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 Tacoma Narrows bridge was built to connect Tacoma with the Olympic peninsula in Washington State. It was a very long suspension bridge with a flexible roadway. In 1940 it collapsed because of strong winds in the Narrows that started an aerodynamic oscillation that finally buckled the roadway.

In the engineering investigations that followed the disaster, it emerged that the engineers who designed the bridge had not done aerodynamic investigations of the design, because none of them were familiar with the techniques and it was not realized that the wind would have such strong dynamic effects.

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, but are likely to follow disciplined investigative methods that draw out the risk areas of a system.

continued on next page

To the Reader, Continued

What is the scope of the SE-CMM?

This first version of the SE-CMM starts with determination of the users' needs and extends through verification of the initial product. This first version focuses on process characteristics. Given sufficient community support, planned expansions will encompass the remaining product life-cycle activities and include 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 will be 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 pilot 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, Suzanne Garcia, may be contacted at

4500 Fifth Ave. (412)268-7625 (voice)
Pittsburgh, PA 15213 (412)268-5758 (fax)
email: smg@sei.cmu.edu

Data rights associated with the SE-CMM

The SE-CMM collaboration members are committed to encouraging free use of the SE-CMM Model Description 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.

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 the initial 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 of the discussion, 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.

Related products

In addition to this document, the SE-CMM Project plans to produce the following documents for public release in early 1995 via the maintenance site for the SE-CMM Project, Carnegie Mellon University's Software Engineering Institute.

Identifier	Name	Description
SECMM-94-06 CMU/SEI-94-HB-05	SE-CMM Appraisal Method Description	The *SE-CMM Appraisal Method Description* 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.
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 Mil-Std-499b, and the Capability Maturity Model for Software, v1.1.

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 instituted 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. Task completion was set at December 1994.

Project organization chart

The following diagram illustrates the project organization chart. It is discussed in the blocks below.

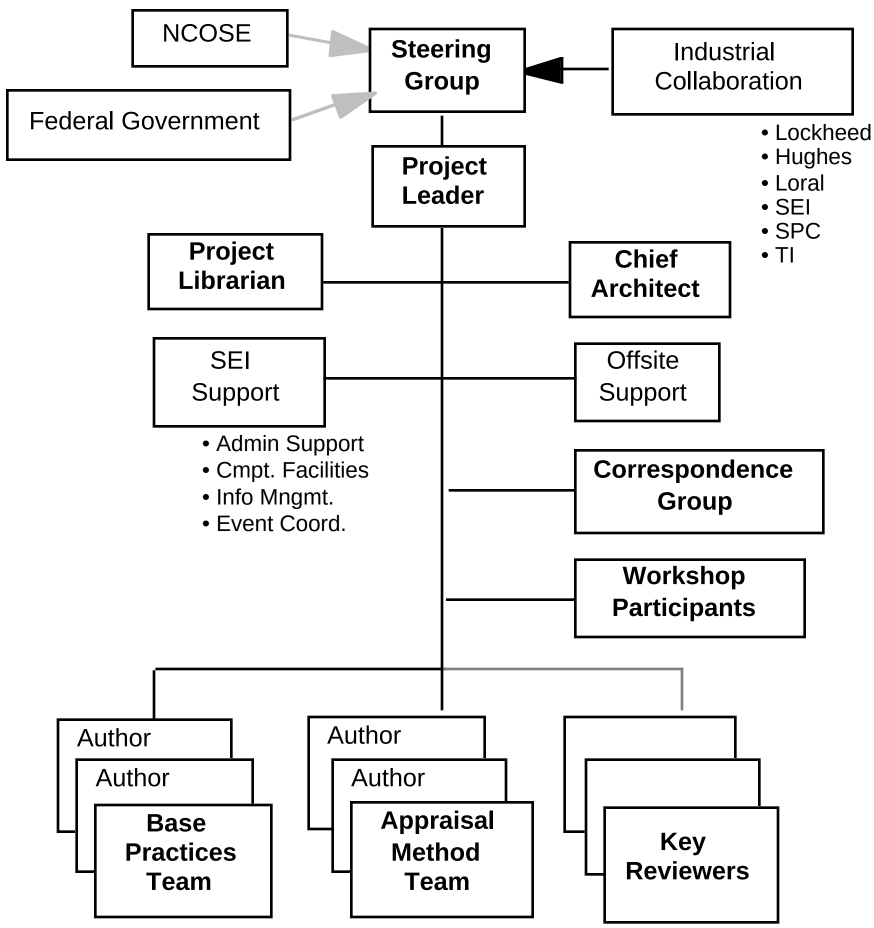

Figure 1-1. SE-CMM Project Organization

continued on next page

1.2 About the SE-CMM Project, Continued

SE-CMM Project composition

The SE-CMM project is run by a steering group which is composed of people from the SE-CMM collaboration, with ex officio members from The National Council on Systems Engineering (NCOSE) and 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 come from GTE, Hughes, Lockheed, Loral, Software Engineering Institute, Software Productivity Consortium, and Texas Instruments, organizations with an established history of good systems engineering performance and/or modeling and assessment methodology.

SE-CMM authors

The authors are listed in alphabetical order in the following table:

Author	Organization
James Armitage, Ph.D.	GTE Government Systems, Pittsburgh, PA
Roger Bate, Ph.D.	Software Engineering Institute, Pittsburgh, PA
Kerinia Cusick	Hughes Aircraft Company, El Segundo, CA
Suzanne Garcia	Software Engineering Institute, Pittsburgh, PA
Robert Jones	Loral Federal Systems Company, Houston, TX
Dorothy Kuhn	Texas Instruments, Inc., 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, Austin Division, Austin, TX

Table 1-2. SE-CMM Authors

continued on next page

1.2 About the SE-CMM Project, Continued

Incorporating community feedback

The SE-CMM 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 are supplemented by an extensive reviewer panel selected from academia, government and industry for their systems engineering expertise. The SE-CMM also includes feedback from two public workshops where early versions of the model were critiqued. In addition, the SE-CMM contains enhancements from three pilot appraisals of organizations using early versions of the model.

Future plans outline

This initial version of 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

- Expand the model to include other phases of the product life cycle such as manufacturing and post-delivery support. This aspect is under consideration for 1995 sponsorship.
- 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.
- Extend the model into addressing the people and technology aspects of product development. This aspect is not under consideration for 1995 sponsorship.

Continued piloting of the model and appraisal method, as well as other industry events, will continue beyond 1994 to obtain feedback and change requests on this first public version of the model.

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 discussions of the model in Chapter 4.

In this chapter

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

2.1 SE-CMM Foundations

Introduction

In this section, the fundamental concepts that have guided the development of the SE-CMM are presented, and the SE-CMM approved requirements related to those concepts are cited. The requirement number most closely related to the discussion is included at the end of the block in parentheses. The complete set of SE-CMM requirements is 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. Subsequent versions of the SE-CMM may expand to include other dimensions, i.e., human resources, and engineering technology.

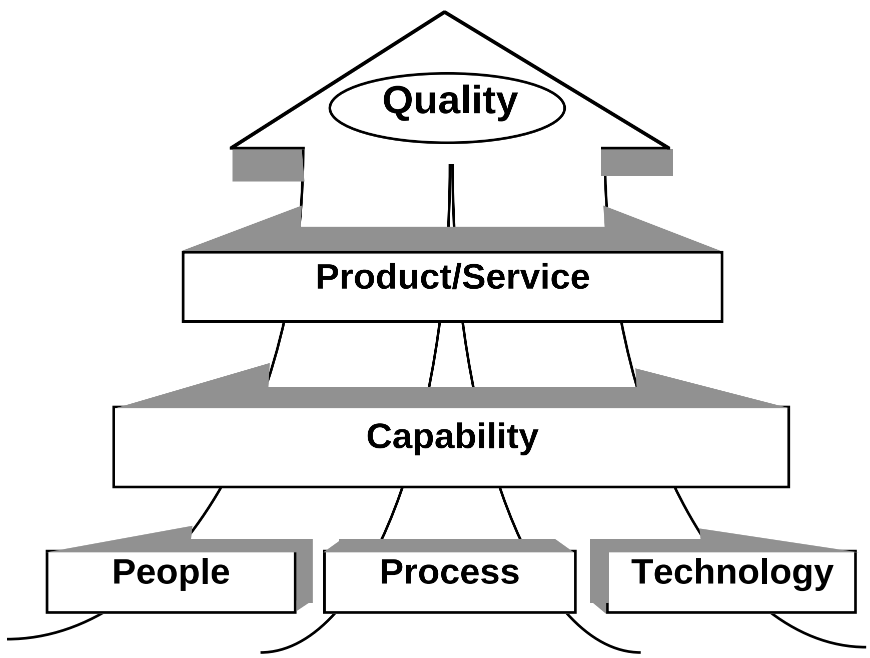

Figure 2-1. Critical Dimensions of Organizational Capability

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 (req't 4.3.1).

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. The current version of the model covers the system life cycle from the customer's identification of need through verification of the initial product. (req'ts 4.4, 6.1.2).

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. There are many such disciplines, and the authors recognize that many systems engineers primarily contribute to the systems development effort via their participation from specialty viewpoints. The model requires the integration of the engineering disciplines and specialties, whichever ones are necessary and appropriate for a particular product development. (req't 4.4)

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 (req't 6.1.1, 4.1)

Flexible architecture

The model architecture, shown in Figure 2-2, below, separates the systems engineering process areas (on the domain side) from the 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

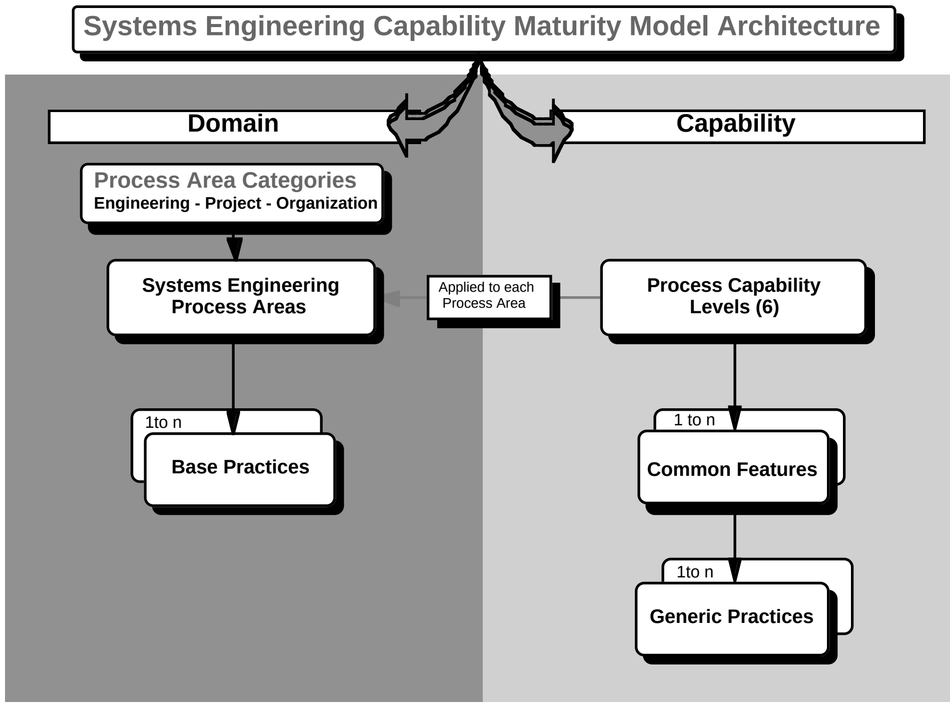

Figure 2-2. Model Architecture

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. The clear distinction between essential, basic systems engineering elements (the domain side) and process management-focused elements (the capability side) facilitates an organized approach to process improvement (req't 6.1.4).

Range of applicability

The SE-CMM has a wide range of applicability. The SE-CMM is 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 based on field data from each application (req't 4.2, 4.5.1).

continued on next page

2.1 SE-CMM Foundations, Continued

Capture and gain leverage from existing & emerging standards

One of the design goals of the SE-CMM effort is 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. For example, the architecture used in the SE-CMM is an adaptation of the ISO SPICE (Software Process Improvement Capability dEtermination) Baseline Practices Guide (BPG). The BPG is a document under development at the time of this writing, and references to it in this text are shown as (SPICE). The version referred to in this document is BPG v1.00a. Information on obtaining the BPG is available from M. Konrad at the SEI in Pittsburgh, PA, or from the SE-CMM Project Office.

SE-CMM-94-09|CMU/SEI-94-TR-26, *Relationships between the SE-CMM and Other Products*, provides cross-reference information between the SE-CMM and related systems engineering and process standards (req't 3.2).

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 to effectively improve and assess the systems and software engineering processes of a project or organization in the future. SECMM-94-09|CMU/SEI-94-TR-26, *Relationship between the SE-CMM and Other Products*, will contain information on this interface (req't 6.2.1.2, 3.2).

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, when applied and interpreted within an organizational and product/project context unique to the business entity using it, will enhance the resulting systems engineering processes without necessarily driving changes in culture or product context. This approach supports the desire to use the SE-CMM in a wide spectrum of organizational contexts. (req't 4.2)

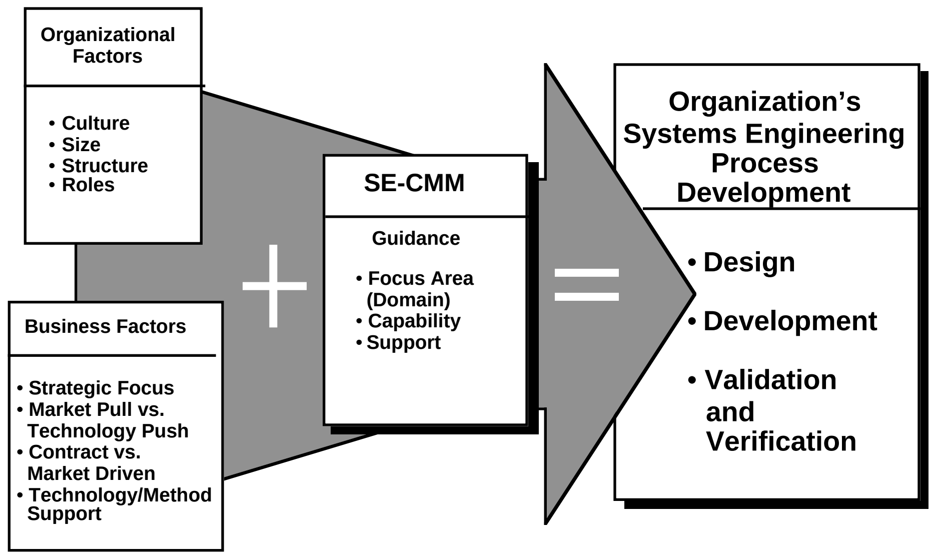

Figure 2-3. Focus of the SE-CMM

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
- Maturity model

Organizations and projects

Two terms are used within the SE-CMM to differentiate different 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 main point of the term "organization" is to connote the fact that an infrastructure to support common strategic, business, and process-related functions exists and must be maintained for the business to be effective in producing, delivering, supporting, and marketing its products.

continued on next page

2.2 Key Concepts of the SE-CMM, Continued

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 the section on domain-specific aspects of the SE-CMM later in this chapter. 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
1) An integrated composite of people, products, and processes that provide a capability to satisfy a need or objective.
2) 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.

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. A system may be a product that is hardware only, hardware/software, software only, or a service. 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

2.2 Key Concepts of the SE-CMM, Continued

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 full definition of process includes not only the activities and input and output artifacts of each activity, but also the mechanisms to control the performance of the activities. A performed process may follow a defined process, but probably not exactly. A performed process may also occur without any pre-defined process.

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, and
- provide information for the next product development or acquisition phase.

The problem and success criteria are defined through requirements analysis, functional or other type of analysis/allocation, and systems analysis. Alternative solutions, evaluation of those alternatives, selection of the best life-cycle balanced solution, and the description of the solution are accomplished through synthesis and systems analysis. System development is controlled by integration, verification/validation and configuration management of the process.

This elaborated definition provides a richer context for understanding the process characteristics under discussion in the SE-CMM. Nevertheless, 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 ISO (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.

continued on next page

2.2 Key Concepts of the SE-CMM, Continued

Process area

A process area (PA) is defined as a purpose and set of related systems engineering process characteristics, which, when performed collectively, can achieve the 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 to be able to claim satisfaction of that PA.

Role independence

The process areas of the SE-CMM group 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 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 appraisal method is further described in Section 3.2)

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.

Maturity model

A 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.

It 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 Aspects 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.

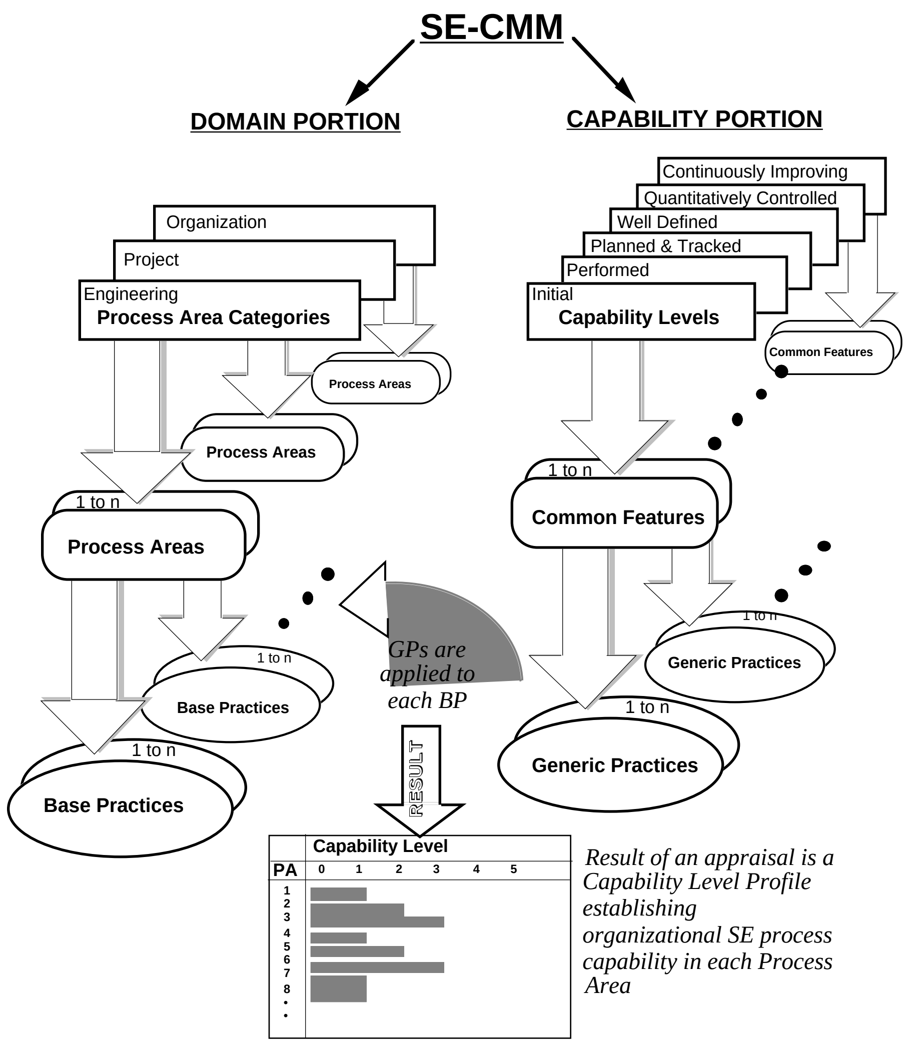

Figure 2-4. Diagram of SE-CMM Architecture

continued on next page

Dual-path architecture

The dual path architecture shown in Figure 2-5 was adopted with only slight modification from that chosen by the International Organization for Standards (ISO) for their Software Process Improvement Capability dEtermination (SPICE) Baseline Practices Guide. It was determined 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

The table below contains the basic definitions of the components of the capability aspect of the SE-CMM. They are further explained in the process capability section later in this chapter, 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 implementation 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 side 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 comprised 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

The table below lists the the capability levels and common features of the capability aspect of the SE-CMM:

Capability Level	Common Feature
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, derived requirements were developed that address particulars implied by the SPICE architecture. These derived requirements reflect mostly issues such as criteria for process area inclusion/exclusion, 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 applies 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 process areas. Each process area is further detailed by several base practices and explanatory notes. There are 17 process areas, which are grouped into 3 process categories: engineering, project, and organization.

The 17 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 of the process areas are likely to 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 someone performs these support tasks. 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.

continued on next page

Architectural components of the domain aspect

The table below 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

Process areas of the domain aspect

The table below lists the 17 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	Define Organization's Systems Engineering Process
Derive and Allocate Requirements	Manage Configurations	Improve Organization's Systems Engineering Processes
Develop Physical Architecture	Manage Risk	Manage Product Line Evolution
Integrate Disciplines	Monitor and Control Technical Effort	Manage Systems Engineering Support Environment
Integrate System	Plan Technical Effort	Manage Systems Engineering Training
Understand Customer Needs and Expectations		
Verify and Validate System		

Table 2-4. SE-CMM Process Areas

continued on next page

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 implied by the capability levels in the SPICE Baseline Practices Guide (BPG) (SPICE), which was adopted by the SE-CMM Project.

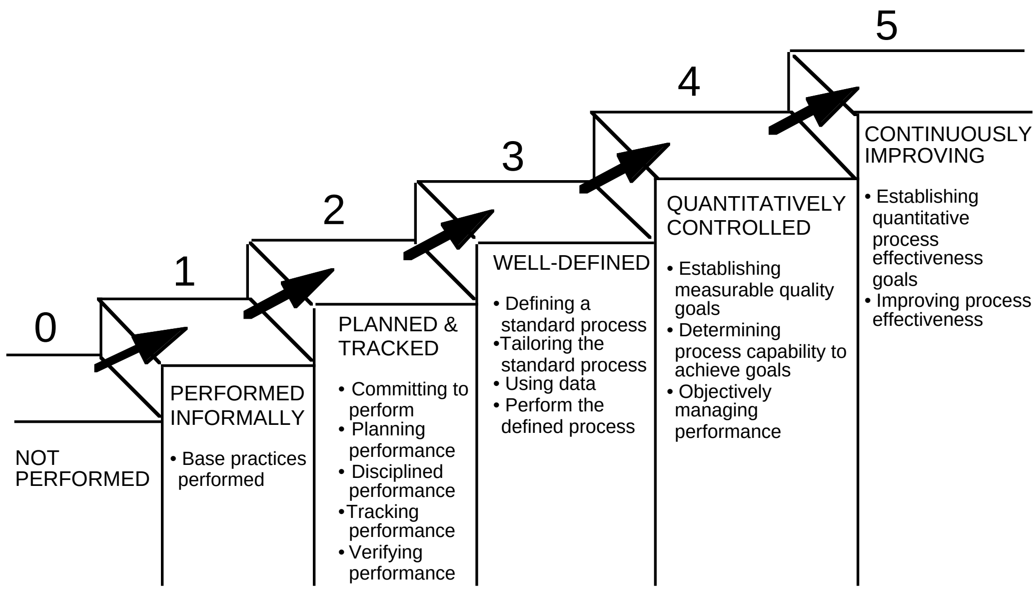

Figure 2-5. Improvement Path for Process Capability

Why group common features by capability level?

The following discussion on the ordering of the common features is adapted from ISO (SPICE) Baseline Practices Guide.

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

continued on next page

Why group common features by capability level?, continued

The ordering of the common features stems from the observation that some implementation and institutionalization practices benefit from the presence of others. This is especially true if institutionalization 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. As an example of this, before institutionalizing a specific estimation process for an entire organization, the organization should first attempt to use the estimation process on a project. 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 and Monitor and Control Technical Effort describe basic measurements that support systems engineering. The base practices of the Ensure Quality process area 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 collects the descriptions of the capability levels together 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.

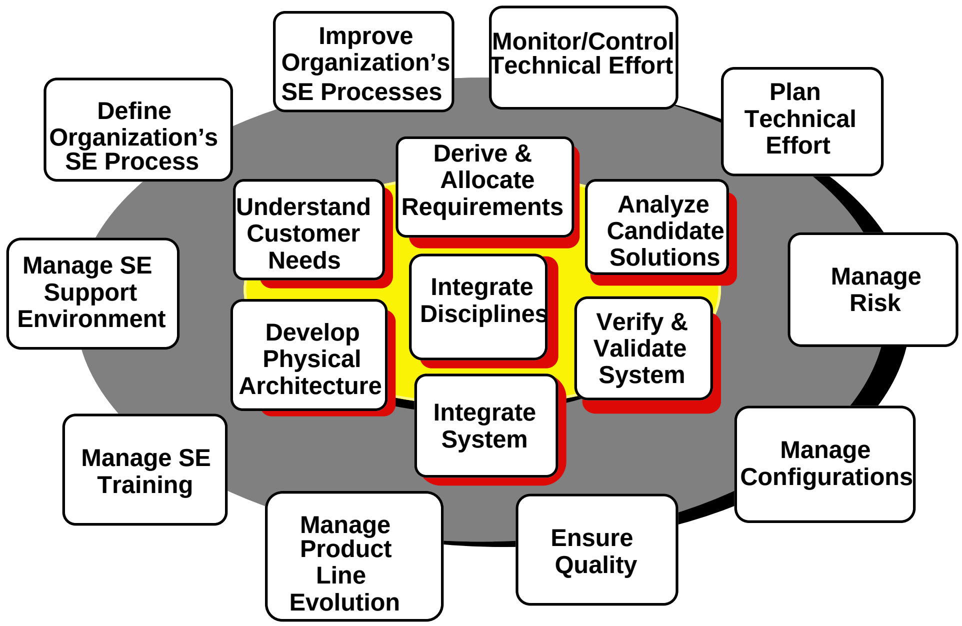

Figure 2-6. SE-CMM Process Areas

continued on next page

2.6 Domain Aspect of the SE-CMM, Continued

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 would, therefore, be used as a guide to the implemented process whenever the work product was 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.

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
- Develop Physical Architecture
- Integrate Disciplines
- Integrate System
- Understand Customer Needs and Expectations
- Verify and Validate system

In Chapter 4B, each of these is described in detail.

continued on next page

2.6 Domain Aspect of the SE-CMM, Continued

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

In Chapter 4B, each of these is described in detail.

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

- Define Organization's Systems Engineering Process
- Improve Organization's Systems Engineering Processes
- Manage Product Line Evolution
- Manage Systems Engineering Support Environment
- Manage Systems Engineering Training

In Chapter 4B, each of these is described in detail.

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. The following is a brief discussion of the rationale for including process areas about which the author team has received such inquiries.

Manage Configurations and Manage Systems Engineering Training 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 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 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. The Manage Risk process area 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.

continued on next page

Balancing the process areas and capability levels

Selection of the process areas to be included within the SE-CMM is a compromise between completeness and having a reasonable number of process areas to deal with when improving and appraising processes. Clearly, the essential elements of systems engineering must be included. In addition, there are activities which, even if they are not performed by system engineers, are crucial to the success of a systems engineering activity. For example, it would be difficult to appraise a systems engineering activity without knowing whether configuration management is consistently practiced and supported. In some cases, activities may be covered in the generic practices, but more detail specific to systems engineering may be desirable. 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.

Some of the process areas were chosen because they are common sources of difficulty in achieving quality results from the systems engineering activities, and thus require special emphasis. Some are 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, 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.

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/or expected by the SE-CMM authors 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, which 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.

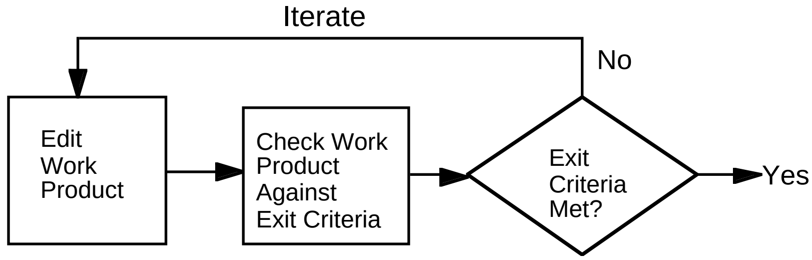

Figure 2-7. Iteration

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.

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.

continued on next page

Control and sequencing example

Figures 2-8 and 2-9 show a more complete example process which contains instances of iteration, concurrency, and recursion. In Figure 2-8 the context of a defined process for developing a system is shown. The activity called Develop System is exploded into greater detail in Figure 2-9.

Iteration is demonstrated in loops involving making a work product, checking or verifying the product and reporting exceptions back to the making activity for correction. One example of this iteration is in the loop Derive Requirements -> Verify Requirements -> Derive Requirements. Another is the overall loop Develop System -> Validate System -> Develop System.

Concurrency is demonstrated in the activities of Derive and Allocate Requirements and the activities of Develop Physical Architecture and Check Feasibility. Notice also that these concurrent activities exchange information as they proceed. Derived Requirements are furnished to Develop Physical Architecture to guide the analysis of candidate solutions. A Structure flows from Develop Physical Architecture to Allocate Requirements to use in the allocation process. Exceptions noted in the Check Feasibility activity are furnished to both Derive Requirements and Develop Physical Architecture so that necessary changes in their work products can be made.

Recursion is shown when the activity Develop System is called upon to develop each of the several subsystems described in the Physical Architecture and the Allocated Requirements. These instances of Develop System can proceed concurrently until they have produced the subsystems for the system. At that point, the concurrent tasks are joined together by Integrate System.

The output of Develop System is an integrated system ready for Validate System.

continued on next page

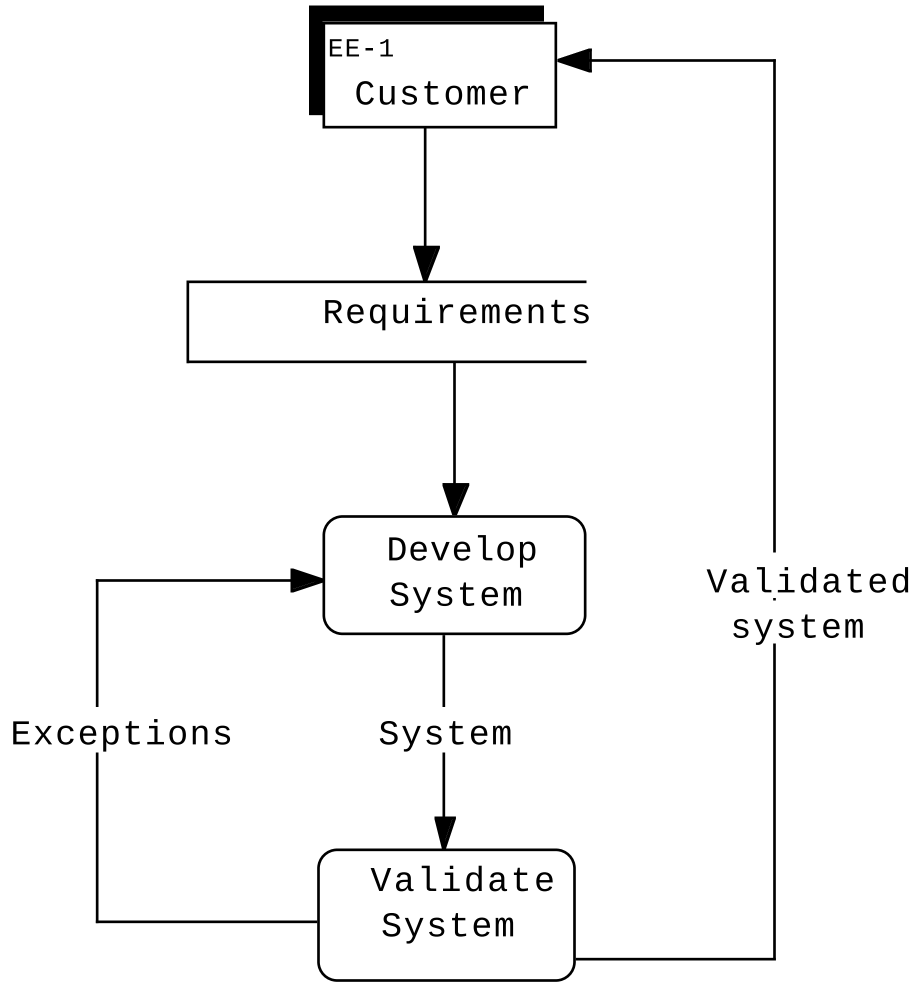

Figure 2-8. Sequencing Concepts Example

continued on next page

2.6 Domain Aspect of the SE-CMM, Continued

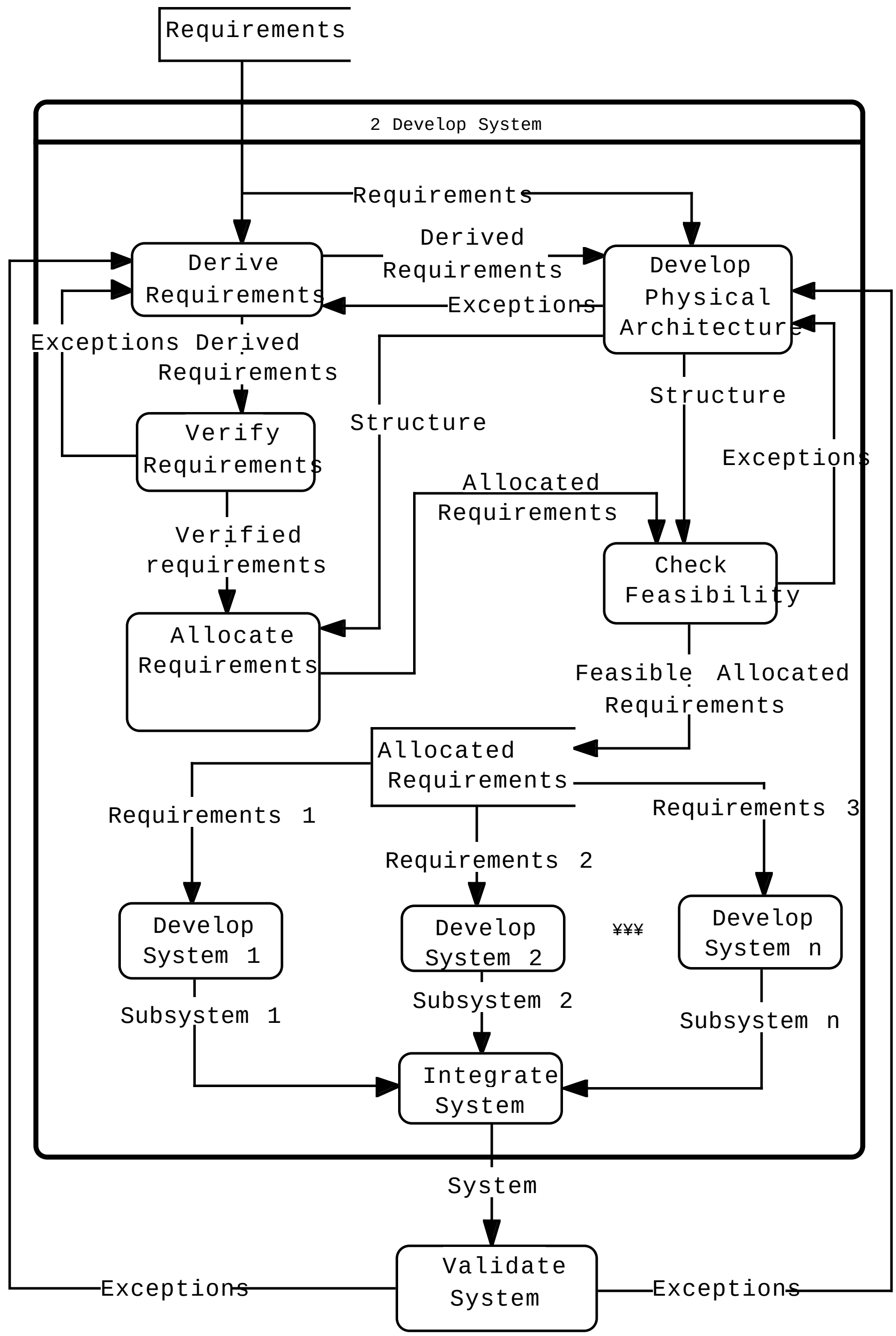

Figure 2-9. Example Process Showing Iteration, Concurrency, and Recursion

Chapter 3: Using the SE-CMM

Introduction

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

In this chapter

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 accomplish product development 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), and
- 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 appropriate use of the SE-CMM 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

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) will be fully described, along with some support materials for conducting appraisals, in SECMM-94-06|CMU/SEI-94-HB-05, *SE-CMM Appraisal Method Description.* (This document will be published early in 1995 by the project via Carnegie Mellon University's Software Engineering Institute, Pittsburgh, PA.) The basic premises of the appraisal method are listed here to provide a context for the reader as to 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, and
- 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 will be 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, others would be in other roles (e.g., configuration managers) that support systems engineering tasks.

continued on next page

Features of the SAM, continued

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.

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.

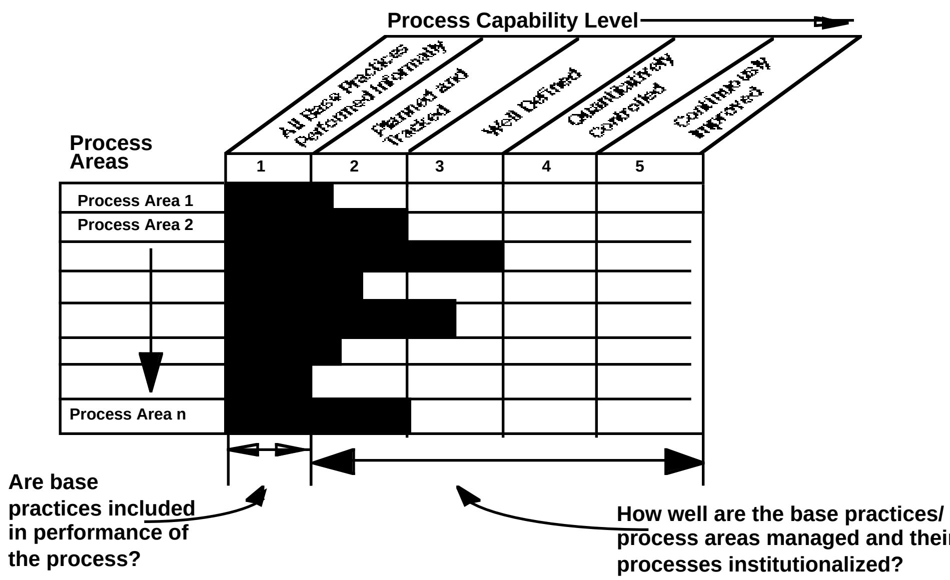

Figure 3-1. Determining Process Capability

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.

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 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 and the generic practice is determining whether or not the project's process for configuration management is resulting in action related to the process area under investigation, e.g., in relation to Derive and Allocate Requirements.

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. Base practices, on the other hand, generally maintain their interpretation over the scope of the model.

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 associated 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, there are several aspects of the SE-CMM that should be considered 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 are considered by the authors 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 process activity related to the unnecessary process area from the organization's systems engineering process for that specific project or the organization, as appropriate. Tailoring should, in all cases, be based on the organization's business goals and customer needs.

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.

continued on next page

3.3 Using the SE-CMM to Support Process Improvement, Continued

Gaining leverage from other experiences, continued

In the case of software process improvement, organizations that have done software process improvements for more than 3 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 the table below:

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 3-2. Process Improvement Principles in the SE-CMM

continued on next page

3.3 Using the SE-CMM to Support Process Improvement, Continued

Some expected results

Based on analogies in the software and other communities, some results that provide leverage to the organization in terms of process and product improvement can be predicted. These are discussed in the blocks 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, higher maturity level organizations 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. There are many sources for designing processes which can be referenced to support an organization's process design needs. This section sets a context for how the SE-CMM could be used in a design activity.

Analyzing your organizational context

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. There are many aspects of process design that are not addressed by the SE-CMM, since they are context specific. Nevertheless, these issues must be addressed when designing or improving processes for your organization. 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.

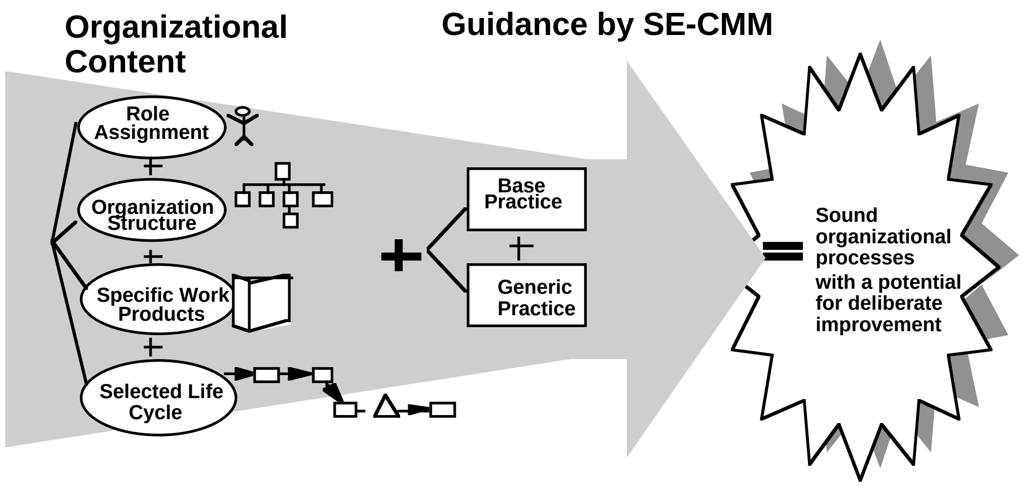

Figure 3-2. Factors for Successful Process Design

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, that is, the practices adapted from the ISO SPICE Baseline Practices Guide that are generic and apply to all processes. The generic practices (GPs) are used in a process appraisal to determine the capability of any process. The generic practices are grouped according to common feature and capability level.

"Process" vs. "process area"

The BPG uses the term "process" where the SE-CMM uses "process area."

Source

This chapter is reproduced with minor adaptations for the SE-CMM from the ISO (SPICE) Baseline Practices Guide v1.00a, with the permission of the BPG technical center manager. The BPG is a work in progress; therefore, the BPG development team would appreciate your comments on the generic practices in order to improve both the BPG and the SE-CMM. Comments on the generic practices may be made to the SE-CMM Project or directly to the BPG technical center manager, Michael D. Konrad, SEI, 4500 Fifth Ave., Pittsburgh, PA 15237; email: mdk@sei.cmu.edu.

In this chapter

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

Adaptations to the BPG

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

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. The performance of these base practices may not be rigorously planned and tracked. Performance depends on individual knowledge and effort. Work products of the process area testify to their performance. 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.

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 the "informal process." The customer(s) of the process area may be internal or external to the organization.

Capability Level 2 - Planned and Tracked

Description

Performance of the base practices in the process area is planned and tracked. Performance according to specified procedures is verified. Work products conform to specified standards and requirements. Measurement is used 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 managed.

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 process area 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: This practice is particularly related to process area 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 process covering a process area may be described in more than one way (e.g., policies, standards, and/or procedures), to cover a process area, and a process description may span more than one process area.

Relationship to other generic practices: This is the “level 2” process description. The process descriptions evolve with increasing process capability (see 3.1.1, 3.1.2, 5.2.3, 5.2.4 for descriptions of this process).

Standards and procedures that describe the process at this level are likely to include measurements, so that the performance can be tracked with measurement (see common feature 2.4).

Relationship to process areas: This practice is related to process areas 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, 5.2.4 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 described in PA 17 - Manage Systems Engineering Training.

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 process area 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.

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

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

Relationship to process areas: The typical practices needed to support systems engineering in the configuration management discipline are described in process area 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 process area 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 process area PA 02 - Develop Functional and Performance 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 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.

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

continued on next page

Common Feature 2.4: Tracking Performance, continued

2.4.2 Tracking Performance. 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 process areas: Project control is described in process area 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, policies, standards, and procedures are being established at an organizational level for common use, and are termed the "standard process definition."

More than one standard process description may be defined to cover a process area, as the processes in an organization need not correspond one to one with the process areas in this capability maturity model. Also, a defined process may span multiple process areas. The SE-CMM does not dictate the organization or structure of process descriptions. Therefore, more than one standard process may be defined to address the differences among application domains, customer constraints, etc. These are termed a "standard process family."

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 process area PA 13 - Define Organization's Systems Engineering Process.

continued on next page

Common Feature 3.1, continued

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

Note: Tailoring the organization's standard process creates the "level 3" process definition. For defined processes at the project level, 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 process area PA 13 - Define Organization's Systems Engineering Process.

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 "defined process" will typically be tailored from the organization's standard process definition. A well-defined process is one with policies, standards, inputs, entry criteria, activities, procedures, specified roles, measurements, validation, templates, outputs, and exit criteria that are documented, consistent, and complete.

Relationship to other generic practices: The organization's standard process definition is described in 3.1.1. The defined process is established through tailoring in 3.1.2.

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

Note: There is no process area for defect reviews, called "peer reviews" in ISO SPICE and the CMM for Software (in this regard, the SE-CMM differs from SPICE and the CMM for Software).

continued on next page

Common Feature 3.2, continued

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

Note: Measurement data that were first collected at level 2 are more actively used by this point, laying the foundation for quantitative management at the next level.

Relationship to other generic practices: This is an evolution of 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).

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 to the tactical needs of the project. The measures referred to here go beyond the traditional end-product measures. They are intended to imply sufficient understanding of the processes being used to enable intermediate goals for work product quality to be set and used.

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 performed in 3.2.1.

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.

Capability Level 5 - Continuously Improving

Description

Quantitative performance goals (targets) for process effectiveness and efficiency are established, based on the business goals of the organization. Continuous process improvement against these goals is enabled by quantitative feedback 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 (organization-level common feature)

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. Improving the standard process attacks common causes of variation.

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

Relationship to process areas: Organizational process improvement is managed in process area 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 pro-active causal analysis activity as well as re-active. 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, 5.2.3, and/or 5.2.4.

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

Note: Both common causes and special causes of variation are implied in this generic practice, and each type of defect may result in different action.

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 (5.2.2) 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: Product technology insertion is managed in PA 15 - Manage Product Line Evolution.

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. They are grouped alphabetically within the engineering, project, and organization categories.

Process Area Format

Current contents

At present, the SE-CMM domain aspect consists of 17 process areas (PAs), each of which contains a number of base practices. Each process area is identified 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

Process Area Format, Continued

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 considered essential to the practice 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 result in the selection of a solution to meet the specified constraints of the situation that generated the need for analysis. Analyze Candidate Solutions involves defining the approach and evaluation criteria for the analysis, as well as for choosing, selecting, and studying the candidate solutions. Communication of the rationale and results of the analysis must also be accomplished.

Process area notes

Analyze Candidate Solutions may be invoked from any of the other process areas. Whenever another activity requires that a choice be made from several alternatives to satisfy one or more constraints, this PA identifies the characteristics that a process for choosing a solution will exhibit.

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 decisions,
- life-cycle cost decisions,
- human factors decisions, and
- 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.

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

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.

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, and
- elicitation of expert judgment.

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.

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
- Trade study candidate analyses.

Notes
An example: 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 clearly the best among the alternatives evaluated, while minimizing the associated risk and uncertainty. The results of the analyses should be incorporated in a decision-making process to select the preferred alternative(s) which will be carried forward in the process.

Typical Work Products
• Trade study.
• Preferred solution rationale.

Notes
The following questions will usually arise when selecting among alternative solutions: (1) How much better is the selected approach to the next best alternative? Is there a significant difference between the results of the comparative evaluation? (2) Have all feasible alternatives been considered? (3) What are the areas of risk and uncertainty?

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 the decision database. The disposition of each alternative under consideration and the rationale for the disposition should be documented in the decision database.

Typical Work Products
• Evaluation of trade study alternatives.
• Mathematical models of appropriate solutions.
• Reports of prototype operation.
• Results of trade-off 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, and
• an engineering log book.

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, 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. This analysis involves addressing the concept of operations, functional partitioning, and performance allocation, as well as capturing the status and traceability of requirements.

Process area notes

The practices in the Derive and Allocate Requirements process area operate in parallel with the practices in Develop Physical Architecture (PA 03). Potential derived requirements are evaluated for feasibility against the functional partitions and are evaluated iteratively against the components of the physical 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 Understand Customer Needs and Expectations or Analyze Candidate Solutions.

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, or performance.

BP.02.03 Partition requirements into groups of requirements 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.

BP.02.06 Allocate requirements to functional partitions, system elements, people, and 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 system stimuli and responses. The stimuli received by the system from users, other systems, or the environment are identified and the system response captured. The captured 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 include
- develop a prototype of the user interface and capture vignettes of user interaction, and
- develop a system simulation.

Development and analysis of operational concepts are valuable tools used in the practices of Understand Customer Needs and Expectations and Derive and Allocate Requirements. 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 processes.

continued on next page

BP 02.02 Identify Key Requirement Issues

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

Description

In analyzing system and derived requirements, requirements are often identified that have an especially strong influence on the cost, development schedule, or performance of a product. The total set of requirements are screened for potential key requirements. These requirements are referred to the process areas Analyze Candidate Solutions and Develop Physical Architecture for cost benefit analyses. The results of analyzing key requirements may be reviewed with the customer using the methods of understanding customer needs and expectations. Key requirements that show a relatively low benefit to cost ratio are candidates for negotiation with the customer. Key requirements that show a high benefit to cost ratio are assessed for level of difficulty and may be subject to risk management considerations.

Typical Work Products

- Key requirements issues.
- Benefit to cost sensitivity analyses for key requirements.

Notes

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

The activities of identifying key requirements are closely related to the activities of the process areas Understand Customer Needs and Expectations, Analyze Candidate Solutions, and Develop Physical Architecture.

continued on next page

BP 02.03 Partition Functions

Partition requirements into groups of requirements 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 include
- group all requirements that apply to user interaction, and
- group all requirements that apply to data storage and retrieval.

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.

Description

The identification of external and internal interfaces is conducted throughout the analysis of system requirements and 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 and 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 partitioning. After functional 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 the Develop Detailed Operational Concept base practice (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.

continued on next page

BP 02.06 Allocate Requirements

Allocate requirements to functional partitions, system elements, people, and support elements to support synthesis of solutions.

Description
The purpose of this practice is to facilitate the separate development of system elements and components at successively lower partitions. Requirements are initially allocated to functional partitions and subfunctions and ultimately to system elements and components. The allocations are performed so that the implementation of allocated derived requirements by the associated system elements is both necessary and sufficient 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 from this joint requirement separate requirements for each system element involved..

Alternatives should be considered relative to allocating 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.
• Requirements allocation attributes.

Notes
Examples 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 uniquely allocate these requirements to the appropriate system element.

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, Derive and Allocate Requirements, and Develop Physical Architecture 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 the characteristics indicating that it can be verified in order to prove that the resulting product meets the intended purpose. Evaluating the feasibility of verifying a potential requirement facilitates producing 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: Assess the verification feasibility for each requirement.

It is important to ensure that requirements verification is performed iteratively and recursively with the practices of verification and validation.

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 called 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 the creation, change, or verification of requirements (especially those of the process areas Understand Customer Needs and Expectations, Derive and Allocate Requirements, Develop Physical Architecture, and Verify and Validate System) 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
The capture of requirements, requirement partitions, derived requirements, requirement allocations, traceability, rationale, and status information, along with the dissemination and control of this information, 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 include
- Enter requirements, their traceability, allocation and status into a requirements data base.
- 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: Develop Physical Architecture

Summary description

The purpose of Develop Physical Architecture is to transform the functional architecture, as defined by the Derive and Allocate Requirements process area, into the physical architecture for the system. It involves deriving the physical architecture requirements, identifying the key design issues, determining the physical structure and interfaces, and allocating the physical architecture requirements. The practices described herein are expected to be performed iteratively until the design is handed off to the implementing or component engineering disciplines.

Process area notes

This process area generates candidate solutions and then makes use of the Analyze Candidate Solutions process area to choose an alternative that meets the needs of Develop Physical Architecture. This process area is performed iteratively with the process areas Understand Customer Needs and Expectations and Derive and Allocate Requirements.

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 physical architecture.
BP.03.02	Identify the key design issues that must be resolved to support successful development of the system.
BP.03.03	Generate physical structure alternative(s) and constraints, and select a solution in accordance with the Analyze Candidate Solutions process area.
BP.03.04	Develop the physical architecture's interface requirements for the chosen physical structure.
BP.03.05	Allocate the physical architecture requirements to the chosen physical structure.
BP.03.06	Maintain requirements traceability for the physical architecture 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 physical architecture by capturing the design results and rationale.

continued on next page

BP 03.01 Derive Physical Architecture Requirements

Derive the requirements for the physical architecture.

Description
This activity makes use of and iterates with a number of other activities, including development of system requirements, and makes use of other states, including the current state of the system functional and physical architectures. 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. Derived requirement types include performance, human interaction, production, maintenance, etc. Derived requirements may be applicable broadly or they may be applicable to specific subsystems or support elements.

Typical Work Products
• Derived architecture requirements.

Notes
Derived requirements for the system's physical architecture apply to the actual subsystems, configuration items, or components as distinguished from functional or notional applicability.

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.

Typical Work Products
• List of key design issues.

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 Physical Structure

Generate physical structure alternative(s) and constraints, and select a solution in accordance with the practices of the Analyze Candidate Solutions process area.

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

Typical Work Products
• Physical structure.
• Subsystems.
• Major assemblies.
• Identified interfaces.

Notes
The identified elements of the system's physical 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 physical structure or its elements may be a requirement.

BP 03.04 Develop Physical Interface Requirements

Develop the physical architecture's interface requirements for the chosen physical structure.

Description
External and internal interfaces are identified that allow development of a complete set of physical architecture requirements. Alternative solutions that satisfy interface constraints are developed. A solution is selected in accordance with the practices of the Analyze Candidate Solutions process area.

Typical Work Products
• Interface requirements.
• User interface requirements.
• Environmental interface requirements.
• Subsystem interface requirements.

continued on next page

BP 03.04 Develop Physical Interface Requirements, continued

Notes
The physical 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 physical architecture. Generally, all or part of the external interface requirements may be known prior to selection of the physical architecture. Internal interface requirements are typically deferred until after selection of the physical architecture.

BP 03.05 Allocate Physical Requirements

Allocate the physical architecture requirements to the chosen physical structure.

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

Typical Work Products
• Allocated requirements.

Notes
Examples 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 Sufficiency and Traceability

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

Description
This practice captures, maintains, and controls the traceability and status of requirements throughout the product life cycle. Derived requirements levied on the physical 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. An assessment of the lower level physical architecture requirements and the validity of their traceability is conducted continuously 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 physical architecture are captured in the same traceability data set.

Examples 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 physical architecture by capturing the design results and rationale.

Description
The captured physical architecture includes the physical 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. The capture, baselining, and dissemination of the physical architecture description is essential to developing and verifying a system that 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, and
• data flow or control flow diagrams.

End of PA 03: Develop Physical 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 forwarded and considered, but the focus on total system development is maintained. These disciplines may include, but are not limited to, marketing, manufacturing, component design, development (e.g., hardware, software, or firmware), reliability, maintainability, 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 environment must persist throughout the system development 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 development staff. The goal is to eliminate nonessential information while providing essential information to members of the development staff, on a timely basis.

Base practices list

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

BP.04.01	Identify the disciplines that are directly or indirectly essential to system development.
BP.04.02	Familiarize individuals involved in the development effort with the various disciplines and their roles in creating a successful system.
BP.04.03	Actively promote cross-discipline understanding within the development team.
BP.04.04	Establish methods for interdisciplinary coordination.
BP.04.05	Establish methods for identifying and resolving interdisciplinary issues.
BP.04.06	Follow established interdisciplinary methods to achieve integrated solutions to identified issues or problems.
BP.04.07	Communicate results of interdisciplinary activities to affected groups.
BP.04.08	Develop project goals and ensure that each project member and direct support person is fully aware of them.

continued on next page

BP 04.01 Identify Essential Disciplines

Identify the disciplines that are directly or indirectly essential to system development.

Description
Efficient and effective systems result from a blending of the efforts of people from many unique disciplines. The earlier that affected disciplines can be identified and their input into the development effort captured, the more satisfying the product will be to both the development and user community.

Typical Work Products
• Roster of essential disciplines.
• List of representatives from each discipline.

Notes
As the development effort proceeds through its life cycle, the number of critical disciplines is a variable. The initial focus should be on attaining complete coverage, not limiting participants. Disciplines not affected will usually eliminate themselves from the roster, over time. However, 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 development life cycle.

BP 04.02 Train Interdiscip-linary Roles

Familiarize individuals involved in the development effort with the various disciplines and their roles in creating a successful system.

Description
Each individual must be introduced to the roles and responsibilities that the various representatives from the essential disciplines have in the development effort. The contribution that each representative makes to the effort should be clear. How the group, as a whole, melds those unique contributions in an effective system solution should be understood and practiced.

Typical Work Products
• Description of roles, responsibilities, and functions of representatives.

continued on next page

BP 04.02 Train Interdisciplinary Roles, continued

Notes
It is extremely important to encourage informal interaction between representatives from the various disciplines. The synergism resulting from social situations (e.g., the discussion of design problems over coffee or lunch) creates an atmosphere of group collaboration rather than group competition and/or placing blame or responsibility for problems on other groups and individuals. Formal meetings being the only venue for communication can often reinforce competitive behaviors. One of the most difficult tasks confronting a systems engineer is to create a collaborative atmosphere in the formal meeting situation.

BP 04.03 Foster Cross-Discipline Understanding

Actively promote cross-discipline understanding within the development team.

Description
Members of the development team need to become familiar with the issues that are important to the disciplines essential to the system development and the effect 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 development effort. 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.

Typical Work Products
• Pamphlets describing each discipline.
• Briefings to familiarize the development team with lessons learned.

Notes
This is often one of the most overlooked areas in the list of systems engineering tasks; yet if 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. The caution is to remember that 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 how proper consideration of each concern has a positive impact on the quality of the group’s product.

continued on next page

BP 04.03 Foster Cross-Discipline Understanding, continued

Examples include
- Hold a meeting at the inception of the project/program at which representatives of the identified development disciplines share their issues.
- Summarize the issues of each discipline in a one- or two-page paper.
- Distribute this paper to all.

BP 04.04 Establish Coordination Methods

Establish methods for interdisciplinary coordination.

Description
In addition to the roles and what information to share, members of the product development team must know how to share knowledge, i.e., the particular nuts and bolts of getting information from an individual or group to others who need it.

Typical Work Products
- Integrated development coordination methods.

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

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

continued on next page

BP 04.05 Establish Resolution Methods

Establish methods for identifying and resolving interdisciplinary issues.

Description
Issues will arise during product development for which there is no simple solution. Pre-determined methods of resolving these issues must be known to the product development staff. Several resolution techniques must be available. 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.

Notes
Examples include
• Pugh's Controlled Convergence technique,
• quality function deployment technique,
• autocratic ediction, and
• arbitration and rules.

BP 04.06 Use Interdiscip-linary Methods

Follow established interdisciplinary methods to achieve integrated solutions to identified issues or problems.

Description
The product development staff must use the established methods to resolve issues. Attempts to circumvent the methods must be discouraged or incorporated (if the alternate method is agreed to be superior).

Typical Work Products
• Reports.

continued on next page

BP 04.07 Communicate Results

Communicate results of interdisciplinary activities to affected groups.

Description
The work of product development is making decisions. These decisions must be communicated to members of the product development staff who must make more decisions.

Typical Work Products
• Results of interdisciplinary activities.

Notes
Examples include
• electronic mail decisions with rationale, and
• use of the facilities of the project's selected automation tool set.

BP 04.08 Develop and Communicate Project Goals

Develop project goals and ensure that each project member and direct support person is 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.

Typical Work Products
• Project objectives.

Notes
Examples include
• a cost/schedule goal,
• a quality/cost goal, and
• 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 life cycle of system development.

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 can appear to be some redundancy between the process characteristics captured in this process area and some of those in Develop Physical Architecture (PA 03). However, the emphasis in PA 03 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.

The process characteristics captured in this process area run concurrently, iteratively, and/or recursively with other process characteristics captured in other process areas.

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 descriptions of the interfaces implied by the system architecture.
BP.05.02	Communicate the interface definitions and coordinate change requests with all system element developers who could be affected by interface changes.
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 requirements 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 which identifies the optimal sequence for receipt, assembly, and activation of the various components that make up the physical architecture of the system.

BP 05.01 Define Interfaces

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

Description
System and subsystem interfaces are defined as early as possible in the development effort. Interface descriptions address logical, physical, electrical, mechanical, and environmental parameters as appropriate. The bulk of integration problems arise from unknown or uncontrolled aspects of interfaces. 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.

Typical Work Products
• Interface descriptions.

continued on next page

BP 05.02 Control Interfaces

Communicate the interface definitions and coordinate change requests with all system element developers who could be affected by interface changes.

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. Designs are audited to verify compliance with the defined interface requirements.

Typical Work Products
• Interface control documents.
• Exception reports.

Notes
The change control and coordination mechanism 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 required elements. Some method of assessing the timeliness of receipt of system elements will need to be in place.

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 by the organization that will assemble the system or sub-system 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
• Validated system elements.
• Exception reports.

Notes
Examples 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 requirements 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 definition. Such verification may be by test, inspection, analysis, etc., and may be executed 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 include
• Elements are inspected and/or tested to verify that the interfaces were implemented in accordance with the defined interface requirements.
• Compliance or deficiency reports are prepared.

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, and is to occur as needed here.

Typical Work Products
• Integration reports.
• Exception reports.

Notes
Examples include
• subsystem build, and
• 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 the assembly of the system elements into the final system is conducted and tested in accordance with a planned strategy. System testing is explicitly addressed in the Verify and Validate System process area, and is to occur as needed here.

Typical Work Products
• Integration reports.
• Integrated system.

Notes
An example: Verify system behavior.

continued on next page

BP 05.08 Develop Integration Strategy

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

Description
Using business as well as technical factors, the strategy must focus on the need 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 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.
- Selected assembly sequence and rationale.

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, and
- 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 their 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.

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 customer needs, expectations, and measures of effectiveness.
BP.06.02	Analyze customer needs and expectations to develop a preliminary operational concept of the system as appropriate.
BP.06.03	Develop a statement of system requirements.
BP.06.04	Obtain concurrence from the customer that the agreed upon customer 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 customer 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 to accomplish this is used throughout the life of the project. During this process, an effort is made to identify any unique end-user needs and expectations which may exist, and to obtain customer approval to include them, or justification for their omission. 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.; and
- observation of existing systems, environments, and workflow patterns.

continued on next page

BP 06.02 Analyze Needs

Analyze customer needs and expectations to develop a preliminary operational concept of the system as appropriate.

Description
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. Customer needs and expectations may need to be probed to determine that adequate understanding and correct prioritization has occurred.

Expression of the logistics concept, support concept, maintenance concept, training concept, 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),
- and prototype.

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, these 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 company standard, or they may be informally captured.

BP 06.04 Obtain Concurrence

Obtain concurrence from the customer that the agreed upon 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, and these understandings to which the customer committed are updated throughout the life of the project.

Typical Work Products
• Validated system requirements.

continued on next page

BP 06.04 Obtain Concurrence, continued

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, and.
- beta tests.

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

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 customer needs are being analyzed and decisions on general approaches are being made. 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.

Notes
Examples of forums to inform the customer include
- working groups,
- normal program reviews,
- payment milestones,
- in-process reviews,
- status meetings,
- weekly telephone conferences,
- focus groups, and
- 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 system element development and integration. The scope of verification covers development of the full system, as well as its production, operation and support. 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.

Process area notes

Means of evaluation associated with verification and validation 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 and validation address the work products of the process areas Understand Customer Needs and Expectations, Analyze Candidate Solutions, Derive and Allocate Requirements, Develop Physical Architecture, and Integrate System. 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 PA 11: Monitor and Control Technical Effort.

Validation is a formal evaluation in the most realistic operational environment achievable, including personnel, procedures, 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 are verified against established criteria or requirements 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.

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.

Typical Work Products

- Master test and evaluation plan.
- System test plan.
- Operational test and evaluation plan.

continued on next page

BP 07.01 Establish Verification and Validation Plans, continued

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.

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 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; and 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.

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 and environment, system configuration, and in the case of testing, inputs, outputs, expected results, and evaluation criteria for each product requirement or group of requirements that the developed system is to be evaluated against.

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 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 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 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 representative operational environment personnel.

---

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
Verification and validation of incremental work products, subsystems, components, and systems is performed, beginning early in project, according to the verification and validation plans and 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 form 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, and reverification of problem areas.

Typical Work Products
- Inspection results.
- Test results.
- System validation data.
- Validation exception reports.

Notes
Example practices include
- Validate system requirements.
- Conduct reviews of 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.

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 produced on a continuous basis if a process exists to continuously measure and improve quality, and this process is adhered to rigorously. Key aspects of the process required to develop high quality systems are measurement, analysis, and corrective action.

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. The support of a quality management process adds confidence for the developers, management, and customers that all aspects of quality management are seriously considered and acted upon by the organization and reflected in its products.

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.

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 development 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 Promote atmosphere that encourages employees to be attentive to quality issues and report quality problems.
BP.08.06 Initiate activities that address identified quality issues or quality improvement opportunities.

continued on next page

BP 08.01 Monitor Conformance to the Defined Process

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

Description
The purpose of this practice is to ensure that the project's execution follows the defined system engineering process. 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.

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

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 allows estimation 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.

continued on next page

BP 08.02 Measure Quality of the Work Product, continued

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, and
  - predicted technical variance.

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 promotes 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 that, in addition to defining the process, can be used to determine which characteristics should be measured, and identify potential sources of variation.
- Statistical process control on process parameters.
- Design for experiments.

continued on next page

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.

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 Foster Quality Environment

Promote atmosphere that encourages employees to be attentive to quality issues and report quality problems.

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. Quality ideas need to be encouraged and a forum needs to exist that allows each employee to raise quality issues freely.

Typical Work Products
- Environment that promotes quality.
- Captured inputs and resolutions from workers.

Notes
A quality environment can be fostered by
- quality circles, and
- a quality assurance group with a reporting chain of command that is independent of the project.

BP 08.06 Initiate Quality Improve-ment 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 are inefficient or jeopardize product or process quality need to be identified and corrected. This would include the identification and elimination, or reduction, of cumbersome or bureaucratic systems.

Typical Work Products
- Recommendations for improvement of 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.

End of PA 08: Ensure Quality

PA 09: Manage Configurations

Summary description

The purpose of Manage Configurations is to maintain data 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 desired to be 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 PA 02, Derive and Allocate Requirements.

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 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 configuration data.
BP.09.04 Control changes to established configuration units.
BP.09.05 Communicate changes to status, proposed changes, and configuration data to affected groups.

continued on next page

BP 09.01 Establish Configuration Management Methodology

Decide among candidate methods for configuration management.

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

The Analyze Candidate Solutions process area 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.

Notes
Example criteria for selecting configuration units at the appropriate work product level, which will affect the level of design visibility, include
- maintaining interfaces at a manageable level,
- unique user requirements such as field replacement,
- new versus modified design, and
- expected rate of change.

Example criteria for determining when to place work products under configuration management include
- portion of the development life cycle that the project is in,
- degree of formalization selected,
- cost and schedule limitations, and
- 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, and
- 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 line replaceable unit (LRU) level.

BP 09.03 Maintain Configuration Data

Maintain a repository of configuration data.

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.

continued on next page

BP 09.03 Maintain Configuration Data, continued

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.

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.

Analyze identified problems with the work product, or the request to change the work product 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, identify a schedule for incorporating the change into the work product and other impacted 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.

continued on next page

BP 09.05 Communicate Configuration Status

Communicate changes to status, proposed changes, and configuration data 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 impacted work products.

Typical Work Products
• Change reports.

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 Plan Technical Effort and Monitor and Control Technical Effort 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 the risk identification, assessment, mitigation, and monitoring activities 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 the risk identification, assessment, mitigation, and monitoring activities 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
The purpose of this base practice is to examine the project objectives, plans, 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.

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 and estimated cost of occurrence, and mitigation strategies. This practice is very useful in improving risk estimates and in reusing successful risk mitigations. [Charette 89]
• 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 (or benefit) of the risks previously identified. 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
The purpose of this base practice is to examine on a regular basis the results of the risk mitigations that have been put into effect and to measure the results and determine whether the mitigations have been successful.

Typical Work Products
• Risk status.

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, 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 resource utilization 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 technical parameter tracking and review activities to determine corrective actions.
BP.11.06 Take corrective actions when actual results deviate significantly from plans.

continued on next page

BP 11.01 Direct Technical Effort

Direct technical effort in accordance with technical management plans.

Description
The purpose of this base practice is to carry out technical management plans created in the Plan Technical Effort process area (PA 12). It involves technical direction of all of the engineering activities of the project.

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 resource utilization against technical management plans.

Description
The purpose of this base practice is to provide current information on resource utilization during the project to help adjust the effort and plans when needed.

Typical Work Products
• Tracks of resource utilization.

Notes
Tracking cost includes comparing the estimates documented in the pro 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 technical parameter tracking and review activities 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.
• Corrective action recommendations.

Notes
Integrate new information collected with historical project data. Identify trends that are hurting the project and new issues that indicate risks to project 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 Control Technical Effort

Take corrective actions when actual results deviate significantly 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 to revise the plan.

Typical Work Products
• Resource allocations.
• Changes to methods and procedures.
• Exception reports.

Notes
This base practice covers whatever actions are needed to correct the problems discovered. The possible actions taken under this base practice are varied and numerous, but changes to the technical management plan are made in the Plan Technical Effort process area.

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 the system development. 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 and the constraints, risks, and goals that define and bound the project. The planning process includes steps to estimate the size of work products and 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 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.
BP.12.11	Obtain commitment to the technical management plans from all affected groups and individuals.

continued on next page

BP 12.01 Identify Critical Resources

Identify resources 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 score 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.

continued on next page

BP 12.02 Estimate Project Scope, continued

Notes
Example practice: Analyze the available project documentation and interview project personnel to obtain what the main technical constraints and assumptions are. 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.

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. In the process definition, include process activities, inputs, outputs, sequences, and quality measures for process and work products. Typical life-cycle models include waterfall, evolutionary spiral, and incremental.

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 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 (such as IEEE P1220), known best practice within the industry segment, or reference models such as the SE-CMM, as well as from 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.

continued on next page

BP 12.05 Identify Technical Activities, continued

For the systems engineering activities, decompose activities planned for the next 3 months until each activity is approximately 2 weeks in duration. Activities 3 to 12 months away should be approximately a month in duration. Activities starting more than a year away can be described at a very high level, approximately 2 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 (PA 04).

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 company that the project needs to interact with in order to be successful. For each group, perform the base practices of PA04, Integrate Disciplines, to define and implement each interface in terms of interaction mechanisms, interaction frequency, and problem resolution mechanisms. Perform the above steps for the systems engineering effort to ensure effective interaction between the systems engineering activities and interfacing groups.

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, procured items, training, and 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 include slack time appropriate for identified risks. Review and commitment on schedules from all affected parties is essential.

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 utilizes 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.

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 selected for monitoring should have pre-planned assessments scheduled at appropriate 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, and
- 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, and
• plans for interacting with other organizations (e.g., subcontractors) performing the technical effort.

continued on next page

BP 12.10 Review Project Plans

Review the technical management plans with all affected groups and individuals.

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. These inputs are incorporated or feedback is provided on rejected or modified inputs in order to build team ownership of the plans. Interim and completed project plans are distributed for review.

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, and
- subcontractors.

Example: 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 each issue until it is resolved.

continued on next page

BP 12.11 Commit to Project's Plans

Obtain commitment to the technical management plans from all affected groups and individuals.

Description
Develop clear objectives and shared understanding of the project's intent throughout the organization. Include the goal of early conflict resolution.

Notes
Example: Document the process for developing the technical management plans and communicate the process to the affected groups. Get buy-in to the planning approach by asking for and resolving concerns about the process. Encourage each group to review the plans in accordance with the base practice above. Work with each group, as needed, to include the applicable portions of the technical management plans in the planning documents for that group.

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 the process that will meet the business goals of the organization, as well as designing, developing, and documenting organizational process assets which are collected and maintained. Process assets is a term used to emphasize the investment nature of defining organizational processes; 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 process are covered in the process area Improve Organization's Systems Engineering Processes.

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 the organization's standard systems engineering process.
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

- Organization's systems engineering process goals.
- Requirements for the organization's standard systems engineering process.
- Requirements for the organization's 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 to be followed during the development of the system. It should contain examples of processes that have been defined, and used together with the measurements of the process execution. 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.

The 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, and
- the project's process measurements.

continued on next page

PA 13: Define Organization's Systems Engineering Process, Continued

BP 13.03 Develop Organization's Systems Engineering Process

Develop the organization's standard systems engineering process.

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.

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

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 the bypassing of substantial and important practices prescribed by organization policy or of standards that must be followed.

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 people; 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.

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 an analysis of 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.

BP 14.02 Plan Process Improvements

Plan improvements to the organization's processes based on an analysis of 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 the improvements that will provide the most improvement 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.

Typical Work Products
• Process improvement plan.

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

continued on next page

PA 14: Improve Organization's Systems Engineering Processes, Continued

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 the incorporation of the improvements in new project’s processes.

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
The process improvements may be useful to existing projects and they can incorporate the useful ones 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.

End of PA 14: Improve Organization's Systems Engineering Processes

PA 15: Manage Product Line Evolution

Summary description

The purpose of Manage Product Line Evolution is to establish and provide the necessary resources for acquiring, developing, and applying technology to a product line for competitive advantage.

Process area notes

The Manage Product Line Evolution process area is needed ". . . to ensure 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 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	Manage the insertion of new technology into product development, marketing, and manufacturing processes.

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 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.

Typical Work Products
• Reviews of product line technology.

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 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.

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 Manage Product Technology Insertion

Manage the insertion of new technology into product development, marketing, and manufacturing processes.

Description
Manage the introduction of new technology into the product lines, including both the 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. The insertion of development and process technology into the environment is executed with a goal of minimizing disruption of development activities while upgrading to make new technology available.

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, organization's structures, policies, procedures, machine shops, chemical process facilities, environment stress facilities, and/or 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 business goals.
BP.16.02 Determine requirements for the organization’s systems engineering support environment based on organizational needs.
BP.16.03 Assess the systems engineering support environment against the support environment requirements.
BP.16.04 Obtain a systems engineering support environment that meets the requirements for supporting systems engineering by using the practices in the Analyze Candidate Solutions process area.
BP.16.05 Tailor the systems engineering support environment to individual project’s needs.
BP.16.06 Insert new technologies into the systems engineering support environment based on the organization's business goals and the projects’ needs.
BP.16.07 Maintain the systems engineering support environment to continuously support the projects dependent on it.
BP.16.08 Monitor the systems engineering support environment for improvement opportunities.

continued on next page

PA 16: Manage Systems Engineering Support Environment, Continued

---

BP 16.01 Maintain Technical Awareness

Maintain awareness of the technologies that support the organization's business goals.

Description
To insert new technology, a sufficient awareness of new technology must be present in the organization. Such awareness may be maintained internally or purchased. Awareness of the current state of the art or state of the practice is a necessary element for rational assessment of improvement options.

Typical Work Products
• Reviews of support environment technology.

Notes
Maintaining awareness may be accomplished by reading industry journals or participating in professional societies.

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 Assess Support Environment

Assess the systems engineering support environment against the support environment requirements.

Description
To insert new technology, the difference between the environment that an organization currently uses and the environment that is available for use must be known.

Typical Work Products
• Systems engineering support environment assessment.

Notes
Independently assess several aspects of the support environment of systems engineering via inspection or survey.

BP 16.04 Obtain Systems Engineering Support Environment

Obtain a systems engineering support environment that meets the requirements for supporting systems engineering 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. Select a solution using the practices in the Analyze Candidate Solutions process area. 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, new facilities, etc.

continued on next page

PA 16: Manage Systems Engineering Support Environment, Continued

BP 16.05 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.

BP 16.06 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.

continued on next page

PA 16: Manage Systems Engineering Support Environment, Continued

BP 16.06 Insert New Technology, continued

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.

BP 16.07 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, 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 power users for selected automation tools,
• develop methodology experts who can be used on a variety of projects, and
• develop process experts who can be used on a variety of projects.

continued on next page

BP 16.08 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 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: Manage Systems Engineering Training

Summary description

The purpose of Manage Systems Engineering Training is to ensure that individuals within the organization have the necessary skill mix to perform their assigned tasks effectively. To achieve this objective, the skill requirements for the systems engineering and related positions within the organization need to be identified, as well as the specific project's or organization's needs such as emergent technology and new products, processes, and policies.

Process area notes

Successful training programs result from an organization's commitment. In addition, successful training programs 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.

Base practices list

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

BP.17.01	Identify training needs throughout the organization using the projects' needs, organizational strategic plan, and existing employee skills as guidance.
BP.17.02	Prepare training materials based upon the identified training needs.
BP.17.03	Train personnel to have the skills and knowledge needed to perform their assigned roles.
BP.17.04	Assess the effectiveness of the training to meet the identified training needs.
BP.17.05	Maintain records of training and experience.
BP.17.06	Maintain training materials in an accessible repository.

continued on next page

BP 17.01 Identify Training Needs

Identify training needs throughout the organization using the projects' needs, organizational strategic plan, and existing employee skills as guidance.

Description
This base practice determines the training that should be offered to provide employees with new skills or maintain an existing skill level. 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 training deficiencies. The organizational strategic plan is used to help identify emerging technologies, and the existing skill level is used to assess current capability.

Identification of training 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 also be offered in the organization's systems engineering process, and in tailoring the process for specific projects.

Typical Work Products
• Organization’s training needs.

Notes
The organization should identify additional training needs as determined from appraisal findings and as part of the defect prevention process. The organization's training plan should be developed and revised according to documented procedure. Each project should develop and maintain a training plan that specifies its training needs.

continued on next page

BP 17.02 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 description and requirements.
• Training material.

Notes
Course description should include
• intended audience,
• preparation for participation,
• training objective,
• length of training,
• lesson plans, and
• criteria for determining the students' satisfactory completion.

The organization should 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; and
• 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.

The organization should review the training material by some or all of the following: instructional experts, subject matters experts, and students from the pilot programs.

continued on next page

BP 17.03 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 that the retention and imparted skill level is the highest possible.
• 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.
• On-line training/customized instruction modules accommodate different learning styles and cultures, in addition to transferring smaller units of knowledge.

BP 17.04 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 17.05 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 17.06 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: Manage Systems Engineering Training

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

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

The following table 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

Appendix A: Change History and Change Request Form,
Continued

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

Issues Form for SECMM-94-04 Version 1.0

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.

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 1.0.

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 w/out 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

Appendix B: Approved Model Requirements, Continued

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, page 1 of 3

continued on next page

Traceability Matrix, cont

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, page 2 of 3

continued on next page

Traceability Matrix, cont

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, page 3 of 3

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)

[FM 770-78] USALMC Army Field Manual, "Systems Engineering," Training Support Center, Ft. Eustis (804-878-4668).

[IEEE 90] *Dictionary of Computing Terms*, IEEE 630-90, 1990.

[IEEE 93] IEEE P1220. *IEEE Standard for Systems Engineering*, Preliminary, 1993.

[MIL-STD-499B] Draft Systems Engineering Standard, AFMC, 1994.

[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.

[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.

continued on next page

Reference list, continued

[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.

[Foster 93] Foster, Kenneth R. "Math, Visualization, and Date Acquisition." *IEEE Spectrum 30* (November 1993): 42-59.

[Hall 62] Hall, Arthur D. *A Methodology for Systems Engineering*. Princeton, N.J.: Van Nostrand Company. 1962.

[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.

[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," *System and Software Requirements Engineering.* Thayer, Richard & Dorfman, Merlin, eds. Washington: IEEE Computer Society Press, 1990: 244-271.

[McAuley 93] McAuley, James E.; & McCumber, William H. eds. *Proceedings of the Third Annual International Symposium of the National Council on Systems Engineering.* 1993.

[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

[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.

[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.

continued on next page

Reference list, continued

[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

The following glossary has been prepared to be applicable to 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 facilitate reader understanding while promoting continuity across the product line.

These definitions are from sources chosen from a wide spectrum of industrial, government, and societal standards, modified only to the extent needed to place them in the SE-CMM context. The basic source of the information has been provided whenever possible.

Definitions with a reference of [SECMM] indicate definitions that were produced by the author team as part of the SE-CMM Project.

Issues Form for SECMM-94-04 Version 1.0

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.

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 1.0.

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 w/out 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.